Al-Mustansiriyah Journal of Science
ISSN: 1814-635X (print), ISSN:2521-3520 (online) Volume 32, Issue 4, 2021 DOI: http://doi.org/10.23851/mjs.v32i4.994

Research Article **Open Access**

Evaluation of Naïve Bayes Classification in Arabic Short Text Classification

Mohammed F. Ibrahim*, Mahdi A. Ali Alhakeem, Nawar A. Fadhil

Middle Technical University (MTU), Baghdad, IRAQ.

*Correspondent contact: mfi@mtu.edu.iq

Article Info

Received
17/05/2021

Accepted
01/08/2021

Published
20/11/2021

ABSTRACT

In the last few years, and due to the vast widespread of social media applications, texts have become more important and get more attention. Since texts, in general, are carrying a lot of information, that can be extracted and analyzed. Several researchers have done significant works in text classification. Within different scripts such as English and other western languages, several challenges and obstacles have been addressed with such a field of research. Regarding the Arabic language, the process is different from other vital languages since Arabic is considered an orthographic language that depends on the word shape. It is not easy to apply the standard text preprocessing techniques since it affects the word meaning. This paper evaluates Arabic short text classification using three standard Naïve Bayes classifiers. In our method, we classify the thesis and dissertations using their titles to perform the classification process. The collected dataset is collected from different repositories by using standard scrapping techniques. Our method classifies the document based on their titles to be placed in the desired specialization. Several preprocessing techniques have been applied, such as (punctuation removal, stop words removal, and space vectorization). For feature extraction, we adopt the TF-IDF method. We implemented three types of Naïve Bayes classifiers, which are (Multinomial Naïve Bayes, Complemented Naïve Bayes, and Gaussian Naïve Bayes). The study results showed that Complemented Naïve Bayes Classifier proposed the best performance with (0.84) of accuracy for the testing phase. The results of the study are promising to be applied with different short text classifications.

KEYWORDS: Arabic Text Classification; Multinomial Naïve Bayes; Complemented Naïve Bayes; Gaussian Naïve Bayes; TF-IDF.

INTRODUCTION

Text classification and categorization have become one of the required fields that are increasingly employed in different aspects, especially when social media applications become daily and extensively used [1]. With such vast data of text, it becomes more important to develop models and techniques to classify the text data and extract the valuable parts. Text classification and clustering recently become the most recent trends that deal with text data due to the variety of their applications such as (e-commerce [2-4], social media [5-7], biometrics [8, 9], healthcare [10, 11]), such field of research have been imposed in the last decade. Recent advancements in machine learning have paved the way for effective automated text classification methods to be proposed. Thus, in our research, we rely on text classification techniques to classify the Arabic documents (thesis and dissertations) depending on the document title.

The main challenge of title-based document classification is that it deals with short texts, which means minor features and information retrieved, which might affect the classification results[12, 13]. In addition, unlike long texts, document titles have a compact representation of the feature due to the academic requirements, which sometimes confuses to identify the document category precisely and raises an overlapping problem.

The context of text classification has been studied and implemented with the most vital languages. Despite that, the Arabic language is considered one of the six official languages endorsed by the United Nations [14]; however, few studies reported the Arabic language in text classification [1, 14, 15]. According to Wikipedia, Arabic is the official language of 25 countries with more than 310

million speakers [16]. Additionally, the middle east and Arab world are getting more attention because of the vital role that Arab play in terms of the world economy [17]. So the Arabic language is becoming more and more essential and receives extensive attention. According to the Internet World Stats [18], the Arabic language ranked as the fourth most language used over the internet with about 185 million internet users. With all these statistics of the Arabic language, it does not witness a significant expansion in terms of Arabic text classification and analysis [14, 19].

Document classification, in general, has been extensively researched in the fields of machine learning, data mining and used in a variety of industrial settings. To construct a collection of training data for a machine learning-based document classification method; documents must be labeled with predefined classes. This training data is then used to create a model that can be used to assign new documents to one or more of the predefined classes.

Several classification algorithms have been utilized to classify the documents, such as Support Vector Machines (SVM) [20] and k-Nearest-Neighbor (KNN)[21]. These algorithms have been used to solve a variety of industrial problems with remarkable effect. In business, machine learning applications often involve the processing of extensive datasets. The use of machine translation on English sentiment methods has been attempted in many studies. This bilingual approach, on the other hand, fails with Arabic due to the linguistic features of Arabic, which, in terms of form and grammar, are fundamentally different from English [22].

In this research, we present a title-based document classification approach to cap this gap, which aims to classify the Arabic documents based on their title. The general specialization of the document depicts the document's classes. Since there is no standard dataset that is directly connected to our study, we rely on our dataset collected using scrapping technique from some Arabic libraries portals. In addition, some languages, such as Persian and Urdu, share a significant portion of the character set. The method we work with can process such similar languages with similar character sets.

The rest of this study is organized as follows: section 2 is dedicated to view the related works, section 3 is devoted to view the dataset collection and preparation steps, while section 4 is dedicated to display the experiment running and classification process of four classification methods, and lastly, section 5 discusses the conclusions and future trends for the paper.

Related Work

Many studies have discussed the issue of automatic text categorization and classification, suggesting various strategies and solutions. This is particularly true in the case of the English language. A paper presented in [23] focuses on a systematic analysis of recent developments in natural language processing concentrate on deep learning. The use of deep learning (DL) for Arabic natural language processing is explained in detail in [24]. The study presented a survey on Arabic studies that deal with Arabic script classification using DL techniques. Even though DL has a significant impact on machine learning, Arabic-related research has a noticeable lack of employing DL for such vital language.

In [25], the authors presented a comparative study for Arabic text categorization. They adopted a dataset for Arabic articles containing 2700 Arabic articles with many classes represented by their type. Five standard text classification methods have been employed in this study. Also, the texts have been preprocessed using stemming and cleaning techniques. The results of the paper yielded that the SVM classifier outperforms the other classifiers.

In [26], a survey for Arabic Sentiment Analysis (ASA) has been presented. The survey focused on delivering an overview of the articles that adopt Arabic text in terms of classification. Obstacles of the Arabic text classification are also addressed in the investigated papers. The study examined many articles that deal with different Arabic text aspects, such as emotions, multilingual SA, and research that adopt dialect analysis.

Research presented in [28] used many classic learning supervised classifiers, including Decision Tree, KNN, SVM, and Naive Bayes. The study concentrated on how pre-processing affects text categorization outcomes. The study method has been implemented using the commonly used Arabic news datasets from the BBC and CNN. The results showed that the preprocessing of the text has a significant impact on the classification performance.

In [27, 28], The authors looked into the effects of stemming, light stemming, and synonyms-clustering on feature space reduction and classification accuracy. In [29], instead of relying on pre-processing and word-counting representations, the study used word and document embedding to identify Arabic documents. Using Doc2Vec to learn and integrate word vectors, the study showed that document embedding outperformed text pre-processing techniques. All previous references addressed the single-label Arabic text categorization issue. However, only a few studies have looked into the problem of categorizing multi-label Arabic texts. In the case of the English language, however, this is not the case. For instance, in [30], a supervised Hebb rule-based feature selection was established for English multi-label text classification. In [31], the effect of pre-processing on text classification was investigated.
The use of six transformation-based approaches for solving the Arabic multi-label text classification problem was explored in Arabic by [32]. On a 10K dataset collected from the BBC Arabic news portal, the label Combination with SVM as a base learner provided the highest output accuracy, according to the findings. In [33], NB, SVM, and KNN classifiers and three feature selection metrics were used to investigate transformation-based approaches using the BBC dataset (Chi-square, mutual knowledge, and odds ratio). [34], On the other hand, the transformation methods were paired with three single-label learning algorithms, namely: (KNN, Random Forest, and Decision trees). Decision trees outperformed both KNN and Random Forest in experiments on a similar dataset collected from the CNN news portal.
A study in [35] suggested an Arabic multi-label text classifier based on lexicons and transformation-based approaches. For Arabic text categorization, [36] developed a multi-label boosting algorithm that is more efficient. Later, they looked at rating functionality to improve their multi-label text categorization boosting algorithm [37]. [38] presented a study to classify books based on the cover image and the title features. The method combines both title and cover features using a logistic regression model to predict the book genre. The study results stated that the model produces better performance when connecting the features than when separating the features.
In [39-41], several studies have been presented, manipulating the text classification in Arabic scripts. These studies involved long text data gained either from Arabic news or from Arabic documents. Different methods were applied, all the results present a plausible performance rate exceeds 84%. Arabic titles have not been explored before in the literature.
From all of the studies above, it can be seen that there are no well-known official datasets for Arabic text classification. In addition to that, there are fewer efforts employed to investigate such a strong language in more details and aspects. Thus, we adopt our dataset for document titles collected using scrap techniques from different online Arabic repositories. For the comparison reasons, we couldn't find any comparative research that deals with document classification, particularly for Arabic scripts. Therefore, our main contribution is to classify Arabic documents based on their titles. Four classification algorithms have been tested to evaluate the accuracy performance. There will be a detailed overview of the method used and the implementation process in the upcoming sections.

Dataset Collection and Preparation

This section is devoted to prepare the dataset that we used in implementing our method. The dataset has been collected by using scraping techniques through python from some online repositories. The dataset is represented by the title of the thesis or dissertation and the general specialization of the document (thesis/ dissertation). The collected dataset was composed of (7500) titles, where the dataset has three attributes (Title, Specialization, Specialization _Code). The Specialization attribute represents the general specialization of the theses/dissertation title. Specialization _Code means the Class_ID, where its data is ranging from 1 to 10. In our dataset, we have (10) classes represented by specialization as described in (Table1) and Figure 1. The reason for selecting the mentioned classes is that they have a higher number of titles than other specializations in the collected dataset. As illustrated in Table 1, the number of labels per class varies among different disciplines, which presents some challenges regarding the classification performance.
The number of characters for each title varies significantly so that in the data set. There are some titles with short text that reach up to (44)

characters. Where almost (25%) of the dataset titles do not exceed (67) characters as stated in Table 2, also, 50% of the dataset at (81) characters, and 75% within the range of (105) characters of length. This issue makes the classification process even more difficult since there are few characters for words, considering the preprocessing steps that may reduce this number of words per title. Eventually, such a variety of the titles' sizes poses some challenges that may affect accuracy.

Table 1. Dataset Description.

Class	General Specialization (Arabic)	General Specialization (English)	# Titles	%
1	الجغرافية	Geography	730	10%
2	الدين	Religion	867	12%
3	الإدارة	Management	905	12%
4	اللغة	Linguistics	507	7%
5	الطب	Medicine	948	13%
6	علوم الحاسبات	Computer Science	439	6%
7	القانون	Legislation and Law	1102	15%
8	التاريخ	History	876	12%
9	الاحياء المجهرية	Microbiology	719	10%
10	الرياضيات	Mathematics	407	5%
Total			**7500**	**100%**

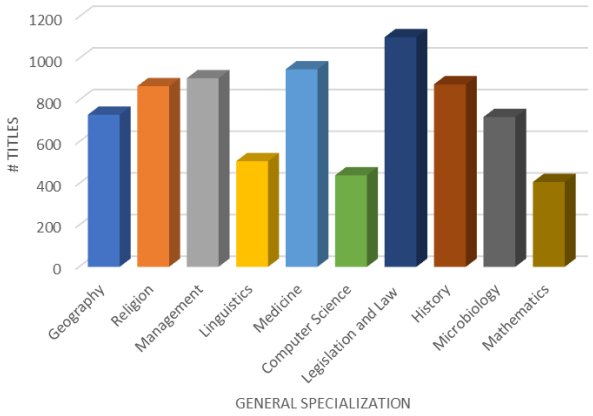

Figure 1. The Dataset Titles Overview.

Table 2. Dataset Overall Characters Description

Title Count	7500
Mean	88
Min	44
25%	67
50%	81
75%	105
Max	251

Data Preprocessing and Feature Selection

In this section, an overview of the text preprocessing is presented. As mentioned previously, our method involves the classification of Arabic theses/dissertations based on their titles. Text pre-processing is a necessary step in cleaning the dataset and, ideally, improving the outcome. The critical problem with the approach we suggest is that the document titles are treated as short texts. As a consequence, there are only a few features extracted from each title. Arabic's orthographic essence varies from that of English and other languages. The Arabic alphabet consists of 28 characters. The shape of the letters varies depending on where they are in the word: beginning, middle, and end; linked to the previous letter or not; and writing is done from right to left. For example, the letter "m" in Arabic is "م" so it can be written in different shapes according to letter's position in a particular word so that (at the beginning of the word: "مـ", middle of the word "ـمـ", end of the word "ـم").

In addition to orthographic nature, the Arabic language also engaged with diacritics, which are some signals placed below or above a particular letter with a single word. Diacritics control the articulation of the letters, as well as the meaning [42]. Such signals also form a challenge when dealing with the preprocessing techniques, where such procedures may eliminate the diacritics signals and eventually change the word meaning [43].

Even though machine learning does not require extensive linguistic skills, Arabic scripts come with their own set of challenges. Since Arabic does not support letter capitalization or has strict punctuation rules, the easy tokenization phase (widely used in machine learning) is difficult for Arabic. This process is simple in English: The sentence has a capital letter at the start and a period at the end. Since Arabic is a morphologically rich language, tokenization is complicated; one Arabic word can contain four tokens [1, 28]. So the preprocessing steps we adopt in our experiments as stated in the upcoming paragraphs. We conducted all the preprocessing and the experiments using Python environment.

A. Removing Punctuations

Since punctuation is not essential in terms of text classification, we perform this step to clean the document title and eliminate the unneeded characters, which may affect the accuracy performance. The punctuation characters are; '〜{|}´_^[//]⑨؟<=>¿:\.-،+*()'/&٪؋#"!'. In addition, we also eliminate the numbers from the documents' titles since they have no significant impact on the document title.

B. Stop Words Removing

Before indexing documents, stop-word elimination is used to exclude un-meaningful terms by removing words with short lengths and specific sequences of characters such as "the," "a," and "of" in the English language. These words are commonly used in the script, which increases the text's size. Excluding them has the advantage of reducing the document's size without sacrificing the valid details required to determine the document's topic. Rare terms, identified as words that appear in a small percentage of the processed documents, are removed [44, 45]. Stop words in Arabic include the word shown in Table 3.

.Table 3. Arabic Stop Words

هذان	لئن	كيفما	عند	خلا	بكما	إنا	إذ
هذه	ليت	لا	غير	دون	بكن	أنا	إذا
هذي	ليس	لاسيما	فإذا	ذا	بل	أنت	إذما
هذين	ليسا	لدى	فإن	ذات	بلى	أنتم	إذن
هكذا	ليست	لست	فلا	ذاك	بما	أنتما	أف
هل	ليستا	لستم	فمن	ذان	بماذا	أنتن	أقل
هلا	ليسوا	لستما	في	ذانك	بمن	إنما	أكثر
هم	ما	لستن	فيم	ذلك	بنا	إنه	ألا
هما	ماذا	لسن	فيما	ذلكم	به	أنى	إلا
هن	متى	لسنا	فيه	ذلكما	بها	أني	التي
هنا	مذ	لعل	فيها	ذلكن	بهم	آه	الذي
هناك	مع	لك	قد	ذه	بهما	آها	الذين
هنالك	مما	لكم	كأن	ذو	بهن	أو	اللاتي
هو	ممن	لكما	كأنما	ذوا	بي	أولاء	اللائي
هؤلاء	من	لكن	كأي	ذواتا	بين	أولئك	اللتان
هي	منه	لكنما	كأين	ذواتي	بيد	أوه	اللتيا
هيا	منها	لكي	كذا	ذي	تلك	آي	اللتين
هيت	منذ	لكيلا	كذلك	ذين	تلكم	أي	اللذان
هيهات	مه	لم	كل	ذينك	تلكما	أيها	اللذين
والذي	مهما	لما	كلا	ريث	ته	إي	اللواتي
والذين	نحن	لن	كلاهما	سوف	تي	أين	إلى
وإذ	نحو	لنا	كلتا	سوى	تين	أين	إليك
وإذا	نعم	له	كلما	شتان	تينك	أينما	إليكم
وإن	ها	لها	كليكما	عدا	ثم	إيه	إليكما
ولا	هاتان	لهم	كليهما	عسى	ثمة	بخ	إليكن
ولكن	هاته	لهما	كم	عل	حاشا	بس	أم
ولو	هاتي	لهن	كم	على	حبذا	بعد	أما
وما	هاتين	لو	كما	عليك	حتى	بعض	أما
ومن	هاك	لولا	كي	عليه	حيث	بك	إما
وهو	اهنا	لوما	كيت	عما	حيثما	بكم	أن
يا	هذا	لي	كيف	عن	حين	بكم	إن

C. Text Feature selection Using TF-IDF

The term frequency-inverse document frequency (TF-IDF) is widely used in information retrieval and text mining to measure the relationship between each term in a series of documents. This method is used for purposes like extracting core terms (keywords) from a text, determining search ranking by comparing degrees of similarity between documents, and so on [46]. In TF-IDF, the TF (Term Frequency) refers to the frequency with which specific words appear in a document. Words with a high TF value are essential. The DF, on the other hand, indicates how many times a particular word appears in a series of documents. It counts the number of times the word appears in several texts, not just one. Since they appear regularly in all texts, words with a high DF meaning that they have little importance. Consequently, the IDF (which is the inverse of the DF) is used to evaluate the significance of terms in all documents. High IDF values in all papers indicate uncommon words, suggesting a rise in relevance [47, 48].

Each document described by a document vector in the vector space model has term weight. (tf1, tf2, tf3, ..., tfn) represent the text vector with term frequency as the term weight. Where tf denotes the term frequency and n is the number of words in the text. The following formula is used to calculate the TF value:

$$TF_i = \frac{tf_i}{((d|\ \ ))} \quad (1)$$

Where TFi is the frequency of a particular word (i) occurrence in a specific text (in our model document title), $(d|\ \ )$: is the total number of words within a specific title

$$IDF(t_i) = log\frac{N}{n_i} \quad (2)$$

N is the total number of titles in the dataset and the number of documents containing the term ti, ni called the document frequency.

$$\text{TF-IDF} = \text{TF} * \text{IDF} \quad (3)$$

By applying the formulas (1), (2), (3), the dataset is ready to be implemented with machine learning classifiers. In this study, we perform four different classifiers, which are (Multinomial Naïve Bayes (MNB), Complemented Naïve Bayes (CNB), and Gaussian Naïve Bayes (GNB). Figure 1 illustrates an overall description of the research process.

Experiment Running and Classification Process

As mentioned before, the dataset size is (7500) titles, representing the Arabic titles for theses and dissertations. We divided our dataset into (0.80, 0.20) for the training and testing, respectively. We rely on the performance accuracy in terms of classifiers comparison for the testing phase. Table 4 describes the experiment setting and the parameters involved in the experiment. As stated in Table (4), we eliminate performing stemming on the processed texts since it might change the word meaning.

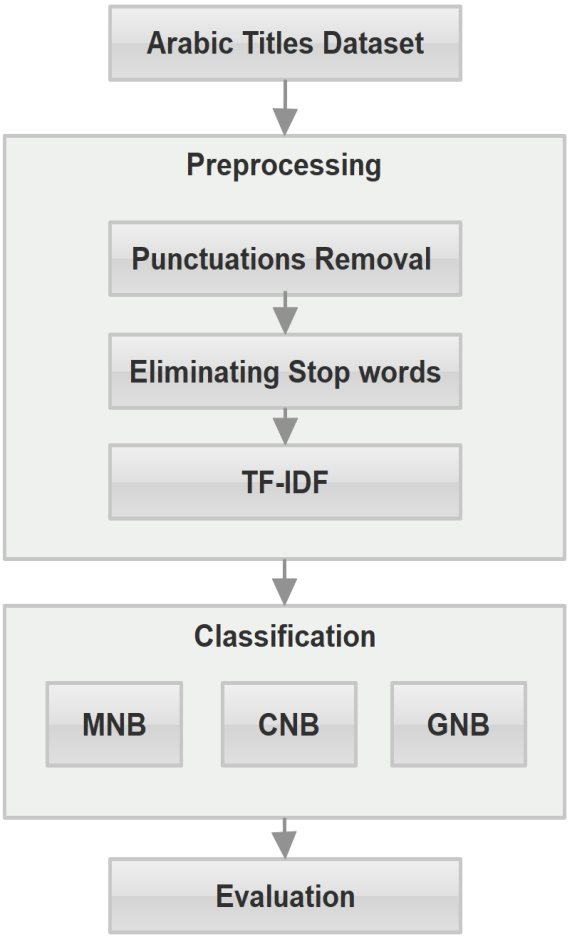

Figure 2. Document Title Classification Diagram.

We mentioned earlier that the Arabic script is based on the orthographic essence so that that stemming can remove the diacritics out of the word, which changes the word meaning. For example, assume the Arabic word (علم); such term has several meanings when adding the diacritics. So that the word (عَلَمٌ) means ("flag"), (عَلِمَ) means ("knew"), (عِلْمٌ) means ("science"), (عَلَّمَ) means ("taught"). Hence, working with such language and applying stemming can grammatically influence the meaning.

Table 4. Experiment's Parameters and Setting

Parameter	Setting
Text Field involved	Title of Theses/Dissertation
Classes	Specialization ID of Theses/Dissertation
Stop words Removing	Yes
Punctuation Removing	Yes
Space Vectorization	Yes
Stemming	No
Feature Selection Method	TF-IDF
Training Size	80%
Testing Size	20%

D. Multinomial Naïve Bayes Classifier (MNB)

MNB has been implemented to be a part of the study methods. The performance accuracy of the test phase using MNB comes at (0.81) as stated in Table 5. Several classes have been precisely classified, but the Linguistics class produced the lowest performance in testing classification with a rate of (0.39).

Table 5. Naïve Bayes Classification Report

Class	F1-Score (MNB)	F1-Score (CNB)	F1-Score (GNB)
Geography	0.90	0.93	0.88
Religion	0.80	0.76	0.66
Management	0.97	0.94	0.87
Linguistics	0.39	0.75	0.73
Medicine	0.77	0.72	0.61
Computer Science	0.65	0.83	0.73
Legislation and Law	0.98	0.96	0.88
History	0.93	0.88	0.81
Microbiology	0.63	0.69	0.59
Mathematics	0.68	0.85	0.80
Accuracy	**0.81**	**0.84**	**0.76**

From the table above, it can be seen that the performance of the Complemented Naïve Bayes (CNB) outperforms the other classifiers with total accuracy of (0.84). At the same time, Gaussian Naïve Bayes (GNB) presents the lowest accuracy rate with (0.76) for the testing phase, Figure 3.

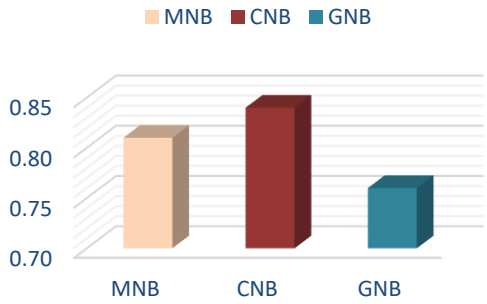

Figure 3. The Accuracy Rate for All of the three Classifiers.

Regarding the data set classes, the classes of (Geography and Legislation) as shown in Figure 4 have presented a stable and convenient performance across the classifiers, which means these classes can be easily classified with different classifiers. It is noteworthy; all other classes have presented a low variety of performance across the classifiers except the (linguistic), which perform very badly with the MNB classifier.

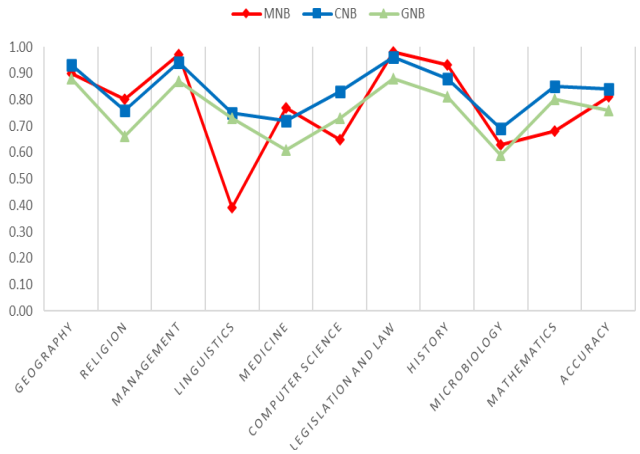

Figure 4. Class-Based Classification Performance.

CONCLUSIONS

From the result of our study, test classification of the Arabic language has some challenges, especially regarding the nature of the orthographic essence that the Arabic language relies on. However, text classification can be noticeably performed with a complex script like the Arabic language with a reasonable performance rate, which is dramatically different from other languages like English, French, and other western languages. According to the results gained, there is a plausible application of such technique in document classification basing on their titles. CNB has higher performance accuracy if compared to other classifiers. CNB can be implemented with other languages close to Arabic in nature, such as Parisian and Urdu. This thing opens the gate toward more practical applications in short text classification and investigates more classification methods to perform the comparison phase.

REFERENCES

[1] A. Elnagar, R. Al-Debsi, and O. Einea, "Arabic text classification using deep learning models," *Information Processing & Management*, vol. 57, no. 1, p. 102121, 2020.

[2] H.-F. Yu, C.-H. Ho, P. Arunachalam, M. Somaiya, and C.-J. Lin, "Product title classification versus text classification," *Csie. Ntu. Edu. Tw*, pp. 1-25, 2012.

[3] Y.-C. Lin, A. Datta, and G. Di Fabbrizio, "E-commerce product query classification using implicit user's feedback from clicks," in *2018 IEEE International Conference on Big Data (Big Data)*, 2018: IEEE, pp. 1955-1959.

[4] M. Skinner and S. Kallumadi, "E-commerce Query Classification Using Product Taxonomy Mapping: A Transfer Learning Approach," in *eCOM@ SIGIR*, 2019.

[5] N. Bel, J. Diz-Pico, M. Marimon, and J. Pocostales, "Classifying short texts for a Social Media monitoring system," *Procesamiento del Lenguaje Natural*, no. 59, pp. 57-64, 2017.

[6] J. Al Qundus, A. Paschke, S. Gupta, A. M. Alzouby, and M. Yousef, "Exploring the impact of short-text complexity and structure on its quality in social media," *Journal of Enterprise Information Management*, 2020.

[7] Z. Alzamil, D. Appelbaum, and R. Nehmer, "An ontological artifact for classifying social media: Text mining analysis for financial data," *International Journal of Accounting Information Systems*, vol. 38, p. 100469, 2020.

[8] S. Ma, X. Sun, J. Lin, and X. Ren, "A hierarchical end-to-end model for jointly improving text summarization and sentiment classification," *arXiv preprint arXiv:1805.01089*, 2018.

[9] A. Abdi, S. M. Shamsuddin, S. Hasan, and J. Piran, "Deep learning-based sentiment classification of evaluative text based on Multi-feature fusion," *Information Processing & Management*, vol. 56, no. 4, pp. 1245-1259, 2019.

[10] T. Baumel, J. Nassour-Kassis, R. Cohen, M. Elhadad, and N. Elhadad, "Multi-label classification of patient notes a case study on ICD code assignment," *arXiv preprint arXiv:1709.09587*, 2017.

[11] A. Blanco, A. Casillas, A. Pérez, and A. D. de Ilarraza, "Multi-label clinical document classification: Impact of label-density," *Expert Systems with Applications*, vol. 138, p. 112835, 2019.

[12] K. Tayal, R. Nikhil, S. Agarwal, and K. Subbian, "Short text classification using graph convolutional network," in *NIPS workshop on Graph Representation Learning*, 2019.

[13] K. Kowsari, K. Jafari Meimandi, M. Heidarysafa, S. Mendu, L. Barnes, and D. Brown, "Text classification algorithms: A survey," *Information*, vol. 10, no. 4, p. 150, 2019.

[14] A. Ghallab, A. Mohsen, and Y. Ali, "Arabic Sentiment Analysis: A Systematic Literature Review," *Applied Computational Intelligence and Soft Computing*, vol. 2020, p. 7403128, 2020/01/29 2020, doi: 10.1155/2020/7403128.

[15] N. Al-Twairesh, H. Al-Khalifa, and A. Al-Salman, "Subjectivity and sentiment analysis of Arabic: trends and challenges," in *2014 IEEE/ACS 11th International Conference on Computer Systems and Applications (AICCSA)*, 2014: IEEE, pp. 148-155.

[16] Wikipedia. "Arabic." Wikimedia Foundation. https://en.wikipedia.org/wiki/Arabic (accessed April 02, 2021).

[17] S. Clerides, P. Davis, and A. Michis, "National sentiment and consumer choice: The Iraq war and sales of US products in Arab countries," *The Scandinavian Journal of Economics*, vol. 117, no. 3, pp. 829-851, 2015.

[18] I. W. Stats. "Top Ten Internet Languages in The World - Internet Statistics." https://www.internetworldstats.com/stats7.htm (accessed April 02, 2021.

[19] W. Zaghouani, "Critical survey of the freely available Arabic corpora," *arXiv preprint arXiv:1702.07835*, 2017.

[20] T. Pranckevičius and V. Marcinkevičius, "Comparison of naive bayes, random forest, decision tree, support vector machines, and logistic regression classifiers for text reviews classification," *Baltic Journal of Modern Computing*, vol. 5, no. 2, p. 221, 2017.

[21] B. Trstenjak, S. Mikac, and D. Donko, "KNN with TF-IDF based framework for text categorization," *Procedia Engineering*, vol. 69, pp. 1356-1364, 2014.

[22] T. Al-Moslmi, M. Albared, A. Al-Shabi, N. Omar, and S. Abdullah, "Arabic senti-lexicon: Constructing publicly available language resources for Arabic sentiment analysis," *Journal of information science*, vol. 44, no. 3, pp. 345-362, 2018.

[23] T. Young, D. Hazarika, S. Poria, and E. Cambria, "Recent trends in deep learning based natural language processing," *ieee Computational intelligenCe magazine*, vol. 13, no. 3, pp. 55-75, 2018.

[24] M. Al-Ayyoub, A. Nuseir, K. Alsmearat, Y. Jararweh, and B. Gupta, "Deep learning for Arabic NLP: A survey," *Journal of computational science*, vol. 26, pp. 522-531, 2018.

[25] I. Hmeidi, M. Al-Ayyoub, N. A. Abdulla, A. A. Almodawar, R. Abooraig, and N. A. Mahyoub, "Automatic Arabic text categorization: A comprehensive comparative study," *Journal of Information Science*, vol. 41, no. 1, pp. 114-124, 2015.

[26] M. Al-Ayyoub, A. A. Khamaiseh, Y. Jararweh, and M. N. Al-Kabi, "A comprehensive survey of arabic sentiment analysis," *Information Processing & Management*, vol. 56, no. 2, pp. 320-342, 2019/03/01/ 2019, doi: https://doi.org/10.1016/j.ipm.2018.07.006.

[27] M. N. Al-Kabi, Q. A. Al-Radaideh, and K. W. Akkawi, "Benchmarking and assessing the performance of Arabic stemmers," *Journal of Information Science*, vol. 37, no. 2, pp. 111-119, 2011.

[28] R. Duwairi and M. El-Orfali, "A study of the effects of preprocessing strategies on sentiment analysis for Arabic text," *Journal of Information Science*, vol. 40, no. 4, pp. 501-513, 2014.

[29] A. El Mahdaouy, E. Gaussier, and S. O. El Alaoui, "Arabic text classification based on word and document embeddings," in *International Conference on Advanced Intelligent Systems and Informatics*, 2016: Springer, pp. 32-41.

[30] H. Wang and M. Hong, "Supervised Hebb rule based feature selection for text classification," *Information Processing & Management*, vol. 56, no. 1, pp. 167-191, 2019.

[31] A. K. Uysal and S. Gunal, "The impact of preprocessing on text classification," *Information Processing & Management*, vol. 50, no. 1, pp. 104-112, 2014.

[32] N. A. Ahmed, M. A. Shehab, M. Al-Ayyoub, and I. Hmeidi, "Scalable multi-label arabic text classification," in *2015 6th International Conference on Information and Communication Systems (ICICS)*, 2015: IEEE, pp. 212-217.

[33] A. Y. Taha and S. Tiun, "BINARY RELEVANCE (BR) METHOD CLASSIFIER OF MULTI-LABEL CLASSIFICATION FOR ARABIC TEXT," *Journal of Theoretical & Applied Information Technology*, vol. 84, no. 3, 2016.

[34] M. A. Shehab, O. Badarneh, M. Al-Ayyoub, and Y. Jararweh, "A supervised approach for multi-label classification of Arabic news articles," in *2016 7th International Conference on Computer Science and Information Technology (CSIT)*, 2016: IEEE, pp. 1-6.

[35] I. Hmeidi, M. Al-Ayyoub, N. A. Mahyoub, and M. A. Shehab, "A lexicon based approach for classifying Arabic multi-labeled text," *International Journal of Web Information Systems*, 2016.

[36] B. Al-Salemi, S. A. M. Noah, and M. J. Ab Aziz, "RFBoost: an improved multi-label boosting algorithm and its application to text categorisation," *Knowledge-Based Systems*, vol. 103, pp. 104-117, 2016.

[37] B. Al-Salemi, M. Ayob, and S. A. M. Noah, "Feature ranking for enhancing boosting-based multi-label text categorization," *Expert Systems with Applications*, vol. 113, pp. 531-543, 2018.

[38] G. R. Biradar, J. Raagini, A. Varier, and M. Sudhir, "Classification of Book Genres using Book Cover and Title," in *2019 IEEE International Conference on Intelligent Systems and Green Technology (ICISGT)*, 2019: IEEE, pp. 72-723.

[39] S. Bahassine, A. Madani, M. Al-Sarem, and M. Kissi, "Feature selection using an improved Chi-square for Arabic text classification," *Journal of King Saud*

University-Computer and Information Sciences, vol. 32, no. 2, pp. 225-231, 2020.

[40] H. Chantar, M. Mafarja, H. Alsawalqah, A. A. Heidari, I. Aljarah, and H. Faris, "Feature selection using binary grey wolf optimizer with elite-based crossover for Arabic text classification," *Neural Computing and Applications*, vol. 32, no. 16, pp. 12201-12220, 2020.

[41] D. AbuZeina and F. S. Al-Anzi, "Employing fisher discriminant analysis for Arabic text classification," *Computers & Electrical Engineering*, vol. 66, pp. 474-486, 2018.

[42] I. A. Doush, F. Alkhateeb, and A. Albsoul, "AraDaisy: A system for automatic generation of Arabic DAISY books," *International Journal of Computer Applications in Technology*, vol. 55, no. 4, pp. 322-333, 2017.

[43] M. Sayed, R. K. Salem, and A. E. Khder, "A survey of Arabic text classification approaches," *International Journal of Computer Applications in Technology*, vol. 59, no. 3, pp. 236-251, 2019.

[44] A. K. Sangaiah, A. E. Fakhry, M. Abdel-Basset, and I. El-henawy, "Arabic text clustering using improved clustering algorithms with dimensionality reduction," *Cluster Computing*, vol. 22, no. 2, pp. 4535-4549, 2019.

[45] J. Ferrero, D. Schwab, and H. Cherroun, "Word embedding-based approaches for measuring semantic similarity of arabic-english sentences," in *International Conference on Arabic Language Processing*, 2017: Springer, pp. 19-33.

[46] S.-W. Kim and J.-M. Gil, "Research paper classification systems based on TF-IDF and LDA schemes," *Human-centric Computing and Information Sciences*, vol. 9, no. 1, p. 30, 2019/08/26 2019, doi: 10.1186/s13673-019-0192-7.

[47] L. Havrlant and V. Kreinovich, "A simple probabilistic explanation of term frequency-inverse document frequency (tf-idf) heuristic (and variations motivated by this explanation)," *International Journal of General Systems*, vol. 46, no. 1, pp. 27-36, 2017.

[48] B. Das and S. Chakraborty, "An improved text sentiment classification model using TF-IDF and next word negation," *arXiv preprint arXiv:1806.06407*, 2018.