

Sentiment Analysis on Moroccan Dialect based on ML and Social Media Content Detection

Mouaad Errami¹, Mohamed Amine Ouassil², Rabia Rachidi³,
Bouchaib Cherradi⁴, Soufiane Hamida⁵, Abdelhadi Raihani⁶

EEIS Laboratory, ENSET of Mohammedia, Hassan II University of Casablanca, Mohammedia 28830, Morocco^{1, 2, 4, 5, 6}

LaROSERI Laboratory, Faculty of Science, Chouaib Doukali University, El Jadida, Morocco^{3, 4}

STIE Team, CRMEF Casablanca-Settat, Provincial Section of El Jadida, El Jadida 24000, Morocco⁴

GENIUS Laboratory, SupMTI of Rabat, Rabat, Morocco⁵

Abstract—As technology continues to evolve, humans tend to follow suit, and currently social media has taken place as the defacto method of communication. As it tends to happen with verbal communication, people express their opinions in written form and through an analysis of their words, one can extract what an individual wants from a product, a topic, or an event. By looking at the emotions expressed in such content, governments, businesses, and people can learn a lot that can help them improve their strategies. Therefore, in this study, we will use different algorithms to improve the Moroccan sentiment classification. The first step is to gather and prepare Moroccan Dialectal Arabic Twitter comments. Then, a lot of different combinations of extraction (n-grams) and weighting schemes (BOW/ TF-IDF) and word embedding for feature construction are applied to get the best classification models. We used Naive Bayes, Random Forests, Support Vector Machines, and Logistic regression and LSTM to classify the data we prepared. Our machine learning approach, which incorporates sentiment analysis, was designed to analyze Twitter comments written in Modern Standard Arabic or Moroccan Dialectal Arabic. As a final benchmark of our paper, we were simply a sliver shy away from the 70% mark in our accuracy by relying on the SVM algorithm. Although not a game-changing result, this was enough to encourage us to continue developing our model further.

Keywords—Sentiment analysis; Arabic Moroccan dialect; tweets; machine learning

I. INTRODUCTION

Natural Language Processing (NLP) attempts to make the machine capable of understanding and generating human language, whether it be written or audible [1] NLP is also one of the most important and challenging areas of artificial intelligence. It has many obstacles [2], especially in Arabic, which will be a topic we discuss later in this article. Sentiment Analysis (SA), or opinion mining, is the mathematical study of people's opinions, emotions, sentiments, ratings, and attitudes about products, services, organizations, individuals, issues, events, topics, and attributes [3]. SA has become one of the most prominent applications of NLP.

In recent years, Machine Learning (ML) and Deep Learning (DL) techniques has been widely used in various classification and prediction problems such as handwritten recognition [4]–[7], medical applications [8]–[13], social media analysis [14]–[16], etc. In particular, applying Arabic

sentiment analysis makes it possible to extract the public's opinion on one or many topics through tweets. The use of social media has become a major contributing factor in the performance of Arabic SA tools [17]. With the rise of social media platforms such as Twitter, Facebook, and Instagram, individuals can now express their opinions and feelings on various topics in real-time. This provides a rich source of data for researchers and businesses to understand the sentiment of the public.

Many researchers have opted to use different approaches like sentiment analysis and opinion mining to classify people's views and comments, ranging from negative, positive, and neutral perspectives. These techniques involve using Machine Learning (ML) algorithms to analyze large amounts of textual data to identify patterns and classify the sentiment expressed in the text. By using these techniques, researchers can understand the public's opinion on a variety of topics, such as politics, sports, entertainment, and social issues. One of the challenges of performing Arabic sentiment analysis is the complexity of the Arabic language. Arabic is a highly inflected language with many variations in grammar and vocabulary across different regions. This makes it difficult to develop accurate sentiment analysis tools that can accurately classify the sentiment of the text. However, recent advances in natural language processing and machine learning techniques have made it possible to overcome these challenges.

The following research is an attempt to apply sentiment analysis on Moroccan people's tweets in order to extract reactions from them by using local idioms and terms in their native dialect as a basis. With the increasing use of social media platforms such as Twitter in Morocco, it has become an important source of information for individuals, businesses, and governments. Therefore, there is a growing need for sentiment analysis to help these groups understand the opinions and emotions expressed by the public. Our goal is to develop a sentiment analysis (SA) model that can accurately classify Arabic Moroccan dialect tweets as negative, positive, or neutral. To achieve this, we gathered a large dataset of Moroccan Dialect Arabic (MDA) tweets from Twitter. However, collecting this dataset was not a straightforward task. Moroccan Arabic is a complex dialect that varies greatly from region to region, and there is no standard written form of the dialect. Therefore, we had to collect tweets that were written in

the dialect and filter them to collect various sorts of terminology to identify MDA efficiently.

To develop our SA model, we employed various techniques such as word recognition, named entity recognition, stop word removal, and stemming to preprocess the collected tweets. We also experimented with different feature extraction techniques such as n-grams and weighting schemes like Bag of Words (BOW) and Term Frequency-Inverse Document Frequency (TF-IDF) to construct effective features for our model. Additionally, we used word embedding to capture the semantic relationships between words and phrases. We employed several machine learning algorithms such as Naive Bayes, Random Forests, Support Vector Machines (SVM), Logistic Regression, and Long Short-Term Memory (LSTM) to classify the tweets based on their sentiment. These algorithms were trained on a portion of the dataset and tested on a separate set of tweets to evaluate their accuracy and effectiveness. Our SA model is unique in that it specifically focuses on Moroccan Dialect Arabic tweets and uses local idioms and terms to improve its accuracy. The results of our experiments are promising, as our model achieved high accuracy in classifying tweets as positive, negative, or neutral.

The structure of this article can be outlined as follows: In Section II, the related work on the topic is discussed to provide context and background for the reader. Section III is dedicated to explaining the methodology used in this research, including the steps taken to collect and prepare the data, the feature selection and extraction techniques used, and the machine learning algorithms applied for classification. The results of the experiment are presented in Section IV, where we discuss the performance of the different models. Finally, in Section V, we conclude the paper by summarizing the findings, discussing their implications, and suggesting directions for future research.

II. RELATED WORKS

We have noticed that the majority of the SA's research is focused on English. As a result, many high-quality frameworks and tools for English text are now available. For other languages, such as Arabic, however, research efforts are still needed to propose tools that are more refined. However, in recent years, many research topics continue to improve upon Arabic sentiment analysis and other dialectal Arabic sentiment analysis. Duwairi and Qarqaz build a Sentiment Analysis (SA) model for Arabic comments on social media platforms [18]. They used word bi-grams as features for representing the text in their model. They also evaluated the performance of different classifiers, including Support Vector Machine (SVM), Naive Bayes (NB), and K-Nearest Neighbour (KNN), with the use of term frequency (TF) and term frequency inverse document frequency (TF-IDF) weighting methods. Furthermore, Ayah Soufan [19] used text data from Twitter and GoodReads to conduct sentiment analysis for the Arabic language. He also worked on two different tasks: binary and multi-class categorization. He noted that the outcomes vary depending on the dataset and model employed. Modern Standard Arabic (MSA) and Arabic dialects differ on all "linguistic representation levels such as morphology, lexicon,

phonology, syntax, semantics, and pragmatics," which is why the results vary depending on the dataset.

In [20], the authors compare different types of ensemble methods. By simply using individual classifiers, on Arabic sentiment analysis, it became apparent that SVM was the better performer, by a decent margin. However, when compared to the performance of a combination of different classifiers, no individual classifier performed nearly as well.

Another work [21] by using NB classifier, described a method for automatically generating a corpus that can be used to train a multilingual sentiment classifier to classify tweets into positive and negative categories.

In [22], the authors collected and labeled a dataset of 17000 Tunisian dialect comments from Facebook, which they used to build a Tunisian dialect sentiment analysis (SA) system using three classification algorithms: support vector machine (SVM), Naive Bayes (NB), and Multi-Layer Perceptron (MLP). They found that their method outperformed models trained on other dialects or on a MSA dataset.

In the study [23], the authors proposed a framework that employs a variety of approaches and efficient models for both Arabic text pre-processing and ASA. In the case of ASA, their study revealed that Deep learning (DL) models are more efficient and accurate than ML (SVM, NB, and ME). In all of the following scenarios, DL models outperformed traditional models: when using unigrams, when using stop words, without stop words, when using stemming, a without stemming. This study demonstrated the significant accuracy and performance potential of DL for ASA.

In another work [24], the authors analyze the sentiment of 4625 comments written in MSA and multidialectal Arabic from Yahoo!-Maktoob using support vector machines (SVM) and naive Bayes (NB) classifiers, which are commonly used for sentiment classification. They tested the classifiers using both balanced and imbalanced datasets, but the results were unsatisfying as the highest accuracy achieved was 70%.

III. MATERIALS AND METHODS

A. Global Overview of the Proposed System

Fig. 1 is a plain summary of the approach we decided to adopt throughout our study.

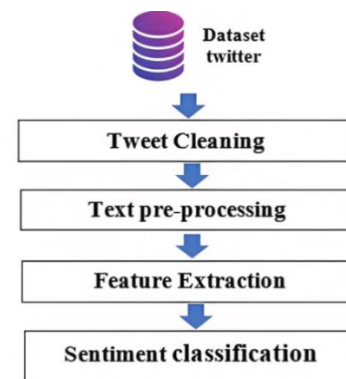


Fig. 1. Simplified Process for SA.

B. Moroccan Dialect Recognition Challenges

Darija is a Moroccan dialect (MD) of Arabic that is widely utilized by Moroccans, brand sites on social media, television programs, and Ads that tend to reach out to the public. It is influenced by various other languages, including Berber, French, Spanish, and Andalusian Arabic, and has its own distinct grammar, vocabulary, and pronunciation. However, Darija maintains its distinctive characteristics in terms of spelling, syntax, vocabulary, and pronunciation [25]. In recent times, the use of written forms of Darija has increased significantly across various platforms. Some difficulties still stand in the face processing Darija, some of which are:

- Domain Dependency [26]: Factors such as culture, context, and basic usage of the language heavily affect performance.
- Code Switching: The French-Spanish colonization of Morocco left a lasting effect on Darija's history, providing her both French and Spanish words/sentences. On top of that, Berber language, which around 40% of Moroccans speak fluently, managed to find a spot in the basic structure of the language. As a result, MSA, Berber, French, Spanish, and English can all be found in Darija writing [27].

- Romanized Arabic: There is no orthographic standard in Darija. It can be written in Arabic [28], Latin, or both (Arabizi). A combination of the two is often used on social media, with numbers being used as substitutes to inexistent letters in the Roman alphabet, such as welcome=Mer7ba = مرحبا, happy= fer7an=سعيد, sleep=n3ass=النوم.

However, the biggest challenge of them all is the limited resources, and the difficulty of extracting datasets.

C. Process of Moroccan Arabic Dialect Tweet for Sentiment Analysis

In this section, we present a Machine Learning (ML) process for SA conducted on twitter comments written in MDA. This process starts by cleaning tweets, pre-processing them and before classification, where we performed a features selection process aiming to reduce the dimensionality and improve the quality of our classification models. Finally, comes the evaluation step where the performance of our model is measured. Fig. 2 describes the proposed ML system of Arabic Tweet for sentiment analysis.

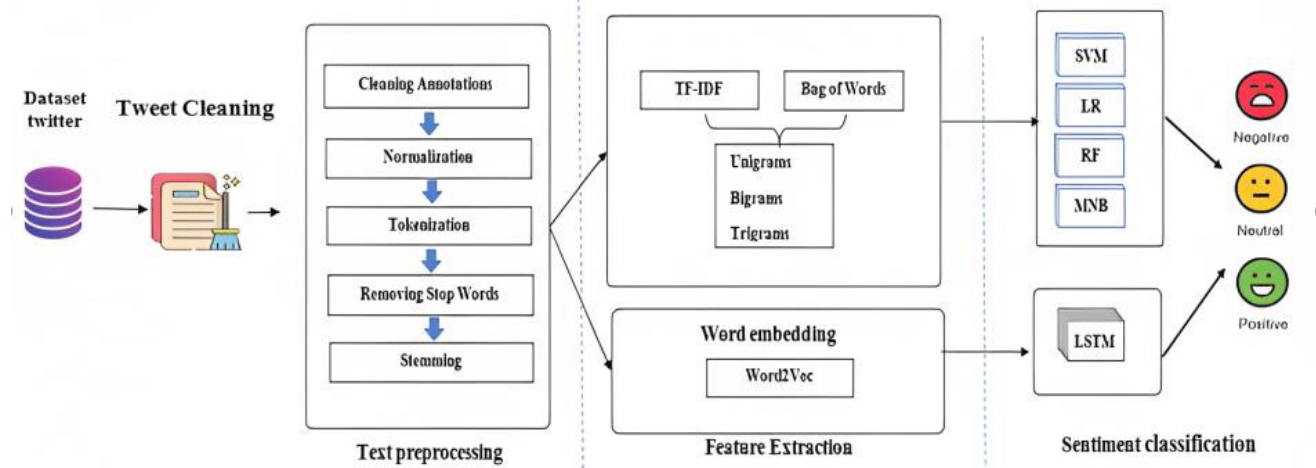


Fig. 2. Sentiment analysis: A ML approach.

D. Dataset Description

In our study, we used a dataset of SA for social media posts in Arabic dialect, publicly available from the Modeling, Simulation and Data Analysis (MSDA)¹. The data (tweets), scrapped from active users located in a predefined set of Arab countries consisting mainly of Lebanon, Algeria, Egypt, Tunisia and Morocco, got narrowed down to tweets relevant to Morocco only. The selected dataset contains 1605 positive tweet, 1620 negative tweet and 1630 neutral tweet. In Table I, we present some illustrative samples.

The word cloud schemes in Fig. 3, Fig. 4, and Fig. 5 present the density of words in relevance to the class from where it originates.

E. Data Pre-processing Steps

1) *Tweet cleaning*: In order to clean our dataset, we had to remove the tweets:

- Written in Roman alphabet.
- Those are spam or insults.
- Composed of a single word.

2) *Cleaning annotations*: In order to rend our tweets more functional, all “http/https” are removed, as well as special symbols (*, &, \$, %, -, _ , . , ! , > <). Whatever we remove, we replace with an empty space. We apply the same for the emoji's.

¹ https://msda.um6p.ma/msda_datasets

TABLE I. ANNOTATED SAMPLE.

Comment	English translation	Sentiment
لن نضيع اوقات اطفالنا بتعليم لهجة ما عندها فائدة و كافي ان كل طفولتهم تروح في التعليم نزيدو لهم لهجة تتعبهم و تمحي هويتهم	We will not waste our children's time by teaching them a dialect that's useless. It's enough that their time is consumed by education, and to top it all we add a dialect that would tire them and erase their identity.	Negative
هاد الولد ممثل المغرب في أكبر مسابقة ديال صناع المحتوى في العالم العربي.. وليوم هو في النهائية	This boy is the Moroccan representative in the biggest content creation contest in the Arabic world, and today he's a finalist.	Neutral
ترامب عين البروفيسور منصف السلاوي للإشراف على تطوير اللقاح , لمغاربة حصلو له مغربي ولا فواش نقولو أمريكي ولا بلجيكي	Trump appointed Prof. Monsif Essalaoui as a supervisor for vaccine development. Moroccans didn't know what to call him 'Moroccan', 'American, or 'Belgium'.	Positive



Fig. 3. Positive class word occurrence.



Fig. 4. Negative class word occurrence.



Fig. 5. Neutral class word occurrence.

3) *Tweets pre-processing*: In order to boost the performance of the SA process, we must perform several pre-processing steps on the collected tweets. These steps are as follows:

- Normalization: Normalization is necessary, as there is no universally accepted rule for the spelling of certain Arabic letters. We perform our normalization by

deleting all unnecessary spaces, and then replacing every un-normalized letter with its normalized version. As indicated in Table II, based on the Pyarabic library, normalization is as such:

- Tokenization: as an integral step in SA, reduces typographical variation of words. It is also necessary since it is required to use techniques such as Feature Extraction [29]. As the Arabic language is known to be hard to deal with, a dictionary of features that can transform words into feature vectors, or feature indexes, is a must. This way, the index of the word is linked to its frequency in the complete Dataset.

TABLE II. NORMALIZATION OF SOME ARABIC LETTERS

UnNormalized Arabic Letters	Normalized Letters
Hamza ء، ة، ة	ء
Alef ا، آ	ا
Lamalef ال، آل	ال
Yah ي	ي
Hah ه	ه
Wow_hamzah و	و

- Removing stop words: Not all stop words are actually helpful in understanding the full meaning of the tweet, which is why deleting some of them is an essential step. Deleting words like articles, prepositions, conjunctions, and pronouns that are overused and do not provide any information about the subject of the documents must be carefully done.
- Data Stemming and Lemmatization: Stemming is a process where, by removing prefixes and suffixes, a word is returned to its root form [30]. Similarly, lemmatization finds the basis of the word while taking into consideration its morphological nature; a meaning conserving lemma is then extracted. We opted to use the following stemmers in our study:
 - ISRI Stemmer [31] is a stemmer that uses the same sequence as other stemmers to derive the root-base of Arabic words². On the other hand, ISRI does not need to confirm the extracted root with a stored root dictionary, as the other stemmer does, because locating the correct root is not critical. ISRI eliminates two- and three-letter prefixes, normalizes Hamza, and eliminates connector letters such ("ثم").
 - Assem's Arabic Light stemmer is frequently used for a variety of text processing tasks, such as text categorization, information extraction, text summarization, and search engine indexing. It is important for researchers to carefully evaluate and compare different stemmers in order to select the one that is most suitable for their specific needs and context³. Table III present examples of executing tasks.

2 <https://github.com/baraaayoubashi/ISRI-Java>

3 https://github.com/assem-ch/arabicstemmer_pythonlibrary

TABLE III. EXAMPLE OF EXECUTING THE PROCESSING TASKS ON A COMMENT

Task	Result	Simulated English result
Original text	تبارك الله مستوى وصل الطالب المغربي ما فهمتش واش خارجيين إصلاحية تحية لجنبييل الذهبي	How disappointiiiiii is the current level of Moroccan students I do not know if they graduated high school or juviiiii. Salute to the golden geneeeration.
Cleaning	تبارك الله مستوى وصل الطالب المغربي ما فهمتش واش خارجيين ثانوية إصلاحية تحية لجنبييل الذهبي	How disappointiiiiii is the current level of Moroccan students. I do not know if they graduated high school or juviiiii. Salute to the golden geneeeration.
Normalizing	تبارك الله مستوى وصل الطالب المغربي ما فهمتش واش خارجيين ثانوية إصلاحية تحية لجيل الذهبي	How disappointiiiiii is the current level of Moroccan students. I do not know if they graduated high school or juviiiii. Salute to the golden geneeeration.
Tokenization	[تبارك 'الله' 'مستوى' 'وصل' 'ما' 'الطالب' 'المغربي' 'فهم' 'واش' 'ثانوية' 'إصلاحية' 'تحية' 'الجيل' 'الذهبي']	['How', 'disappointing', 'is', 'the', 'current', 'level', 'of', 'Moroccan', 'students', 'I', 'do', 'not', 'know', 'if', 'they', 'graduated', 'high', 'school', 'or', 'juvie.', 'Salute', 'to', 'the', 'golden', 'generation']
Stop words Removal	[بركة 'الله' 'مستوى' 'وصل' 'الطالب' 'المغربي' 'فهم' 'واش' 'ثانوية' 'إصلاحية' 'تحية' 'الجيل' 'الذهب']	['disappointing', 'current', 'level', 'Moroccan', 'students', 'graduated', 'high', 'school', 'juvie', 'Salute', 'golden', 'generation']
Stemming	['مغرب' 'طلب' 'صل' 'مستوى' 'بركة' 'إصلاح' 'ثان' 'خارج' 'الذهب' 'جيل']	['disappoint', 'current', 'level', 'Moroccan', 'student', 'graduat', 'high', 'school', 'juvi', 'Salut', 'golden', 'gener']

- Tashaphyne is a stemmer tool specifically designed for analyzing the sentiment of Arabic roots [32]. It has been shown to be effective at mapping Arabic words to their basic roots for sentiment analysis tasks, resulting in significant improvements compared to baseline performance [33].

F. Features Extraction

N-gram is a traditional method for identifying formal terms in tweets that takes into account the occurrences of N-words in the tweet. The value of n is used to refer to N-grams of larger sizes [34]. During our research, we used unigrams, bigrams, and trigrams to get the best results. Table IV is the perfect example on the results obtained by applying this method on our tweets.

We employed two alternative feature extraction algorithms: Bag of Words [35] generates a vector that represents the frequency of occurrences of words. The Bow model, frequently used in text processing to classify and recognize documents, is known for simplicity and effectiveness. The TF-IDF was created to address the limitations of the more basic strategy of counting each term's occurrences, in which highly common terms end up with very big values and uncommon words (which are normally a good discriminant between texts) can be buried in the noise. The TF-IDF approach is used to weight terms based on their relevance within the document and corpus [36]. Words with a high TF-IDF score are those that appear frequently in a document but not across the corpus.

TABLE IV. FEATURES EXTRACTION

Task	Result	Simulated English result
Cleaned text	['مغرب' 'طلب' 'صل' 'مستوى' 'بركة' 'إصلاح' 'ثان' 'خارج' 'الذهب' 'جيل']	['disappoint', 'current', 'level', 'Moroccan', 'student', 'graduat', 'high', 'school', 'juvi', 'Salut', 'golden', 'gener']
Uni-gram n=1	['مستوى' 'صل' 'طلب' 'مغرب' 'إصلاح' 'خارج' 'بركة' 'جيل']	['disappoint', 'current', 'level', 'Moroccan', 'student', 'graduat', 'high', 'school', 'juvi', 'Salut', 'golden', 'gener']
Bi-gram n=2	['مستوى صل' 'صل طلب' 'طلب' 'مغرب' 'مغرب' 'إصلاح' 'إصلاح بركة' 'خارج' 'خارج' 'جيل']	['disappoint current', 'current level', 'level moroccan', 'moroccan student', 'student graduat', 'graduat high', 'high school', 'school juvi', 'juvi salut', 'salut golden', 'golden gener']
Tri-gram n=3	['مستوى صل طلب' 'صل طلب' 'طلب' 'مغرب' 'مغرب' 'إصلاح' 'إصلاح بركة' 'خارج' 'خارج' 'جيل']	['disappoint current level', 'current level moroccan', 'level moroccan student', 'moroccan student graduat', 'student graduat high', 'graduat high school', 'high school juvi', 'school juvi salut', 'juvi salut golden', 'salut golden gener']

Term Frequency is calculated by dividing the total number of words in a document by the number of times each term appears in the document.

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}} \quad (1)$$

Inverse Document Frequency is the log of the number of documents divided by the number of documents that contain the word.

$$w_{i,j} = tf_{i,j} \times \log\left(\frac{N}{df_i}\right) \quad (2)$$

t_{ij} = Occurrences of i in j .

df_i = Documents containing i .

N = Total number of documents.

This increases the weight of rare words across all documents in the corpus. Note that when we compute the TF-IDF for every word in every document of a corpus, it will form a matrix of shape (documents * vocabulary).

Word embedding methods turn words into digital vectors. These vectors include more or less information about the semantics and syntax of the word depending on the model employed and the context in which it was used [37]. There are numerous word embedding strategies available.

Word2vec is a renowned word-embedding model developed at Google in 2013. It incorporates a neural network layer to either predict adjacent words to the target word (context) in the case of the skip gram architecture, or the word from its neighbors in the case of the CBOW (Continuous bag of words). Word2vec's input and output are a one-time encoding of the dataset's vocabulary words. During training, a window moves through the corpus [38], training the neural network to predict surrounding words or a target word for each word by assigning a probability to the words. After training,

the network layer is the vector representation of a word. In our case, we implemented a Wikipedia-made word2vec model for Arabic language.

G. The Used Classifiers

The data was classified using four supervised machine-learning algorithms: Naive Support Vector Machine classifier (SVM), Multinomial Naive Bayes (MNB), Logistic regression (LR), Random forest (RF), and one DL algorithm: Long Short-Term Memory (LSTM). We aim to answer the following question: Can ensemble learning (combining different classification algorithms) improve Arabic sentiment classification?

In what follows, we explain those algorithms:

1) *Support vector machine*: Support vector machine is a supervised learning technique commonly used to classify jobs. In a high-dimensional space, it is extremely efficient. It also works well when the number of dimensions is more than the number of data. SVM is a discriminative classifier whose basic principle is to construct decision boundaries that distinguish between a set of objects belonging to various classes [39].

2) *Multinomial Naive Bayes (MNB)*: Multinomial Naive Bayes is a supervised machine-learning algorithm that relies on annotated data for training. It is based on the Bayes theorem and is well suited for high-dimensional input data like text. It is a generative model that relies on conditional probabilities and assumes that the features are conditionally independent [40]. Despite its simplicity, it is a very effective algorithm and can outperform more complex classifiers on short datasets. It is also relatively robust, easy to implement, and fast to run.

3) *Logistic regression*: The statistical model of logistic regression is used to investigate the associations between a set of qualitative variables X and a qualitative variable Y. A logistic function is used as a link function in the generalized linear model. The optimization of the regression coefficients in a logistic regression model can also be used to forecast whether an event will occur (value of 1) or not (value of 0). This outcome is always between 0 and 1 [41]. The event is more likely to happen if the anticipated value is above a certain threshold, but not if it is below the same threshold. The hypothesis representation of logistic regression defined as follow:

$$h\theta(X) = \frac{1}{1+e^{-(\beta_0+\beta_1X)}} \quad (3)$$

The cost function of LR is defined as follow:

$$J(\theta) = \frac{1}{M} \sum_{i=1}^M \text{cost}(h\theta(x^{(i)}), y^{(i)}) \quad (4)$$

With M is the size of training set.

4) *Random forest*: Random forest is a consensus approach for solving regression and classification problems in supervised machine learning (ML). Each random forest is

made up of several decision trees that work together to provide a single prediction [42].

5) *Long short-term memory*: Long short-term memory (LSTM) networks are a type of recurrent neural networks (RNN) that have a longer memory than traditional RNNs. They are well suited for learning from large, complex data sets that have long delays between important events. LSTM units are used to build the layers of an RNN, which is then called an LSTM network. LSTMs allow RNNs to retain information over a long period of time [43] by using a memory cell that is similar to computer memory, where it can read, write, and delete information as needed. This memory cell is gated, meaning that it decides whether to store or delete information based on its perceived importance, as determined by learned weights. There are three gates in an LSTM: an input gate, a forget gate, and an output gate. Fig. 6 illustrates a LSTM memory cell structure. These gates control whether new input is accepted (input gate), whether information is suppressed because it is not important (forget gate), or whether it is allowed to influence the output at the current time step (output gate) [44]. The importance of each piece of information is determined by the learned weights of the LSTM, which are updated as the algorithm learns over time.

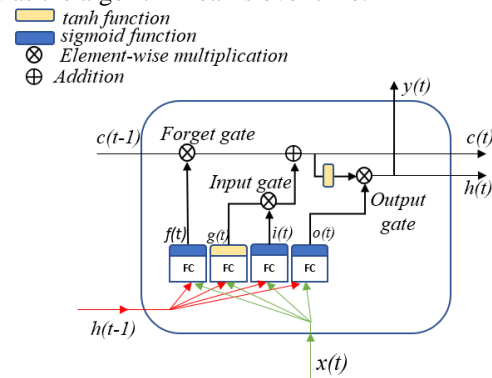


Fig. 6. Single LSTM memory cell.

IV. RESULTS AND DISCUSSION

A. Confusion Matrix and Performance Evaluation Measures

A confusion matrix, also known as a contingency table, is a tool used to evaluate the accuracy of a machine learning model's predictions in classification problems. It compares the model's predictions to reality and shows how often the predictions are correct. To understand how a confusion matrix works, it is important to familiarize oneself with the four key terms: true positive (TP), true negative (TN), false positive (FP), and false negative (FN). These terms represent the different outcomes of a prediction and are defined as follows:

- **TP (True Positives)**: the cases where the prediction is positive, and where the actual value is actually positive. Example: the doctor tells you that you are pregnant, and you are indeed pregnant.
- **TN (True Negatives)**: the cases where the prediction is negative, and where the actual value is actually

negative. Example: the doctor tells you that you are not pregnant, and you are indeed not pregnant.

- FP (False Positive): Cases where the prediction is positive, but the actual value is negative. Example: the doctor tells you that you are pregnant, but you are not pregnant.
- FN (False Negative): Cases where the prediction is negative, but the actual value is positive. Example: the doctor tells you that you are not pregnant, but you are pregnant. Various measures can be derived from a confusion matrix. In this work, we utilized four scoring performance metrics: accuracy, precision, recall, and F1 score. The following equations provide these metrics:

$$F_1 \text{ score} = \frac{2 * (\text{precision} \times \text{recall})}{\text{precision} + \text{recall}} \quad (5)$$

$$\text{Recall} = \frac{TP}{FN+TP} \quad (6)$$

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (7)$$

$$\text{Precision} = \frac{TP}{TP+FP} \quad (8)$$

B. Training of ML Models Results

Our implementation and training of the model heavily relied on the Scikit-Learn library. Vectorization, pipeline and training were imported from its sub libraries. Similarly, our different scripts (based on SVM, Random Forest, Naïve Bayes,

and Logistic regression) were applied and tuned using this library. GridSearchCV adjusted the parameters, and followed up with training and comparison; for LSTM, it was the TensorFlow library. We also used Google Colab platform to train our models.

C. Testing of ML Models Results

From Table V we are able to gauge the average accuracy obtained by using different algorithms on top of two separate vectorization methods (BOW, TF-IDF), which is in and of itself topped off with stemming algorithms, including the ISRI Stemmer, Assem's Arabic Light Stemmer and Tashafyn light Stemmer. Several tests were carried out on the dataset in order to assess the influence of stemming after it had been pre-processed (without undergoing stemming). Based on the outcomes of these tests, we can safely deduce that the SVM algorithm is the highest performer, boasting an accuracy of 0.68. Combining ISRI Stemmer, TF-IDF, Uni-gam + Bi-gam, and the SVM classifier resulted in this accuracy. However, when looking at the Tashafyn light stemmer system, we obtained the highest accuracy after using a mix of Bag of Words, Uni-gam + Bi-gam + Tri-Gam, and the SVM algorithm once again. This accuracy clocked at 0.678. As we compare these results with those obtained when using no Stemming, we notice that stemming proved itself an indispensable method that reliably improves performance and accuracy.

TABLE V. CLASSIFICATION OF EXPERIMENTATION RESULTS

Stemming Mechanism	Classifier	TF-IDF			Bag of Words			Word Embedding
		Uni-gram	1g+2g	1g+2g+3g	Uni-gram	1g+2g	1g+2g+3g	
No-stemming	LR	63.00%	63.00%	65.00%	64.00%	64.00%	65.00%	-
	RF	59.00%	58.00%	57.00%	59.00%	57.00%	57.00%	-
	NB	60.00%	59.00%	58.00%	60.00%	59.00%	58.00%	-
	SVM	64.00%	65.00%	64.00%	63.00%	64.00%	64.00%	-
	LSTM	-	-	-	-	-	-	61.50%
Tashafyn Light Stemmer	LR	66.39%	66.59%	66.80%	66.39%	66.25%	66.00%	-
	RF	61.51%	61.30%	60.14%	63.16%	61.30%	61.70%	-
	NB	63.78%	63.51%	63.23%	63.78%	63.30%	64.00%	-
	SVM	66.39%	68.52%	67.42%	65.49%	67.14%	67.80%	-
	LSTM	-	-	-	-	-	-	63.00%
ISRI Stemmer	LR	66.87%	67.36%	67.14%	66.87%	66.59%	66.94%	-
	RF	63.23%	62.34%	62.40%	63.57%	63.09%	61.85%	-
	NB	63.37%	62.82%	62.34%	63.37%	62.82%	62.34%	-
	SVM	67.62%	68.59%	68.27%	67.88%	67.56%	67.13%	-
	LSTM	-	-	-	-	-	-	63.90%
Assem's Arabic Light Stemmer	LR	64.05%	65.01%	65.15%	64.06%	64.95%	65.22%	-
	RF	63.24%	61.37%	60.82%	63.30%	61.51%	60.93%	-
	NB	62.54%	61.85%	61.37%	62.54%	61.58%	61.37%	-
	SVM	65.43%	65.15%	65.08%	63.98%	63.36%	64.88%	-
	LSTM	-	-	-	-	-	-	62.00%

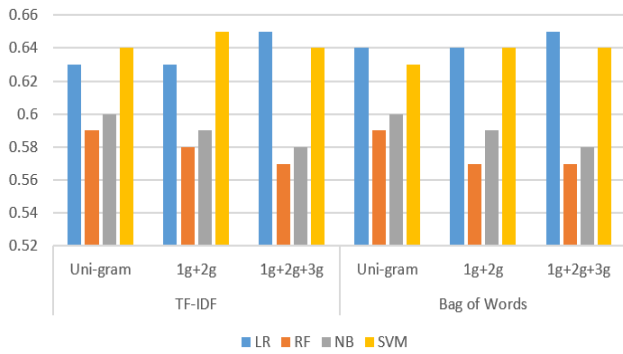


Fig. 7. Classifiers' accuracy with no stemmers.

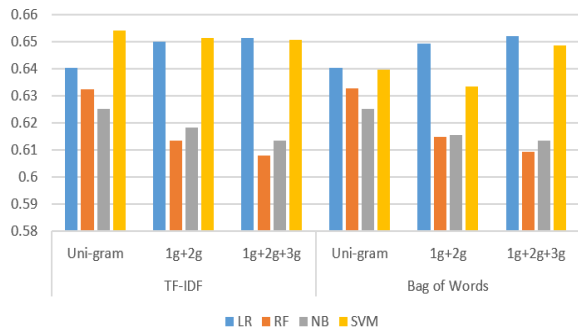


Fig. 8. Classifiers' accuracy with assem's arabic light stemmer.

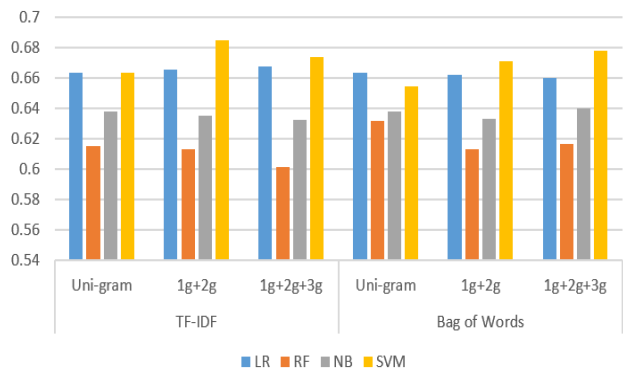


Fig. 9. Classifiers' accuracy with tashafyn light stemmers.

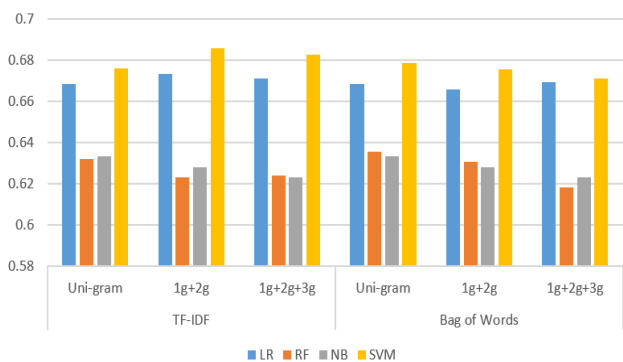


Fig. 10. Classifiers' accuracy with ISRI stemmers.

Fig. 7, Fig. 8, Fig. 9 and Fig. 10, which managed to better visualize the performance spread using the different stemmers, show a clear pattern where SVM and LR are the top performers even when taking into account the different feature extraction methods implemented (TF-IDF and Bag of Words). We managed to push the SVM accuracy all the way to 68.59% using the ISRI Stemmer as well as the combination of TF-IDF and 1g-2g feature extraction methods (these same methods also resulted in the highest value for LR: 67.36%). It should also be noted that despite implementing three different types of stemmers, we do not see very strong variations in the accuracy score across the board. This is largely because these custom-made stemmers for the Arabic language still leave a lot to be desired in terms of efficiency and performance.

Table VI summarizes our top scorer's (ISRI Stemmer+ (Uni-gram+Bigram) + SVM+TD-IDF (feature extraction)) Performance evaluation measures- the Precision, Recall, and F1-score for each one of our selected sentiments (positive, negative, or neutral). We notice that our model is extremely accurate when classifying neutral sentiments, as it has the highest precision score of 0.74.

TABLE VI. PERFORMANCE EVALUATION METRICS

Model		Precision	Recall	F1-score
ISRI Stemmer +(Uni-gram+Bigram)+SVM+TD-IDF (feature extraction)	neg	0.64	0.64	0.64
	neu	0.74	0.74	0.74
	pos	0.66	0.66	0.66

The confusion matrix of the model that had better performance in the detection sentiment analysis is shown in Fig. 11. On the diagonal, we can see 260 neutral sentiments were correctly predicted as neutral, and 198 positive sentiments were also correctly predicted as positive, and the same can be said for the 168 negative sentiment. On the other hand, it is obvious that the outliers of the matrix are heavy on numbers, as the maximum that a model should have wrongly predicted should not exceed 5-10.

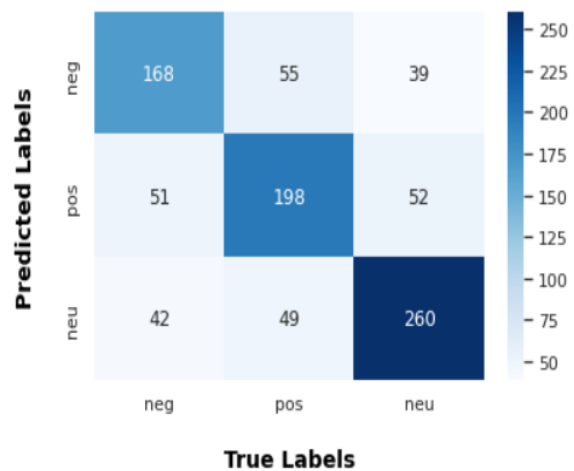


Fig. 11. Confusion matrix of the model (ISRI Stemmer+ (Uni-gram+Bigram) + SVM+TD-IDF (feature extraction)).

For LSTM model that we adapted for our research, we managed to achieve the results described in Table VII. We immediately notice that the Accuracy, Recall, and F1-score all receive a slight bump in their values as we apply the different Stemmers. However, for our Precision score, our max value remains the one where no Stemmer is applied.

TABLE VII. LSTM'S MODELS PERFORMANCE

Stemmers	Accuracy	Recall	F1-score	Precision
No-stemming	61.50%	68.57%	67.29%	66.06%
Tashafyn Light Stemmer	63.00%	72.83%	68.97%	65.49%
ISRI Stemmer	63.90%	73.10%	69.25%	65.79%
Assem's Arabic Light Stemmer	62.00%	71.70%	68.30%	65.22%

D. Discussion

The ROC curve is an evaluation metric commonly used to plot the True Positive Rate vs. the False Positive Rate. While this metric is mainly used for classification problems that are binary, we managed to push it in order to encompass out multiclass classification problem, all by using the one vs. Rest technique (We calculate the AUC-ROC curve by considering each label independently). In Fig. 12, based on the results from the SVM model, we see how our Neutral label is our best performer, while the Negative and Positive labels are lagging slightly behind. While the Negative and Positive labels intertwine at different thresholds, the Neutral label maintains an advantage throughout. Our results are far from perfect, but they are not disappointing as there is still much to improve upon our study.

The support vector machine (SVM) algorithm outperformed all other classifiers on all datasets, showing a

significant difference in performance. SVM is a popular choice for sentiment analysis studies due to its various advantages, such as its ability to handle high-dimensional spaces efficiently and its robustness when working with a sparse set of samples. It also considers all features relevant.

As one can tell from Table VIII, LSTM was nowhere near as powerfully performant as the SVM algorithm. Though Word Embedding is said to be more advanced compared to the bag of words method, in our case where we are processing the Darija Language, it is actually held back by the fact that the Dictionary used in the Word2Vec method is based on the Arabic language and not Darija. This means that when classifying Darija words, many are found to be unclassifiable.

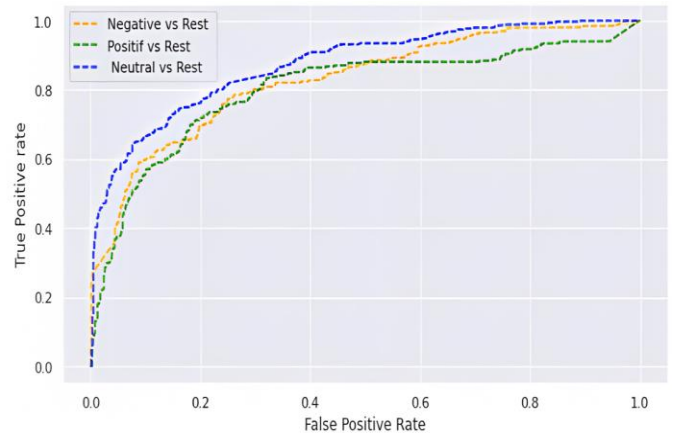


Fig. 12. ROC curves one vs rest.

TABLE VIII. SUMMARY OF ALL THE EXPERIMENTAL FINDINGS PROVIDED IN THIS RESEARCH

Reference	Dataset	Feature Representation	Classification Algorithm	Best Accuracy
[18]	Scrapping dataset	(TF) and term frequency inverse document frequency (TF-IDF)	Supervised Classification(SVM, NB, and K-nearest neighbour (KNN)	67,19%
[19]	QRCI , ArTwitter, ASTD Comb,LABR	Word embedding	SVM,NMB,LSTM,CNN	92,4 %
[20]	MSAC (Moroccan Sentiment Analysis Corpus)	Bag of words	Naive Bayes classifier (NB), Support Vector Machine classifier (SVM), Maximum Entropy (ME)	82.5 %
[22]	TSAC (Tunisian Sentiment Analysis Corpus)	None	(SVM) and Naive Bayes (NB)	76,41%
[24]	Yahoo!-Maktoob	None	SVM, NB	68.2%
This work	dataset of Sentiment Analysis for social media posts in Arabic dialect from MSDA	(TF) and term frequency inverse document frequency (TF-IDF), Word embedding Bag of words, N-gram	LR, RF, MNB, SVM, LSTM	68,59%

V. CONCLUSION AND PERSPECTIVES

In this study, we went through the steps and methods that allowed us to test the different approaches of sentiment analysis' performance and efficiency. Our Dataset, consisting of 4855 tweets/comments split into three balanced groups of negative, positive and neutral sentiments, underwent multiple thorough processes in order to take advantage of it. We began by incorporating a variety of pre-processing techniques (stemming, normalization, tokenization, stop words, etc.) to improve Sentiment Analysis of Moroccan Tweets. Then, in the

hopes of drawing out the maximum possible accuracy, five classification algorithms: NB, LR, RF, SVM, and LSTM, were combined as an application of the Ensemble Method. Finally, we delved into a comparison of three types of FE method: BOW, TF-IDF and WE. The results for individual classifiers made it clear that SVM performed on a higher level compared to the other algorithms. There was also a noticeable performance boost when it came to using Stemmers vs. using no stemmers; the only drawback being an increase in the computational time. Our next step is to use other DL models such as BERT

We plan to further improve our SA model by incorporating more advanced deep learning techniques, such as BERT, and exploring other FE methods that may provide better accuracy. We also aim to expand our dataset to include more diverse sources of Moroccan Arabic language and include more topics to enhance the model's performance on a wider range of subjects. Additionally, we plan to extend our research to cover other Arabic dialects to provide more comprehensive sentiment analysis for the Arabic language. Finally, we hope to collaborate with other researchers in the field to develop a standardized evaluation framework for Arabic sentiment analysis to facilitate comparison and benchmarking between different approaches and models.

REFERENCES

- [1] R. Mihalcea, H. Liu, and H. Lieberman, "NLP (Natural Language Processing) for NLP (Natural Language Programming)," in *Computational Linguistics and Intelligent Text Processing*, vol. 3878, A. Gelbukh, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 319–330. doi: 10.1007/11671299_34.
- [2] S. Elkateb, W. Black, and D. Farwell, "Arabic WordNet and the challenges of Arabic," p. 10.
- [3] L. Yue, W. Chen, X. Li, W. Zuo, and M. Yin, "A survey of sentiment analysis in social media," *Knowl Inf Syst*, vol. 60, no. 2, pp. 617–663, Aug. 2019, doi: 10.1007/s10115-018-1236-4.
- [4] S. Hamida, B. Cherradi, O. El Gannour, O. Terrada, A. Raihani, and H. Ouajji, "New Database of French Computer Science Words Handwritten Vocabulary," in *2021 International Congress of Advanced Technology and Engineering (ICOTEN)*, Taiz, Yemen, Jul. 2021, pp. 1–5. doi: 10.1109/ICOTEN52080.2021.9493438.
- [5] S. Hamida, B. Cherradi, and H. Ouajji, "Handwritten Arabic Words Recognition System Based on HOG and Gabor Filter Descriptors," in *2020 1st International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET)*, Meknes, Morocco, Apr. 2020, pp. 1–4. doi: 10.1109/IRASET48871.2020.9092067.
- [6] S. Hamida, B. Cherradi, O. Terrada, A. Raihani, H. Ouajji, and S. Laghmati, "A Novel Feature Extraction System for Cursive Word Vocabulary Recognition using Local Features Descriptors and Gabor Filter," in *2020 3rd International Conference on Advanced Communication Technologies and Networking (CommNet)*, Marrakech, Morocco, Sep. 2020, pp. 1–7. doi: 10.1109/CommNet49926.2020.9199642.
- [7] S. Aqab and M. Usman, "Handwriting Recognition using Artificial Intelligence Neural Network and Image Processing," *IJACSA*, vol. 11, no. 7, 2020, doi: 10.14569/IJACSA.2020.0110719.
- [8] B. Cherradi, O. Terrada, A. Ouhmida, S. Hamida, A. Raihani, and O. Bouattane, "Computer-Aided Diagnosis System for Early Prediction of Atherosclerosis using Machine Learning and K-fold cross-validation," in *2021 International Congress of Advanced Technology and Engineering (ICOTEN)*, Taiz, Yemen, Jul. 2021, pp. 1–9. doi: 10.1109/ICOTEN52080.2021.9493524.
- [9] O. El Gannour et al., "Concatenation of Pre-Trained Convolutional Neural Networks for Enhanced COVID-19 Screening Using Transfer Learning Technique," *Electronics*, vol. 11, no. 1, p. 103, Dec. 2021, doi: 10.3390/electronics11010103.
- [10] N. A. Ali, A. E. abbassi, and B. Cherradi, "The performances of iterative type-2 fuzzy C-mean on GPU for image segmentation," *J Supercomput*, vol. 78, no. 2, pp. 1583–1601, Feb. 2022, doi: 10.1007/s11227-021-03928-9.
- [11] H. Moujahid et al., "Combining CNN and Grad-Cam for COVID-19 Disease Prediction and Visual Explanation," *Intelligent Automation & Soft Computing*, vol. 32, no. 2, pp. 723–745, 2022, doi: 10.32604/iasc.2022.022179.
- [12] O. Daanouni, B. Cherradi, and A. Tmiri, "Predicting diabetes diseases using mixed data and supervised machine learning algorithms," in *Proceedings of the 4th International Conference on Smart City Applications*, Casablanca Morocco, Oct. 2019, pp. 1–6. doi: 10.1145/3368756.3369072.
- [13] S. Hamida, O. El Gannour, B. Cherradi, A. Raihani, H. Moujahid, and H. Ouajji, "A Novel COVID-19 Diagnosis Support System Using the Stacking Approach and Transfer Learning Technique on Chest X-Ray Images," *Journal of Healthcare Engineering*, vol. 2021, pp. 1–17, Nov. 2021, doi: 10.1155/2021/9437538.
- [14] D. Irawan, D. I. Sensuse, P. A. W. Putro, and A. Prasetyo, "Public Response to the Legalization of The Criminal Code Bill with Twitter Data Sentiment Analysis," *IJACSA*, vol. 14, no. 2, 2023, doi: 10.14569/IJACSA.2023.0140236.
- [15] M. Amine, H. Ait, R. Moulouki, S. Nkiri, and M. Azouazi, "An Improved Social Media Analysis on 3 Layers: A Real Time Enhanced Recommendation System," *ijacsa*, vol. 9, no. 2, 2018, doi: 10.14569/IJACSA.2018.090234.
- [16] M.-A. Ouassil, B. Cherradi, S. Hamida, M. Errami, O. E. Gannour, and A. Raihani, "A Fake News Detection System based on Combination of Word Embedded Techniques and Hybrid Deep Learning Model," *IJACSA*, vol. 13, no. 10, 2022, doi: 10.14569/IJACSA.2022.0131061.
- [17] M. Maghfour and A. Elouardighi, "Standard and Dialectal Arabic Text Classification for Sentiment Analysis," in *Model and Data Engineering*, vol. 11163, E. H. Abdelwahed, L. Bellatreche, M. Golfarelli, D. Méry, and C. Odonez, Eds. Cham: Springer International Publishing, 2018, pp. 282–291. doi: 10.1007/978-3-030-00856-7_18.
- [18] R. M. Duwairi and I. Qarqaz, "Arabic Sentiment Analysis Using Supervised Classification," in *2014 International Conference on Future Internet of Things and Cloud*, Barcelona, Spain, Aug. 2014, pp. 579–583. doi: 10.1109/FiCloud.2014.100.
- [19] A. Soufan, "Deep Learning for Sentiment Analysis of Arabic Text," in *Proceedings of the ArabWIC 6th Annual International Conference Research Track on - ArabWIC 2019*, Rabat, Morocco, 2019, pp. 1–8. doi: 10.1145/3333165.3333185.
- [20] A. Oussous, A. A. Lahcen, and S. Belfkih, "Improving Sentiment Analysis of Moroccan Tweets Using Ensemble Learning," in *Big Data, Cloud and Applications*, vol. 872, Y. Tabii, M. Lazaar, M. Al Achhab, and N. Enneya, Eds. Cham: Springer International Publishing, 2018, pp. 91–104. doi: 10.1007/978-3-319-96292-4_8.
- [21] A. E. Abdouli, L. Hassouni, and H. Anoun, "Sentiment Analysis of Moroccan Tweets using Naive Bayes Algorithm," vol. 15, no. 12, p. 11, 2017.
- [22] S. Medhaffar, F. Bougares, Y. Estève, and L. Hadrach-Belguith, "Sentiment Analysis of Tunisian Dialects: Linguistic Ressources and Experiments," in *Proceedings of the Third Arabic Natural Language Processing Workshop*, Valencia, Spain, 2017, pp. 55–61. doi: 10.18653/v1/W17-1307.
- [23] A. Oussous, F.-Z. Benjelloun, A. A. Lahcen, and S. Belfkih, "ASA: A framework for Arabic sentiment analysis," *Journal of Information Science*, vol. 46, no. 4, pp. 544–559, Aug. 2020, doi: 10.1177/0165551519849516.
- [24] M. N. Al-Kabi, N. A. Abdulla, and M. Al-Ayyoub, "An analytical study of Arabic sentiments: Maktoob case study," in *8th International Conference for Internet Technology and Secured Transactions (ICITST-2013)*, London, United Kingdom, Dec. 2013, pp. 89–94. doi: 10.1109/ICITST.2013.6750168.
- [25] S. Harrat, K. Meftouh, and K. Smaïli, "Maghrebi Arabic dialect processing: an overview," vol. 1, no. 1, p. 8, 2018.
- [26] A. Lighthart, C. Catal, and B. Tekinerdogan, "Systematic reviews in sentiment analysis: a tertiary study," *Artif Intell Rev*, vol. 54, no. 7, pp. 4997–5053, Oct. 2021, doi: 10.1007/s10462-021-09973-3.
- [27] Y. Samih and W. Maier, "An Arabic-Moroccan Darija Code-Switched Corpus," p. 6.
- [28] R. Tachicart, K. Bouzoubaa, and H. Jaafar, "Lexical differences and similarities between Moroccan dialect and Arabic," in *2016 4th IEEE International Colloquium on Information Science and Technology (CiSt)*, Tangier, Morocco, Oct. 2016, pp. 331–337. doi: 10.1109/CIST.2016.7805066.
- [29] J. J. Webster and C. Kit, "Tokenization as the initial phase in NLP," in *Proceedings of the 14th conference on Computational linguistics -*, Nantes, France, 1992, vol. 4, p. 1106. doi: 10.3115/992424.992434.

- [30] H. A. Almuzaini and A. M. Azmi, "Impact of Stemming and Word Embedding on Deep Learning-Based Arabic Text Categorization," *IEEE Access*, vol. 8, pp. 127913–127928, 2020, doi: 10.1109/ACCESS.2020.3009217.
- [31] K. Taghva, R. Elkhoury, and J. Coombs, "Arabic stemming without a root dictionary," in *International Conference on Information Technology: Coding and Computing (ITCC'05) - Volume II*, Las Vegas, NV, USA, 2005, pp. 152–157 Vol. 1. doi: 10.1109/ITCC.2005.90.
- [32] "T. Zerrouki, (2010). 'Tashaphyne, Arabic Light Stemmer/Segment'. <http://tashaphyne.sourceforge.net>."
- [33] Y. A. Alhaj, J. Xiang, D. Zhao, M. A. A. Al-Qaness, M. Abd Elaziz, and A. Dahou, "A Study of the Effects of Stemming Strategies on Arabic Document Classification," *IEEE Access*, vol. 7, pp. 32664–32671, 2019, doi: 10.1109/ACCESS.2019.2903331.
- [34] A. Dey, M. Jenamani, and J. J. Thakkar, "Senti-N-Gram: An n -gram lexicon for sentiment analysis," *Expert Systems with Applications*, vol. 103, pp. 92–105, Aug. 2018, doi: 10.1016/j.eswa.2018.03.004.
- [35] D. Mohey, "Enhancement Bag-of-Words Model for Solving the Challenges of Sentiment Analysis," *ijacsa*, vol. 7, no. 1, 2016, doi: 10.14569/IJACSA.2016.070134.
- [36] B. Das and S. Chakraborty, "An Improved Text Sentiment Classification Model Using TF-IDF and Next Word Negation," 2018, doi: 10.48550/ARXIV.1806.06407.
- [37] B. Oscar Deho, A. William Agangiba, L. Felix Aryeh, and A. Jeffery Ansah, "Sentiment Analysis with Word Embedding," in *2018 IEEE 7th International Conference on Adaptive Science & Technology (ICAST)*, Accra, Aug. 2018, pp. 1–4. doi: 10.1109/ICASTECH.2018.8506717.
- [38] Y. Wang, Z. Li, J. Liu, Z. He, Y. Huang, and D. Li, "Word Vector Modeling for Sentiment Analysis of Product Reviews," in *Natural Language Processing and Chinese Computing*, vol. 496, C. Zong, J.-Y. Nie, D. Zhao, and Y. Feng, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 168–180. doi: 10.1007/978-3-662-45924-9_16.
- [39] D. Dqg, "Sentiment Analysis Using Support Vector Machine," p. 5, 2019.
- [40] M. Abbas, A. Kamran, Memon, A. A. Jamali, Saleemullah Memon, and Anees Ahmed, "Multinomial Naive Bayes Classification Model for Sentiment Analysis," 2019, doi: 10.13140/RG.2.2.30021.40169.
- [41] W. P. Ramadhan, S. T. M. T. Astri Novianty, and S. T. M. T. Casi Setianingsih, "Sentiment analysis using multinomial logistic regression," in *2017 International Conference on Control, Electronics, Renewable Energy and Communications (ICCREC)*, Yogyakarta, Sep. 2017, pp. 46–49. doi: 10.1109/ICCEREC.2017.8226700.
- [42] P. Karthika, R. Murugeswari, and R. Manoranjithem, "Sentiment Analysis of Social Media Network Using Random Forest Algorithm," in *2019 IEEE International Conference on Intelligent Techniques in Control, Optimization and Signal Processing (INCOS)*, Tamilnadu, India, Apr. 2019, pp. 1–5. doi: 10.1109/INCOS45849.2019.8951367.
- [43] X. Bai, "Text classification based on LSTM and attention," in *2018 Thirteenth International Conference on Digital Information Management (ICDIM)*, Berlin, Germany, Sep. 2018, pp. 29–32. doi: 10.1109/ICDIM.2018.8847061.
- [44] Venkatesh, S. U. Hegde, Z. A. S, and N. Y, "Hybrid CNN-LSTM Model with GloVe Word Vector for Sentiment Analysis on Football Specific Tweets," in *2021 International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT)*, Bhilai, India, Feb. 2021, pp. 1–8. doi: 10.1109/ICAECT49130.2021.9392516.