Using Machine Learning and TF-IDF for Sentiment Analysis in Moroccan Dialect an Analytical Methodology and Comparative Study

Boudhir Anouar Abdelhakim(✉), Ben Ahmed Mohamed, and Ayanouz Soufyane

SSET Research Team C3S Laboratory FSTT, Abdelmalek Essadi University, Tangier, Morocco
{aboudhir,mbenahmed}@uae.ac.ma

Abstract. The Moroccan dialect is a linguistic area that presents special difficulties because of its complex morphology and wide range of influences. This study offers a novel technique to sentiment analysis in this dialect. Our work focuses on using machine learning methods in conjunction with Natural Language Processing (NLP) techniques, namely Term Frequency-Inverse Document Frequency (TF-IDF) for feature extraction, to effectively classify sentiment.

Given the scarcity of resources and standardized forms in Moroccan dialect, conventional sentiment analysis methods are less effective. To address this, our methodology involves rigorous preprocessing steps, including normalization, tokenization, and stemming, ensuring the refinement of input data for the machine learning models. The study utilizes a dataset comprising Moroccan tweets, classified into positive and negative sentiments, to train and test the models.

We use algorithms such as Decision Tree, Support Vector Machine, and Logistic Regression, and assess their performance using metrics like accuracy, precision, recall, and F-1 score. Our findings highlight the varying effectiveness of these models in handling sentiment analysis for a morphologically rich and unstructured language like Moroccan dialect.

This research not only contributes to the field of sentiment analysis in underrepresented languages but also opens avenues for further exploration using more advanced NLP tools and deep learning techniques. It underscores the potential and challenges of applying machine learning to dialect-specific sentiment analysis, providing valuable insights for future research in this domain.

Keywords: Sentiment Analysis · TF-IDF · Feature extraction · Machine learning · Moroccan dialect

1 Introduction

Currently, social media are really increasing in the daily life. It is a way for freely expressing our opinions about so many things or thematics. And those opinions represent a goldmine for businesses in particular, because by having a review from a client on any product proposed, it will help managers to decision making. Justly, Sentiment

M. Ben Ahmed et al. (Eds.): SCA 2023, LNNS 906, pp. 342–349, 2024.
https://doi.org/10.1007/978-3-031-53824-7_31

analysis can perform this task. By retrieving precious insights from messages, carrying out some actions for making data useful and finally knowing the polarity of texts, Sentiment Analysis became unavoidable for any industry who wants to stay ahead of their competitors. Nevertheless, when it comes to deal with some unstructured languages like Moroccan dialect; it becomes very difficult. Because of that, we proposed to effect feature extraction with TF-IDF, which can allow us to achieve the application of machine learning models like SVM or logistic regression for better classifying the sentiments.

The remaining paper is structured as: Sect. 2 present some challenges encountered in the case of Moroccan Dialect sentiment analysis, Sect. 3 refers to our methodology, Sect. 4 is about the application of classification algorithms, Sect. 5 shows the metrics used for evaluating our models, Sect. 6 highlights our results and a comparative study to another research and finally we conclude in Sect. 7.

2 Sentiment Analysis in Moroccan Dialect

The term “darija” also refers to Moroccan dialect. There is no standard written form for the dialect, and it can differ from one place to another. Also, darija is frequently expressed joined to other languages like English, French or Spanish, it can even written in many languages like Arabic, Latin or Arabizi, thus it becomes laborious to analyze the texts and makes obligated to translate and normalize in one only language.

Another hindrance is the normalized Arabic letters like ؤ، ى، ء **Hamza,** أ، ا **Alef or** ال، أل **lamalef** and the numbers presented in the orthographic darija like in Mer7ba = welcome or fer7an = happy.

And we can note finally and unfortunately the lack of specific resources to darija or the difficulty for finding the data to analyze.

3 Proposed Approach

In this section, we unveil the different steps for carrying out our sentiment analysis. We displayed in the following image an overview (Fig. 1).

3.1 Dataset

Normally, concerning this work, we were supposed to use the Moroccan Sentiment Twitter Dataset (MSTD), but the github page for accessing to this dataset wasn’t available. Thus, we retrieved a dataset which is a collection of tweets, from another github page[1], and the latter is called Moroccan Sentiment Analysis Corpus. It is annotated and has two classes of sentiment: positive and negative.

[1] https://raw.githubusercontent.com/ososs/Arabic-Sentiment-Analysis-corpus/master/MSAC.arff.

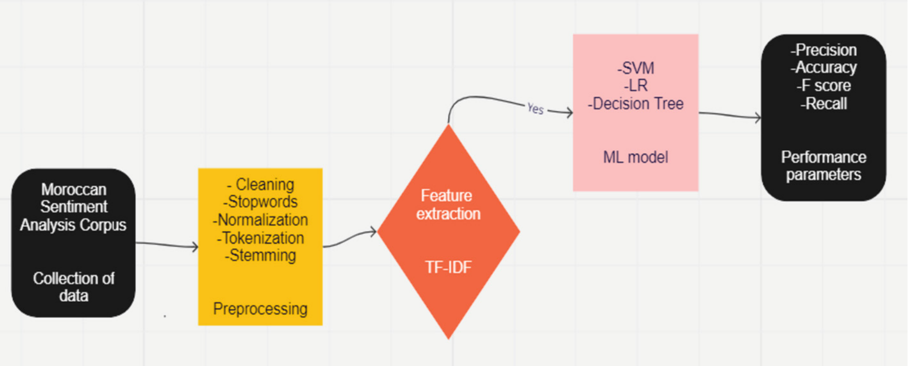


Fig. 1. Methodology

3.2 Preprocessing Techniques

Because Garbage in equals Garbage out, it is extremely crucial to perform well the preprocessing task, in particular in the case of this language. For dealing with, we followed this architecture described below.

Firstly, we started by cleaning the dataset. We removed anything which hasn't many importance for the analysis and the prediction of the polarity, like special characters, diacritics, repeated characters, numbers, punctuation and also emoji.

Secondly, we have sometimes some words which are common in any language, like prepositions, conjunctions or pronouns, but doesn't add much information to the text, they are known as stop words. We removed them by using the NLTK library which contains a built-in stop words.

Thirdly, we worked on the issue of normalization. Certain thoughts are expressed in unconventional ways. For example, some words have repeated letters, like "مبروووووووك"instead of "مبروك",which indicates congratulations, emotions such as “ههههههـ”which indicate laughing. Others include common spelling errors or accents. The normalization process aids in bringing the texts into compliance with accepted practices.

Fourthly, we performed the tokenization task. It aims to divide the text into pieces of data called tokens. Those tokens contain the essential information important for the analysis.

Finally, we ended by stemming task. The stemming process is used to change different tenses of words to its base form, this process is thus helpful to remove unwanted computation of words. For doing this, we applied a stemming technique called Light Stemming by using a specific library called Tashaphyne.

3.3 Feature Extraction

One of the most crucial processes to take in order to comprehend the context of the material we are working with better is feature extraction. We must convert the original text into its features so that it may be utilized for modeling after it has been cleaned. Put another way, in order to provide machine learning algorithms with numerical features, we must extract features from the raw text. We decide to use TF-IDF, or term frequency inverse document frequency, to do this.

A popular method in NLP for assessing a word's importance in a document or corpus is called TF-IDF. In essence, it compares a word's frequency within a particular document to its frequency over the entire corpus to determine how important it is. The fundamental premise is that a word is especially significant in a document if it appears more frequently inside it but less frequently throughout the corpus.

4 Classification Algorithms

Here, we show the ML model used for classifying our reviews based on their sentiments.

4.1 Logistic Regression

The method of modeling the probability of a discrete result given an input variable is known as logistic regression. A binary outcome, or something that can have two values, such as true or false, yes or no, and so on, is what most logistic regression models represent. Another name for this approach is Maximum Entropy.

4.2 Support Vector Machine

Backing A supervised technique called Vector Machine is employed for problems involving both classification and regression. Its goal is to locate a hyperplane that clearly classifies the data points in an N-dimensional space, where N is the number of features.

4.3 Decision Tree

The way this algorithm operates makes it incredibly efficient. The main concept is to partition the dataset into more manageable groups while concurrently creating the corresponding tree piece by piece. This is capable of handling numerical and categorical data.

5 Performance Parameters

In this party, we expose the metrics used for judging or estimating the quality of our ML models.

5.1 Accuracy

As indicated in Eq. 1 or, more accurately, 1.a, this is the ratio of true positives plus true negatives to the true positives plus true negatives plus false positives plus false negatives. It determines the proportion of cases that are correctly classified.

$$Accuracy = \frac{(\text{True positive} + \text{True negative})}{True\,positive + True\,negative + False\,positive + False\,negative} \tag{1}$$

$$Accuracy = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}} \tag{1.a}$$

5.2 Precision

Precision attempts to answer the following question:

What percentage of positive identifications were true positives?

Precision is defined as the ratio of expected positive observations to the total number of positive observations. It is computed as follows:

$$Precision = \frac{\text{True positive}}{\text{True positive } + \text{ False positive}} \tag{2}$$

$$Precision = \frac{\text{Relevant retrieved instances}}{\text{All retrieved instances}} \tag{2.a}$$

5.3 Recall

Ratio of correctly predicted positive observations to all observations in actual class yes is known as recall. It is computed as follows:

$$Recall = \frac{\text{True positive}}{\text{True positive} + \text{False negative}} \tag{3}$$

$$Recall = \frac{\text{Relevant retrieved instances}}{\text{All relevant instances}} \tag{3.a}$$

5.4 F-1score

Weighted average of recall and precision is called f-score. More important parameter than accuracy when having an uneven class distribution in data. It is calculated as follows:

$$F - score = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4}$$

6 Results and Comparison Analysis

6.1 Results of the Work

In this work, we wished to use Term Frequency In-verse Document Frequency (TF-IDF) for the features extraction step in order to approach the analysis of Moroccan dialect sentences. Next, we used three machine learning algorithms—Support Vector Machine, Logistic Regression, and Decision Tree—for categorization. Using the 80–20 approach, we divided the data into training and testing sets. Ultimately, we assessed our models' performance using criteria including accuracy, precision, recall, and F-1 score. (Table 1).

It is evident that a model's performance might differ based on the metric, albeit the discrepancy between the outcomes is not very great.

Table 1. Classification Results

	SVM	LR	DT
Precision	Neg:81% Pos:81%	Neg:80% Pos:81%	Neg:83% Pos:61%
Recall	Neg:85% Pos: 76%	Neg85:% Pos:74%	Neg:53% Pos:87%
F-1 score	Neg:83% Pos: 78%	Neg:82% Pos:77%	Neg:64% Pos:71%
Accuracy	81%	80%	68%

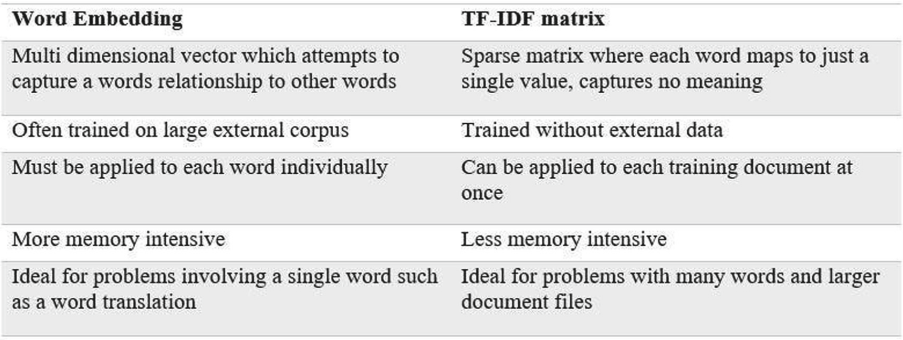
6.2 Comparison Analysis

In this section, we perform a brief comparison between our work and a follow-up study that used Word embedding, Arabert for feature extraction, and various deep learning models for classification on an identical item.

It's important to clarify right away that we didn't utilize the same dataset. The MSTD that the authors utilized is larger than ours. While our dataset only contains two classes—positive and negative—MSTD is a collection of 12K Moroccan tweets that were annotated with four distinct classes: 2769 negative, 866 positives, 6378 objective, and 2188 sarcastic.

Second, as the following graphic illustrates, there are a lot of distinctions between TF-IDF and Word embedding. (Table 2).

Table 2. TF-IDF vs Word Embedding.



Word Embedding	**TF-IDF matrix**
Multi dimensional vector which attempts to capture a words relationship to other words	Sparse matrix where each word maps to just a single value, captures no meaning
Often trained on large external corpus	Trained without external data
Must be applied to each word individually	Can be applied to each training document at once
More memory intensive	Less memory intensive
Ideal for problems involving a single word such as a word translation	Ideal for problems with many words and larger document files

It is evident from the differences that word embedding performs better than TF-IDF and can produce the most accurate results when it comes to relevant classification by machine learning models.

Lastly, we used the Accuracy metric to show how our models performed differently from their models (Fig. 2).

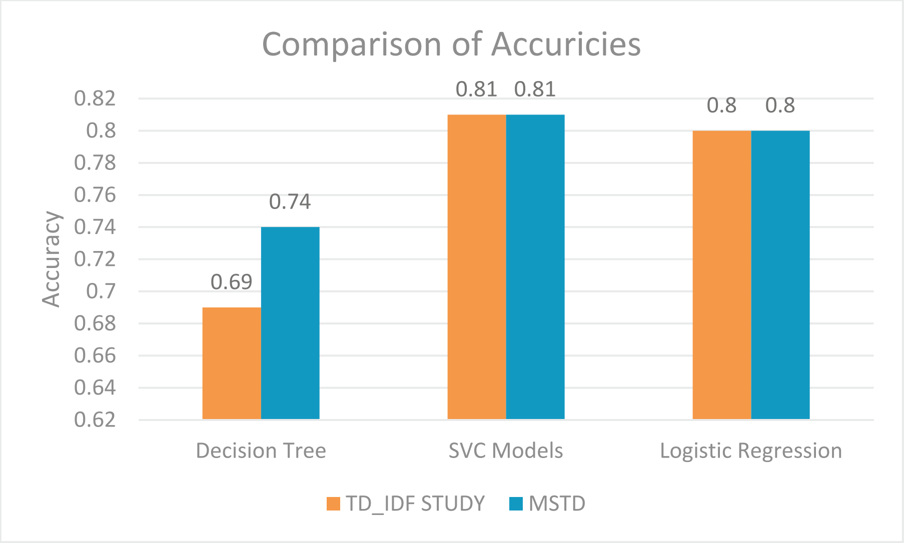


Fig. 2. Performance based on accuracy

7 Conclusion

In conclusion, our study makes significant strides in the axe of sentiment analysis for the Moroccan dialect, a linguistically complex and underrepresented language in computational linguistics. By adapting and applying NLP techniques, specifically TF-IDF for feature extraction, coupled with machine learning techniques including decision trees, logistic regression, and support vector machines, we have demonstrated a viable approach to classifying sentiments in Moroccan dialect texts.

Our findings reveal that while each algorithm has its strengths and limitations, they collectively offer promising avenues for accurately discerning sentiment in a dialect that presents unique challenges due to its unstructured nature and lack of standardization. The preprocessing steps, including tokenization, normalization, and stemming, were crucial in refining the data for more effective analysis.

This research contributes to the broader understanding of sentiment analysis in dialects and minority languages, highlighting the importance of tailored approaches for such linguistic contexts. It also underscores the potential of machine learning in uncovering insights from dialect-specific data, which is often overlooked in mainstream NLP research.

Looking forward, there is ample scope for enhancing this research by integrating more advanced NLP tools and exploring deep learning models like Long Short Term Memory or Neural Networks. Such future endeavors could further refine the accuracy and efficiency of sentiment analysis in the Moroccan dialect and other similar languages, potentially expanding the applicability of NLP in diverse linguistic landscapes.

References

The International Conference on Digital Age & Technological Advances for sustainable

Development: 'Sentiment Analysis through Word embedding using AraBERT: Moroccan Dialect use case", ©2021

Ravinder Ahuja, Aakarsha Chug,Shruti Kohli,Shaurya Gupta, and Pratyush Ahuja: "The impact of Features extraction on the sentiment analysis", International Conference on

Pervasive Computing Advances and Applications – PerCAA 2019

Mouaad Errami, Mohamed Amine Ouassil, Rabia Rachidi, Bouchaib Cherradi, Soufiane Hamida, Abdelhadi Raihani: " Sentiment Analysis on Moroccan Dialect based on

ML and Social Media Content Detection", (IJACSA) International Journal of Advanced Computer Science and Applications, vol. 14, No. 3, 2023

Soukaina MIH1, Brahim AIT BEN ALI , Ismail EL BAZI , Sara AREZKI , Nabil LAACHFOUBI.: " MSTD: Moroccan Sentiment Twitter Dataset", (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 11, No. 10, 2020

MOUHOUBI Azzedine,GHEFFARI Mohammed Abdelfattah,' Analyse de sentiments dans la langue arabe en utilisant differentes approches, presente en vue de l'obtention du diplome de Master,2019–2020

EL BERGUI Adam :"Sentiment Analysis for Moroccan Dialect",September 2019