Arabic Text Classification

1. Introduction

Finding useful knowledge on a given subject in a vast volume of online textual data that is rapidly growing is a difficult challenge. To solve this issue, organize data into predetermined categories could help. Algorithms of text classification are the basis of many applications for natural language processing, such as text description, query response, detection of spam, and visualization of text.
While Arabic language on the internet is rising increasingly, its content is still as poor as 3 percent. For researchers and developers, the recent rapid growth is a convincing incentive to develop successful frameworks and tools to advance study in Arabic NLP. The automated mapping of texts to predefine marks or classes is text categorization. Text categorization refers to the process of grouping text or documents into classes or categories according to their content. Text categorization process consists of three phases which are: preprocessing, feature extraction and classification. In comparison to the English language, just few studies have been done to categorize and classify the Arabic language. For a variety of applications, such as text classification and clustering, Arabic text representation is a difficult task because Arabic language is noted for its richness, diversity, and complicated morphology. The process of binary classification requires a collection of two classes where as a multi classification process operates on more than two types of data gathered for assigning them to an unseen text. Categorization of text may be manual or automated. Since the early days, manual text classification has been the central role of classifying library meaning. Automatic text classification is performed primarily by computing device using classification algorithms. Automatic document categorization gained more importance in view of the plethora of textual documents added constantly on the web. As a result of the rise of the Internet and Web 2.0, unimaginable amount of data is constantly on the rise, which is produced by several sources including social media users. The presence of such unstructured data makes a great resource for data processing and management in order to extract useful information. One important task is text classification and clustering, which is a field of research that gained more momentum in the last few years. The recent advances in Machine Learning paved the road for proposing successful automatic text categorization systems. The terms text categorization and text classification are used interchangeably to indicate the process of predicting predefined categories or domains to a given document. The automated categorization process may report the most relevant single category or multiple close ones. For the huge amount of available documents (or text) on the internet, manual classification by domain experts becomes ineffective and unfeasible. Therefore, automated classifiers had become not only an alternative but a necessity utilizing machine learning algorithms. However, the unstructured textual documents have to be represented in a format compatible with machine learning algorithms such as numeric vectors. Text categorization is well studied in several languages and in particular the English language. Despite of the importance of Arabic language being the fourth used language on the Internet and 6th official language reported by United Nations (Eldos (2003)), few research attempts are reported on the Arabic language text classification as detailed in the next section. According to Wikipedia, as of 2018, there are 25 independent nations where Arabic is an official language and the number of Arabic speakers reach 380 million. With the rise of Arabic data on the internet, the need for an effective and robust automated classification system becomes a must. The research attempts at addressing this problem for Arabic text are limited to using classical machine learning classifiers and were conducted on small and mostly unavailable datasets (i.e., not freely available for download). Methods of text classification are used in several applications, like e-mail search, filtering of spam and classification of news.

https://data.mendeley.com/datasets/57zpx667y9/2

2. Arabic Language Characteristics

Arabic language is spoken by more than 250 million. Letters of Arabic language consist of 28 letters plus hamza. Arabic language letters are written from right to left. One of the main characteristics of Arabic language is that its letters has different forms and shapes depending on the position of the letter.one of the excellent merits of Arabic language is that majority of Arabic word has a root. Besides that, majority of Arabic root words are consisting of three letters. Representing words with its root helps in reducing the number of words. Arabic features are assorted in abundance aspects compared to English language. Arabic is a global language that is commonly used and has considerable variations compared to the most common, such as Spanish, English and Chinese. There are several forms of grammatical, variations of synonyms word, and numerous meanings of word in the Arabic language, which differ based on factors such as order of the word. although such difficulties, the work on natural language processing (NLP) with Arabic has been minimal especially compared to the English language. Arabic language is the fifth most commonly spoken language in the world and the fifth most frequently used on the internet. More than 422 million speak Arabic language (by more than 6.0% of the global population. The requirements of Arabic language are not resolved by several tools, packages and APIs in information retrieval and natural language processing applications. To make these tools, software packages to handle Arabic language data, modifications and additional work are necessary. Arabic language written from right to left and it includes 28 different characters for the same letter, with varies formulations depending on position of the letter in word. In addition, there are diacritics for example small characters that may be added to a letter either as subscript or superscript to add distinct spelling, grammatical formulation, and these diacritics are widely found in formal Arabic, often indicating the letter as well as the whole word.

3. Methodology

Model The goal of text classification is to create a model that used to classify different text documents to its predefined classes. Figure 1 represents the classification model phases.

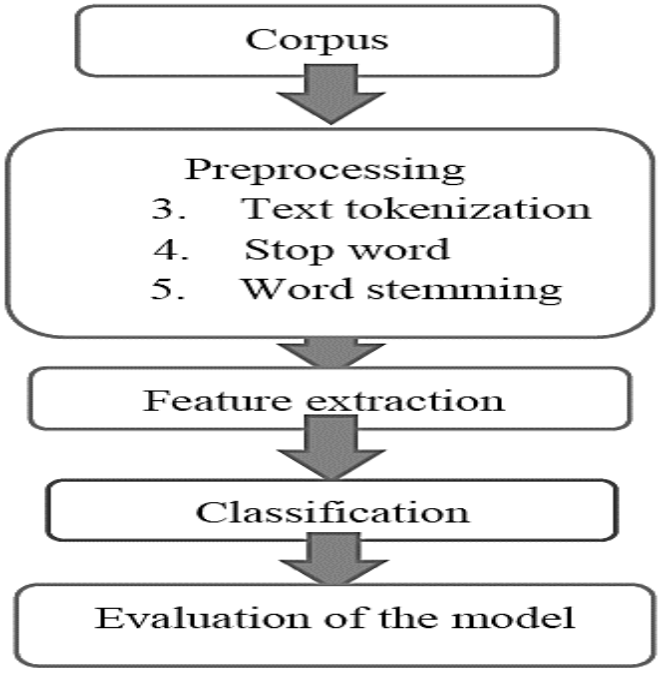

Figure.1.1 Classification General phases.

https://data.mendeley.com/datasets/57zpx667y9/2

3.1. Text Classification Datasets

There are several reference data sets for the processing of data that are publicly usable for English text classification. Unfortunately, an open access of standard dataset for the Arabic language are not aware to us. The Arabic Corpus Open-Source is freely available, but not organized. The bulk of researchers in Arabic text classification assembled their test corpora from the online dataset of Arabic news.

3.2. The Preprocessing

Some preprocessing is required to deal with text data to select features which are semantically represent the document and remove other features that are not. This process which extracts important features that represent training dataset is named Feature Extraction (FE). The primary goal of the preprocessing phase is to minimize the space of testing and to decrease the rate of error. Data preprocessing involves tokenization of text, removal of Stop-words, and term stemming. After preprocessing, the dataset is presented in a shape appropriate for the feature selection stage.

3.3. Feature Extraction

This stage includes taking stemming words and transform them into features to be used by the classifier. In the section below, we are giving a summarized introduction to text classification features used in this survey.

- Chi Square Is a common method of collection of features that can be independently evaluated with regard to categories by computing the statistics of chi square. This suggests that the chi squared value analyze relationship between word and category. If the word is distinct from the category, the score would then be equal to 0, else it is 1. If the word has a higher chi-square value, that means it is more informative .
- Information Gain Information gain method could be easier than the chi square. The fundamental principle is that for each feature that can represent discrimination between categories, we just have to determine the score, the features are then categorized according to this value and then only certain top-ranking ones are preserved.
- Term Frequency Inverse Document Frequency (TF-IDF) It works by measuring how many times the word (relative frequency) in a text compared to the inverse ratio of a word over the whole corpus. This measure, of course, decide how relevant a given word is in a specific text. It is expected that words exist in single or multiple documents would have higher TF-IDF numbers than prepositions which are the common words .
- Word Embedding It is a text representation that convert text into a numerical vector in vector space which represents both of syntactic and semantic characteristics of text. The word embedding models which recently provide enhanced results compared to bag of words which still used in some of natural language processing tasks. Bag of words model represents count of tokens in the text in which the location of the word is ignored in context of others. Word2vec and Glove are the most popular models for word embedding.

3.4. Machine Learning Algorithms

In standard machine learning applications, each text document in an annotated of dataset training is then converted into a numerical representation of vectors related to text categorization. Upon the text vectorization techniques, like the Term Frequency Inverse Document Frequency (TF-IDF) algorithm, the words and phrases inside the documents are known as variables or features and scores are allocated. The mostly used machine learning algorithms are: Naive Bayes Classifier , Support Vector Machines Classifier, Logistic Regression, Decision Trees, Rule Induction Classification, and K-Nearest Neighbor (KNN)

3.5. Deep Learning Algorithms

Deep learning refers to a vast number of machine learning approaches and frameworks that have the advantage of employing multiple levels of hierarchical nonlinear data processing. Based on the intended application of the architectures and techniques, such as synthesis/generation or identification/classification. Comparing deep learning and traditional machine methods it could be instructive to suggest parallels to a regression model in learning algorithms. Users should define a specific model when running a regression as logistic regression or linear for example. To optimize accuracy and supply data for an outcome (dependent) variable and related data for predictor (independent) variables, the regression algorithm will then fit parameters into the model. The regression algorithm would then fit the parameters to the model to optimize accuracy and supply data for the result.

Typically, neural networks consist of neurons operating together to form a layer. To form the network, multiple layers are then connected. The neural networks with hidden layers are Deep Neural Networks (DNN) that are deep and rich. The hidden layers are extra layers that are applied to the network to add additional processing, when the task is very difficult for a tiny network. The number of hidden layers will reach a one hundred or more. DNN are known to be creative and have excellent precision. There are several forms of DNN, many of them are alerted to function on image data, and many of them are texts sources of data. The mostly used deep learning algorithms are: Multilayer Perceptron Network (MLP), Convolution Neural Networks (CNN) , Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM) ,

Capsule Neural Networks, Gated Recurrent Unit (GRU), Bidirectional Long Short-Term Memory Networks (BiLSTM), CNN–BiLSTM Networks, and BiLSTM– CNN Networks.

SANAD: Single-label Arabic News Articles Dataset for automatic text categorization (Report)

1. Datasets

SANAD Dataset is a large collection of Arabic news articles that can be used in different Arabic NLP tasks such as Text Classification and Word Embedding. The articles were collected using Python scripts written specifically for three popular news websites: AlKhaleej, AlArabiya and Akhbarona. All datasets have seven categories [Culture, Finance, Medical, Politics, Religion, Sports and Tech], except AlArabiya which doesn't have [Religion]. SANAD contains a total number of 190k+ articles.

1.1.Distribution

SANAD corpus is a large collection of Arabic news articles that can be used in several NLP tasks such as text classification and producing word embedding models. AlKhaleej and Akhbarona-Alanba datasets have seven categories, which are: Culture, Finance, Medical, Politics, Religion, Sports and Technology. As for AlArabiya dataset, it has six categories: Culture, Finance, Medical, Politics, Sports and Technology. SANAD has a total number of 194,797 articles categorized and formatted as shown in Fig.1. In general, SANAD adopted the annotation of each article as appeared in its news portal source. Only one collection of articles is manually re-labeled to enrich the 'politics' category in AlArabiya dataset. The distribution of articles per category for each dataset is summarized in Table 1 and Fig. 2, Distribution of articles per label for each dataset, and a list of examples from the datasets is presented in Fig. 3.

AlArabiya\Finance\00002.txt

قال الرئيس التنفيذي للشركة السعودية للكهرباء زياد الشيحة، في مقابلة عبر الهاتف مع قناة "العربية"، إنه لأول مرة في تاريخ الشركة تراجع استهلاك المملكة في فترات الذروة. وأضاف الشيحة أن التراجع الذي حدث في استهلاك المملكة في 2016، مقارنة مع عام 2015، دفع الشركة لمراجعة السعات المطلوبة لدى دراسة المشاريع الجديدة. وأكد أن مسألة انخفاض الحمل الذروي لأول مرة في تاريخ الشركة عن العام جلعنا نراجع المحطات المستقبلية والتي ستكون بعقود شراء الطاقة، وهذا سيكون لمشاريع الإنتاج وتتم مراجعة السعات المطلوبة خاصة مع قلة الحمل الذروي في 2016. وستزود الشركة السعودية للكهرباء شركة زين السعودية بشبكة الألياف البصرية الممتدة لـ 60 ألف كم. وأوضح الشيحة أن "قطاع التوليد سيطرح للخصخصة كما هو معلن، ونعمل على الموضوع بشكل متوازن وشبه يومي". وكانت خسائر شركة السعودية للكهرباء قد تفاقمت بأكثر من 60%، في الربع الأخير من العام الماضي، مقارنة بالربع المماثل من عام 2015، لتبلغ 2.34 مليار ريال. من ناحية أخرى، ارتفعت أرباح الشركة بنسبة 37%، خلال العام الماضي، مقارنةً بعام 2015، لتبلغ 2.1 مليار ريال. وأرجعت الشركة تفاقم الخسائر الفصلية إلى ارتفاع تكلفة المبيعات نتيجة الزيادة في أسعار الوقود وارتفاع المصاريف التشغيلية.

Fig. 1. An example of an Article.

Table.1. Distribution of articles per category.

Label	AlArabiya	Akhbarona-Alanba	AlKhaleej
Finance	30,076	9,280	6,500
Sports	23,058	15,377	6,500
Culture	5,619	6,746	6,500
Tech	4,411	12,199	6,500
Politics	4,368	13,979	6,500
Medical	3,715	12,947	6,500
Religion	–	7,522	6,500
Total	71,247	78,050	45,500

https://data.mendeley.com/datasets/57zpx667y9/2

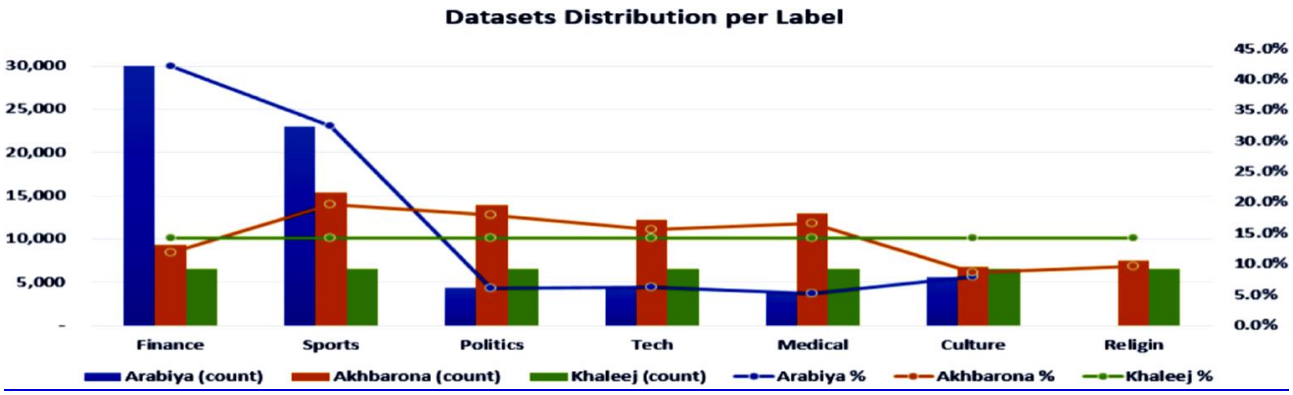

Fig. 2. Distribution of articles per label for each dataset.

Label	Dataset	Example Article Text
Finance	AlArabiya	أظهرت بيانات من إدارة الجمارك، اليوم الجمعة، أن واردات الصين من النفط الخام الإيراني في ديسمبر ارتفعت بنسبة 19.1% عن مستواها قبل عام لتصل إلى 2.57 مليون طن أو 604740 برميلا يوميا.
Tech	AlArabiya	بدأت شبكة التدوين المصغر #تويتر بإجراء تعديلات جديدة على سياسة الخصوصية بما يشمل التحكم بالبيانات التي يتم جمعها بواسطة الشبكة وذلك من أجل إظهار إعلانات تناسب اهتمامات المستخدم.
Sports	AlKhaleej	حقق فريق مانشستر سيتي حامل اللقب فوزه السادس على التوالي في الدوري الإنجليزي الممتاز لكرة القدم بتغلبه على ضيفه كريستال بالاس بثلاثة أهداف نظيفة في المرحلة السابعة عشرة من المسابقة.
Culture	AlKhaleej	تحاول رواية نصف مواطن محترم، للكاتب السعودي هاني نقشبندي الصادرة مؤخراً عن دار الساقي، أن تطرح بأسلوب فانتازي المكانة التي يحظى بها المواطن العربي، وكيف ينظر إليه الحاكم.
Politics	Akhbarona-Alanba	غادر الرئيس الفرنسي, السيد فرانسوا هولاند, مرفوقا بالسيدة فاليري تريرفيلر, مساء اليوم الخميس, المغرب في ختام زيارة رسمية للمملكة استغرقت يومين.
Medical	Akhbarona-Alanba	يعد اللفت من احد اشهر الخضار الجذرية ذات القيمة الغذائيه العالية، والغنية بالفيتامينات والمعادن ومضادات الاكسدة القوية والمتنوعة والتي منحته فوائد عديدة للجسم والمناعة.
Religion	Akhbarona-Alanba	من نوائب الدهر في عصرنا الحالي الغُلو في المهور وتجهيزات الزواج والتي تَئِن منها مجتمعاتٌ عربيةٌ ثريةٌ فما بالنا بالمجتمعات الأقل دخلاً بل التي أصبحت متوسطاتها تتراوح بين الستر والفقر.

Fig. 3. Illustrative examples from each category from the 3 datasets.

https://data.mendeley.com/datasets/57zpx667y9/2

1.2.Experiments dataset

A subset Khaleej (45500 articles in 7 categories) of SANAD. Labels are categorized in: Culture, Finance, Medical, Politics,Religion,Sports,Tech.

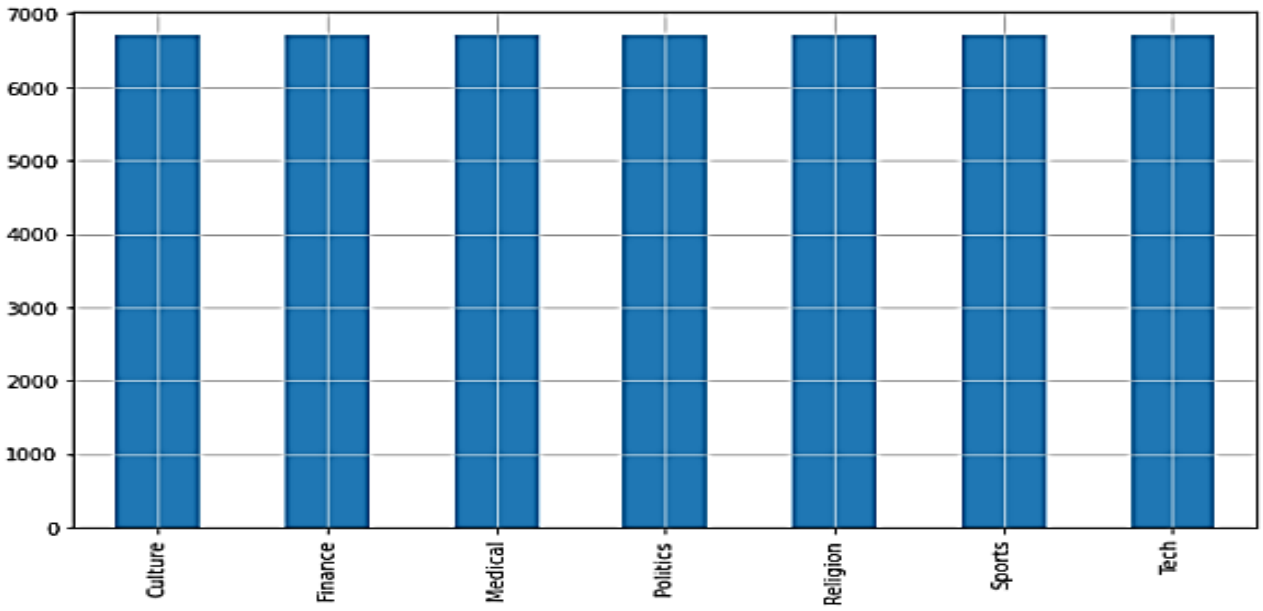

Fig. 4. Dataset Khaleej

2. Experiments

2.1.The first experiment: Machine Learning

- Arabic Text Preprocessing

Fig. 5. Extracting data from the folders and files

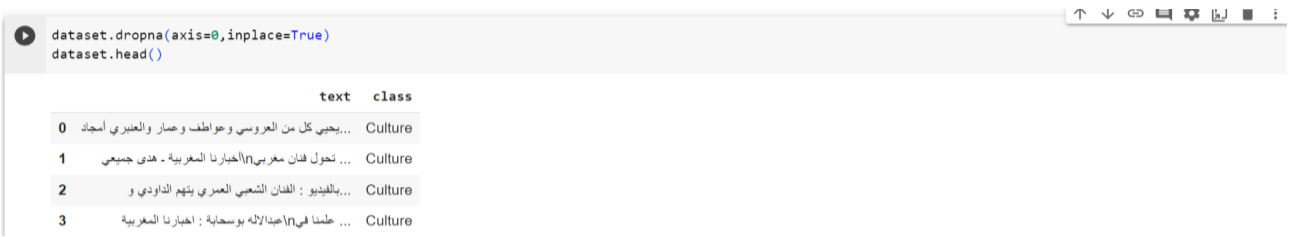

Fig.6. Read Dataset

https://data.mendeley.com/datasets/57zpx667y9/2

```
[ ] dataset['word_count']= dataset['text'].apply(lambda x:len(str(x).split(" ")))
    dataset['char_count']= dataset['text'].str.len()
    dataset['avg_char_per_word'] = dataset['text'].apply(lambda x: avg_word(x))
    dataset['stopwords']=dataset['text'].apply(lambda x: len([y for y in x.split() if y in stop]))
    dataset=dataset.sort_values(by='word_count',ascending=[0])
    dataset.head()
```

	text	class	word_count	char_count	avg_char_per_word	stopwords
36350	تلتقي إسبانيا مع إيطاليا مساء اليوم في المبارا...	Sports	5786	33339	4.762185	1161
8355	انطلاقاً من حرص إدارة سوق أبوظبي للأوراق المال...	Finance	3999	24373	5.095024	792
6685	سجل اجمالي موجودات البنوك التجارية السعودية تم...	Finance	3720	20464	4.501344	656
35160	تضم المجموعة الأولى كلاً من سويسرا وتشيكيا وال...	Sports	3658	22262	5.086113	592

Fig. 7. Exploratory Data Analysis

```
[ ] # Remove stopwords
    def removeStop(text):
      tmp=word_tokenize(text)
      text=" ".join([w for w in tmp if not w in stop and len(w) >=2])
      return text

    dataset['noStop_article']=dataset['text'].apply(lambda x: removeStop(x))
    dataset.head()
```

	text	class	word_count	char_count	avg_char_per_word	stopwords	noStop_article
3914	من الخطاب الملكي في 9 مارس، إلى الانتخابات ا"]...	Politics	4890	29109	5.360779	939	من الخطاب الملكي مارس، الانتخابات التشريعية ي'...
4337	الحمد لله نحمده، ونستعينه، ونستغفره، ونعوذ ب"]...	Religion	4327	23823	4.505893	998	الحمد لله نحمده، ونستعينه، ونستغفره، ونعوذ با"...
241	شهد المشهد الثقافي والفني المغربي خلال العام"]...	Culture	4177	26898	5.439789	805	شهد المشهد الثقافي والفني المغربي خلال العا ``...
4184	إن الحمد لله نحمده ونستعينه ونستغفره ، ونعوذ"]...	Religion	3667	18242	3.974911	772	إن الحمد لله نحمده ونستعينه ونستغفره ونعوذ با"...
4182	العالم الإسلامي حُذع بإيران وبحزب الله وأحدا"]...	Religion	3316	18249	4.503619	800	العالم الإسلامي حُذع بإيران وبحزب الله وأحدا'...

Fig. 8. Remove stopwords

```
[ ]     text = text.replace(u"\u0670", "`")  # dagger 'alif
        return text

    # aggregate all preprocessing steps into one column for the next step
    dataset['normalized_article']=dataset['noStop_article'].apply(lambda x:normalize(x))
    dataset.head()
```

	text	class	word_count	char_count	avg_char_per_word	stopwords	noStop_article	normalized_article
3914	من الخطاب الملكي في 9 مارس، إلى"]... الانتخابات ا	Politics	4890	29109	5.360779	939	من الخطاب الملكي مارس، الانتخابات التشريعية'... ي	من الخطاب الملكي مارس، الانتخابات التشريعية'... ى
4337	الحمد لله نحمده، ونستعينه، ونستغفره، ونعوذ"]... ب	Religion	4327	23823	4.505893	998	الحمد لله نحمده، ونستعينه، ونستغفره، ونعوذ'... با	الحمد لله نحمده، ونستعينه، ونستغفره، ونعوذ با"...
241	شهد المشهد الثقافي والفني المغربي خلال"]... العام	Culture	4177	26898	5.439789	805	شهد المشهد الثقافي والفني المغربي خلال `` ...العا	شهد المشهد الثقافي والفني المغربي خلال `` ...العا
4184	إن الحمد لله نحمده ونستعينه ونستغفره ،"]... ونعوذ	Religion	3667	18242	3.974911	772	إن الحمد لله نحمده ونستعينه ونستغفره ونعوذ'... با	ان الحمد لله نحمده ونستعنه ونستغفره ونعوذ با"...

Fig.9. Normalization

```
# Noise Removal
def clean(text):
  # Remove extra whitespace
  text=re.sub('\s+',' ',text)
  # Remove numbers
  text=re.sub('\d+',' ',text)
  return text
dataset['abstract_article']=dataset['clean_article'].apply(lambda x: clean(x))
dataset.head()
```

	text	class	word_count	char_count	avg_char_per_word	stopwords	noStop_article	normalized_article	clean_article	abstract_article
3914	من الخطاب"] الملكي في 9 مارس، إلى الانتخابات ا...	Politics	4890	29109	5.360779	939	من الخطاب الملكي مارس، الانتخابات التشريعية ي...	من الخطاب الملكي مارس، الانتخابات التشريعية ى...	من الخطاب الملكي مارس الانتخابات التشريعية يوم...	من الخطاب الملكي مارس الانتخابات التشريعية يوم...

Fig. 10. Abstract article (Remove Punctuations and Noise Removal (Remove extra whitespace, Remove numbers))

https://data.mendeley.com/datasets/57zpx667y9/2

Fig. 11. Preparing the Dataset

- Applying ML

```
[23] clf = RandomForestClassifier(n_estimators=100)
     clf.fit(clfx_train,clfy_train)
```

```
RandomForestClassifier
RandomForestClassifier()
```

```
[24] print(clf.score(clfx_test,clfy_test))
```

```
0.9652014652014652
```

Fig. 12. Random Forest Classifier

```
logR=Pipeline([('vect',CountVectorizer()),
               ('tfidf',TfidfTransformer()),
               ('clf',LogisticRegression())
               ])

# logR=Pipeline([('vect',CountVectorizer(binary=True)),
#               ('tfidf',TfidfTransformer()),
#               ('clf',LogisticRegression())
#               ])

logR.fit(x_train,y_train) # takes 31s to run
y_pred_logR=logR.predict(x_test)

print(f'Accuracy: {accuracy_score(y_pred_logR,y_test)}')
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/linear_model/_logistic.py:458: ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
  n_iter_i = _check_optimize_result(
Accuracy: 0.9764102564102564
```

Fig. 13. Logistic Regression

```
naiveB=Pipeline([('vect',CountVectorizer()),
                 ('tfidf',TfidfTransformer()),
                 ('clf',MultinomialNB())
                ])

naiveB.fit(x_train,y_train)
y_pred_naiveB=naiveB.predict(x_test)
print(f'accuracy: {accuracy_score(y_pred_naiveB,y_test)}')
```

```
accuracy: 0.9604395604395605
```

Fig. 14. Naive Bayes

https://data.mendeley.com/datasets/57zpx667y9/2

```
[35] # Train
     model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['categorical_accuracy'])
     model.fit(x_train_tok, y_train_encoded_, batch_size=100, epochs=5, verbose=1, validation_split=0.1)

     Epoch 1/5
     36/36 [==============================] - 154s 4s/step - loss: 0.3064 - categorical_accuracy: 0.9258 - val_loss: 0.2111 - val_categorical_accuracy: 0.9650
     Epoch 2/5
     36/36 [==============================] - 149s 4s/step - loss: 0.0071 - categorical_accuracy: 0.9989 - val_loss: 0.2139 - val_categorical_accuracy: 0.9625
     Epoch 3/5
     36/36 [==============================] - 146s 4s/step - loss: 2.8109e-04 - categorical_accuracy: 1.0000 - val_loss: 0.1988 - val_categorical_accuracy: 0.9675
     Epoch 4/5
     36/36 [==============================] - 145s 4s/step - loss: 1.2105e-04 - categorical_accuracy: 1.0000 - val_loss: 0.1987 - val_categorical_accuracy: 0.9675
     Epoch 5/5
     36/36 [==============================] - 145s 4s/step - loss: 6.9604e-05 - categorical_accuracy: 1.0000 - val_loss: 0.1983 - val_categorical_accuracy: 0.9675
     <keras.callbacks.History at 0x794cbccbfc40>

[36] # Test
     eval_val = model.evaluate(x_test_tok, y_test_encoded_, verbose=0)
     print("Loss\t\t" , 'categorical_accuracy\t')
     print(eval_val)

     Loss            categorical_accuracy
     [0.1825205534696579, 0.9660000205039978]
```

Fig. 15. Neural Network

Table.2. Machine Learning Results

Model	Accuracy	Our Accuracy
Random Forest Classifier	89.47%	96.52%
Logistic Regression	93.73%	97.64%
Naive Bayes	92.39%	96.04%
Neural Network	92.69%	96.60%

2.2.The second experiment: Machine Learning and Deep Learning

```
data= pd.DataFrame(columns=['News','Type'])
df=pd.read_csv('/content/drive/MyDrive/sanad-dataset.csv')
data['News']=df['text']
data['Type']=df['class']
data.head()
```

	News	Type
0	...يدين الشباب في هذا الزمن للتكنولوجيا ومستجداته	Culture
1	...نعت النخبة السياسية والثقافية الموريتانية الرا	Culture
2	...فتحت الأمانة العامة لجائزة الشيخ زايد العالمية	Culture
3	... الشارقة: عثمان حسن لن نجد نقاداً فلسفياً أكثر	Culture
4	...تستضيف دار أوبرا دبي عند الساعة السادسة من مسا	Culture

Fig.16. Read Dataset

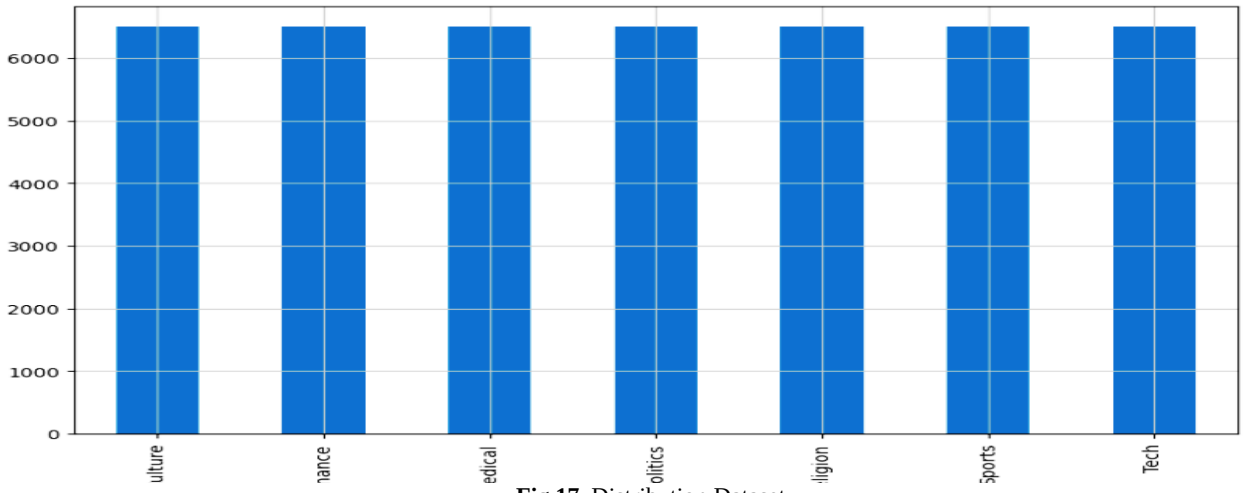

Fig.17. Distribution Dataset

https://data.mendeley.com/datasets/57zpx667y9/2

- **Applying ML and DL**

```
GaussianNB()

[19] y_pred = classifier.predict(X_test)

nb_score = accuracy_score(y_test, y_pred)
nb_score

0.9050949050949051
```

Fig.18. GaussianNB

```
[36] rnn_model = Sequential()
rnn_model.add(Embedding(MAX_NB_WORDS, EMBEDDING_DIM, input_length=X_final.shape[1]))
rnn_model.add(SpatialDropout1D(0.2))
rnn_model.add(SimpleRNN(128, dropout=0.2, recurrent_dropout=0.2))
rnn_model.add(Dense(7, activation='softmax'))
rnn_model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])

epochs = 5
batch_size = 64
rnn_history = rnn_model.fit(X_train, y_train, epochs=epochs, batch_size=batch_size,validation_split=0.2)
```

```
Epoch 1/5
382/382 [==============================] - 76s 193ms/step - loss: 1.9124 - accuracy: 0.2132 - val_loss: 1.8144 - val_accuracy: 0.2383
Epoch 2/5
382/382 [==============================] - 74s 193ms/step - loss: 1.7708 - accuracy: 0.2865 - val_loss: 1.7331 - val_accuracy: 0.3184
Epoch 3/5
382/382 [==============================] - 72s 189ms/step - loss: 1.6871 - accuracy: 0.3201 - val_loss: 1.5808 - val_accuracy: 0.3702
Epoch 4/5
382/382 [==============================] - 71s 186ms/step - loss: 1.5532 - accuracy: 0.3798 - val_loss: 1.4726 - val_accuracy: 0.4328
Epoch 5/5
382/382 [==============================] - 73s 191ms/step - loss: 1.3622 - accuracy: 0.4640 - val_loss: 1.4019 - val_accuracy: 0.4358
```

```
rnn_accuracy = rnn_model.evaluate(X_test,y_test)
print('RNN Model Evaluation\n Loss: {:0.3f}\n Accuracy: {:0.3f}'.format(rnn_accuracy[0],rnn_accuracy[1]))
```

```
470/470 [==============================] - 10s 21ms/step - loss: 1.4032 - accuracy: 0.4325
RNN Model Evaluation
 Loss: 1.403
 Accuracy: 0.433
```

Fig.19.RNN

```
[29] lstm_model = Sequential()
lstm_model.add(Embedding(MAX_NB_WORDS, EMBEDDING_DIM, input_length=X_final.shape[1]))
lstm_model.add(SpatialDropout1D(0.2))
lstm_model.add(LSTM(128, dropout=0.2, recurrent_dropout=0.2))
lstm_model.add(Dense(7, activation='softmax'))
lstm_model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])

epochs = 5
batch_size = 64

lstm_history = lstm_model.fit(X_train, y_train, epochs=epochs, batch_size=batch_size,validation_split=0.2)
```

```
Epoch 1/5
382/382 [==============================] - 306s 796ms/step - loss: 0.9743 - accuracy: 0.6714 - val_loss: 0.3088 - val_accuracy: 0.9216
Epoch 2/5
382/382 [==============================] - 269s 703ms/step - loss: 0.2813 - accuracy: 0.9227 - val_loss: 0.4171 - val_accuracy: 0.8457
Epoch 3/5
382/382 [==============================] - 266s 697ms/step - loss: 0.2342 - accuracy: 0.9328 - val_loss: 0.3450 - val_accuracy: 0.9000
Epoch 4/5
382/382 [==============================] - 273s 715ms/step - loss: 0.1197 - accuracy: 0.9656 - val_loss: 0.4034 - val_accuracy: 0.8691
Epoch 5/5
382/382 [==============================] - 269s 706ms/step - loss: 0.0933 - accuracy: 0.9751 - val_loss: 0.2216 - val_accuracy: 0.9482
```

```
[35] lstm_accuracy = lstm_model.evaluate(X_test,y_test)
print('LSTM Model Evaluation\n Loss: {:0.3f}\n Accuracy: {:0.3f}'.format(lstm_accuracy[0],lstm_accuracy[1]))
```

```
470/470 [==============================] - 32s 68ms/step - loss: 0.1961 - accuracy: 0.9530
LSTM Model Evaluation
 Loss: 0.196
 Accuracy: 0.953
```

Fig.20.LSTM

https://data.mendeley.com/datasets/57zpx667y9/2

```
[38] gru_model = Sequential()
     gru_model.add(Embedding(MAX_NB_WORDS, EMBEDDING_DIM, input_length=X_final.shape[1]))
     gru_model.add(SpatialDropout1D(0.2))
     gru_model.add(GRU(128, dropout=0.2, recurrent_dropout=0.2))
     gru_model.add(Dense(7, activation='softmax'))
     gru_model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])

     epochs = 5
     batch_size = 64

     gru_history = gru_model.fit(X_train, y_train, epochs=epochs, batch_size=batch_size,validation_split=0.2)
```

```
Epoch 1/5
382/382 [==============================] - 241s 625ms/step - loss: 0.8177 - accuracy: 0.7033 - val_loss: 0.2785 - val_accuracy: 0.9175
Epoch 2/5
382/382 [==============================] - 235s 616ms/step - loss: 0.1979 - accuracy: 0.9411 - val_loss: 0.1839 - val_accuracy: 0.9478
Epoch 3/5
382/382 [==============================] - 243s 637ms/step - loss: 0.0861 - accuracy: 0.9761 - val_loss: 0.1932 - val_accuracy: 0.9552
Epoch 4/5
382/382 [==============================] - 235s 617ms/step - loss: 0.0458 - accuracy: 0.9877 - val_loss: 0.1747 - val_accuracy: 0.9569
Epoch 5/5
382/382 [==============================] - 240s 630ms/step - loss: 0.0226 - accuracy: 0.9948 - val_loss: 0.1985 - val_accuracy: 0.9536
```

```
gru_accuracy = gru_model.evaluate(X_test, y_test)
print('GRU Model Evaluation\n Loss: {:0.3f}\n Accuracy: {:0.3f}'.format(gru_accuracy[0],gru_accuracy[1]))
```

```
470/470 [==============================] - 38s 82ms/step - loss: 0.1836 - accuracy: 0.9542
GRU Model Evaluation
 Loss: 0.184
 Accuracy: 0.954
```

Fig.21. GRU

Table.3. Machine Learning and deep learning Results

Model	Accuracy	Our Accuracy
Naive Bayes	83.00%	90.50%
RNN	85.00%	43.30%
LSTM	91.00%	95.30%
GRU	90.00%	95.40%
BERT	90.00%	
DistilBERT	89.37%	
RoBERTa	78.70%	

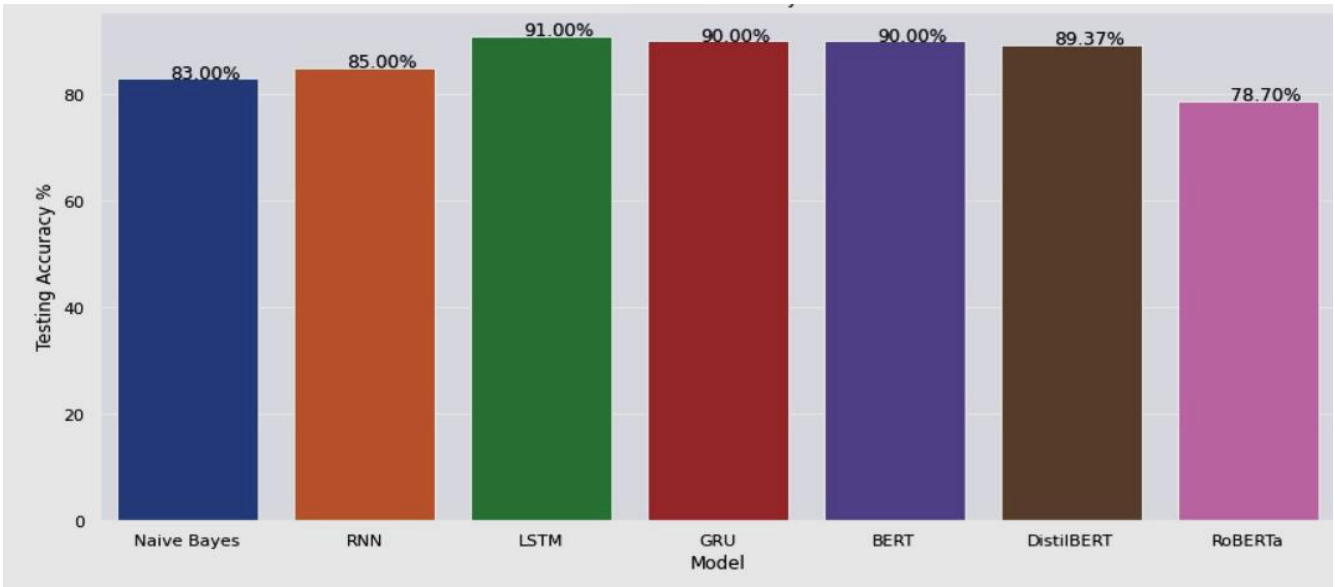

Fig.22. Machine Learning and deep learning Result

https://data.mendeley.com/datasets/57zpx667y9/2

3. Results

Table.4. Machine learning and deep learning

Model	Experiment	ML or DL	Accuracy	Accuracy
Random Forest Classifier	First	ML	89.47%	96.52%
Logistic Regression	First	ML	93.73%	97.64%
Naive Bayes	First	ML	92.39%	96.04%
Neural Network	First	ML	92.69%	96.60%
Naive Bayes	Second	ML	83.00%	90.50%
RNN	Second	DL	85.00%	43.30%
LSTM	Second	DL	91.00%	95.30%
GRU	Second	DL	90.00%	95.40%
BERT	Second	DL	90.00%	-
DistilBERT	Second	ML	89.37%	-
RoBERTa	Second	ML	78.70%	-

4. Future Directions

Because of the importance of the issue of processing natural languages at this time, especially the Arabic language, which suffers from great poverty in the field of natural language processing, we will focus in the future on work in this field and apply learning transfer models to the classification of Arabic texts. We will also focus on generating the Arabic text and recommend the rest of our colleagues. Addressing this area.

5. Conclusion

Algorithms of text classification are the basis of many applications for natural language processing, such as text description, query response, detection of spam, and visualization of text. Arabic language on the internet is rising increasingly, but its content is still as poor as 3 percent. Few studies have been done to categorize and classify the Arabic language. In the first experiment, the highest result they achieved in the Logistic Regression algorithm was 93.73%accuracy, and the highest result they achieved in the Logistic Regression algorithm was 90% accuracy. In the second experiment, the highest result they achieved in the Logistic LSTM was 91.00%accuracy, and the highest result they achieved in the GRU algorithm was 95.40%accuracy.

Resource

[1] Abdulghani, F. A., & Abdullah, N. A. (2022). A survey on Arabic text classification using deep and machine learning algorithms. *Iraqi Journal of Science*, 409-419.

[2] Al-Tamimi, A. K., Bani-Isaa, E., & Al-Alami, A. (2021, March). Active learning for Arabic text classification. In *2021 International Conference on Computational Intelligence and Knowledge Economy (ICCIKE)* (pp. 123-126). IEEE.

[3] Al-Salemi, B., Ayob, M., Kendall, G., & Noah, S. A. M. (2019). Multi-label Arabic text categorization: A benchmark and baseline comparison of multi-label learning algorithms. *Information Processing & Management*, *56*(1), 212-227.

[4] Einea, O., Elnagar, A., & Al Debsi, R. (2019). Sanad: Single-label arabic news articles dataset for automatic text categorization. Data in brief, 25, 104076.

[5] Elnagar, A., Al-Debsi, R., & Einea, O. (2020). Arabic text classification using deep learning models. Information Processing & Management, 57(1), 102121.

[6] Alalyani, N., & Marie-Sainte, S. L. (2018). NADA: New Arabic dataset for text classification. International Journal of Advanced Computer Science and Applications, 9(9). https://data.mendeley.com/datasets/57zpx667y9/2