BAU Journal - Science and Technology

Volume 3 | Issue 1
ISSN: 2706-784X

Article 10

December 2021

A NOVEL ARABIC CORPUS FOR TEXT CLASSIFICATION USING DEEP LEARNING AND WORD EMBEDDING

Roua A. Abou Khachfeh
PhD Student, Faculty of Science, Beirut Arab University, Beirut, Lebanon, roua.aboukhachfeh@bau.edu.lb

Islam El Kabani
Assisstant Professor, Faculty of Science, Alexandria University, Alexandria, Egypt,
islam.kabani@alexu.edu.eg

Ziad Osman
Professor, Faculty of Engineering, Beirut Arab University, Beirut, Lebanon, zosman@bau.edu.lb

A NOVEL ARABIC CORPUS FOR TEXT CLASSIFICATION USING DEEP LEARNING AND WORD EMBEDDING

Abstract

Over the last years, Natural Language Processing (NLP) for Arabic language has obtained increasing importance due to the massive textual information available online in an unstructured text format, and its capability in facilitating and making information retrieval easier. One of the widely used NLP task is “Text Classification”. Its goal is to employ machine learning technics to automatically classify the text documents into one or more predefined categories. An important step in machine learning is to find suitable and large data for training and testing an algorithm. Moreover, Deep Learning (DL), the trending machine learning research, requires a lot of data and needs to be trained with several different and challenging datasets to perform to its best. Currently, there are few available corpora used in Arabic text categorization research. These corpora are small and some of them are unbalanced or contains redundant data. In this paper, a new voluminous Arabic corpus is proposed. This corpus is collected from 16 Arabic online news portals using an automated web crawling process. Two versions are available: the first is imbalanced and contains 3252934 articles distributed into 8 predefined categories. This version can be used to generate Arabic word embedding; the second is balanced and contains 720000 articles also distributed into 8 predefined categories with 90000 each. It can be used in Arabic text classification research. The corpus can be made available for research purpose upon request. Two experiments were conducted to show the impact of dataset size and the use of word2vec pre-trained word embedding on the performance of Arabic text classification using deep learning model.

Keywords

Arabic text categorization, Deep learning, Word embedding, Arabic corpus

1. INTRODUCTION

Automatic Text Classification is the process of automatically assigning a text document to a set of pre-defined classes, by employing a specific machine learning technique, based on the content and the extracted features. It is an essential component in many applications such as Subject Categorization, Spam Detection, Sentiment Analysis of reviews or opinions, Language Identification, authorship recognition of documents, etc. The importance of text classification increases due to the huge textual information available online and its capability in facilitating and making information retrieval easier [1].

Standard methods of text classification consist of two main steps. The first step deals with representing documents with high-dimensional feature vectors. The commonly used model is the Bag of Words model (BOW) where unigrams, bigrams, n-grams or some other designed patterns are extracted as features. Furthermore, several feature selection methods, such as Term Frequency-Inverse Document frequency (TF-IDF), singular value decomposition (SVD) are needed to reduce dimensionality of data and take only the most important features for classification. To reduce the high dimensionality of word representations, a new approach "word embedding" has been proposed [2]. It is a distributed representation of words learned with neural network based models in a way that alleviates the data sparsity problem and capture meaningful syntactic and semantic consistencies: each word corresponds to a point in a feature space, so that similar words get to be closer to each other in that space. The second step involves training classifiers by using machine learning techniques. Many conventional machine learning techniques were used for text classification such as Naïve Bayes (NB), Support Vector Machine (SVM) and shallow Artificial Neural Network (ANN). The difference between them lies in the type and number of features used in classification as well as their accuracy results obtained. However, these techniques were limited in their ability to process natural data in their raw and they required careful feature engineering and significant domain expertise to build a feature extractor able to transform the data from raw into an appropriate internal representation or feature vector. In addition, these techniques handle only supervised data. Moreover a lot of data currently available online are unlabeled. In 2006, Deep Learning (DL), a new subset of machine learning research has evolved with the objective of moving it closer to one of its original goals: Artificial Intelligence. DL based models do not require traditional, task-specific feature engineering. These models are based on forming a hierarchy of concepts by learning multiple levels of data representations corresponding to different levels of abstraction. Deep learning techniques achieved a high performance in a broad variety of applications such as computer vision, speech processing and natural language processing. In addition, it is started outperforming traditional machine learning techniques for many driving factors: a lot of data that have been hugely available online, new techniques and the faster machines with multicore CPU/GPU that favor deep learning[3].

Many researches have been worked on text classification for English and western languages. However, researches on text classification for Arabic language are limited. Arabic Text Classification remains a challenging task due to the complexity and rich morphological structure of the Arabic language.

The purpose of this work is two-fold. First we build a large free access corpus that could be used as a rich representative resource in different Arabic NLP research areas such as Arabic Text Categorization as well as producing word-embedding models (using word2vec, fastText, etc.). Second, we use this corpus to investigate the performance of deep learning model for Arabic text classification. We further employ word embedding to enhance the classification performance.

2. LITERATURE REVIEW

Several researches addressed the problem of text classification using different technics. In [4], the authors investigated Naïve Bayesian method (NB) and Support Vector Machine algorithm (SVM) approaches to text categorization. The data set used in this work, collected from the Saudi Newspapers, consists of 5121 documents belonging to seven categories. Several preprocessing steps have been conducted on the data set including the removal of Arabic function words, normalization of letters and filtering of non-Arabic texts. The text has been represented using BOW model with standard normalized feature weighting. An average F_1 score of 77.85% and 74% have been obtained for SMV and NB respectively.

In [5], the authors addressed the problem of BOW model that ignores important semantic relationships between terms. They used the Bag of Concept (BOC) text representation model for Arabic texts. The BOC model is used to generate a vector space model using Wikipedia as a knowledge source for building the model. Two datasets were used in this work. The first dataset was collected from online Arabic newspaper archives and consists of 1445 documents falling into nine different categories. The second dataset is Saudi Newspapers dataset consisting of 5121 documents falling into seven categories. Naïve Bayes and Random Forest Classifier have been employed to evaluate the efficiency of the proposed model compared to the basic BOW model. The results of their experiments showed that the suggested BOC model achieved an improvement in the classification accuracy by 10% with respect to the BOW model.

In [6], the authors presented a recurrent convolutional neural network for text classification without human-designed features. First, the authors applied a bi-directional recurrent structure to learn word representations by capturing the contextual information. Then, they employed a max-pooling layer that selects the features playing key roles in text classification. The Skip-gram model was used for pre-training the word embedding. The authors performed the experiments using four datasets: 20Newsgroups (English dataset) Fudan Set (Chinese dataset), ACL Anthology Network (English dataset), and Sentiment Treebank (English dataset). The results of their experiments showed that the proposed method achieved an improvement in the classification performance compared to the baselines and state-of-the-art methods on several datasets.

In [7], the authors in this research aimed to follow a process for handling unsupervised data by using a pre-training technique. They used Word2Vec to quantify words and generates word embedding. Then, large unsupervised texts were quantified using the vectors of words. To obtain a better initial value of weight in the network, pre-training used denoising autoencoder were conducted in the neural network to handle unsupervised data. Same process is performed for fine-tuning conducted to learn the network by using supervised data. For the analysis and experiments conducted in this research, they used Japanese text data delivered by Japanese websites. The number of records used was 1,728,942 records, and the vocabulary size was 218,295. As a result of this research, the proposed method achieved superior accuracy (78%) compared to the Naive Bayes method (68%).

In [8], the authors conducted several comparative experiments on English texts in order to investigate how effective is the use of word embedding on text classification. The results of their research showed that the models employed with word embedding, on English texts, outperform the models that uses traditional methods.

3. REVIEWING EXISTING CORPORA

Different studies are conducted on text classification for Arabic language. In these studies, researchers experimented different classification algorithms on corpora already existing or they created their own corpus.

However, the majority of the Arabic Corpora that are currently available online for Arabic Text classification research have a small number of documents and some of them contains redundant articles. In addition, the classes assigned to articles are imbalanced. These issues reduce the classification accuracy, mainly for the trending machine learning technics.

To analyze the commonly used Arabic corpora, we have implemented a C# console application that reads the articles from a given corpus and compare their content in order to find the redundant articles among them (i.e. the files having exactly the same content). In addition, it counts the words in each file to discover if there are articles with no entries.

Table 1 provides the commonly datasets used in Arabic text categorization research. From the number of records of each dataset, the following drawbacks can be reflected. Khaleej-2004 dataset is small, and imbalanced. Watan-2004 dataset is small and contains 3421 redundant articles out of 20291. BBC dataset is small, imbalanced and contains 508 redundant articles out of 4763. CNN dataset is small and imbalanced. OSAC dataset is small, imbalanced and contains 1166 redundant articles out of 22429. DAA dataset is small. BINIZ dataset is imbalanced, contains 6142 redundant articles out of 111728 and it also includes empty articles. RTA News dataset is imbalanced and includes 40 categories where some of them are very closely related. SANAD dataset is imbalanced and contains 1115 redundant articles out of 193807.

Table 1: Commonly datasets used in Arabic text categorization research

Dataset	# of Categories	Categories	# of Records
Khaleej-2004 (2004) [9]	4	Economy (909), International News (953), Local News (2398), Sports (1430)	Total Records: 5690 Redundant Records: 5 Distinct Records: 5685
Watan-2004 (2004) [10]	6	culture (2782), economy (3468), International (2035), Local (3596), Religion (3860), Sports (4550)	Total Records: 20291 Redundant Records: 3421 Distinct Records: 16870
BBC [11]	7	Middle East News (2356), World News (1489), Economic & Business (296), Sport (219), Newspaper (49), Science & Technology (232), Mix (122)	Total Records: 4763 Redundant Records: 508 Distinct Records: 4255
CNN [11]	6	Business (836), Entertainment (474), Middle East News (1462), Science & Technology (526), Sport (762), World News (1010)	Total Records: 5070 Redundant Records: 1 Distinct Records: 5069
OSAC (2010) [11]	10	Economic (3102), History (3233), Education & Family (3608), Religious & Fatwas (3171), Sport (2419), Health (2296), Astronomy (557), Law (944), Stories (726), Cooking Recipes (2373)	Total Records: 22429 Redundant Records: 1166 Distinct Records: 21263
Diab Dataset DAA (2014) [12]	9	Art (300), Economy (300), Health (300), Law (300), Literature (300), Politics (300), Religion (300), Sport (300), Technology (300)	Total Records: 2700 Redundant Records: 3 Distinct Records: 2697
BINIZ Mohamed Arabic Classification Dataset (2018) [13]	5	culture (13738), diverse (16728), economy (14235), politic (20505), sport (46522)	Total Records: 111728 Redundant Records: 6142 Distinct Records: 105586
RTA News (2018) [14]	40	Oil Markets (1247), Discoveries (408), Ukrainian crisis (502), Syrian Crisis (2104), Medical Research (393), Crimes (400), diseases (322), etc.	Total Records: 23837 Redundant Records: 0 Distinct Records: 23837
SANAD (2019) [15]	7	Culture (18865), Finance (45856), Medical (23162), Politics (24847), Religion (14022), Sports (45313), Technology (22857)	Total Records: 194922 Redundant Records: 1115 Distinct Records: 193807

4. BUILDING THE CORPUS

The process of building our corpus consists of two steps: the first step is the data collection by crawling several news portals. The second step is a basic preprocessing step for the data collected.

4.1 Data Collection

Our automated news portals crawler is implemented as a C# console application to extract the specific HTML data from 16 news portals mentioned in Table 2. It is based on analyzing the DOM structure of the source page for each HTML page and using XPath to query parts of the structure in order to extract specific data for the news collected (like title, description, author, publication date, etc.). The data collected are saved in SQL database for later retrieval.

Table 2: News portals crawled to collect our data

News portal	**URL**
Aitnews	https://aitnews.com/latest-it-news
Akhbarona	https://www.akhbarona.com/
Albawaba	https://www.albawaba.com/
Al-hayat	http://www.alhayat.com/
Almaghribtoday	https://www.almaghribtoday.net/
Annahar	https://www.annahar.com/
Altibbi	https://www.altibbi.com/
Arabstoday	https://www.arabstoday.net/
Elfann	https://www.elfann.com/
Eliktisad	https://www.eliktisad.com
Elnashra	https://www.elnashra.com/
Elsport	https://www.elsport.com
Hekmah	http://hekmah.org/
Sehhanews	http://sehhanews.com/
Sohati	https://www.sohati.com/
youm7	https://www.youm7.com/

Fig.1 represents the algorithm used for the Automated Data Collection Process. After crawling the news portals, the articles are retrieved from the database using a C# console application. For each news portals, a folder is created with the name of the news portal itself and the articles are saved inside subfolders entitled with the name of each category or subcategory. Table 3 shows the number of articles collected from each news portal.

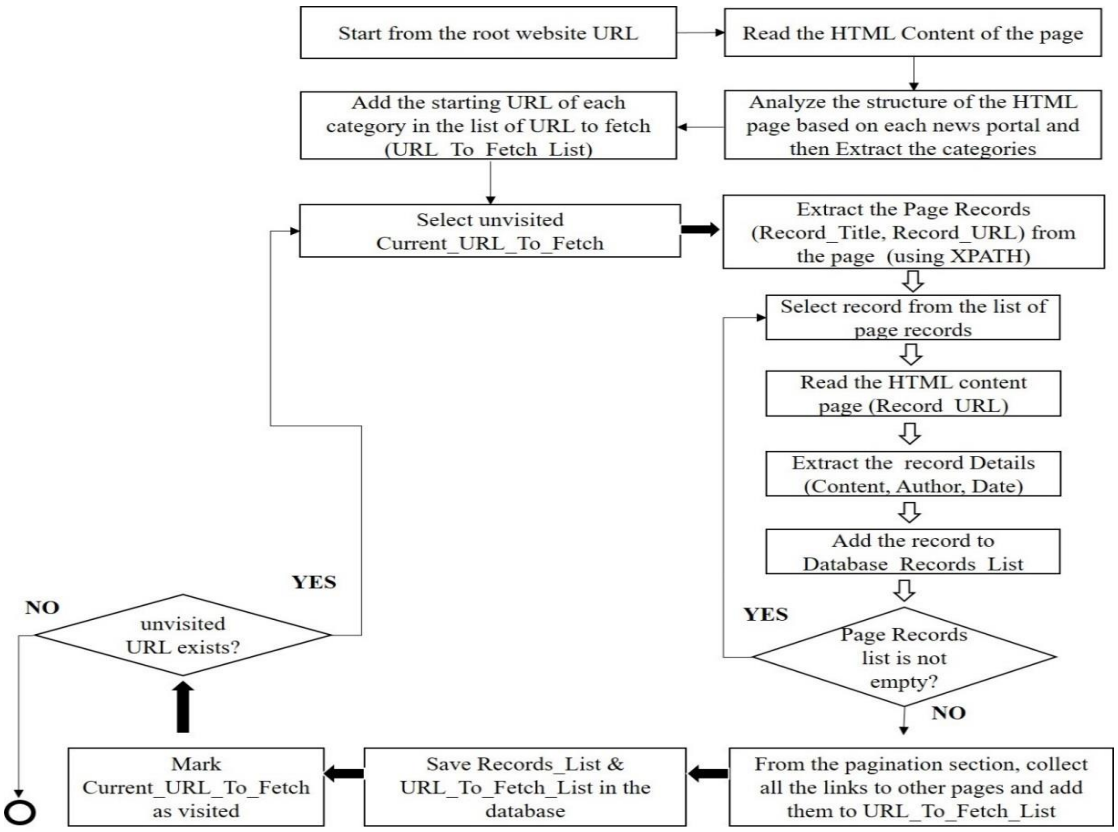

Fig.1: Algorithm for Data Collection Process

Table 3: Number of articles collected from each news portal

News portal	**Number of articles collected**
Aitnews	30257
Akhbarona	67788
Albawaba	33025
Al-hayat	33756
Almaghribtoday	338282
Annahar	14059
Altibbi	7826
Arabstoday	194757
Elfann	135028
Eliktisad	212493
Elnashra	273511
Elsport	261290
Hekmah	1054
Sehhanews	12083
Sohati	6168
youm7	1631557
Total	**3252934**

From this corpus, we have selected the articles having words count greater than 30. The articles are then grouped into 8 categories: Art (فن), Economy(اقتصاد) , Judiciary and Accidents (قضاء وحوادث), Science & Technology (علوم وتكنولوجيا) , Sport (رياضة), Health (صحة) , Politics (سياسة) , Culture (ثقافة) . In addition, the articles having the same content are eliminated to avoid duplication. We end up with an imbalanced dataset of 2243616 articles. Table 4 shows the distribution of articles per category and the average number of words per category.

Table 4: Distribution of articles per category and the average number of words per category for the imbalanced corpus

Category	Number of articles	Average Count words/article
Art	319363	140
Economy	473946	182
Judiciary and Accidents	173190	129
Science & Technology	107741	211
Sport	529613	145
Health	180529	252
Politics	328401	194
Culture	130833	237
Total	2243616	

Since each category has different number of articles, and to be aligned with the category having the minimum number of records, we have selected the top 90000 articles from each category sorted by the maximum number of words to have a balanced dataset with 720000 records suitable to be used for Arabic text classification research as shown in Table 5.

Table 5: Distribution of articles per category and the average number of words per category for the balanced corpus

Category	Number of articles	Average Count words/article
Art	90000	268
Economy	90000	439
Judiciary and Accidents	90000	182
Science & Technology	90000	236
Sport	90000	315
Health	90000	356
Politics	90000	366
Culture	90000	303
Total	720000	

4.2 Data Preprocessing

To have a clean Arabic text in our corpus, several preprocessing steps were applied to our corpus using python. These steps include filtering non-arabic character, removing special characters and punctuation and normalizing text (this includes, as example, replacing the letters "إ آ أ" with the letter "ا").

5. TEXT CATEGORIZATION: EXPERIMENTS AND RESULTS

In this paper we have conducted two basic experiments to test the influence of the size of the dataset and the use of pre-training word embedding on the performance of text classification. The experiments are done on a server with processor Intel(R) Xeon(R) CPU E5-2697 v4 @ 2.30 GHz 2.30 GHz, NVIDIA Quadro K1200 GPU with VRAM 4GB and RAM 256 GB.

In both experiments, we trained a basic deep learning model (a shallow convolutional neural network "CNN"). The architecture of this shallow CNN consists of an embedding layer to enhance the computational efficiency, a dropout layer for reducing overfitting, followed by a 1D convolutional layer with kernel size of 2 and 50 filters, followed by a global max-pooling layer, then a vanilla hidden layer and finally the fully-connected softmax layer responsible for text classification.

As for the first experiment conducted to test the influence of the size of the datasets on the performance of the classification, we have trained the model on 4 version of datasets with different sizes (10000 records, 100000 records, 200000 records, 400000 records) selected from the whole dataset we have collected.

As for the second experiment, a new pre-trained model is created based on an existing word embedding model build by Soliman et.al. [17] using twitter dataset. We have trained this existing model by using Wikipedia Arabic dump and the entire dataset we have collected. Continuous bag of word (CBOW) model using word2vec [2] is used as a strategy to generate word embedding representations. The embedding layer is then initialized by loading the pre-trained model created instead of being initialized with random weights. This experiment is conducted only on the two smallest datasets (10000 records and 100000 records) due to computation limitations on the server we are working on.

For the performance measure, F_1 measure, a popular metric, is used for the evaluation of the classifier. It provides a way to combine both precision and recall into a single measure that captures both properties.

- Precision (P) is the ratio of how much of the predicted labels is correct:
  $P = \frac{T_p}{T_p+F_p}$ where T_p denotes the count of true positives and F_p denotes the count of false positives.
- Recall (R) is the ratio of how many of the actual labels were predicted:
  $R = \frac{T_p}{T_p+F_n}$ where T_p denotes the count of true positives and F_n denotes the count of false negatives.
- F_1 measure is the harmonic mean of precision and recall:
  $F_1 = 2*\frac{P*R}{P+R}$

Table 6 and Fig.2 illustrate the F_1 measure results of the first experiment. The results show that the performance of the classifier model increases with the increase in the dataset size for all the categories. This increase is well noted for all categories specifically for Economy (from 0.81 to 0.86), Politics (0.78 to 0.85) and Culture from (0.76 to 0.85). Furthermore, Sport category achieved the highest F_1 measure among all categories ranging from 0.95 (for the smallest dataset) to 0.97 (for the biggest dataset). The average F_1 measure range from 0.86 (for the smallest dataset) to 0.90 (for the biggest dataset).

Table 6: F_1 Measure Results of the First Experiment

Category	10000	100000	200000	400000
Economy	0.81	0.83	0.85	0.86
Art	0.86	0.87	0.88	0.88
Judiciary & Accidents	0.90	0.91	0.92	0.92
Science & Technology	0.88	0.90	0.90	0.91
Sport	0.95	0.97	0.97	0.97
Health	0.92	0.93	0.93	0.94
Politics	0.78	0.81	0.83	0.85
Culture	0.76	0.82	0.83	0.85
Average	**0.86**	**0.88**	**0.89**	**0.90**

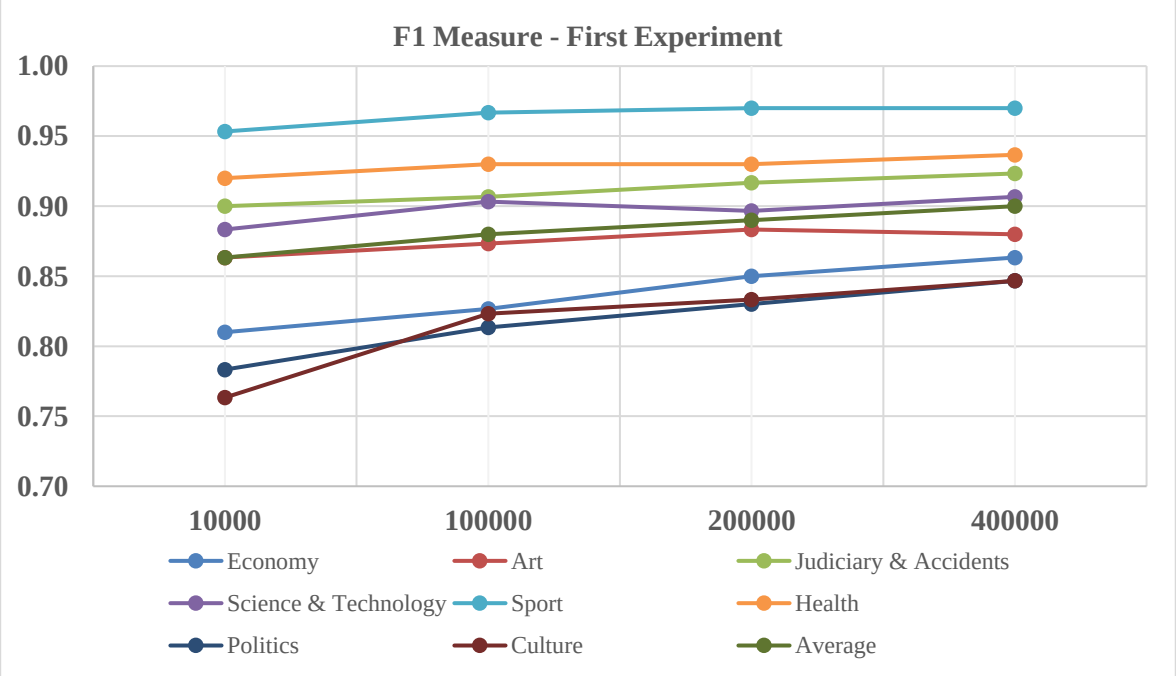

Fig.2: F_1 Measure of the First Experiment showing better performance with the increase of dataset size

As for the second experiment the results obtained for the two datasets (10000 and 100000), in Table 7, Fig.3, Table 8 and Fig.4, show an improvement in the performance of the classification model for all categories when using a pre-trained word embedding model in the embedding layer. The average F_1 measure increases from 0.86 to 0.89 for the dataset with size 10000 and from 0.88 to 0.92 for the dataset with size 100000. This improvement is due to the fact that pre-trained word embedding, when trained on a large dataset, are able to capture both syntactic and semantic meaning of words and then improving the overall performance of the classification model.

Table 7: F_1 Measure Results of First Experiment versus Second Experiment for the Dataset of Size 10000

Category	First Experiment	Second Experiment
Economy	0.81	0.83
Art	0.86	0.89
Judiciary & Accidents	0.90	0.92
Science & Technology	0.88	0.93
Sport	0.95	0.97
Health	0.92	0.93
Politics	0.78	0.80
Culture	0.76	0.83
Average	**0.86**	**0.89**

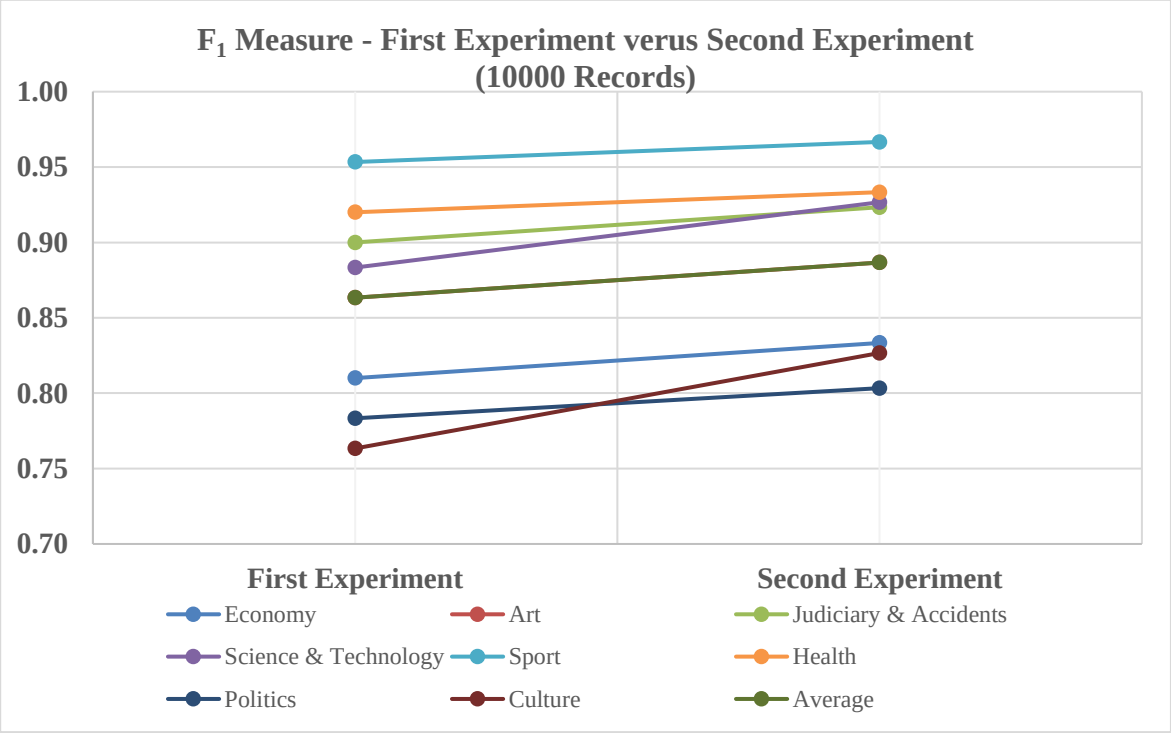

Fig.3: Comparison of F_1 Measure Results between First and Second Experiment for the dataset with size 100000

Table 8: F_1 Measure Results of First Experiment versus Second Experiment for the Dataset of Size 10000

Category	First Experiment	Second Experiment
Economy	0.83	0.89
Art	0.87	0.91
Judiciary & Accidents	0.91	0.96
Science & Technology	0.90	0.95
Sport	0.97	0.98
Health	0.93	0.95
Politics	0.81	0.88
Culture	0.82	0.86
Average	**0.88**	**0.92**

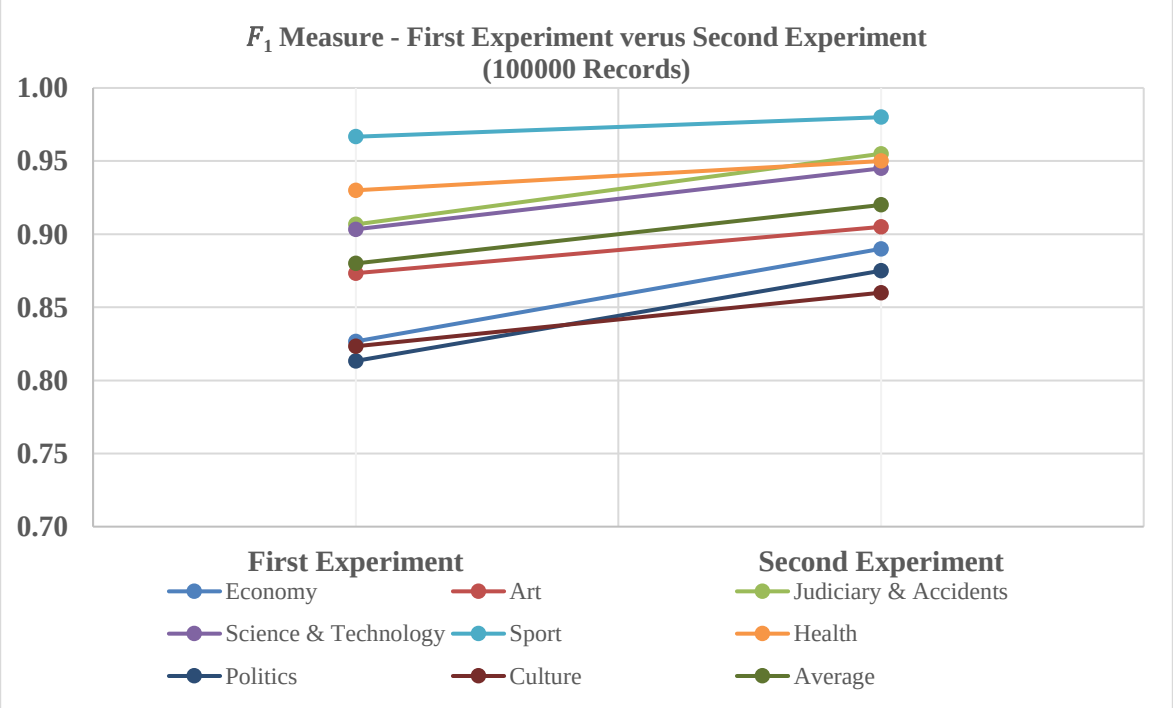

Fig.4: Comparison of F_1 Measure Results between First and Second Experiment for the dataset with size 100000

6. CONCLUSION

Arabic Text classification is an active research nowadays. However there is a lack of voluminous datasets for training an Arabic text classifier. For this purpose, we have created our own dataset. It is available in two versions. The first one is imbalanced and can be used in several NLP research areas such as generating Arabic word-embedding. The second version of this dataset is balanced and it is suitable for using deep learning technics in contrast to the small Arabic corpus currently available.

The results of the experiments conducted reflects the impact of dataset size on training deep learning algorithm as using more training data improves the performance of the classification model. In addition, using word2vec to obtain vector representations for words also enhanced the classification performance.

REFERENCES

- Elhassan, R., & Ahmed, M. (2015). Arabic Text Classification Review *International Journal of Computer Science and Software Engineering (IJCSSE).*
- Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.
- Brown, L. (2015). Deep learning with GPUs. *Larry Brown Ph. D., Johns Hopkins University*.
- Alsaleem, S. (2011). Automated Arabic Text Categorization Using SVM and NB. *Int. Arab J. e-Technol.*, *2*(2), 124-128.
- Alahmadi, A., Joorabchi, A., & Mahdi, A. E. (2014, October). Arabic Text Classification using Bag-of-Concepts Representation. In *KDIR* (pp. 374-380).
- Lai, S., Xu, L., Liu, K., & Zhao, J. (2015, February). Recurrent convolutional neural networks for text classification. In *Twenty-ninth AAAI conference on artificial intelligence*.
- Kato, R., & Goto, H. (2016, March). Categorization of web news documents using word2vec and deep learning. In *Proceedings of the 2016 International Conference on Industrial Engineering and Operations Management Kuala Lumpur, Malaysia*.
- Kilimci, Z. H., & Akyokuş, S. (2019, July). The Analysis of Text Categorization Represented With Word Embeddings Using Homogeneous Classifiers. In *2019 IEEE International Symposium on INnovations in Intelligent SysTems and Applications (INISTA)* (pp. 1-6). IEEE.
- Abbas, M., Smaïli, K., & Berkani, D. (2011). Evaluation of topic identification methods on Arabic corpora. *JDIM*, *9*(5), 185-192.
- [Abbas, M., & Smaili, K. (2005, September). Comparison of topic identification methods for arabic language. In *Proceedings of International Conference on Recent Advances in Natural Language Processing, RANLP* (pp. 14-17).
- Home. (n.d.). Retrieved from https://sourceforge.net/projects/ar-text-mining/files/Arabic-Corpora/
- Abuaiadah, D., El Sana, J., & Abusalah, W. (2014). On the impact of dataset characteristics on arabic document classification. *International Journal of Computer Applications*, *101*(7).
- Boukil, S., Biniz, M., El Adnani, F., Cherrat, L., & El Moutaouakkil, A. E. (2018). Arabic text classification using deep learning technics. *International Journal of Grid and Distributed Computing*, *11*(9), 103-114.
- Al-Salemi, B., Ayob, M., Kendall, G., Mohd , N., Shahrul, A. (2018). RTAnews: A Benchmark for Multi-label Arabic Text Categorization. *Mendeley Data, v1*
- Einea, O., Elnagar, A., & Al Debsi, R. (2019). SANAD: Single-label Arabic News Articles Dataset for automatic text categorization. *Data in brief*, *25*, 104076.
- DataSet for Arabic Classification. (2018, July 30). Retrieved from www.data.mendeley.com
- Soliman, A. B., Eissa, K., & El-Beltagy, S. R. (2017). Aravec: A set of arabic word embedding models for use in arabic nlp. *Procedia Computer Science*, *117*, 256-265.