Arabic News Articles Classification Using Vectorized-Cosine Based on Seed Documents

Mohamed T Elhadi[1]

1- Computer Technology Department, Faculty of Information Technology, Zawia University, Libya.
(mtelhadi@yahoo.com)

Received (2019-02-11) Accepted (2019-04-24)

Abstract: *Besides for its own merits, text classification (TC) has become a cornerstone in many applications. Work presented here is part of and a pre-requisite for a project we have overtaken to create a corpus for the Arabic text process. It is an attempt to create modules automatically that would help speed up the process of classification for any text categorization task. It also serves as a tool for the creation of Arabic text corpora. In particular, we create a text classification process for Arabic news articles downloaded from web news portals and sites. The suggested procedure is a pilot project that uses some human predefined set of documents that have been assigned to some subjects or categories. A vectorized Term Frequency, Inverse Document Frequency (TF-IDF) based information processing was used for the initial verification of the categories. The resulting validated categories used to predict categories for new documents. The experiment used 1000 initial documents pre-assigned into five categories of each with 200 documents assigned. An initial set of 2195 documents were downloaded from a number of Arabic news sources. They were pre-processed for use in testing the utility of the suggested classification procedure using the cosine similarity as a classifier. Results were very encouraging with very satisfying precision, recall and F1-score. It is the intention of the authors to improve the procedure and to use it for Arabic corpora creation.*

Keywords: *Arabic text classification, TFIDF-Vector space model, news articles, Corpora creation.*

How to cite this article:
Mohamed T Elhadi. Arabic News Articles Classification Using Vectorized-Cosine Based on Seed Documents. J. ADV COMP ENG TECHNOL, 5(2) Spring : 107-116

I. INTRODUCTION

THE abundance of electronic documents can be a valuable resource for many tasks, provided that useful and efficient tools and techniques are found and available. Electronic texts made available on the web are prohibitive and ever increasing by the day. The web is considered the major source of documents with its various topics and numerous and divergent relationships [1]. It is estimated that about 2.5 billion GBs of data are created per day in the new research literature, videos, tweets, blogs and text documents. Approximately 90% of the world's data has been created in the last few years [2]. The availability and accessibility of text are changing the field of natural language processing (NLP) from its theoretical scope into linguistic and practical text processing modulus [3].

It is estimated that around 80% of all information is unstructured, with the text being one of the most common types of unstructured data. Thus, making text-analyzing, understanding, organizing, and sorting hard and time-consuming tasks [4]. Such tasks span multiples of fields and are applied in a variety of areas including document filtering, automated metadata generation, document organization, digital libraries, spam filtering, online news, word sense disambiguation, information retrieval, topical crawling, real-time file sorting, topic identifications, social media monitoring, voice of customer, brand monitoring, and customer service. Many single individuals and organizations are turning to TC to help structure the text and analyze it in more cost-efficient ways to help improve decision-making and automate many processes in broader and deeper tasks. It is being applied in many contexts and used in many applications, to help in commercial tasks such as understanding customer behavior using relevant data coming from sources such as social media conversations, and emails. Many search engines, news portals, and e-commerce sites classify content and products to facilitate the search and navigation [4,5,6].

In addition to many other tools and techniques, TC stands out as an important human cognitive ability that is directly usable and related to most text processing tasks and applications.

Principally, TC is the activity of assigning some topic names (labels), representing relevant categories to a piece of text based on its content [2]. The need for and the ability to do TC automatically is essential for the proper handling and efficient utilization of documents and media available in digital format.

TC is an interesting and important task, which falls in the crossroads of many thriving and important fields including machine learning, text processing, natural language understanding and data mining with the aim of helping extract valuable information from large amounts of data.

TC, not only important as text analysis and processing on its own right, but also as a core component in many text processing and natural language applications. It has been applied to a variety of languages with English taking the lion share. Most of the efforts on TC work have been devoted to the classification of English and Latin texts [1,2,7,9]. It is also spreading to other world languages, Arabic among them.

The Arabic language has been receiving more attention by the world in general and by the scientific communities in particular, as it is the main language in the Arab world and the secondary language in many other countries. Arabic is the official language of 25 countries and is spoken by over 250 million just in the Arab world alone and by close to 400 million all over the world. The US Department of Cultural Affairs categorized Arabic, among other world languages, as a critical language. The United Nations heavily emphasizes the social and political importance of these languages. It lists Arabic as one of the six official languages of the United Nations. [1].

Arabic text classification (ATC) is a subfield of TC that is experiencing lots of interest nowadays and is catching up with its counterpart languages [5].

Experiments conducted and work presented in this paper is an attempt to create a simple text classification procedure that would help speed up the process of classification. In particular, this work was very much motivated by the need for a clear process and simple tool that would help create Arabic datasets (corpora). We intended to validate this procedure to be able to use it as a front end for the creation of news articles corpus to be used for Arabic text processing analysis and applications. We created a text classification process for Arabic news articles downloaded from the web.

The suggested procedure discussed here is only an initial pilot project that uses some predefined set of documents (seed documents) that were pre-assigned to some categories by a human. A vector space model that uses Term Frequency, Inverse Document Frequency (TF-IDF) was adopted to represent and score the importance of words in a document based on how frequently they appear across multiple documents. The vector space model along with cosine similarity [4] was used for the creation of a set of representative vectors for the initial verification of the categories. The resulting validated categories were then used for the prediction of categories for new documents using the text and pre-defined categories available from [12], which has five categories of Arts, Politics, Economics, Science, and Sports. Each subject tag or category was manually assigned a set of 200 documents. A test set of 2240 cases was downloaded from a number of Arabic news sources including Aljazeera, CNN, BBC, Al-Aharam, Al-Hayat, and Yahoo. The test set was manually investigated, refined and categorized before using it to test the utility of the suggested procedure. Very encouraging results were obtained as is explained in the rest of this paper.

The paper contains five sections beyond this introductory section. Section II is coverage of important concepts and techniques along with related work. Section III is a description and a presentation of the data sets used for initial categories and the newly collected test set. Section IV is a description of the suggested procedure and used tools and techniques. Section V is a presentation of results along with validation measurements and a discussion of results. Section VI is the conclusion and future work. Finally yet importantly is the list of references consulted in this work.

II. RELATED WORK

TC aims at helping users extract valuable information from a large amount of data using a variety of methods and techniques [1,7,9]. Many methods, techniques, and algorithms have been applied to the problem of classification with varying accuracy and efficiency including, among many others, Support Vector Machine (SVM), K-Nearest Neighbor (KNN), Artificial Neural Networks, Naïve Bayes Classifier, and Decision Trees [4,5,13].

In general, two different ways are used: manual and automatic classification in TC. The manual method involves a human annotator who studies and interprets the content of the text and categorizes it accordingly. This is not only time-consuming and expensive but also unscalable. It is, however, the best and most trusted.

The automatic methods, on the other hand, applies different techniques derived from machine learning and natural language processing. It is not as best and as trusted as the manual, but it is much faster and certainly more cost-effective and scalable. Unless for a small collection of several hundred documents or so, automatic classification is the viable option.

Historically different approaches to automatic text classification have been grouped in many different ways. One such grouping is based on the type of underlying, the model used [9] and it classifies TC into the following:

1. Rule-based systems, which uses a set of handcrafted linguistic rules made of an antecedent and a predicted category. It uses semantically relevant elements of a text to identify relevant categories based on content.

2. Machine Learning based systems in which the system learns the different associations between pieces of text to make classifications using a provided set of past observations using pre-labeled examples as training data.

3. Hybrid systems in which a combination of both 1 and 2 is used.

The predominantly preferred methods involve some machine learning or a combination of machine learning and other techniques with machine learning technique being the core. Organizations turn to TC using machine learning for a number of apparent reasons. Firstly, for scalability which allows easy analysis of millions of documents in a cheaper and much faster way. Secondly, for real-time analysis in situations that require immediate analysis and decision making to enable institutions to identify needed information and take proper action in a timely manner. Thirdly, for a consistent criterion by avoiding many errors and mistakes normally done when humans are involved. Machine learning applies the same criteria to all of the data, thus allowing humans to reduce errors with centralized text classification models [9].

1. Arabic Language

Arabic Language consists of 28 alphabet characters and written from right to left. Its letters have different styles when appearing in a word depending on the letter position. Arabic words have two genders, feminine and masculine; three numbers, singular, dual, and plural; and three grammatical cases, nominative, accusative, and genitive. Words are classified into three main parts of speech, nouns (including adjectives and adverbs), verbs, and particles. All verbs and some nouns are morphologically derived from a list of roots. Words are formed by following fixed patterns. The prefixes and suffixes are added to the word to indicate its number, gender, and tense [10].

The Arabic language has rich morphology and a complex orthography. Different shapes and diacritics of the Arabic language make parsing difficult task due to this complex morphology. Limited access to technology has hindered research in automation and utilization of Arabic. A large percentage of Arabic speakers do not have the ability to read or understand

English. Furthermore, more than 3% of the internet content is Arabic content, which puts it in the fourth rank. Even though Arabic text classification (ATC) is quite limited [1,8,9,14-16], still there are many applicable machine learning algorithms and techniques that can be used for TC.

2. Arabic Text Classification

Arabic text classification (ATC) is similar to TC in other languages and shares the same underlying concepts, motivations, and use. The nature of Arabic language, however, poses a number of challenges due to being orthographic with diacritics making it less ambiguous and more phonetic; its complex morphology and its words which are usually derived from simple bare three-letter verb form (roots). Arabic has a broken plural that does not obey normal morphological rules; existing stemmers that often do not resemble the singular form do not handle them. It also has short vowels, which give different pronunciation. The vowels are grammatically required but omitted in written Arabic texts. Arabic synonyms are widespread which makes exact keyword match inadequate for Arabic retrieval and classification [1,9,11]

Several approaches were proposed for Arabic TC using Naïve Bayes algorithm [17], KNN [16,18-22], SVM [9,19-27], Decision Tree [9,28-30], Manhattan Distance and Dice Measures [15], Polynomial Neural Networks (PNNs) [5], regression model [31,32], and many other techniques and algorithms [9,33]. Fewer studies have however focused on the performance of automatically classifying Arabic language using Arabic corpora [14,15]. Many reasons have been cited demanding that the Arabic language needs enormous processing to construct an accurate categorization system [3].

Among the applied approaches for Arabic Text categorization, a number of recent studies have been proposed [9,25,34-42].

Each work contains a combination of tools and methods and concentrates on some aspect. Mostly are hybrid methods that combine one or more machine learning methods or concenter on some aspect of the process like representation or classifications. Following is a presentation of some of the work found in the filed with highlights of methods used and or aspect address and some results and conclusions drawn by their respective authors. The list in not in any particular order but similar work is presented consecutively.

An intelligent TC for Arabic language approach was proposed in [25] using the statistical n-gram stemmer, a hybrid approach of Document Frequency Thresholding and Information Gain for feature selection, a normalized TF-IDF for term weighting, and Rocchio classifier for classification. The authors demonstrate the accuracy of about 98%.

Another similar work using a hybrid approach proposed combining Document Frequency Thresholding with an embedded information gain criterion of the decision tree algorithm [28]. It used 373 scientific documents fitting in eight categories and 435 literary documents fitting in 14 categories. The authors showed an accuracy of 93% for the scientific corpus and 91% for the literary corpus.

Another intelligent Arabic TC used k-nearest neighbor and Rocchio classifiers and different term weighting schemes with light stemming was performed [8]. Their results show that Rocchio classifier performed better than k-nearest neighbor classifier.

Another comparative study was conducted using KNN and SVM based on full-word features with TF-IDF as the weighting the results [19]. It showed that both methods were of high performance with SVM showed a better micro average F1 and prediction time.

In [15], a machine learning approach for classifying Arabic documents is presented mapping each document by locating the N-gram frequency technique. The classification was achieved by computing the Manhattan distance between the profile of the instance to be classified and the profiles of all the instances in the training set.

Using statistical methods in several Arabic datasets, a tool was implemented for feature extraction and selection using SVM and C5 algorithms. C5.0 classifier showed better results in terms of classification than SVM [36].

In an experiment consisting of text classification system using CHI statistics as a feature extraction method and SVM classification model for TC on a dataset collected from online Arabic newspaper archives consisting of nine classification categories was performed [40]. Results showed an F-measure of 88.

A document indexing method to create, in an automatic manner, an index Arabic books-based on text summarization was conducted [41]. Experimental results in terms of accuracy and performance showed that the proposed method could effectively replace the human time-consuming effort for indexing a large number of documents or books.

A study conducted using stemming and light stemming techniques as feature selection techniques, K-nearest neighbors (KNN) as a classifier [43]. Results reported indicated that light stem was superior over stemming in terms of classifier accuracy.

Authors in [44] proposed a distance-based classifier for categorizing Arabic text. Each category was represented as a vector of words in an m-dimensional space, and documents are categorized based on their closeness to feature vectors of categories.

Use of the rooting algorithm with Naïve Bayes classifier on 300 documents belong to 10 categories and achieves 62.23% of classification accuracy was introduced [8] while an NB and SVM were used on different Arabic datasets with experimental results showing that the SVM algorithm outperforms the NB in F1 measure, Recall and Precision measures [37].

KNN and distance-based classifiers were used and compared in their performances [17].

The authors of [35] used a rule-based approach with rule induction, decision trees, and hybrid classification data mining algorithms approach to Arabic classification. Results showed that the most applicable algorithm was the hybrid approach.

A centroid-based technique was used on a corpus containing a set of 1400 Arabic text documents covering seven distinct categories [52]. The results showed approximate figures of 90.7%, 87.1%, 88.9%, 94.8%, and 5.2% of Micro-averaging recall, precision, F-measure, accuracy, and error rates respectively.

Authors in [42] proposed a hybrid approach based on n-grams and the OKAPI model for the indexing and classification of an Arabic corpus. They took into account the concept of the semantic vicinity of terms and the use of a radial basis modeling.

A new Frequency Ratio Accumulation Method (FRAM) approach using a mathematical model to combine the categorization task with the feature selection task was introduced [38]. The performance of FRAM classifier was compared with three classifiers based on the Bayesian theorem, namely the Simple NB, MNB, and MBNB. Experimental results showed that the FRAM has outperformed the simple NB, MNB, and MBNB achieving 95.1% macro-F1 value by using a unigram word-level representation method.

3. TC Evaluation, and Validation

Evaluation involves estimating to what extent, if any, the suggested procedure is capable of predicting the class labels of the unseen instances. The ability of a classifier to accurately predict is measured by its predictive accuracy computed on the testing examples. Although predictive accuracy is simple and practical, it has been shown to be insufficient and at times not suitable [35,45].

Standard performance metrics that are normally used include Precision, Recall, and F-Score. Other measures suggested and used by researchers have included Receiver Operating Characteristics (ROC) analysis [46,47]. These methods are mainly used to visualize relative performance to determine the "best" classifier based on a comparison of the rate of correctly classified positive examples to the rate of incorrectly classified positive examples for a particular class [48]. The Area under the ROC Curve (AUC) [35,49] produces a single scalar measure to rank classifiers based on how they dominate each other. The ROC curves are shown to be insensitive to the cost of classification, therefore, Cost Curves were proposed in [64] to introduce costs as a factor in comparing the performance of classifiers.

An important consideration that is not addressed by previously stated methods has to do with the important question addressed in [51]: given a classifier and its results on a particular collection of data, how confident are we on the computed precision, recall, or F-score? Many Statistical attempts have used probability to measure confidence in results. Authors of [52] show that the underlying statistical method of Tango's produces more reliable and consistent confidence intervals with good coverage probability.

III. DATASETS AND PROGRAMMING TOOLS

Having sufficient and genuine data is very critical for training, testing, and evaluations. A number of datasets are available for development and testing, particularly for English and Latin text. Arabic, however, suffers from lack of freely available datasets or corpus; thus, this work intends, once ready and validated, to help in the development of freely available annotated Arabic language corpus. A manually classified small set of text documents adopted from [12] is used as a seed for the development of the procedure and consequently for the creation of the sought corpus. Next sections highlight the seed text set that is used for the development of the tool and its validation.

As far as tools and programming environment, we have used a number of packages to apply the proposed procedure. In particular, I have used Python-based Anaconda IDE along with the many useful packages including basic utilities for processing, sklearn, vector space model and cosine, and as mentioned newspaper package. Most of the work made of scripts I have written myself as part of the Arabic corpus creation project currently underway.

1. Seed Documents and Predefined Categories

A set of 1000 documents hand prepared and manually categorized [12] were further refined and used as a seed text for further development, refinement, and validation to be used as a representative set for each category.

For the purpose of prototyping of this work and the adopted procedure, nothing was changed neither to the available text documents nor to the defined categories or assignment even though we believe that further refinement and structuring of the categories is needed and can improve the results.

In this study, 3524 documents were crawled, downloaded and scraped using the Python-based Newspaper module [53]. After pre-processing of the downloaded documents, removing duplicates, non-Arabic text, and very short advertisement listings and newspaper information, 2210 documents were left for use for the experiment. This set was thoroughly hand reviewed to decide on a topic for each document to be compared to the automatic results once done. Most documents were easily decided as belonging to some relevant category. Few documents, however, were indescribable to the human reviewer and where removed. Few other documents could have belonged to more than one category and were given one or the other. A final set of 2195 documents were manually categorized and used for testing and evaluation.

Table 1 next shows the percentages of documents of different categories in the test collection.

TABLE I. THE COMPOSITION OF TEST COLLECTION IN TERMS OF DIFFERENT CATEGORIES

Category	Arts	Econ	Politics	Sciences	Sports	**Total**
Docs	558	316	997	141	183	2195
Perc	25.4%	14.4%	45.4%	6.42%	8.34	100%

The percentages reflect natural sizes of documents available from the web with the highest percentage as expected in politics followed by arts and economics. Sciences and sports accounted for much smaller parts of sets.

IV. SUGGESTED PROCEDURE

The used procedure bases its representation of the document's text on vector space with TF-IDF for both documents and categories. Cosine similarity is used to measure the level of belongings (similarity) between the unknown documents and the category representative. A document can be represented in many ways. One common and simple option is using a bag of words. Then a document can be converted into a vector where each dimension corresponds to a word. Non-appearing words are assigned a value of zero to that dimension. Appearing words would be assigned a value that corresponds to the number of times that a word appears in the document. Normalized techniques such as TF-IDF can be used where the number of times the word appears in all the documents. Different sets of documents can have a different distribution of words, the TF-IDF vector representations of documents depend upon the particular document set selected. Intuitively, when a word shows up frequently in a document, that word becomes important and it is assigned a high score. However, a word that appears in many documents is less important as it is not a unique identifier and thus is assigned a low score. This way a common word (normally termed stop words) like "this" and "for" along with domain terms not particular to any category appear in many documents and will be scaled down. Words that appear frequently in a single document will be scaled up [45].

The suggested procedure goes through three stages, each of which is made of a number of steps as depicted in Figure 1 and 2 below and is further explained in the following paragraphs.

1. Stage One: Categories preparation and refinement

Categories preparation, refinement and validation is the first stage involving some initial pre-processing and preparation of predefined categories representative. This stage is made of the following steps:

1. Merging of each of the sets of documents (200 for each category) into a single document representing the categories as pre-decided by a human. The result is just a large document compromised of the 200-document attached to each category in no particular order.

2. More refining and validation of the categories sets was performed by running each document and testing it against the defined category representatives. This is done for all the available 1000 seed documents. A new category representative is re-defined or refined to contain the newly decided documents. The process of refinement continues until all the 1000 document are properly categorized or no more documents can be correctly categorized.

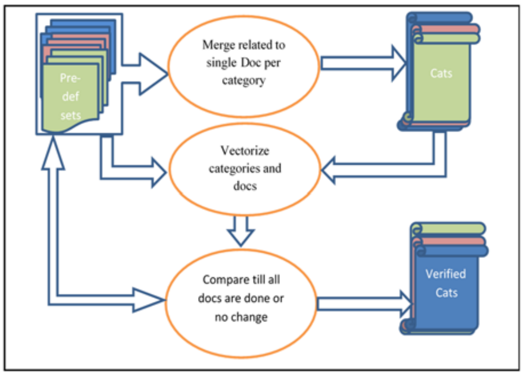
Pre-def sets
Merge related to single Doc per category
Cats
Vectorize categories and docs
Compare till all docs are done or no change
Verified Cats

Figure 1. Stage one: Categories refinements and evaluation

The process was able to rightly re-assign each document to its pre-defined category. All documents were rightly re-assigned except for a few that were left out before refinements could produce no more re-assignments with 98.7% was rightly re-assigned.

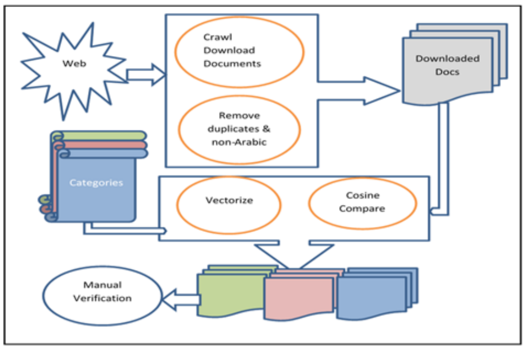

Figure 2. Stage two: New documents download, preparation, and evaluations

2. Stage Two: New documents downloading and preparations

In this stage, new documents were downloaded using Newspaper Scraper [52]. For the purpose of validation, only 3524 documents were downloaded and filtered in the following steps:

1. Downloading of the HTML version of the news articles.

2. Scrapping the text of the downloaded documents.

3. Removal of any duplication and non-Arabic documents or parts of.

4. Removal of very short documents representing advertisements and instructions for the user but not at all of the news items.

5. Hand investigation and categorization of the documents to be used for verification in the next stage.

This stage resulted in a set of documents ready to be used for text classification. The set of documents were pre-categorized by humans and were ran on the classifier. The results were further compared and evaluated. A set of 2195 documents were downloaded, prepared and hand pre-compiled as belonging to which category. The set of documents are then fed to the next stage one at a time using the same procedure of combined IDFIF and cosine similarity. This is considered the final stage where newly downloaded, prepared and hand-categorized documents were subjected to the same procedure using TF-IDF and cosine similarity based on the predefined categories representative from previous stages. The result is an automatic classification of each document.

3. Stage Three: Manual comparison and evaluation of the results

In this stage, the automatically classified documents were manually compared against the hand categories classification in order to evaluate the results obtained. Further discussions of the results are presented in the next section.

V. RESULTS, EVALUATIONS AND ANALYSIS

As shown in table II and III, the obtained results were surprisingly quite good. Precision ranged from 95.52 in the arts category to 99.02% in politics, whereas recall ranged from 95.57% in the Art category to 100% in Sports. F1 Measure ranged from 94.03% in the Science category to 99.19 in the Sports category.

TABLE II. CONFUSION MATRIX

Categories		Positive	Negative	Total
Art	Positive	555	26	581
	Negative	3	1604	1607
	Total	558	1630	2188
Economics	Positive	316	3	319
	Negative	3	1604	1607
	Total	319	1607	1926
Politics	Positive	997	10	1007
	Negative	11	1882	1893
	Total	1008	1892	2900
Science	Positive	141	1	142
	Negative	15	2053	2068
	Total	156	2054	2210
Sport	Positive	183	3	186
	Negative	0	2011	2011
	Total	183	2014	2197

The overall measures were very satisfactory but rather extraordinary for most of the cases. The overall average was 0.97 for F1-score. The best results were for the sports category. This is very encouraging and shows that the adopted procedure, as simple as it is, can serve the purpose of classification for corpus collection.

Future work will better confirm the results based on a bigger and more complete dataset.

An important observation worth mentioning is that even though the seed documents are several years old, they still worked well as category presentative.

Future work to be done will address a number of issues including:

1. Accumulative refinement of the categories to reflect newly classified documents' content;

2. Detailed comparison of the procedure to other procedures used in the literature using the same experiment setting;

3. Continuation of the creation of sizable Arabic text corpus with variation on categories to include more categories and subcategories.

TABLE III. STANDARD EVALUATION METRICS

Category	PRECISION	RECALL	F1
Arts	0.9552	0.9946	0.9745
Economics	0.9902	0.9557	0.97
Politics	0.99	0.989	0.9895
Sciences	0.9921	0.8936	0.9403
Sports	0.9839	1	0.9919
Averages	0.98228	0.96658	0.97324

VI. CONCLUSION

TC has become a cornerstone in many applications and is experiencing increased interest by researchers. Work presented here is part of a project undertaken with the aim of the creation of a corpus for Arabic text process. The suggested procedure is a pre-requisite for the automatically creation of such dataset. The work is an attempt to create a simple TC module that would help speed up the process of classification. It also serves as a procedure to use for the creation of Arabic text corpora. In particular, we create a text classification process for Arabic news articles downloaded from web news portals and sites. The suggested procedure is a pilot project that used a seed of human predefined and manually labeled set of documents. A hybrid vectorized Term Frequency, Inverse Document Frequency (TF-IDF) based information processing model and cosine similarity was used for the representation and verification of the initial categories based on the seed dataset. The resulting validated categories were then used for predicting categories for new documents. The experiment used one thousand initial set of documents pre-assigned into five categories each with 200 example documents [12]. A newer set of 2195 documents was downloaded and prepared from online Arabic news sources. The set was pre-processed for use in testing the utility of the procedure. As presented above results were very encouraging with very satisfying precision, recall and F1 values. It is the intention of the authors to improve the procedure and to use it for Arabic corpora creation.

REFERENCES

1. Olayah F., Alromina W. Automatic Machine Learning Techniques (AMLT) for Arabic Text Classification Based on Arabic Term Collections. Journal of Theoretical & Applied Information Technology. 2018 Jun 30;96 (12).

2. Mirończuk MM, Protasiewicz J. A recent overview of the state-of-the-art elements of text classification. Expert Systems with Applications. 2018 Sep 15;106:36-54.

3. Schneider, S. The biggest data challenges that you might not even know you have, https://www.ibm.com/blogs/watson/2016/05/biggest-data-challenges-might-not-even-know/

4. Shahmirzadi O, Lugowski A, Younge KA. Text Similarity in Vector Space Models: A Comparative Study. Available at SSRN 3259971. 2018 Sep 15.

5. Al-Tahrawi MM, Al-Khatib SN. Arabic text classification using Polynomial Networks. Journal of King Saud University-Computer and Information Sciences. 2015 Oct 1;27(4):437-49.

6. Zhu Z. Improving Search Engines via Classification. The University of London. 2011 May.

7. Ahmed M, Elhassan R. Arabic Text Classification review. International Journal of Computer Science and Software Engineering. 2015 Jan 31;4(1):1-5.

8. Syiam MM, Fayed ZT, Habib MB. An intelligent system for Arabic text categorization. International Journal of Intelligent Computing and Information Sciences. 2006 Jan 1;6(1):1-9.

9. Mohammad AH, Alwada'n T, Al-Momani O. Arabic text categorization using support vector machine, Naïve Bayes and neural network. GSTF Journal on Computing. 2018 Jan 23;5(1).

10. Elhassan R, Ali M. Arabic Text Classification Process. International Journal of Computer Science and Software Engineering. 2017 Nov 1;6(11):258-65.

11. Jindal V. A Personalized Markov Clustering and Deep Learning Approach for Arabic Text Categorization. In Proceedings of the ACL 2016 Student Research Workshop 2016 (pp. 145-151).

12. Bani-Ismail, B, Al-Rababah, K, Shatnawi, S., The effect of full word, stem, and root as index-term on Arabic information retrieval, Global Journal of Computer Science and Technology, 2011

13. Agarwal R, Dhar V. Big data, data science, and analytics: The opportunity and challenge for IS research (2014).

14. A. M. El-Halees, "Arabic text classification using maximum entropy," IUG Journal of Natural Studies, vol. 15, no. 1, 2015.

15. L. Khreisat, "A machine learning approach for Arabic text classification using N-gram frequency statistics," Journal of Informatics, vol. 3(1), pp. 72-77, 2009.

16. Al-Shalabi R, Kanaan G, Gharaibeh M. Arabic text categorization using KNN algorithm. In Proceedings of the 4th International Multiconference on Computer Science and Information Technology 2006 Apr 5 (Vol. 4, pp. 5-7).

17. Duwairi, R.M.: Arabic text categorization. Int. Arab J. Inf. Technol. 4(2), 125–132 (2007) 17

18. R. Al-Shalabi and R. Obeidat, "Improving knn Arabic text classification with n-grams based document indexing," in Proceedings of the Sixth International Conference on Informatics and Systems, Cairo, Egypt, 2008, pp. 108–112.

19. Hmeidi I, Hawashin B, El-Qawasmeh E. Performance of KNN and SVM classifiers on full word Arabic articles. Advanced Engineering Informatics. 2008 Jan 1;22(1):106-11.

20. Masand, B., Linoff, G. and Waltz, D., 1992, June. Classifying news stories using memory-based reasoning. In Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval (pp. 59-65). ACM.

21. Lam, W. and Ho, C.Y., 1998, August. Using a generalized instance set for automatic text categorization. In Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval (pp. 81-89). ACM.

22. Yang Y, Pedersen JO. A comparative study on feature selection in text categorization. In Icml 1997 Jul 8 (Vol. 97, pp. 412-420).

23. El-Halees AM. A comparative study on Arabic text classification. A comparative study on Arabic text classification. 2008;30(2).

24. Al-Saleem S. Associative classification to categorize Arabic data sets. Int. J. Acm Jordan. 2010;1(3):118-3.

25. Mohamed A Mesleh A. Chi Square Feature Extraction Based SVMs Arabic Language Text Categorization System. Journal of Computer Science. 2007;3(6):430-5.

26. Chantar HK, Corne DW. Feature subset selection for Arabic document categorization using BPSO-KNN. In Nature and Biologically Inspired Computing (NaBIC), 2011 Third World Congress on 2011 Oct 19 (pp. 546-551). IEEE.

27. Joachims T. Text categorization with support vector machines: Learning with many relevant features. In European conference on machine learning 1998 Apr 21 (pp. 137-142). Springer, Berlin, Heidelberg.

28. Harrag, F., El-Qawasmeh, E., & Pichappan, P. (2009). Improving Arabic text categorization using decision trees.

Networked Digital Technologies, 110-115.

29. Lewis DD, Ringuette M. A comparison of two learning algorithms for text categorization. In Third annual symposium on document analysis and information retrieval 1994 Apr 11 (Vol. 33, pp. 81-93).

30. Apte C, Damerau F, Weiss S. Text mining with decision rules and decision trees. IBM Thomas J. Watson Research Division; 1998 Jun.

31. Fuhr N, Hartmann S, Lustig G, Schwantner M, Tzeras K, Knorz G. AIR/X: A rule-based multistage indexing system for large subject fields. In Intelligent Text and Image Handling-Volume 2 1991 Apr 2 (pp. 606-623.

32. Yang Y, Chute CG. An example-based mapping method for text categorization and retrieval. ACM Transactions on Information Systems (TOIS). 1994 Jul 1;12(3):252-77.

33. Dumais S, Platt J, Heckerman D, Sahami M. Inductive learning algorithms and representations for text categorization. In Proceedings of the seventh international conference on Information and knowledge management 1998 Nov 1 (pp. 148-155). ACM.

34. Alshammari R. Arabic Text Categorization using Machine Learning Approaches. Inter J. of Advanced Computer Science and Applications. 2018 Mar 1;9(3):226-30.

35. Abu-Errub A. Arabic text classification algorithm using TFIDF and chi-square measurements. International Journal of Computer Applications. 2014 Jan 1;93(6).

36. Farghaly A, Shaalan K. Arabic natural language processing: Challenges and solutions. ACM Transactions on Asian Language Information Processing (TALIP). 2009 Dec 1;8(4):14.

37. Liu RL. Context recognition for hierarchical text classification. Journal of the American society for information science and technology. 2009 Apr;60(4):803-13.

38. Sharef BT, Omar N, Sharef ZT. An automated Arabic text categorization based on the frequency ratio accumulation. Int. Arab J. Inf. Technol. 2014 Mar 1;11(2):213-21.

39. Alsaleem S. Automated Arabic Text Categorization Using SVM and NB. Int. Arab J. e-Technol. 2011 Jun;2(2):124-8.

40. Abu-Errub A. Arabic text classification algorithm using TFIDF and chi-square measurements. International Journal of Computer Applications. 2014 Jan 1;93(6).

41. Noaman HM, Elmougy S, Ghoneim A, Hamza T. Naive Bayes classifier based Arabic document categorization. In Informatics and Systems (INFOS), 2010 The 7th International Conference on 2010 Mar 28 (pp. 1-5). IEEE.

42. Duwairi R, Al-Refai M, Khasawneh N. Stemming versus light stemming as feature selection techniques for Arabic text categorization. In Innovations in Information Technology, 2007. IIT'07. 4th International Conference on 2007 Nov 18 (pp. 446-450). IEEE.

43. Duwairi RM. Machine learning for Arabic text categorization. Journal of the American Society for Information Science and Technology. 2006 Jun;57(8):1005-10.

44. Ling CX, Huang J, Zhang H. AUC: a better measure than accuracy in comparing learning algorithms. In Conference of the Canadian society for computational studies of intelligence 2003 Jun 11 (pp. 329-341). Springer, Berlin, Heidelberg.

45. Provost FJ, Fawcett T. Analysis and visualization of classifier performance: Comparison under imprecise class and cost distributions. In KDD 1997 Aug 14 (Vol. 97, pp. 43-48).

46. William C, Robert ES, Singer Y. Learning to order things. Journal of Artificial Intelligence Research. 1999;10:243-70.

47. Elazmeh W, Japkowicz N, Matwin S. Confidence Interval for the Difference in Classification Error. In American Association for Artificial Intelligence.

48. Caruana R, Niculescu-Mizil A. Data mining in metric space: an empirical analysis of supervised learning performance criteria. In Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining 2004 Aug 22 (pp. 69-78). ACM.

49. Drummond C, Holte RC. What ROC Curves Can't Do (and Cost Curves Can). In ROCAI 2004 Aug 22 (pp. 19-26).

50. Goutte C, Gaussier E. A probabilistic interpretation of precision, recall, and F-score, with implication for evaluation. In European Conference on Information Retrieval 2005 Mar 21 (pp. 345-359). Springer, Berlin, Heidelberg.

51. Newcombe RG. Two-sided confidence intervals for the single proportion: comparison of seven methods. Statistics in medicine. 1998 Apr 30;17(8):857-72.

52. Ou-Yang L. Newspaper: Article scraping & curation. Python Library. Retrieved. 2013.