



Volume 2	Issue 1	February (2022)	DOI: 10.47540/ijias.v2i1.432	Page: 60 – 67
----------	---------	-----------------	------------------------------	---------------

## Sentiment Analysis and Topic Modeling on Arabic Twitter Data during Covid-19 Pandemic

Nassera Habbat<sup>1</sup>, Houda Anoun<sup>1</sup>, Larbi Hassouni<sup>1</sup>

<sup>1</sup>Ecole Supérieure de Technologie Hassan II University, Morocco

**Corresponding Author:** Nassera Habbat; Email: [nassera.habbat@gmail.com](mailto:nassera.habbat@gmail.com)

### ARTICLE INFO

**Keywords:** Latent Dirichlet Allocation, Sentiment Analysis, Topic Modeling, Twitter.

*Received* : 04 January 2022

*Revised* : 08 February 2022

*Accepted* : 10 February 2022

### ABSTRACT

Twitter Sentiment Analysis is the task of detecting opinions and sentiments in tweets using different algorithms. In our research work, we conducted a study to analyze and compare different Algorithms of Machine Learning (MLAs) for the classification task, and hence we collected 37 875 Moroccan tweets, during the COVID-19 pandemic, from 01 March 2020 to 28 June 2020. The analysis was done using six classification algorithms (Naive Bayes, Logistic Regression, Support Vector Machine, K-Nearest Neighbors, Decision Tree, Random Forest classifier) and considering Accuracy, Recall, Precision, and F-Score as evaluation parameters. Then we applied topic modeling over the three classified tweets categories (negative, positive, and neutral) using Latent Dirichlet Allocation (LDA) which is among the most effective approaches to extract discussed topics. As result, the logistic regression classifier gave the best predictions of sentiments with an accuracy of 68.80%.

### INTRODUCTION

Twitter is a social networking site that allows its users to express their thinking, interests, and views on different topics. A large number of users in the world generally, and Morocco, in particular, use this micro-blogging website as a way to transmit their views freely; these opinions contain negative, positive, and neutral sentiments on a specific subject.

Sentiment Analysis (SA) in general, is the way of detecting and categorizing the polarity of a given text at document, phrase, and sentence level, and predicted performance of different text classifiers using various natural language processing (NLP) techniques.

There is a lot of research on sentiment analysis in social media, especially on Twitter. We present in this section some existing work. In (Maheshwari *et al.*, 2019), the authors classified opinions about iPhone 6, using Naïve Bayes classifier with different training datasets and comparing it with baseline algorithm that grouped the tweets in negative and positive classes with 88.32% of accuracy. In the same way, the authors in (Tiwari *et*

*al.*, 2019) analyzed opinions of travelers about an airline company and classified them into negative and positive, in their experiments, they used BIRCH clustering (Balanced Iterative Reducing and Clustering Using Hierarchies) as a hierarchical clustering method and association rule mining to find the association between variables and for analysis the dataset.

The sentiment analysis on Twitter is used in different fields especially marketing (Smailovic *et al.*, 2014) and policy (Ringsquandl and Petkociv, n.d.). Many algorithms are used to detect sentiment from the text, in (Smailovic *et al.*, 2014), the authors used SVM (Support Vector Machine) to classify 152,570 tweets concerning eight companies, as positive, negative, or neutral. While, the authors in (Fatahillah *et al.*, 2017) utilized naïve Bayes to group Indonesian tweets into positive and negative speeches, and they found in their research that Naive Bayesian classifier achieves high accuracy (93%) and speed to be comparable with selected neural network and decision tree classifiers. However, the authors in (Devika, n.d.) had as an outcome of their experiments, that

Random forest gives better and stable results than Naive Bayes and KNN (K-Nearest Neighbors) in detecting spammers and non-spammers in English tweets concerning Apple, the comparison between classifiers is done based on their evaluation parameters (Accuracy, Precision, F-Measure, and Recall). Using those measures, the authors (Ahuja *et al.*, 2019) found that logistic regression is the best classifier for sentiment analysis compared to five algorithms; K-Nearest Neighbors, Decision Tree, Support Vector Machine, Naive Bayes, and Random Forest. They applied their experiments on SS –TWEET (Sentiment Strength Twitter Dataset) labeled manually, and they extracted features using two techniques: N-Gram and Term Frequency – Inverse Document Frequency (TF-IDF), then they concluded that TF-IDF is the best technique of features with the word-level performance of SA of 3-4% superior to that of N-gram.

Some researchers focused on analyzing the content of tweets (topic modeling) in addition to sentiment analysis using various methods. LDA is one of the most popular techniques in text mining; it can offer good descriptions of broader topics compared with NMF (Non-negative Matrix Factorization) (O’Callaghan *et al.*, 2015). However, concerning short texts, the authors in (Ramage *et al.*, 2009), were the first to experiment with the use of topic modeling to perform classification tasks.

Generally, most researchers opt for LDA as a basic and generative probabilistic model used for topic modeling associated with NLP and text mining in the data science field. In (Zhang *et al.*, 2018), the authors used LDA to extract topics from chemotherapy radiation therapy or chemotherapy surgery tweets, between individuals and organizations in 2009/2010. To assess perceptions about chemotherapy of patients, the method was also combined with another method (Hajjem and Latiri, 2017), without modifying the basic

mechanism of LDA. In addition to the medical domain, LDA was used in the political domain (Hagen, 2018), 87% of LDA-generated topics were meaningful to human judges, and therefore LDA was considered as an efficient method of topic modeling.

In our study, we used as a dataset about 37 875 tweets, collected from 01 March 2020 to 28 June 2020, during the COVID-19 pandemic, the six classifiers used are, Naive Bayes (NB), Logistic Regression (LR), Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Random Forest classifier (RF), and Decision Tree (DT). Then the evaluation performance of each model is determined based on precision, accuracy, recall, and F-measure. Thereafter the topics under each category (positive, neutral, and negative tweets) are extracted using the LDA model.

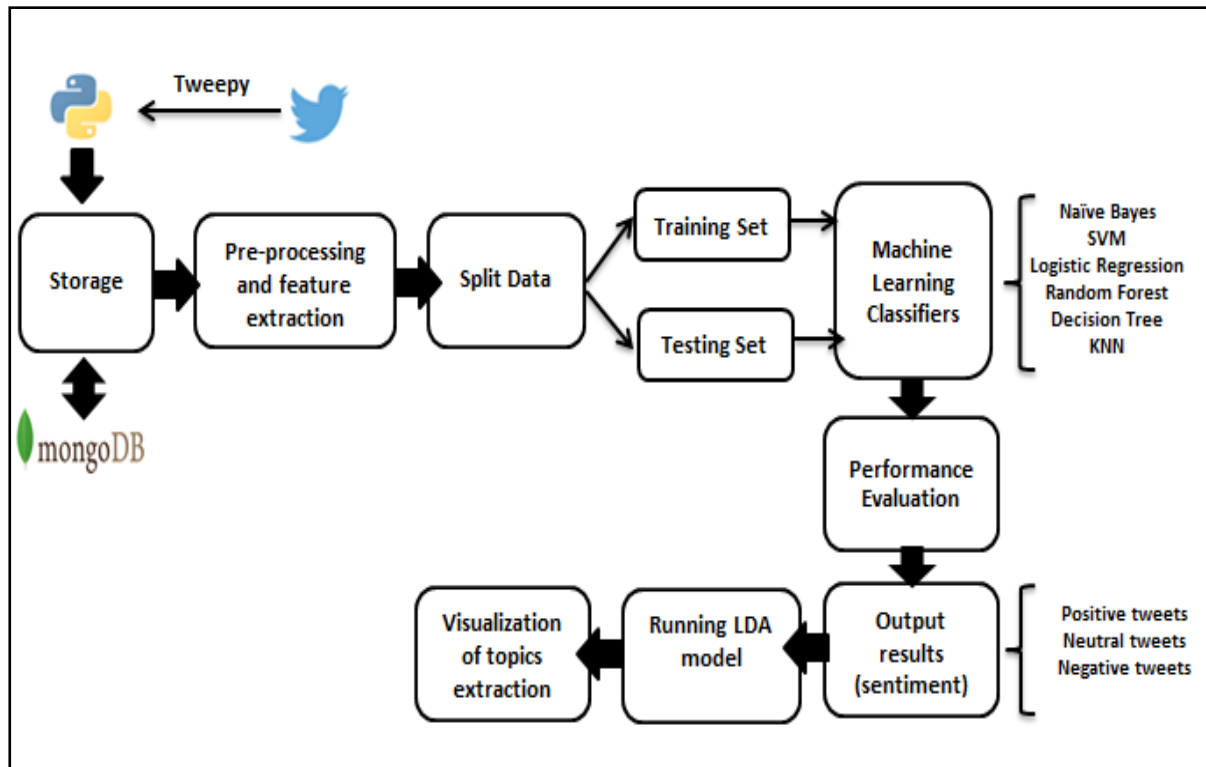
The global contribution is as follows. Section 2 summarizes related works. Algorithms and evaluation parameters are described in Section 3. Section 4 exposes the experimental results. We finish with a conclusion where we close this paper and outline the future work.

## **METHODS**

### **Proposed architecture**

The proposed architecture (illustrated in Fig.1) is realized following the next steps: Step-1 Getting a dataset. We retrieved tweets using the Tweepy library (“Tweepy,” n.d.) and used MongoDB (“MongoDB: the application data platform | MongoDB,” n.d.) database to store the collected tweets. Step-2 The tweets were pre-processed to be suitable for the extraction of the feature phase. Step-3 After preprocessing this data was passed in the trained classifier, to classify them into negative, positive, or neutral class based on trained outputs. Step-4: Finally, we extracted topics in the three classes of tweets using LDA.

Figure 1. Proposed Architecture



## Sentiment Analysis

### Naïve Bayes

Naïve Bayes (Fatahillah et al., 2017) is the supervised MLA used for classification, based on Bayes' Theorem. The Naive Bayes algorithm is facile to apply and practical for large datasets in particular. Bayes theorem allows to calculation from  $P(y)$ ,  $P(x)$ , and  $P(x|y)$ , a posterior probability  $P(y|x)$ , as shown in the following expression:

$$P(\beta|x) = \frac{P(x|\beta)P(y)}{P(x)} \quad (1)$$

$$P(y|x) = P(x_1|y) \times P(x_2|y) \times \dots \times P(x_n|y) \times P(y) \quad (2)$$

Where:

$P(y|x)$  : the posterior likelihood of class (target  $y$ ) taking into consideration the predictor (attributes  $x$ ).

$P(x|y)$  : the likelihood of predictor given class.

$P(y)$  : the prior likelihood of  $y$  class.

$P(x)$  : the prior likelihood of predictor.

### Support Vector Machine (SVM)

SVMs ("1.4. Support Vector Machines," n.d.) are an ensemble of supervised learning algorithms, they are used for regression, classification, and outliers' detection. SVM is represented in 2 vectors

where each vector is of size  $k$ . it is a classifier that partition the data taking into consideration that the margin should be maximum.

The aim of an SVM defined on an  $n$ -dimensional vector space is to get a surface in  $n$ -dimensional space separating the points of data in that space into multiple classes (Kelleher *et al.*, n.d.). In two dimensions, this surface is often a straight line. In three dimensions, the support vector machines often find a plane. In general, the support vector machines find a hyperplane. These surfaces are optimal in the sense that based on the information available to the machine; it optimizes the separation of the  $n$ -dimensional spaces. In general, SVM can be used to partition a space into any number of classes by generalizing the task itself.

### Logistic regression

Logistic regression (Lutfullaeva *et al.*, 2018) is among the most popular algorithms for classification problems; it is a transformation of linear regression using the sigmoid function. In general, LR is used to relate one categorical dependent variable to one or more independent variables, and its equation has the following form:

$$\frac{1}{1+e^{-z}} \quad (3)$$

Where  $z$  represents columns of vectors containing independent variables  $v$  (feature vector) and their corresponding weights  $w$  for  $n$  variables:

$$z = w_0 + w_1 v_1 + w_2 v_2 + \dots + w_n v_n \quad (4)$$

### Decision tree

A Decision tree (Ansari *et al.*, 2020) is a classification algorithm that uses a decisions tree graph and its possible results. The main concept is to partition the dataset into small subsets and in parallel, a tree associated is incrementally created. This gives in a tree containing decision and leaf nodes. A node of decision has two or more branches, and the leaf node interprets the decisions. The highest node looks like the root node. Then we found the smallest tree that fits the data.

### K Nearest Neighbor

The K Nearest Neighbor (KNN) (Imandoust and Bolandraftar, 2013) method is a classification technique based on the nearest learning models in the space of feature. The KNN is the simplest method of classification when the distribution of the data is not well known. This rule keeps all training data in the learning period and attributes it to every item of a class, which is defined in the training set as the most presented label of its  $k$ -nearest neighbors. Furthermore, the distance between the existing data points and test points, whom we want to determine class, is computed based on Euclidean distance or another distance measurement such as Manhattan distance. The  $k$  nearest neighbors initially decided, will adopt the class of new points of data, and then the class will be defined by the majority voting.

### Random forest

Random forest classifier (RFC) (Al Amrani *et al.*, 2018) was proposed by Adèle Cutler and Leo Breiman in 2001, it is part of the learning automatic technique that mixes two notions: “bagging” and random subspaces. The multiple decision trees used to train RFC based on softly different data subsets, this classifier consists of tree-structured classifiers collections  $\{f(x, \beta_i), i=1, \dots\}$  where the  $\{\beta_i\}$  are random vectors independently and identically distributed and every tree gives a part of the vote for the popular class (input  $x$ ). It is noted that reducing overfitting and variance, gives RFC more precise predictions in comparison with simple Decision Tree models.

### Evaluation parameters

To get performance conclusions from different algorithms, and thus determine their accuracy, we used evaluation parameters such as precision, accuracy, recall, and F-measure. The calculation of these parameters is based on the False Statement. For example, an expression is indicated as positive even if it is negative or an expression is announced neutral even though it is negative or positive. To uncover the False Statement (Fitriet *et al.*, 2019) we must make a performance evaluation. There are four classifications of performance evaluation:

1. True Positive : T\_Pos
2. False Positive : F\_Pos
3. True Negative : T\_Neg
4. False Negative : F\_Neg

### Accuracy

The accuracy measurement is the correctly predicted instances divided by the total observations.

$$\text{Accuracy} = \frac{T\_Pos + T\_Neg}{T\_Pos + T\_Neg + F\_Pos + F\_Neg} \quad (5)$$

### Precision

The precision measurement is a value of positive prognostic. It is the percentage of total relevant results correctly classified by the result of the algorithm.

$$\text{Precision} = \frac{T\_Pos}{T\_Pos + F\_Pos} \quad (6)$$

### Recall

The Recall is the ratio of correctly classified positive observations to all observations in the specific class.

$$\text{Recall} = \frac{T\_Pos}{T\_Pos + F\_Neg} \quad (7)$$

### F-measure

It is a measure of the accuracy of the test, and it is defined as the average of the recall and precision.

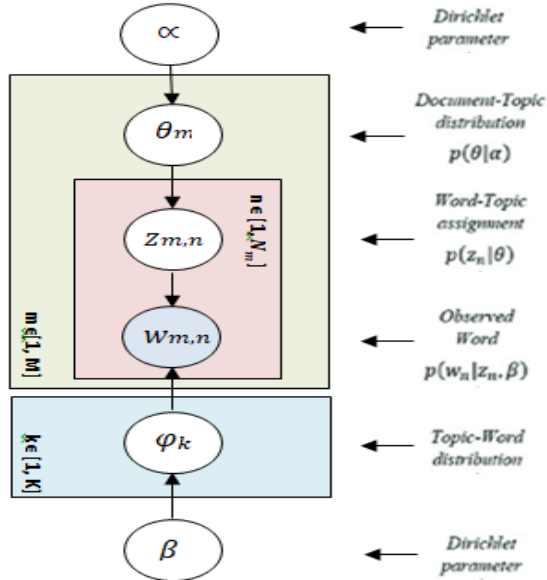
$$F_{\text{Measure}} = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} \quad (8)$$

**Topic Modeling**

*Latent Dirichlet Allocation (LDA)*

LDA (Blei, n.d.) is a generative statistical method utilized for the extraction of the topics in a specific collection of text. Those topics are identified based on the probability of words co-occurrences holder in them.

Figure 2. Graphical model of LDA



As shown in Figure 2, the boxes in form of “plates” present the following replicates: The green plate interprets documents, the red plate marks the repeated selection of words and topics in the document, and the blue part denotes the latent topics concealed in the collection of documents.

We explain formally the following items: The basic element of distinct data is the Word which is from a vocabulary indicated by  $\{1, \dots, V\}$ . Unit-basis vectors ( $v$ -th word) is used to represent these words, and these vectors have only component equal to one ( $w^v = 1$ ) and others equal to zero ( $w^u = 0$ ) for  $u \neq v$ .

A sequence of  $N$  words constitute a document, presented by  $D = (w_1, w_2, \dots, w_n)$ , and  $w_n$  is the  $n$ -th word in the sequence of the document  $D$ .  $M$  documents collection  $d_1$  forms a corpus, denoted by  $C = (d_1, d_2, \dots, d_M)$ .

The successive procedures for every document in the archive of text are presented as follows:

1. For the  $m$  document  $d$  for  $M$  ranging from  $1$  to  $M$  in the ensemble  $M$  document-corpus, determine  $\theta_m \sim \text{Dirichlet}(\alpha)$ ;
2. For every word  $w_{m,n}$  in the  $d$  document:

- a. Determine assignment of topic  $z_{m,n} \sim \text{Multinomial}(\theta_m)$ ;
  - b. Get correspondent distribution of topic  $\varphi_{z_{m,n}} \sim \text{Dirichlet}(\beta)$ ;
  - c. Sample a word  $w_{m,n} \sim \text{Multinomial}(\varphi_{z_{m,n}})$ .
- By repeating the processes in the generative operation described on the top for  $M$  times within every corresponding to a document, which is shown in Fig.1 (where  $\alpha$  and  $\beta$  are 2 hyper-parameters prior to Dirichlet), clearly, we get the probability of the corpus  $D$ :

$$P(D | \alpha, \beta) = \prod_{m=1}^M \int P(\theta_m | \alpha) F(\theta, \varphi) d\theta_m \tag{9}$$

Where:

$$F(\theta, \varphi) = \prod_{n=1}^N \sum_{z_{m,n}} P(z_{m,n} | \theta_m) P(w_{m,n} | \varphi_{z_{m,n}}). \tag{10}$$

**RESULTS AND DISCUSSION**  
**Data**

In this work, we collected 37 875 tweets published from 01 March 2020 to 28 June 2020 by Moroccan users and we stored them in the MongoDB database. Twitter Platform provides Twitter API to pull data from Twitter. For that, we created an account on: <https://apps.twitter.com>, and then we had the permission to access the database using 4 privy keys (access token, access secret token, consumer key, and consumer secret key) and pick up tweets using the REST API.

We filtrate tweets by place and language to obtain Moroccan tweets written in one of the standard languages (Arabic, French, or English), then we used the Python library Tweepy to handle these data. Finally, to store the collected data in a MongoDB database we utilized the Python library Pymongo (Siddharth *et al.*, n.d.).

**Preprocessing data**

We used various NLP techniques to preprocess the stored tweets. We present under, the applied procedures:

1. Translation of Arabic and French tweet to English using a python code based on google translate.
2. Cleaning up irrelevant data, to have relevant tweet data, this clean consists of elimination of hyperlinks, usernames with preceding “@”, hashes with preserving of tags?

3. Changing to lower case all letters.
4. Elimination of stopwords and punctuation using library of NLTK Python (“NLTK : Natural Language Toolkit,” n.d.) that gives a list of stop-words and punctuation characters for a lot of languages.
5. Tokenization consists of breaking sentences into small sections. For example; "Never give up" becomes: {'Never', 'give', 'up', '!'}.
6. Stemming consists of removing suffixes and affixes and obtaining the word base, For instance; “studying” to “study”, “student” to “study”.
7. Lemmatization of words using WordnetLemmatizer which provides for each word part of speech tag (POS TAG) (e.g., noun, verb, etc) using pos\_tag technique of NLTK library. For example; “studied” to “study”, “learned” to “learn”.

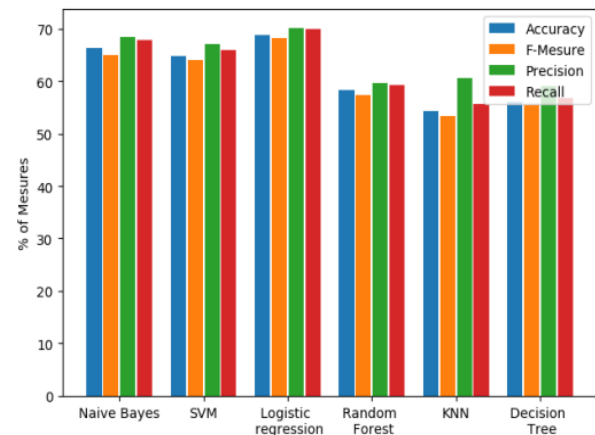
**Sentiment analysis**

After preprocessing data, we extracted features using the TF-IDF, which is an important statistical measure, having as output a matrix of term frequency versus inverse document frequency, and a collection of raw documents as input.

We used the training corpus of Sentiment140 (“For Academics - Sentiment140 - A Twitter Sentiment Analysis Tool,” n.d., p. 140) as training data, it is more relevant to use due to its large number of data (1,600,000 tweets), these tweets are labeled as following; 0 for negative tweets, 2 represent neutral expressions, and positive tweets are represented by number 4.

We separated this training set with a testing set; 80% of training data and testing data is 20 %. Then we implemented the classification on training data with six algorithms. Thereafter we applied to the testing data, the classification results from the training data. Finally, we achieved a performance evaluation to compare the six classifiers. We used scikit-learn (“scikit-learn: machine learning in Python — scikit-learn 1.0.2 documentation,” n.d.) to conduct these experiments.

Figure 3. Performance evaluation of different classifiers



Results from experiments (Figure 3) show that a Logistic regression classifier achieves remarkably better results than other algorithms in sentiment analysis with precision, recall, f-measure, and accuracy percentages of 70.20%, 69.90%, 68.28%, and 68.8 %, respectively.

Figure 4. Sentiment analysis on Moroccan tweets.

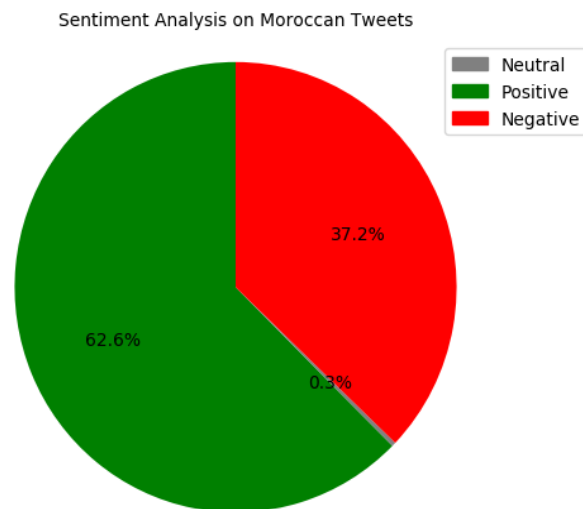


Figure 4 shows that Moroccan users post more positive tweets (62.6% of Positive tweets).

**Topic modeling**

To find out the hidden topics existing in the classified tweets using Logistic regression classifier, we implemented LDA model; firstly, we generated a dictionary from the corpus with the help of the gensim package (“gensim • PyPI,” n.d.). The dictionary established consists of a collection, which contains unique terms in the collection of documents. After that, we created a document-term matrix using the dictionary to use it in the model.

Table 1. The detected topics on positive, negative, and neutral tweets.

Sentiment / Topics	Topic 1	Topic 2	Topic 3
Positive	['love', 'safe', 'interesting', 'great', 'perfect', 'happy', 'thank', 'welcome', 'super', 'nice']	['update', 'recovery', 'right', 'back', 'soon', 'together', 'money', 'life', 'time', 'work']	['beautiful', 'marrakech', 'morocco', 'nature', 'amazing', 'memory', 'picture', 'rabat', 'agadir', 'casablanca']
Neutral	['sahara', 'laayoune_western', 'story', 'true', 'morocco', 'people', 'meaning', 'ramadan', 'episode', 'quarantine']	['wind', 'clouds_humidity', 'humidity', 'temperature', 'atmosphere', 'rabat', 'marrakech', 'cloud', 'humidity_wind', 'Agadir']	['course', 'picture', 'play', 'heart', 'graphic', 'music', 'story', 'lecture', 'life', 'home']
Negative	['sad', 'year', 'hesitation', 'opinion', 'choice', 'hard', 'cry', 'heart', 'imperfection', 'broken']	['guy', 'attack', 'police', 'black_live', 'racism', 'matter', 'urgent', 'wrong', 'black', 'trump']	['virus', 'time', 'long', 'thing', 'still', 'minute', 'incredible', 'patient', 'corona', 'end']

As exposed in Table 1, we presented the three most frequent topics extracted by the LDA model (each topic comprising of 10 words) concerning the three categories (positive, neutral, and negative) classified using logistic regression. For instance; Positive tweets are about feeling in Topic 01, Topic 02 interprets life, and Topic 03 is about travel and holiday. Concerning neutral tweets are about events in Morocco in Topic 01, Topic 02 is about the weather, and Topic 03 interprets hobbies. While negative tweets are about feeling in Topic 01, Topic 02 interprets George Floyd death in the USA, and the third Topic is about the corona virus.

**CONCLUSION**

In this research paper, we have applied six various machine learning algorithms to classify Moroccan Twitter data during the period of containment due to COVID-19, to determine the general emotion, and extract discussed topics under each category (positive, negative, and neutral tweets), using LDA model. Thus we concluded after doing sentiment analysis of these tweets, that the logistic regression classifier yielded the best prediction of sentiments with better performance for all four evaluation measurements namely; accuracy, precision, recall, and f-measure. We also used the generative statistical model LDA to discover topics distribution within Moroccan tweets, and so a better understanding of the Moroccan mood. In future

work, we plan to use deep learning algorithms (e.g. CNN, LSTM, Bi-LSTM ...).

**REFERENCES**

1. Support Vector Machines [WWW Document], n.d. Scikit-Learn. URL <https://scikit-learn/stable/modules/svm.html> (accessed 1.3.22).
2. Ahuja, R., Chug, A., Kohli, S., Gupta, S., Ahuja, P. (2019). The Impact of Features Extraction on the Sentiment Analysis. *Procedia Computer Science*. 152, 341–348.
3. Al Amrani, Y., Lazaar, M., El Kadiri, K.E. (2018). Random Forest and Support Vector Machine-based Hybrid Approach to Sentiment Analysis. *Procedia Computer Science*. 127, 511–520.
4. Ansari, M.Z., Aziz, M.B., Siddiqui, M.O., Mehra, H., Singh, K.P. (2020). Analysis of Political Sentiment Orientations on Twitter. *Procedia Computer Science*. 167, 1821–1828.
5. Blei, D.M., n.d. *Latent Dirichlet Allocation* 30.
6. Devika, R., n.d. Comparative Study of Classifiers in Twitter Spam Dataset using Naive Bayes, KNN, And Random Forest 10.
7. Fatahillah, N.R., Suryati, P., Haryawan, C. (2017). Implementation of Naive Bayes classifier algorithm on social media (Twitter) to the teaching of Indonesian hate speech, in 2017 *International Conference on Sustainable*

- Information Engineering and Technology (SIET)*. Presented at the 2017 International Conference on Sustainable Information Engineering and Technology (SIET), IEEE, Malang, pp. 128–131.
8. Fitri, V.A., Andreswari, R., Hasibuan, M.A. (2019). Sentiment Analysis of Social Media Twitter with Case of Anti-LGBT Campaign in Indonesia using Naïve Bayes, Decision Tree, and Random Forest Algorithm. *Procedia Computer Science*. 161, 765–772.
  9. For Academics - Sentiment140 - A Twitter Sentiment Analysis Tool [WWW Document], n.d. URL <http://help.sentiment140.com/for-students> (accessed 1.3.22).
  10. Gensim • PyPI [WWW Document], n.d. URL <https://pypi.org/project/gensim/> (accessed 1.3.22).
  11. Hagen, L. (2018). Content analysis of e-petitions with topic modeling: How to train and evaluate LDA models? *Information Processing Management*. 54, 1292–1307.
  12. Hajjem, M., Latiri, C. (2017). Combining IR and LDA Topic Modeling for Filtering Microblogs. *Procedia Computer Science*. 112, 761–770.
  13. Imandoust, S.B., Bolandraftar, M., 2013. Application of K-Nearest Neighbor (KNN) Approach for Predicting Economic Events: Theoretical Background 3, 7.
  14. Kelleher, J.D., Namee, B.M., D’Arcy, A., n.d. Fundamentals of Machine Learning For Predictive Data Analytics 31.
  15. Lutfullaeva, M., Medvedeva, M., Komotskiy, E., Spasov, K. (2018). Optimization of Sentiment Analysis Methods for classifying text comments of bank customers. *IFAC-PaperOnline*. 51, 55–60.
  16. Maheshwari, S., Shukla, S., Kumari, D. (2019). *Twitter Opinion Mining Using Sentiment Analysis 10*.
  17. MongoDB: the application data platform | MongoDB [WWW Document], n.d. URL <https://www.mongodb.com/> (accessed 1.3.22).
  18. NLTK :: Natural Language Toolkit [WWW Document], n.d. URL <https://www.nltk.org/> (accessed 1.3.22).
  19. O’Callaghan, D., Greene, D., Carthy, J., Cunningham, P. (2015). An analysis of the coherence of descriptors in topic modeling. *Expert System with Applications*. 42, 5645–5657.
  20. Ramage, D., Hall, D., Nallapati, R., Manning, C.D. (2009). Labeled LDA: a supervised topic model for credit attribution in multi-labeled corpora, in *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing Volume 1 - EMNLP ’09*. Presented at the 2009 Conference, Association for Computational Linguistics, Singapore, 248.
  21. Ringsquandl, M., Petkociv, D., n.d. *Analyzing Political Sentiment on Twitter 8*.
  22. Scikit-learn: machine learning in Python — scikit-learn 1.0.2 documenta-tion [WWW Document], n.d. URL <https://scikit-learn.org/stable/> (ac-cessed 1.3.22).
  23. Siddharth, S., Darsini, R., Sujithra, D.M., n.d. *Sentiment Analysis on Twitter Data Using Machine Learning Algorithms In Python 15*.
  24. Smailovic, J., Grčar, M., Lavrac, N., Žnidaršic, M. (2014). Stream-based active learning for sentiment analysis in the financial domain. *Information Sciences*. 285, 181–203.
  25. Tiwari, P., Yadav, P., Kumar, S., Mishra, B.K., Nguyen, G.N., Gochhayat, S.P., Singh, J., Prasad, M. (2019). Sentiment Analysis for Airlines Services Based on Twitter Dataset, *Social Network Analytics*. 149–162.
  26. Tweepy [WWW Document], n.d. URL <https://www.tweepy.org/> (ac-cessed 1.3.22).
  27. Zhang, L., Hall, M., Bastola, D. (2018). Utilizing Twitter data for analysis of chemotherapy. *International Journal of Medical Informatics*. 120, 92–100.