IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2022.Doi Number

WeVoTe: A Weighted Voting Technique for Automatic Sentiment Annotation of Moroccan Dialect Comments

YASSIR MATRANE, FAOUZIA BENABBOU, AND ZOUHEIR BANOU

Laboratory of Information Technology and Modeling, Faculty of Sciences Ben M'Sick, Hassan II University of Casablanca, Casablanca 20000, Morocco
Corresponding author: Yassir Matrane (e-mail: yassermatrane@gmail.com).

ABSTRACT Sentiment analysis represents the systematic procedure of independently discerning polarity inherent in a textual document. A multitude of sectors can derive substantial advantages from this specialized domain. Conducting sentiment analysis (SA) involves various phases, with the initial step being the annotation process, which is often time-consuming and laborious. Within this framework, there exists a notable scarcity of existing research works. The complexity of this task becomes more difficult when analyzing texts in 'Darija', a form of the Moroccan dialect (MD). In our research endeavors, we introduced a novel automatic annotation methodology designed explicitly for sentiment analysis within the Moroccan dialect. A pivotal aspect of our contribution is the refinement of the stacking approach, utilizing a weighted voting technique for enhanced predictive accuracy. Our advanced method starts with the training of various neural network models across six unique MD datasets. The selection of these neural network architectures was underpinned by a comprehensive grid search procedure. Conclusively, it was discerned that models predicated on Recurrent Neural Networks (RNNs) outperformed others. Subsequent to this, we deployed an augmented stacking model, grounded in the aforementioned weighted voting technique. This model leverages the predictions generated by the neural networks as inputs. It then employs the mode of these inputs as an output, which feeds directly into a meta-classifier, which in turn produces the coefficients. These coefficients are then multiplicatively combined with the initial neural network predictions to derive the finale outputs. To evaluate the efficiency of our proposed methodology in annotating the six datasets, each dataset was isolated as a test while the remaining five served as training sets. Consequently, within the set of six datasets, the annotation results of three datasets have outperformed the established standards, attaining agreement rate percentages of 87.54% for MSAC, 91.25% for FB, 85.10% for MSDA, and 83.60% for MSTD, all of which represent new achievements in the literature.

INDEX TERMS Sentiment analysis, Arabic dialect, Automatic annotation, labeling technique, Machine learning.

I. INTRODUCTION

It is crucial in today's society to analyze the emotions expressed on social media. This is accomplished using a technique known as sentiment analysis, which identifies the polarity, intent, or intensity of an opinion. Beginning with text annotation, text preprocessing, text extraction, and classification models, multiple phases are applied to the sentiment analysis procedure. However, the efficacy of sentiment analysis is contingent on multiple factors, including the language employed and the method of annotation, which is the process of labeling data. Due to the fact that social media platforms such as Facebook, Instagram, YouTube, and Twitter host the vast majority of the data that forms the foundation for opinions expressed in Modern Standard Arabic (MSA) and colloquial Arabic, a correct analysis can be validated by their essential transparency. Analysis of unstructured languages, particularly Arabic dialects, can be challenging due to obstacles such as complex morphology and the wide variety of dialects spoken in each Arabic-speaking country.

It is worthwhile mentioning that sentiment analysis is the broader task of determining sentiment in text data, involving training a model on labeled data. The annotation process is a specific step within sentiment analysis, where human or automatic methods are used to assign sentiment labels on the whole dataset.

The Moroccan dialect, known as "Darija", presents unique challenges in terms of automatic processing due to its complexity. In this context, Darija incorporates a mix of Arabic, Berber, French, and Spanish elements. This blending of languages makes Darija distinctively diverse and multifaceted, as it not only uses vocabulary from these languages but also integrates their phonetic and syntactic feature. Additionally, the influence of Berber languages (Amazigh) is profound, in both the structure and vocabulary of Darija, which is not a characteristic seen in other Arabic dialects. Furthermore, it has the characteristic of being writable in both Arabic and Latin, which is called Arabizi. Several studies have endeavored to classify sentiments in this dialect. In the study titled [1] the research utilized the MSAC dataset, comprising approximately 1,014 positive and 1,022 negative reviews. The most effective method for the NB classifier involved the use of TF-IDF weighting, light stemming, and both uni-gram and bi-gram combinations, achieving an accuracy of 86.02%. Moreover, using the ElecMorocco2016 dataset, which has 3,673 positive and 6,581 negative comments, the logistic regression classifier attained an accuracy of 81.17% when applying TF-IDF, light stemming, and uni-gram features. The [2] study focused on the MSDA dataset, sourced from active users from Arab countries like Lebanon, Algeria, Egypt, Tunisia, and particularly Morocco. The dataset was refined to encompass only Moroccan-relevant tweets, resulting in 1,605 positive, 1,620 negative, and 1,630 neutral tweets. After preprocessing, the SVM classifier combined with the ISRI stemmer and TF-IDF for feature extraction yielded the best performance, with an accuracy rate of 68.59%. Lastly, the work of [3] utilized a dataset extracted from 50 YouTube channels, amassing a total of 20,000 comments labeled either positive or negative. By leveraging the CNN architecture and implementing preprocessing techniques, the study achieved a commendable accuracy of 90%.

The text annotation stage is more difficult for Arabic than for other languages due to the language's numerous difficulties. Due to the complex morphology and distinctive properties of Arabic dialects, morphological analysis is a difficult endeavor, particularly in terms of derivations and inflections. According to [4], [5] and [6], Arabic dialects lack a standardized written format, and their forms vary across Arab countries, making them challenging to process mechanically. In addition, as stated in [7], Arabic has numerous regional and national dialects spoken throughout the Arab globe. There are three methods to annotate Arabic text for sentiment: automatically, semi-automatically, and manually. In the manual approach, which relies on a team of data annotators with specialized knowledge to manually annotate text with high accuracy, crowdsourcing is utilized. As humans perform manual annotation, it can be accurate but is time-consuming and costly, making it difficult to scale for large datasets. Therefore, there is a need for automatic annotation techniques capable of accurately assigning polarity identifiers (positive or negative) to text. It is common practice to have multiple people annotate the same text in order to obtain the highest quality annotation feasible. Cross-annotation could be utilized to determine the inter-rater reliability. According to [8], Cohen's Kappa calculates an inter-rate score between two annotators in this situation, taking into consideration chance agreement. Active learning is the basis for the semi-automatic technique, which allows learning algorithms to collaborate with annotators to designate samples with the desired results. By limiting manual intervention to the most informative samples, the primary objective of this method is to reduce the cost of data annotation. In this context, the authors of this work [9], developed a method of active learning that significantly reduces the cost and effort of annotation by allowing researchers to create large, high-quality annotated datasets with only a few manual annotations. In addition, unsupervised learning requires the creation of algorithms or models that can recognize patterns or structures within unlabeled data. Without the need for human-labeled data, these models can then be used to autonomously annotate the text. The primary advantage of unsupervised learning is that it can manage large amounts of data without the labor-intensive and time-consuming process of manual annotation.

Because of the limited resources and complex morphology of Arabic dialects, semi-automatic and automatic annotation approaches are still understudied in the literature. This methodology is not confined to Moroccan dialect but can be adapted for any language, demonstrating the versatility of unsupervised learning in various tasks like sentiment analysis and topic modeling. Our method's essence lies in its wide applicability and ability to handle the multifaceted nature of languages, particularly in complex linguistic contexts like Darija, highlighting its potential in broader linguistic analysis applications. This work's main contributions are as follows:

- This research work critically examines the current state of the art in Arabic dialect sentiment analysis by conducting a comprehensive survey of prior studies, focusing on both semi-automatic and automatic annotation techniques. Moreover, this research contributes significantly by highlighting the pivotal role of the annotation phase in achieving more accurate sentiment analysis results.

- The primary contribution encompasses the formulation of a sophisticated stacking methodology grounded in a weighted voting mechanism. To refine this method, several neural network models were trained on an assortment of six unique Moroccan dialect datasets. By integrating BERT-based feature extraction into the stacking methodology, the neural network models can leverage these advanced feature representations. In this context, the efficacy of

BERT-based approach stems from its ability to capture context from both left and right sides of a token in the input sequence. This bidirectional context capture is far more comprehensive than traditional unidirectional approaches, making BERT particularly adept at understanding nuances in language, which is critical in dialect analysis. Moreover, BERT's pre-training on vast amounts of text data allows it to effectively extract subtle language features, enhancing the neural network models' ability to discern intricate patterns in the Moroccan dialect datasets. Selection of these networks was steered by an exhaustive grid search process, culminating in the identification of RNN-based models as superior performers. Subsequently, we introduced an enhanced stacking technique, which leverages predictions from the neural network models as input data. This model harnesses the mode of these predictions to feed into a meta-classifier. The purpose of this is twofold: firstly, to generate specific coefficients, and secondly, to multiply these coefficients with the original neural network model predictions to yield final outputs.

The rest of the paper is organized as follows. In section 2, we present the state of the art of automatic and semi-automatic annotation approaches in the context of Arabic dialects. Afterward, we describe the proposed approach in section 3. Consequently, section 4 describes and analyzes the experimental results of the annotation approach. Finally, we conclude the paper and give directions to future work in Automatic labeling approaches for Arabic dialects.

II. RELATED WORK

For many tasks, there are large amounts of unlabeled data available, but it is expensive to provide the annotations required to use them as training sets. Therefore, several research works have been carried out utilizing automatic annotation approaches. In this literature review, we present the studies that investigated semi-automatic and automatic approaches in the annotation phase of Arabic dialect sentiment analysis.

The authors of [10], utilized a balanced dataset of 500 positive and 500 negative Facebook messages from Algeria. Native speakers performed the annotation to guarantee the quality of the lexicon-based annotation. In this context, the authors created a sentiment lexicon by translating the English lexicon, SOCAL [11], resulting in a lexicon of 1745 terms (968 negative, 771 positive, and 6 neural). To score a message, the sentiment scores of each term in the message that is present in the lexicon are averaged. The polarity scoring algorithm considered a simple rule-based light stemmer that manages DALG prefixes and suffixes, as well as the negation problem, resulting in an F1-score of 42.3% for the best performance. In contrast, the authors of this study [12], used Ar-SeLn [10], a conventional Arabic sentiment lexicon, and an Emoji Polarity Lexicon to incorporate the list of emojis used for tweet collection into the lexicons. The accuracy of the autonomous data annotation method was then evaluated by randomly extracting 1,000 tweets from each class. These samples were hand-labeled by two Arabic-native speakers. The classification error rate for positive classes was 5.6%, negative classes 4.2%, and neutral classes 11.6%. In this context, [13] conducted a sentiment analysis of Moroccan dialect by annotating tweets based on the assumption that the emotion symbols in a tweet convey the overall sentiment of the message. Consequently, their accuracy was 69%. Another method of automatic annotation applied to a health-related Twitter dataset by [14], consisted of two steps: emoji labeling and semantic labeling. These steps are based on two dictionaries that comprise emojis and words with the appropriate polarity. This strategy was effective in enhancing the training evaluation and final predictions, resulting in an F1-score of 82%. In [15], this work presents a lexicon-based approach to build an annotated corpus for classifying the emotional state of Arabic messages using an Algerian Facebook dataset with 3048 messages (1488 positive and 1560 negative). The authors developed an algorithm that considers the morphology of Arabic and its dialects, along with a lexicon that was automatically made by using an existing English mood lexicon. The corpus that was made has a right annotation rate of 85.17%. In this context, the research work of [12], presents an automatic method for annotating an Arabic dataset with tweets and emoticons for sentiment analysis. The authors employ a lexicon-based method that makes use of the first publicly available large-scale sentiment lexicon [10], in addition to a list of emojis and their respective polarity (Sentiment of emojis). Additionally, the authors also took into consideration the negation of the text, which can influence the polarity of sentiments. They extracted 1,000 tweets at random from each class (positive, negative, and neutral) and have manually annotated them by two native Arabic speakers. The validation of this automatic method reveals an acceptable classification error rate, with the maximum value for the neutral set being 11%. The authors of [16], developed a web-based tool for Arabic sentiment analysis that obtained varying degrees of accuracy for different polarities using a lexicon-based approach for automatic annotation. This study classified 8,000 tweets written in various Arabic dialects, predominantly Egyptian, into four categories: positive, negative, both, and neutral. In addition to applying a stemming technique, the dataset was cleaned by removing digits, punctuation, URLs, hashtags, usernames, retweets, and spaces. To vectorize the data, N-gram feature extraction was utilized. Furthermore, a lexicon of words constructed from a dataset of tweets and then labeled with polarity scores by annotators in order to be utilized for automatic annotation. The accuracy of this annotation method was 40%, 64%, 72%, and 90% for negative, positive, neutral, both, and average polarities, respectively. For a lexicon-based annotation in a study published in [17], the authors used a dataset of 1,000 sentences from the Shami corpus [18], which comprises Levantine dialects. To assure the accuracy of the automatic annotation, a native Levantine speaker manually annotated the sentences. The LABR lexicon from [19] includes negative, positive, and negated terms. The Moarlex lexicon [20] has only positive and negative terms. In contrast, when these sentiment lexicons were used to automatically annotate one thousand sentences, the inter-annotator agreement between the human annotator and the automatic annotation was as low as 80%. The Authors in [21] seek to discern the sentiments and perceptions of the Arab community about the Russo-Ukraine War by analyzing tweets. During preprocessing, irrelevant data was discarded, followed by normalization and lemmatization using the ‘Farasa’ lemmatizer. For validation, a set of 400 tweets were manually annotated for sentiment orientation and compared with the outcomes from the n-grams lexicon-based technique. Four reviewers labeled these tweets as either ‘Pro-Russia’ or ‘Pro-Ukraine’, with consensus achieved through majority vote. Remarkably, 81.25% of these manual labels aligned with the n-grams lexicon method. After filtering tweets that ambiguously criticized either Russia or Ukraine, this consistency rose to 85.43%. Additionally, from a sample of 1000 tweets, 780 were tested using an LSTM model, and 220 were trained using the lexicon method. Four reviewers and the LSTM or lexicon models in 70.1% of cases provided the same annotations. By enlarging the training set to encompass about 80% of the entire dataset, the LSTM model's accuracy surged to 77.3%, signifying the benefits of a larger dataset in enhancing model performance. This indicates that current sentiment lexicons may be inadequate for accurately annotating Levantine dialects.

A semi-automatic annotation approach was conducted by [22], in which a Nave Bayes classifier was used to annotate a new Saudi dialect dataset for sentiment analysis. The objective of this work is to reduce the human effort and time required for the labeling procedure. Consequently, this method obtained an accuracy of 83%. In this context, the authors of [23] present a self-learning strategy for expanding a manually annotated 15,000-tweet dataset. Each unlabeled tweet was mechanically annotated by passing it through three classifiers built using FastText as a word embedding. If a tweet's confidence value was greater than or equal to a threshold across all classifiers, it was added to the training data and removed from the unlabeled data. To increase the quantity of labeled data, they repeated this procedure three times. In the previous iteration, the exhaustive corpus included over 3.2 million and 4.5 million tweets for two-way and three-way sentiment classification, respectively. After acquiring each of the expanded annotated datasets, the authors trained an LSTM deep learning model and assessed it on two benchmark datasets to ensure the quality of their proposed annotation method. Their approach improves sentiment two-way classification results from 80.37% to 87.4% using the SemEval 2017 dataset [24] and from 79.77% to 85.51% using the ASTD dataset [25]. It achieved a three-way classification accuracy of 69.4% with the SemEval 2017 dataset and 68.1% with the ASTD dataset. This research [26] devised a method for classifying a subset of 3063 Facebook posts, including 1021 negative, 1021 positive, and 1021 neutral posts, using ensemble domain adaptation. This subset of 60,000 posts was extracted from the larger set. The objective was to attribute sentiment labels to unannotated posts. To determine the most effective classifier for each dataset, the authors applied nine classifiers to five datasets from prior studies [25], [27], [28], [29], and [19] to determine the most effective classifier for each dataset. The best classifier from each dataset was then used to train the data and predict the target dataset. A system of majority voting was used to determine the final label. Logistic Regression with Bag-of-Words (BoW) feature representation obtained the highest performance on the target dataset, with an accuracy of 88.47%, an F1-score of 88%, and a recall of 88.47%.

The authors of [30] utilized multiple approaches for annotation. Primarily, they compared emoji-based annotation and a self-training approach based on remote supervision with human annotation using a subset of 8000 dialectical Arabic tweets. The accuracy of the emoji-based approach attained 77.2%. Using the unigram features as a starting point, they constructed a sentiment analysis machine learning model using a Linear Support Vector classifier as well as a more complex model that employs word and character grams. In numerous experiments designed to reduce human effort, they achieved an accuracy of 86% by using self-training methods. Therefore, remote supervision proved to be an effective method for autonomously annotating an enormous number of samples. In this study [31], a method was proposed for autonomously classifying an Algerian sentiment corpus into positive and negative statements using a created lexicon of words and their respective scores. They used a variety of classifiers to validate the annotated corpus and evaluate its efficacy on two separate test sets. External test set represents a collection of messages that have been manually annotated, whereas internal test set represents a portion of the automatically annotated Algerian corpus. Each test dataset is composed of Arabic and Arabizi datasets. External Arabic test set and internal Arabizi test set F1-scores are up to 72% and 78%, whereas internal Arabic and internal Arabizi test set F1-scores are up to 68% and 68%, respectively. In a study conducted by [32], an extensive corpus of Tunisian dialect sentiments was extracted from the website goodreader.com. The authors utilized techniques for automatic annotation based on the ratings assigned to the evaluations. Following the same methodology as in [19], they regarded positive reviews with ratings of 4 or 5; negative reviews with ratings of 1 or 2; and neutral reviews with ratings of 3. In the same context, another study conducted by [33], collected more than 370000 Tunisian evaluations from the website booking.com. This corpus was mechanically annotated according to the user's review rating on a scale from 1 to 10. Each review has a positive, negative, or neutral annotation.

Table 1 provides an overview of the annotation approaches proposed for Arabic dialect. In summary, a significant portion of studies focused on Arabic dialects, largely employing machine learning techniques, particularly unsupervised and ensemble learning, for a semi-automatic sentiment analysis. In contrast, most automatic annotations leveraged lexicon-based methods, with emojis playing a pivotal role in labeling tweets. For studies drawing reviews from websites, user ratings were the primary annotation references. The three predominant automatic annotation strategies encompass lexicon-based, ensemble-based and user review ratings. Notably, the evaluation of annotation methodologies was often overlooked in these works.

Smaller datasets benefit from the simplicity of lexicon-based annotation due to their limited token count. Conversely, larger datasets, abundant in examples, facilitate the development of more comprehensive models post-training. Yet, annotating them with lexicon-based techniques demands substantial effort, even when handling specific dialectal terms like (بغيت، عاوز، بدي،). The lexicon's quality and volume play crucial roles in the annotation's outcome, particularly in addressing out-of-vocabulary tokens. As the lexicon expands, the likelihood of encountering an Out of Vocabulary (OOV) token diminishes. Some researchers have also gravitated towards unsupervised learning for annotating extensive datasets, while others hinged on metrics like user feedback on review platforms or social media engagement metrics.

TABLE 1: ANNOTATION APPROACHES FOR ARABIC DIALECT SENTIMENT ANALYSIS WORKS

Ref	Dataset	Language	Annotation approach	Technique
[10]	400 Arabic Treebank	Multi-dialects + MSA	Automatic	Lexicon-based
[12]	3000 Twitter	Multi-dialects	Automatic	Lexicon-based
[23]	4.5M Twitter	Multi-dialects	Semi-automatic	Unsupervised learning
[13]	930 Twitter	Moroccan dialect	Automatic	Lexicon-based
[15]	3048 Facebook	Algerian dialect	Automatic	Lexicon-based
[31]	8000 Facebook	Algerian dialect	Automatic	Lexicon-based
[26]	3063 Facebook	Tunisian dialect	Automatic	Ensemble method
[32]	510600 Website	Tunisian dialect	Automatic	User rates
[33]	370000 websites	Tunisian dialect	Automatic	User rates
[16]	8000 Twitter	Egyptian dialect	Automatic	Lexicon-based
[17]	1000 Twitter	Levantine dialect	Automatic	Lexicon-based
[22]	3495 Twitter	Saudi dialect	Semi-automatic	Unsupervised learning
[34]	3495 Twitter	Saudi dialect	Semi-automatic	Unsupervised learning

The annotation techniques may seem effective, although, a deeper analysis of social media reveals the following drawbacks:

1. Lexicon-based technique involves constructing a lexicon.
2. User rates technique relies on human intervention, moreover, it is vulnerable to biasing by trolls which can lead to false positives.
3. The reader may misinterpret some satiric content; consequently, their reaction may not reflect their true opinion.
4. Controversial topics make posts prone to likes/dislikes based on individual's point of view of the matter.
5. The method's effectiveness is highly dependent on the presence of an indicator in the same platform data is collected from. The absence of such indicators or their presence in a different multiplicity (such as Facebook reactions) is likely to impact the annotation process, since most of these reactions are not clear indications of the underlying sentiment, or sometimes carry a neutral sentiment (example: Wow reaction).
6. Automatic annotation can be less accurate than manual annotation, but it is faster and more scalable.

In the field of sentiment analysis, the application of a lexicon containing word polarities for annotating textual content is typically categorized as an automated annotation technique. Nevertheless, the success of such automated annotation is contingent upon both the comprehensive nature and the caliber of the lexicon employed.

There may be instances in which manual verification or correction is required, rendering the process semi-automatic.
In addition, using user ratings to annotate review sentiments is considered an automatic annotation. This is because positive

and negative sentiment labels are assigned based on user ratings without immediate human input. However, this method assumes that user ratings accurately reflect the sentiments expressed in review text, which is not always the case. Even if the procedure is automated, human review or intervention may be required to ensure annotation accuracy, particularly if user ratings and review sentiments do not align. In such cases, the method may be described as semi-automatic. The precise terminology depends on the level of human participation in the procedure.

III. METHODOLOGY

In this section, we will present the proposed methodology to automatically annotate a Moroccan dialect dataset with either positive or negative labels. As shown in Figure 1, the annotation process begins by collecting a set of Moroccan dialect datasets publicly available for sentiment analysis purpose. These datasets are preprocessed to remove any irrelevant information before being transformed into the vector format using a two pre-trained word embedding models. Afterwards, BiGRU, CNN-BiGRU, LSTM, CNN-LSTM, BiLSTM, CNN-BiLSTM classifiers were trained using the training datasets individually. The baseline models will then be used to predict labels for a new unlabeled dataset. The resulting votes from the classifiers are aggregated using two weighted-average-based techniques, which forms the core contribution presented in this paper.

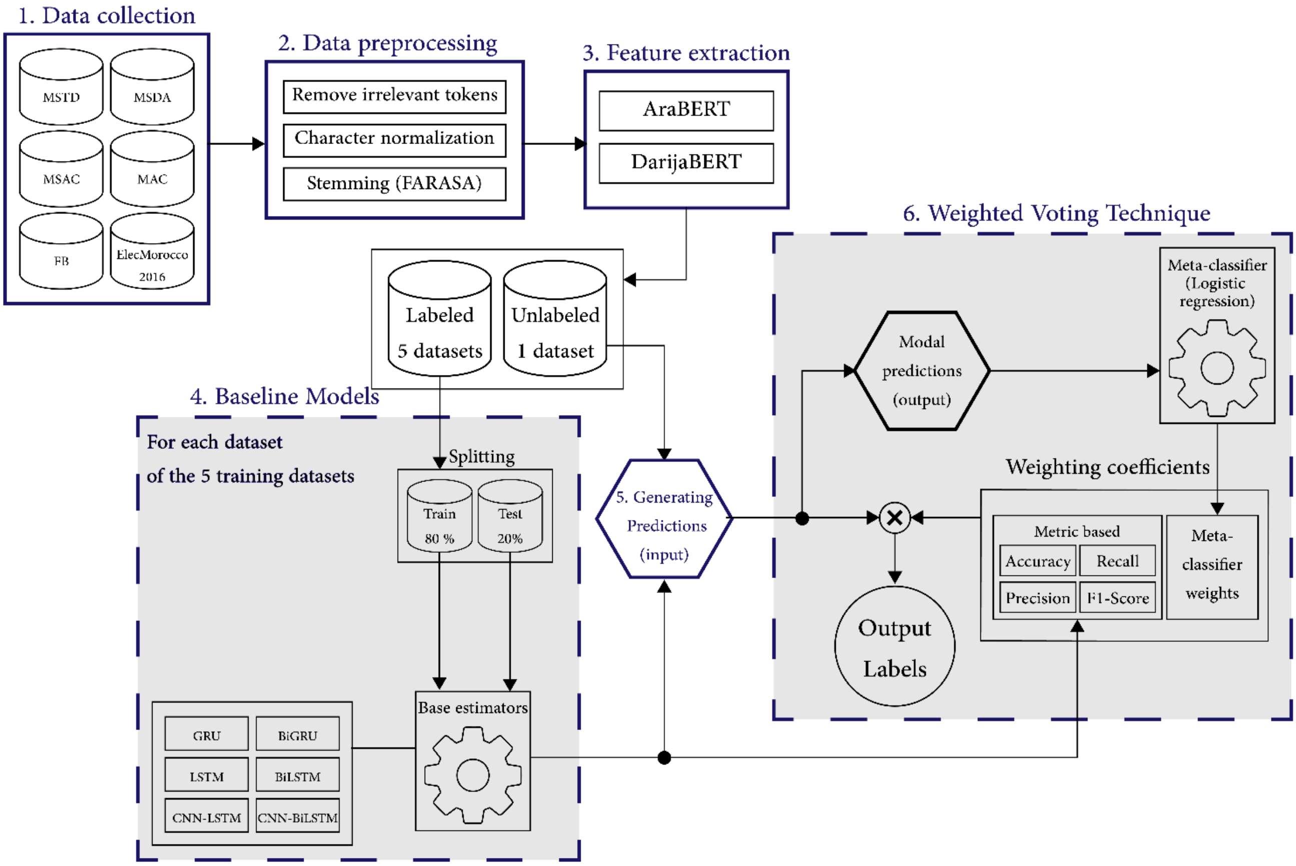

FIGURE 1. **Automatic Annotation Methodology Overview**

Each of the six steps of the automatic annotation of Moroccan dialect approach will be explained in details in the next sections.

A. DATA COLLECTION

We collected six distinct open-access datasets for two-way sentiment classification purpose. The Moroccan Sentiment Analysis Corpus (MSAC) was gathered by [35] from web blogs and discussion forums that are either heavily frequented by Moroccans or housed in Morocco. The data set consists of user reviews and comments from Hespress (HES) website, and Facebook (FB), Twitter (TW), and YouTube (YT) social networks. It is a multi-domain corpus made up of text from the sports, social, and political domains that uses a variety of vocabulary. MSAC contains about 981 of positive reviews and

994 of negatives written in Moroccan Dialect (MD). The ElecMorocco2016 dataset [36] was manually annotated and collected from several social media websites, including FB, Twitter, YouTube, and Hespress. Only comments regarding politics, particularly the 2016 Moroccan general election, written in MD are included in the dataset. ElecMorocco2016, includes 3673 comments that are positive and 6581 that are negative. The Moroccan Sentiment Twitter Dataset (MSTD) dataset was collected from Twitter and manually annotated [37]. The MSTD dataset includes almost 12K tweets divided into four classes, we considered only two classes: negative (2769) and positive (866), all of which are related to sports, arts, politics, education, and other social concerns. The fourth set of data was gathered by [38] from the public Facebook pages of Moroccan online news sources and covers a wide range of subjects, including politics, the economy, sport, religion, and society. A manual annotation task was conducted on a total of 9901 collected comments, comprising both MSA and Arabic dialect (MD) comments. This dataset includes 2858 negative and 684 positive comments. By using the trendy hashtags, the Moroccan Arabic Corpus (MAC) dataset was collected from active Moroccan Twitter accounts [39]. The 18000 valid tweets in the final corpus were manually annotated, with a mixture of MSA and MD tweets divided into four classes: positive (9897), neutral (4039), negative (3508), and mixed (643). Only MD and two classes (positive and negative) were considered, yielding 2990 positive and 1309 negative tweets. MSDA [40] is the latest dataset designed specifically for three Natural language processing (NLP) tasks: Sentiment Analysis, Topic Detection, and Dialect Classification. Hence, the dataset consists of tweets belonging to different topics, mainly Politics, Health, Social, Sport, and Economics. For this study, the dataset was narrowed down to include only tweets in the Arabic dialect (MD), and we focused solely on the positive and negative classes. Table 2 provides an overview of the characteristics for the datasets used in the study.

TABLE 2: CHARACTERISTICS OF DATASETS USED IN THE STUDY

Reference	Dataset	Size		Sources	Topics
		Positives	Negatives		
[36]	ElecMorocco2016	3673	6581	FB, TW, HES, YT	Politics
[38]	FB	684	2858	FB	Politics, economy, sport, religion, society
[39]	MAC	2987	982	TW	-
[35]	MSAC	981	994	FB, TW, HES, YT	Sports, social, politics
[40]	MSDA	397	1200	TW	Politics, Health, Social, Sport, Economics
[37]	MSTD	868	2773	TW	Sports, arts, politics, education, social

According to Table 2, only ElecMorocco2016 is considered mono-topic, as it covers only political social media posts. This specific attribute of a dataset can be a direct cause of having more topic-specific words. Other datasets are general-purpose datasets containing user comments, collected mainly from Twitter. As per the size, ElecMorocco2016 is a relatively the largest dataset, having as much as twice the number of negative tweets compared to positive tweets. This pattern can be seen in all the other datasets, except for MAC, where positive tweets are much more abundant.

B. DATA PREPROCESSING

Text preprocessing is a crucial step in the [1]downstream tasks such as sentiment analysis, text classification, information retrieval, machine translation, and many other NLP applications. Therefore, we will present the different text preprocessing techniques used in the methodology as presented in Figure 2. Data cleaning is a crucial step for NLP tasks, particularly sentiment analysis in Arabic dialect. The pipeline consisted of removing HTML content, URLs, mentions, punctuations, numbers, laughter such as ‘ههه’, lengthening, one-letter words, repeated letters, diacritics, and non-Arabic letters. Stop word removal may appear to be a crucial stage in data preprocessing, nevertheless, we have decided not to include it because it removes context-sensitive tokens such as negators, which can change the polarity of the text, for instance, adverbs like “لا” or “ماشي”. After testing a variety of stemmers, we have opted to employ the Farasa stemmer [41], given that Moroccans frequently use modern standard Arabic words.

[1] https://msda.um6p.ma/msda_datasets

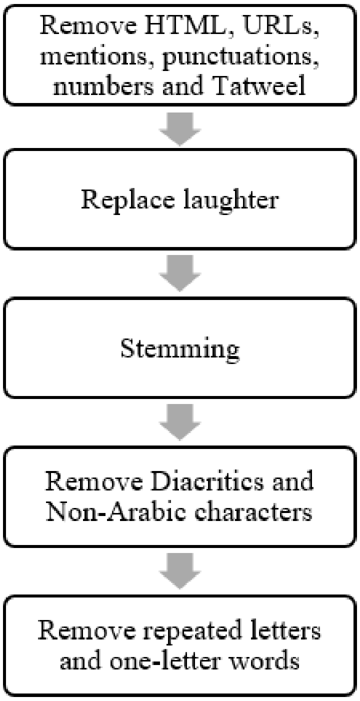
Remove HTML, URLs, mentions, punctuations, numbers and Tatweel
Replace laughter
Stemming
Remove Diacritics and Non-Arabic characters
Remove repeated letters and one-letter words

FIGURE 2. **Preprocessing steps**

Then, we normalized the text by unifying each of these multiple Arabic letters (أ, ة, ي, و) into a single shape. The letters (أ, إ, آ) are converted into (ا) ,(ى, ي) are converted into (ي), the letter (ة) is converted into (ه), and the letters ‘ئ and ؤ are converted to (ء).

C. FEATURE EXTRACTION

Feature extraction constitutes the conversion of raw text into a series of characteristics that can serve as the basis for training a machine learning model. The objective of this process is to pinpoint relevant text patterns and attributes indicative of the expressed sentiment and subsequently represent these patterns in a format that a classifier can interpret. Feature extraction can be achieved either via frequency-based embedding (FBE) or prediction-based embedding (PBE). Over time, FBE gained traction until the emergence of PBEs, which have since seen an uptick in usage. This can likely be attributed to the superior performance demonstrated by PBEs, given their capacity to incorporate semantic and contextual features. The study conducted in [42] showed that PBE techniques could result in higher performances when they are combined with deep-learning models, which in turn perform higher than machine-learning and lexicon-based approaches as per the same study. For these reasons, we adopted two feature extractors in these experimentations: namely AraBERT [43] and DarijaBERT [44]. The choice of these two pre-trained language models is made based on the results obtained in sentiment analysis tasks in previous research works. In [37], the Logistic Regression classifier have reached an accuracy of 77.6% on MSTD dataset by using a frequency-based extractor (TF-IDF), while in [45], a bidirectional LSTM-based classifier performed 83.24% of accuracy on the same dataset using AraBERT. On the other hand, DarijaBERT is based on both MSA and Maghrebin lexicons, making its vocabulary much richer than AraBERT's. Table 3 presents an overview of both feature extractors.

TABLE 3: FEATURE EXTRACTOR CHARACTERISTICS

Model	**AraBERT**	**DarijaBERT**
Vocabulary size	64K	80K
#Tokens	2.5B	100M
Dataset Size	77GB	691MB
#Params	136M	147M
#Training Steps	3M	235K
Embedding size	1024	768

AraBERT is a cutting-edge language model that has been trained on a wide range of Arabic text corpora, partly in Arabic dialect. The training data originates from a variety of sources, including websites, news stories, and Wikipedia. We have chosen to work with AraBERT-Large in this study since it is a larger model than BERT-base and has more parameters, which means it has more capacity to learn complex relationships in the data. DarijaBERT is the first Open Source BERT model for the "Darija" Moroccan Arabic dialect. It has the same architecture as BERT-base, except it does not have the Next Sentence Prediction (NSP) objective. This model was trained on a total of 3 million Darija dialect sequences, totaling 691MB of text or 100M tokens. Despite its significantly larger size, the dataset used to train AraBERT resulted in a vocabulary of 64000 tokens, while DarijaBERT's contains 80000, as Moroccan Dialect has no spelling rules, it is legit to assume that the difference is due to tokens originating from Darija being written in several forms. The authors of DarijaBERT's work have suggested 3 variants of the model: other than the regular DarijaBERT, DarijaBERT-Arabizi is conceived to deal with Arabizi, which is written in Latin characters, and DarijaBERT-mix which can handle both Arabic and Arabizi lexicons. For the purpose of this experiment, we have opted for DarijaBERT-base for the mere reason that the experimentation data is primarily written in Arabic dialects.

D. BASELINE MODELS

To address the challenge of selecting the best algorithms to form the baseline classifiers, we decided to combine BERT-based feature extractor with deep learning models, as research has shown that this approach consistently produces high-performance scores in this previous work [45]**.** As shown in Table 4, we have explored 120 neural network architectures that utilize different variants of RNN layers and the most important parameters such as number of layers and the layer type. This latter has the capability to capture contextual information from the sequence of tokens forming each comment. As a result, a total of 720 different architectures was used for every possible combination of parameters indicated in table 4. These tests were conducted via Grid Search to ensure the testing of every possible combination.

TABLE 4: GRID SEARCH SPACE HYPERPARAMETER VALUES

Parameter	**Values**
Is bidirectional	False, True
Initializer	glorot_uniform, glorot_normal
Number of convolution layers	0,1,2,3,4
Recurrent layer type	GRU, LSTM, SimpleRNN
Global Average Pooling	False, True
Optimizer	Adam

During the training phase, we have split datasets into a training set (80%) and a testing set (20%), and 15% of the training set was then used for validation. Moreover, we have added EarlyStopping and ModelCheckpoint callbacks to mitigate the problem of overfitting, as well as Adam as an optimizer, since it can converge faster than other optimizers while also ensuring an adaptive learning rate. To reduce the inference time during the annotation phase, we have chosen the top 5% performing architectures to execute the voting process, this will also reduce the likelihood of biasing the proposed ensemble approach. Table 5 describes the six selected architectures. The vectorized datasets were fed into the selected architectures, leading to the training of 30 models specifically designed to classify the polarity of a Darija comment. As indicated in Table 5, glorot_uniform was proven to be the best initializer in the sentiment analysis task. Additionally, GlobalAveragePooling was only added in one architecture, while in other cases the last recurrent layer is always set not to return sequences. After applying grid search, we selected 6 architectures presented in Table 5. Afterwards, we fit 30 models on the 5 training datasets. Figure 3 presents the step 4 of the whole approach (see figure 1).

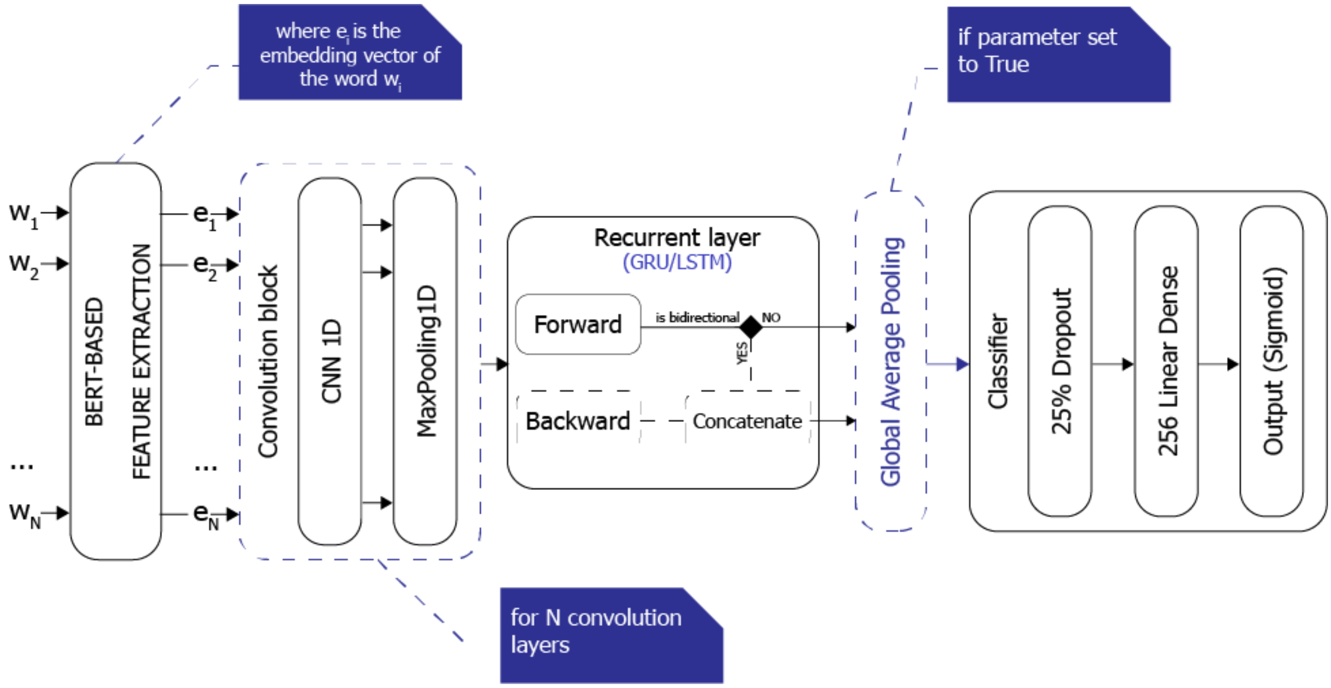

FIGURE 3. **General architecture of baseline models.**

TABLE 5: SELECTED MODELS

Model	Is bidirectional	Initializer	Number of convolution layers	Recurrent layer type	Global Average Pooling
BiGRU	True	glorot_uniform	0	GRU	False
CNN-BiGRU	True	glorot_uniform	3	GRU	False
LSTM	False	glorot_uniform	0	LSTM	False
CNN-LSTM	False	glorot_uniform	3	LSTM	False
BiLSTM	True	glorot_uniform	0	LSTM	False
CNN-BiLSTM	True	glorot_uniform	2	LSTM	True

To clarify table 5, we present a generalized schema of our architectures. Figure 3 is designed to provide a high-level overview of how RNN/CNN neural networks are trained to classify text inputs transformed to vectors using BERT-based feature extractors. As indicated in Figure 3, each token w_i is embedded into a numerical representation e_i using the BERT-based feature extractor, in our experimental framework, we have tested both AraBRT and DarijaBERT in order to capture more tokens from Darija dialect. Then, embedded vectors are used as inputs to train classifiers on categorizing positive and negative. Each model is composed of a stack of convolution blocks, as indicated in Table 5, ranging between 0 to 3. Convoluted outputs constitute the inputs of the recurrent layer in the architecture. In some experiments, we have set the latter to return output sequences which are then averaged by a GlobalAveragePooling layer, such as the case of the CNN-BiLSTM model, whereas other models are configured to directly aggregate the outputs upon computation on the RNN layer level. Finally, we apply a dropout layer of 25% to eliminate potentially noisy features, pass the outputs to a dense layer and calculate the output class using a single-neuron dense layer with an activation function.

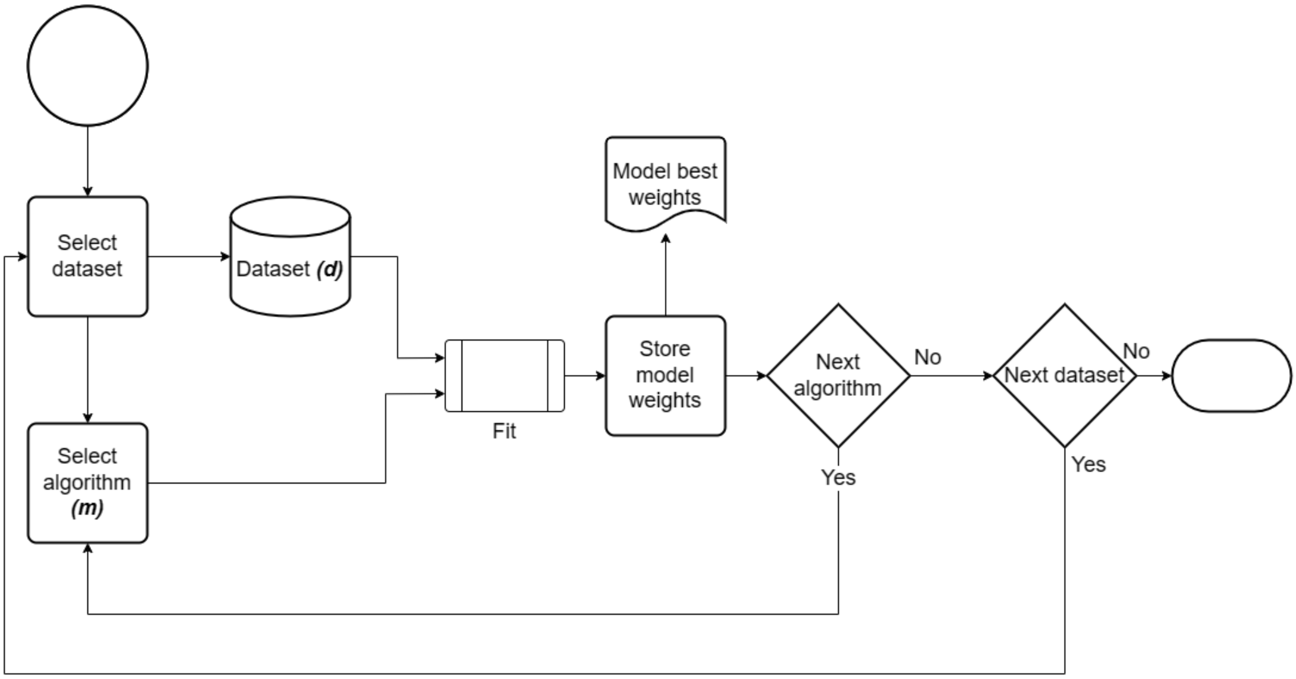

FIGURE 4. Baseline model training steps

The process is repeated for each dataset with the six classifiers. Hence, for each dataset, we trained and saved six distinct models BiGRU, CNN-BiGRU, LSTM, CNN-LSTM, BiLSTM and CNN-BiLSTM. After this step, a total of 30 pre-trained models were obtained. The ModelCheckpoint callback was used to monitor validation loss in order to select the model best weights. By incorporating EarlyStopping and learning rate reduction, we aimed to prevent overfitting and optimize the models' generalization abilities.

E. GENERATING PREDICTIONS

After training baseline models, the fifth step was to produce the predictions from the unlabeled dataset, as explained in Figure 4. Model predictions will later be used as inputs to a linear model whose task is to predict the likeliest label for each comment. It is also paramount to note that we have run the labeling process on each dataset using these models except those trained on the dataset itself. The reason behind this is mainly to avoid biasing the voting technique with a classifier that already knows the patterns of the dataset to be labeled. As shown in Figure 5, the first step of the annotation process is to load the unlabeled dataset, which we will denote by **Δ**. Then for each pre-trained model denoted as m, we load and predict the output classes of **Δ** if m was not trained on **Δ** itself. Predictions of **Δ**’s rows are then stored in order to be used later for the last step of WeVoTe approach

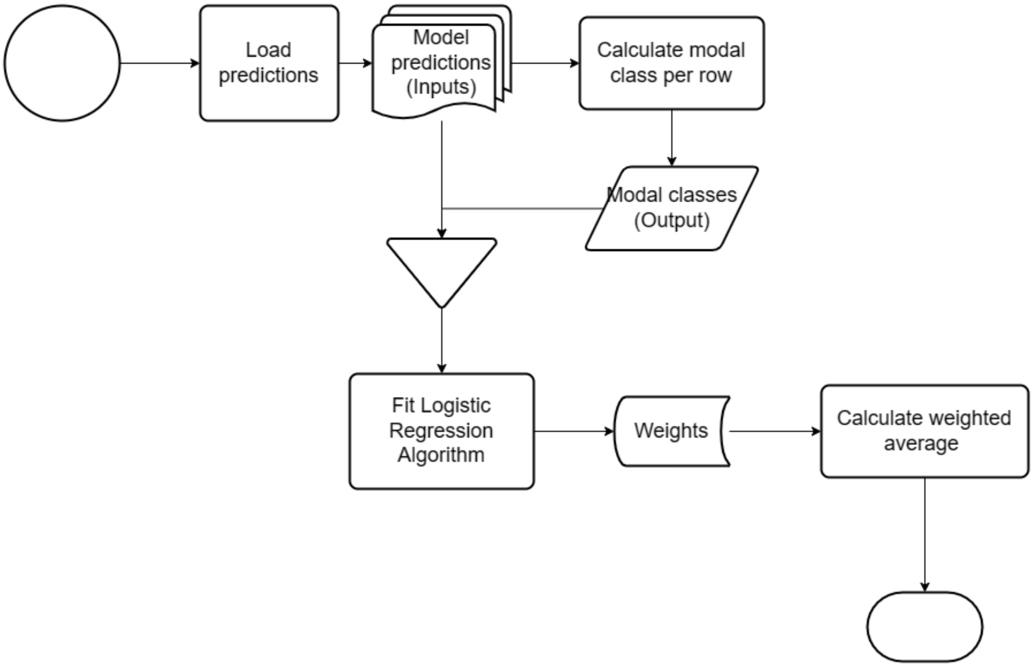
Load predictions
Model predictions (Inputs)
Calculate modal class per row
Modal classes (Output)
Fit Logistic Regression Algorithm
Weights
Calculate weighted average

FIGURE 5. **Generating predictions step for an unlabeled dataset**

In order to evaluate the proposed approach, predicted labels will be compared against ground-truth labels and the corresponding agreement rate will be used to assess the efficiency of each approach. This step involves generating the predictions of the whole unlabeled dataset (Δ) using the generated baseline models that previously fit on five labeled datasets, therefore, each comment of (Δ) has 30 predictions. These predictions will be processed to generate the final label using metric-based and logistic regression based weighting techniques.

F. Weighted Voting Technique

In this section, we present the final step of the propose approach, in which we aggregate the produced predictions from step 5 for labeled dataset into a single value reflecting the estimated polarity score for a comment. The proposed approach relies on its core on computing a weighted arithmetic mean of predicted labels of an unlabeled dataset, where weights can be based on respective performance metrics of each base estimator, or retrieved by training a meta-classifier to predict the class where a comment is likely to belong given base estimator outputs. Coefficients of the trained meta-classifier are used as weights for the arithmetic mean calculation.

1) UNWEIGHTED VOTING TECHNIQUE

This approach was adopted in most literature works; thus, we have decided to use it so we compare its accuracy to the novel approach proposed in this paper. By assuming all models are equally reliable since they were trained in the same conditions and datasets, each output would have the same importance, regardless of how good the model is in distinguishing positive comments from negatives. Predictions are aggregated by calculating their average, as explained by Equation 1; finally, the value of $f_{\Delta(x)}$ is rounded to the closest integer value.

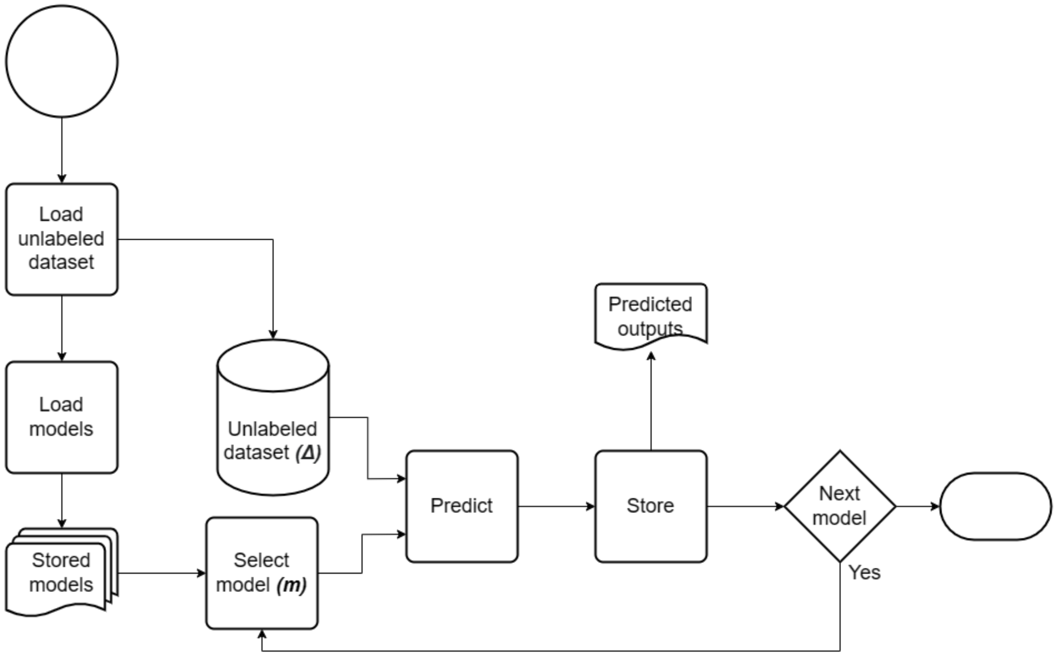

FIGURE 6. Vote aggregation step

$$f_{\Delta}(x) = \frac{\sum_{m\in M}\sum_{d\in D}^{d\neq\Delta} pred_{(m,d)}(x)}{N} \qquad \textit{Equation (1)}$$

Where N is the number of predictors that were not trained on the target dataset ***Δ***, and pred(m, d) is the function that predicts the class of a comment x, using a model m trained on each dataset d from the set of datasets D. For each dataset except the one we are trying to label, we aggregate all of the model's predictions via the average function, where all weights are equal, thus giving the same importance to all the classifiers. The function starts by adding up prediction values from each model, which we denote by $\text{pred}_{(m,\ d)(x)}$ for each model m trained on dataset d, where d is different from the testing set ***Δ***, Then we divide by the N, the total number of models used for labeling.

2) METRIC-BASED WEIGHTING

Similarly, this approach consists of aggregating model outputs by averaging them. However, the difference resides in using weights for the models. This approach assumes that models are not equally performant, so we are using their performance metrics during the training of the baseline models since weights tend to lower the importance of underperforming models. In this experiment, we initially considered using accuracy as a weighting coefficient, but the notable imbalance in the datasets imposes the usage of precision, recall, and most importantly F1-score as coefficients as well in order to alleviate the bias that can be caused by this imbalance. This is explained in Equation 2.

$$f_{\Delta}(x) = \frac{\sum_{m\in M}\sum_{d\in D}^{d\neq\Delta} w_{m,d} * pred_{m,d}(x)}{\sum_{m\in M}\sum_{d\in D}^{d\neq\Delta} w_{m,d}} \qquad \textit{Equation (2)}$$

$w_{m,d}$ is the performance score of the model, depending on the weighting approach that is being used, which in this case are the performance metrics.

3) LOGISTIC REGRESSION-BASED WEIGHTS

Unlike the previous approaches and those of the literature review in the context of the voting techniques used for automatic annotation. The contribution of this novel approach relies on fitting the predictions of the baseline models as inputs in addition to the most frequent prediction (mode) as the target variable, using a ML algorithm as a meta learning. Since this algorithm, aims to find the best coefficients that can match the most predicted label for each text (Equation 3).

$$f_{\Delta}(x) = \frac{\sum_{m\in M}\sum_{d\in D}^{d\neq\Delta} \theta_{m,d} * pred_{m,d}(x)}{\sum_{m\in M}\sum_{d\in D}^{d\neq\Delta} \theta_{m,d}} \qquad \textit{Equation (3)}$$

Parameters $\theta_{m,d}$: the coefficients found of the fitted logistic regression model.

Figure 6 presents the process we suggest to label new dataset using Logistic Regression weighting. The process starts by loading the annotations of each model for the dataset targeted. Next, it determines the most frequent annotation by rounding the prediction scores to the nearest integer value. Subsequently, the Logistic Regression model is trained to predict the most frequent label based on the predictions made by the models. LR coefficients are then used as weights to calculate the average annotation of a given comment, resulting in a newly assigned label.

The fundamental element of this methodology lies in the ability of logistic regression to accurately identify coefficients that delineate the correlation between individual classifiers and the resultant vote. This process prioritizes predictors with greater dependability relative to a preponderance of forecasted labels. Concurrently, it minimizes the influence of less pertinent models by allocating them lower coefficients. Additionally, unlike metric-based weighting, logistic regression coefficients are insensitive to the performance of the models; consequently, non-generalized or overfitted models do not mislead them.

IV. Experimental results and analysis

In this section, we present and analyze the results of the overall steps of the annotation approach in accordance with a set of sensitive parameters, which we will present throughout the analysis. This will help future readers reproduce the experimentation in their research works. For each dataset, we have trained the six architectures mentioned in the methodology section, while considering using both AraBERT and DarijaBERT on preprocessed and non-preprocessed versions of the datasets, as well as the weighted voting techniques. The reason behind choosing to experiment with two different feature extractors is that word-level processing is prone to producing out-of-vocabulary tokens, consequently reducing the classifier's performance.

A. Baseline models results

In this section, we present the best accuracy scores obtained for each model being trained on every dataset in this experiment. The comparison was done after experimenting with both AraBERT and DarijaBERT, as well as datasets being either preprocessed or un-preprocessed. As indicated in Table 6, the best performance achieved in each of ElecMorocco2016, FB, and MAC was obtained via embedding the non-processed version of respective datasets using DarijaBERT. However, for MSAC and MSDA datasets, the best accuracy scores were reached using AraBERT. MSTD on the other hand was the only dataset that required preprocessing and recorded the highest accuracy with the DarijaBERT embedding.

In addition, BiLSTM model often outperforms other architectures, while LSTM model has not been the best predictor in any case. This is due to the bidirectional nature of the model, which helps capture more contextual information from the comment. Generally, the only case where a unidirectional model records the best performance is in the MAC dataset. However, the usage of CNN within an architecture is proven to be effective only in the case of LSTM itself, as it increases its performance by 0.43%, while it reduces the accuracies of BiGRUs and BiLSTMs respectively by 0.06% and 0.13%. This loss of performance is caused by the loss of sequential information due to the use of CNN, which captures position-invariant features in the data.

TABLE 6: MAX ACCURACIES OF BASELINE MODELS

Dataset	BiGRU		CNN-BiGRU		LSTM		CNN-LSTM		BiLSTM		CNN-BiLSTM	
	AraBERT	DarijaBERT	AraBERT	DarijaBERT	AraBERT	DarijaBERT	AraBERT	DarijaBERT	AraBERT	DarijaBERT	AraBERT	DarijaBERT
ElecMorocco 2016	82.69	81.57	82.44	82.30	82.54	80.00	82.64	80.30	82.93	**83.27**	82.30	81.47
FB	85.80	**90.13**	87.40	89.70	86.80	88.43	85.40	89.00	87.80	89.42	87.00	**90.13**
MAC	89.02	89.55	88.77	90.55	87.64	91.06	87.53	**92.44**	88.27	90.68	89.28	90.81
MSAC	88.35	85.82	88.10	88.60	88.86	84.81	89.11	86.10	**89.62**	87.34	**89.62**	86.10
MSDA	**91.25**	87.40	**91.25**	89.42	89.06	88.28	89.06	89.04	90.62	89.79	90.62	88.66
MSTD	85.10	86.01	83.81	85.05	83.53	85.32	83.81	85.60	83.67	**86.83**	85.18	86.15

It is worthwhile to mention that the proposed architectural models have attained state-of-the-art (SOA) results on several datasets, notably ElecMorocco2016 and MSTD, as detailed in Table 6. Compared to references [36] and [37], our models show an average accuracy improvement of 3%. Furthermore, in the case of MSAC [35], FB [38], and MAC [39], we observed an average performance enhancement of 6%. Most notably, for MSDA [40], there was a significant performance boost of 14%. These improvements serve to strengthen the efficacy of the proposed annotation methodology.

B. WeVoTe Approach results

In the following subsections, we will present the overall results to prove the efficiency of the approach, and analyze the effects of the most important criteria that influence the proposed approach in order to narrow the field of possible approaches, as well as help future researchers. These criteria include dataset-related characteristics, such as the size and the balancing of both positive and negative classes, as well as investigating the impact of the embedding and the preprocessing we suggested. After aggregating each dataset's predicted labels using one of the weighted voting techniques, we compared the resulting labels against the original labels of the dataset itself in order to measure the agreement rate of the annotation approach by calculating the percentage of how many times the annotation was correct. Table 7 summarizes the performance scores of the annotation approach with different sets of parameters. In this context, our objective was to determine the label's value. Initially, we aggregated the outputs of the baseline models by computing their mean. Nevertheless, less optimal models unevenly influenced the derived labels, even though these models were given the same level of consideration as the more advanced foundational models. Consequently, we transitioned to a metric-driven technique, emphasizing the outputs from higher-performing models over their less efficient counterparts. Although this method outperformed the non-weighted average, the limited range of metric values prevented its universal application. Thus, we considered an alternative method that autonomously ascertains the optimal weights using a meta-classifier.

TABLE 7: ANNOTATION AGREEMENT RATE SCORES

Dataset		Feature extractor	Weighting Approach					
			Without	**Accuracy**	**Precision**	**Recall**	**F1-Score**	**Logistic Regression**
ElecMorocco2016	Preprocessed	AraBERT	77.82	77.75	77.76	77.77	77.76	**77.96**
		DarijaBERT	78.5	78.58	78.54	78.64	78.62	**79.12**
	Unpreprocessed	AraBERT	79.03	79.01	79.00	79.15	79.12	**79.19**
		DarijaBERT	79.35	79.31	79.33	79.38	79.36	**79.65**
FB	Preprocessed	AraBERT	87.38	**87.38**	87.38	87.38	87.38	**87.38**
		DarijaBERT	89.53	**89.67**	89.61	89.55	89.58	88.23
	Unpreprocessed	AraBERT	87.63	87.58	**87.63**	87.61	87.63	**87.63**
		DarijaBERT	91.25	91.19	**91.25**	91.22	91.22	91.11
MAC	Preprocessed	AraBERT	76.34	76.41	76.36	**76.46**	76.44	75.25
		DarijaBERT	73.13	73.61	73.61	**73.81**	73.74	69.37
	Unpreprocessed	AraBERT	77.22	77.25	77.22	**77.4**	77.35	75.76
		DarijaBERT	84.5	84.73	84.66	**84.93**	84.91	82.36
MSAC	Preprocessed	AraBERT	86.01	**86.16**	86.11	86.16	86.11	86.01
		DarijaBERT	83.68	83.68	83.68	83.73	**83.73**	83.07
	Unpreprocessed	AraBERT	87.49	87.49	87.49	87.54	**87.54**	87.34
		DarijaBERT	86.28	86.23	86.33	86.33	**86.33**	86.03
MSDA	Preprocessed	AraBERT	81.65	81.84	81.97	81.97	81.84	**82.22**
		DarijaBERT	82.34	82.4	82.34	82.34	82.34	**84.85**
	Unpreprocessed	AraBERT	84.85	84.97	84.91	84.85	84.85	**85.72**
		DarijaBERT	80.28	80.15	80.15	80.15	80.15	**83.22**
MSTD	Preprocessed	AraBERT	83.19	83.19	**83.22**	83.19	83.19	83.14
		DarijaBERT	83.52	83.49	**83.60**	83.55	83.55	83.33
	Unpreprocessed	AraBERT	81.32	81.35	81.3	81.38	81.38	**82.18**
		DarijaBERT	79.15	79.35	79.26	79.15	79.32	**83.16**

[2] https://github.com/YassirMatrane/WeVote-Datasets

It is worthwhile to demonstrate an illustrative example of the proposed approach using an unlabeled comment “ عجبتني الفكرة ديالك”, which will be fed to the 36 models, resulting 36 predictions of 30 positives and 6 negatives (see figure 4), which in turn will be treated by the stacking technique (see figure 5). The outcome of the whole pipeline execution returned a positive label. Similarly, providing the labeling system with the comment “واش هادي سياسة تاع دولة كتحتارم راسها؟” resulted in a negative label.

It is constructive to note that the evaluation of the proposed annotation techniques was applied on the whole dataset. In this context, some of these results surpassed the results of benchmark results of sentiment analysis works, which were evaluated only on a portion of dataset.

The best performance recorded in ElecMorocco2016 is 79.16%, while other datasets mostly reach a top score ranging between 83% and 86%, except for FB where the highest performance goes up to 91%. By looking at the best combination between preprocessing and feature extraction, we notice that most datasets have been more accurately annotated when they are directly embedded with DarijaBERT. The two exceptions were MSTD, which required preprocessing to strengthen the prediction process even further, and MSDA which recorded a higher performance using AraBERT. We have experimented two weighting approaches, the first is based on multiple metrics (accuracy, recall, precision and F1-score), while the other is based on a meta classifier (LR), which makes it six different weighting techniques to identify the polarity of a certain comment, which performance scores are presented in Table 7 and summarized in Figure 7.

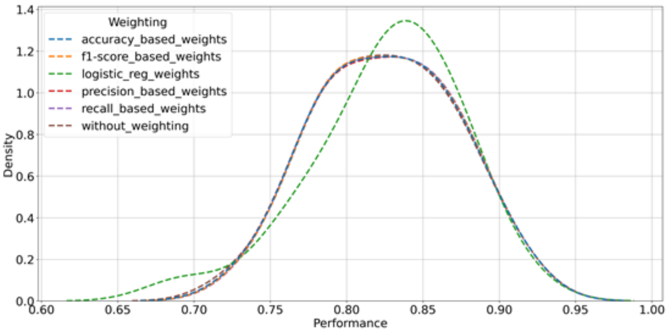

FIGURE 7. **Accuracy variation depending on weighting techniques**

As shown in Figure 7, the distribution of performance scores recorded while using logistic regression-based weights (presented in green) are higher estimator by an average of 2%, compared to the other techniques, which give almost identical results. This is caused by weights being chosen by the LR estimator itself, hence finding the best parameters according to the most frequent annotation given by DL architectures. We justify behavior similarity across every approach by the fact that they are all based on measuring the performance of the same experiment, using metrics that rely on the same four variables: True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN). In general, models with good performances have these metrics converge towards 1, which results in almost equal coefficients, consequently, no weighting is applied to the votes. Meanwhile, Logistic regression finds the best linear mapping between each classifier’s output and the majority vote. This has 2 main advantages:

- Models trained on imbalanced datasets are likely to be overfitting, hence, the previously used performance metrics may not be entirely reliable.
- An external fact other than performance may be the best coefficient to use while annotating a dataset via weighted voting. A linear model can be a great tool to consider to find these weights which do not have to be explainable.

Figure 8 shows the percentage of cases where each approach has the highest performance.

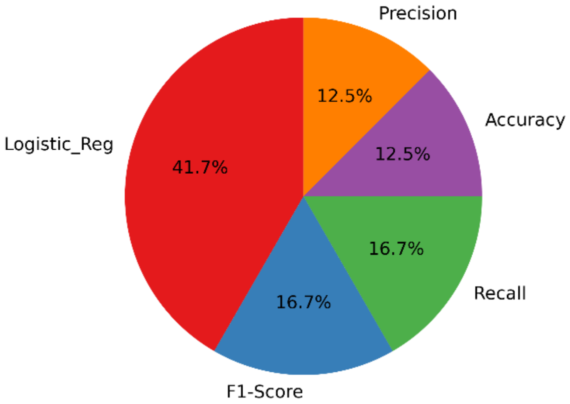

FIGURE 8. **Most performant approaches**

The Logistic-Regression-based weighting approach has outperformed any other approach in 41% of cases as shown in Figure 8, which further proves the strength of the proposed approach. The remaining cases can be interpreted by the imbalance in the dataset, such as in MAC, which constitutes the 16.7% of cases. The recall measure has the highest performance, as well as the F1-Score, which is 75% of the time the highest performing approach in MSAC, the most balanced dataset.

C. Dataset effect

In this subsection, we explore different features of each dataset to study their effect on the overall performance of the automatic annotations’ techniques. These features include average comment length, dataset size, out-of-vocabulary word rate, and positive to negative comments rate. Figure 9 presents the overall performance of automatic annotation techniques applied to each dataset.

As we can see, the highest performances were recorded on FB, followed by MSAC and MSTD, while MAC has a relatively low performance. Moreover, MSAC and ElecMorocco2016's scores are more stable, thus less affected by the model. Since Arabic dialects have a high level of inflection and spelling differences, we decided to study the out of vocabulary (OOV) rate of each feature extractor in order to supervise the loss of information caused by the absence of a spelling standard in Moroccan dialect. Moreover, our approach might be impacted by the imbalanced dataset in which the models were trained, therefore, we included in our study the ratio of positive comments to negative ones. Furthermore, the comment length may impact long-range dependencies between highly influential tokens. The most important findings can be linked to the previously mentioned features. Table 8 summarizes the statistics of the accuracy variation per dataset, in order to reinforce the plot findings.

The datasets differ on several aspects, such as the size and the out of vocabulary in the used feature extractors. Table 9 summarizes the values of investigated features per dataset.

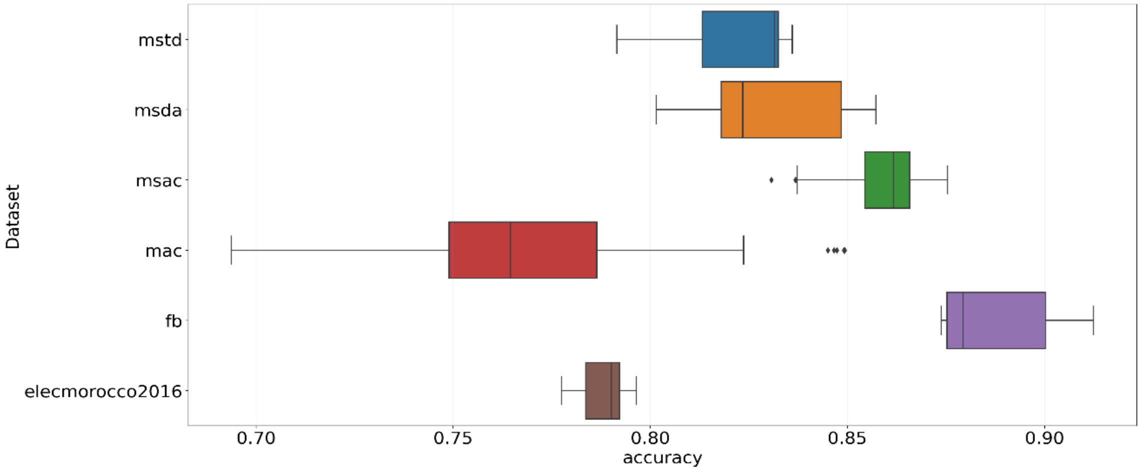

FIGURE 9. Annotation accuracy variation per dataset

TABLE 8: SUMMARY STATISTICS FOR PERFORMANCE PER DATASET

Dataset	Mean	Std	Min	25%	50%	75%	Max
ElecMorocco2016	78.74%	0.62	77.75%	78.36%	79.01%	79.22%	79.65%
FB	88.89%	1.59	87.38%	87.53%	87.93%	90.03%	91.25%
MAC	77.62%	4.39	69.37%	74.89%	76.45%	78.64%	84.93%
MSAC	85.86%	1.45	83.07%	85.44%	86.16%	86.58%	87.54%
MSDA	82.60%	1.79	80.15%	81.79%	82.34%	84.85%	85.72%
MSDT	82.02%	1.66	79.15%	81.32%	83.15%	83.25%	83.60%

TABLE 9: VALUES OF INVESTIGATED FEATURES PER DATASET.

Dataset	POS/NEG ratio	Size		OOV rate		Average comment length
		Positives	Negatives	AraBERT	DarijaBERT	
ElecMorocco2016	0.56	3673	6581	0.62	0.51	14.57
FB	0.24	684	2858	0.67	0.41	16.75
MAC	3.04	2987	982	0.52	0.28	8.40
MSAC	0.99	981	994	0.50	0.42	12.51
MSDA	0.33	397	1200	0.57	0.40	15.15
MSTD	0.31	868	2773	0.66	0.45	15.87

The OOV rates in AraBERT are generally higher than DarijaBERT, since the latter has a larger vocabulary, thus capturing more words. We can also see how MAC has the highest positive-to-negative ratio value, specifically, for each negative comment there are 3 positives, which is the opposite for other datasets, being more dominated by negative tweets. The average comment length on the other hand is highly impacted by the nature of platforms from which comments were collected. For instance, the only dataset which has no comments collected from Twitter is FB and it has the highest average comment length. The character limit imposed by the platform of Twitter is then highly impacting the content of studied datasets. We can conclude from Table 8 that MAC has a higher positive-to-negative ratio, while it has a low performance compared to the other datasets. Thus, the performance drop is explained by using models trained on highly imbalanced datasets where positive tweets constitute the majority class, as shown in Figure 10.

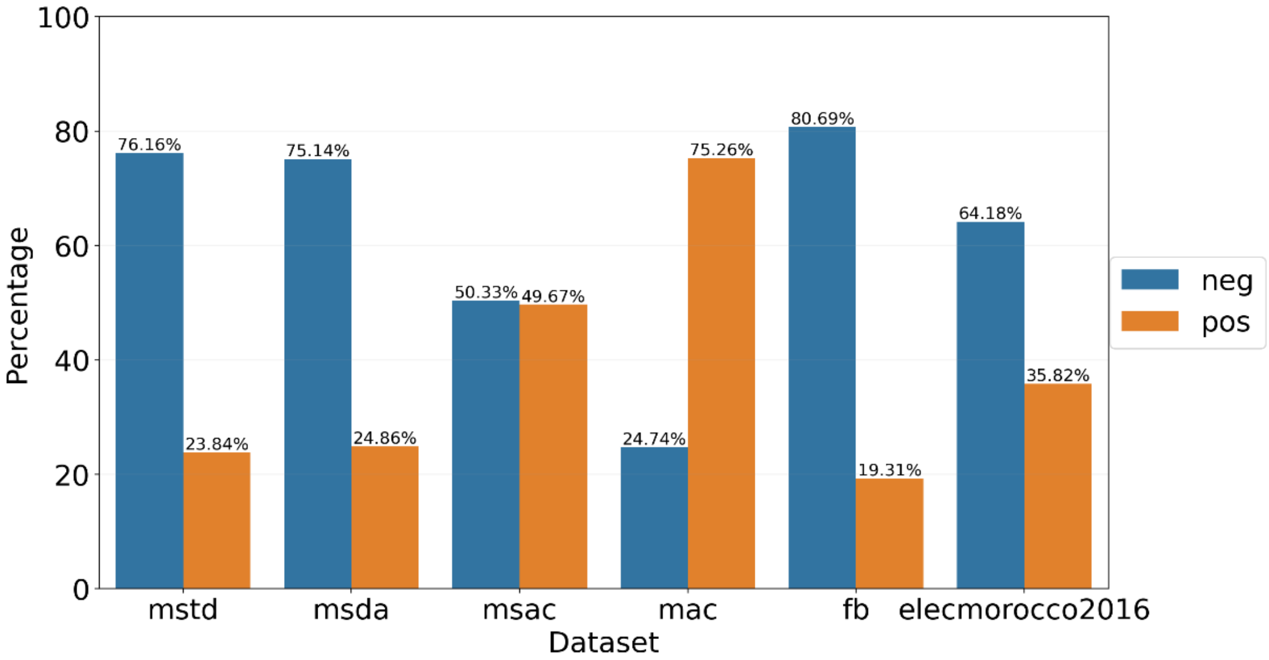

FIGURE 10. The distribution of sentiment labels in each dataset

Figure 10 suggests that, five of six benchmark datasets are biased towards negative samples, with the highest rate in FB, which has 80% negative labeled data points, followed by MSTD and MSDA, where 75% of data is negative.

ElecMorocco2016 comes next with 64% of negative comments. MSAC is the most balanced dataset with 49.67% and 50.33% of negative and positive comments respectively, which is reflected in the stable annotation performance despite not being the highest. The reason is as indicated in Table 2, MSAC is a multi-topic dataset that contains commonly used words in the Moroccan dialect. On the other hand, ElecMorocco2016 is the second-most balanced dataset, however, it has a relatively low annotation accuracy score. This may be explained by the fact that ElecMorocco2016 is a politics-related dataset, moreover, it is the largest among the six chosen datasets. This can result in a high rate of words related to politics-lexicon which do not figure in the rest of the datasets. In an opposite scenario, a dataset containing tweets from various topics may be used in the training phase to label a domain-specific sentiment dataset. However, this could potentially result in a drop in performance as technical words with high polarity are likely to be abundant in the unlabeled dataset. FB dataset is expectedly the highest-performing dataset since it is the most biased towards negatives, hence capturing True Negatives is way easier compared to the rest of the datasets. To conduct a more detailed evaluation of each model's performance using these metrics, we undertook an analysis of the confusion matrix of annotations for each dataset, as depicted in Table 10.

TABLE 10: CONFUSION MATRIX VALUES FOR EACH DATASET

Dataset	True Negatives	False Negatives	False Positives	True Positives
ElecMorocco2016	60.85%	17.92%	3.34%	17.90%
FB	79.07%	9.46%	1.64%	9.83%
MAC	22.07%	19.70%	2.67%	55.56%
MSAC	44.93%	8.67%	5.43%	40.97%
MSDA	64.33%	6.52%	10.85%	18.31%
MSTD	69.30%	11.07%	6.87%	12.75%

In table 10, we present the percentages of each dataset's comprehensive annotations. Figure 10 demonstrates that, on average, models can identify negative comments (TN) significantly more easily than positive comments (TP), with the exception of MAC. True Positive and False Negative rates recorded in MSTD, FB, and ElecMorocco2016 are quite comparable, necessitating a closer examination of the influence of both parameters on the overall labeling quality. Even though ElecMorocco2016 is more balanced, identifying positives is more difficult for models trained on it than on FB dataset. Nevertheless, considering the extent of each dataset according to Table 2, the three largest datasets are ElecMorocco2016, MSTD, and FB. The small discrepancy between (TP) and (FN) for ElecMorocco2016 can be explained by the size, leading one to believe that the size has a direct effect on (TP) and (FN) values. Nonetheless, the smaller difference observed between FB and MSTD, despite the minor size difference, refutes this hypothesis. FB dataset is significantly more unbalanced than MSTD and less effective at identifying positive tweets. MSAC has a nearly identical proportion of 44.93% TN and 40.97% TP; therefore, the ratio of positive to negative comments appears to be the most influential variable in this study.

D. Preprocessing and feature Extraction effects

Preprocessing data is a step that aims to improve the quality of data by eliminating noisy elements, such as stop words from the text, or normalizing other tokens to their initial states, for instance, resetting verbs to their infinitive form. As colloquial Arabic has no specific spelling rules, a standard form of writing is likely to decrease the probability of OOV token presence, which can be an existing word written in a different spelling. For this reason, we decided to study the effect of preprocessing on the overall performance of our classifiers. In our case, usage of preprocessing caused an average decrease in performance by 2% as shown in Figure 11. The reason is that many tokens after being processed with acoustic and rule-based functions may have introduced new forms, which are likely not to exist in each vocabulary. If this were to occur, several tokens would identify as OOV, thus reducing the performance of the model.

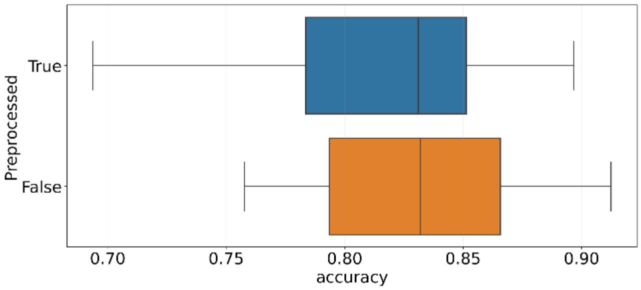

FIGURE 11. Accuracy variation depending on preprocessing

In figure 11, we present the variation of accuracy scores in respect of the application of preprocessing. In our case, usage of preprocessing caused an average decrease in performance by 2% as shown in Figure 11. The reason is that many tokens after being processed with acoustic and rule-based functions may have introduced new forms which are likely not to exist in each vocabulary. If this were to occur, several tokens would identify as OOV, thus reducing the performance of the model. The average accuracy is unaffected by preprocessing, however, the first and third quartiles show that preprocessing decreases the overall performance of labeling in general by a factor of 1%, a decrease in performance is probable in this context, especially when the studied dataset is in an unstructured dialect such as Moroccan dialect, which has no defined linguistic rules to follow.

To further validate the previous finding, we studied the same accuracy variation for each dataset in Figure 12 to explain in more detail this phenomenon and show the difference of performance for each dataset.

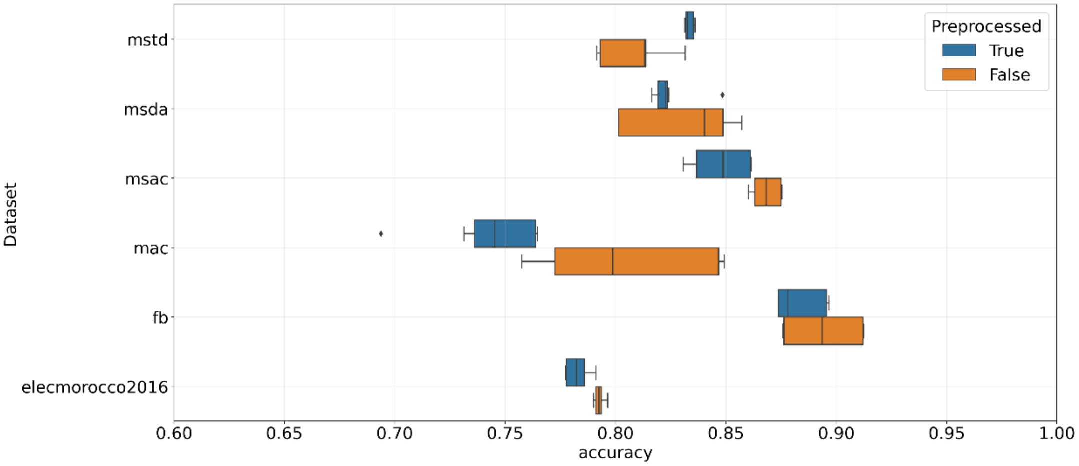

FIGURE 12. **Accuracy variation depending on preprocessing for each dataset**

Figures 11 and 12 demonstrate that non-preprocessed data can yield superior results. However, the improvement in efficacy varies from dataset to dataset. The MAC dataset, for instance, performs 6% better without preprocessing, whereas ElecMorocco2016 only improves by 1%. The unstructured nature of Moroccan Dialect, the type of content comprising each dataset, and the average comment length may have contributed to this performance drop. MAC has the shortest average comment length among benchmark datasets; consequently, comments are susceptible to further reduction due to preprocessing steps that involve token deletion, such as stopword and emoji removal. Consequently, some remarks may contain few words, making them difficult to categorize. In terms of the provenance of comments, datasets such as ElecMorocco2016 and MSDA are the least affected because their content is derived from a formal source: Hespress, as opposed to other datasets in which the majority of comments are typed by the average user and therefore do not adhere to any linguistic rules that can lead to high-quality preprocessing. As this study concentrates on the classification of Moroccan dialect text, we should normally employ a Darija-trained feature extractor, such as DarijaBERT. However, it is well known that Darija is primarily derived from Arabic, so we decided to also test AraBERT to determine which method is most effective. Almost every preprocessing technique (Stop words, Word Embeddings, and Stemming) is designed for MSA, so they perform poorly on dialect-based datasets. Nonetheless, DarijaBERT produced the greatest results (figure 14), as its vocabulary is better suited to the datasets we chose.As the preprocessing phase is likely to produce out-of-vocabulary (OOV) tokens, we have studied the effect of such an outcome regarding each feature extractor. For each dataset, we calculated the amount of OOVs before and after the preprocessing, as presented in Figure 13.

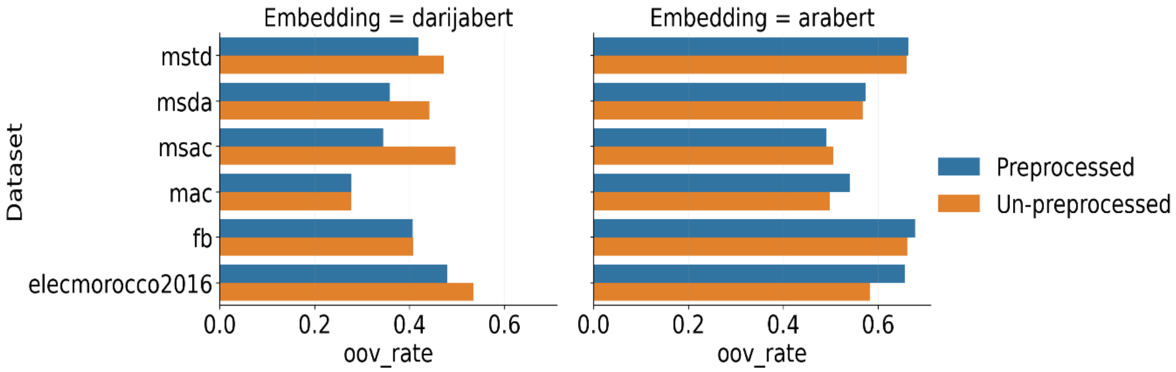

FIGURE 13. **OOV rates per dataset**

Regarding OOVs, the preprocessing pipeline has been shown to be effective as it has helped reduce OOVs by processing various cases (Ex: conjugated verbs, plurals, etc.). The remaining OOVs are likely misspellings or proper names. Moreover, the preprocessing reduces the rate of OOV utterances in every dataset, particularly when DarijaBERT is used as the feature extractor; however, we cannot demonstrate the same for AraBERT embedding as it was not designed for colloquial Arabic. Even more OOV tokens emerged in some datasets that are most likely dialectal words in their original form that do not appear in AraBERT's vocabulary. This observation can be explained by the nature of the topics included in each dataset, for example: ElecMorocco2016 is a topic-centered dataset whose primary topic is politics; consequently, the majority of its terms are derived from MSA, rendering the preprocessing technique ineffective with respect to OOVs when the employed vocabulary is itself in MSA. The vocabulary of DarijaBERT (80,000) is significantly larger than that of AraBERT (64,000), resulting in a reduced rate of OOVs when compared to AraBERT combined with preprocessing (a smaller reduction in OOVs rate). Some datasets contain Arabizi and words from foreign languages (Example: MSTD), causing many tokens to be OOVs and unusable by the used pipeline (Example: "Ach bghiti" cannot be stemmed). Since DarijaBERT contains the majority of AraBERT's tokens, as well as a few colloquial words and affixes, its vocabulary is anticipated to be larger. Figure 14 demonstrates that the use of a large vocabulary reduces the likelihood of obtaining OOVs, thereby preventing information loss, which influences the quality of annotation.

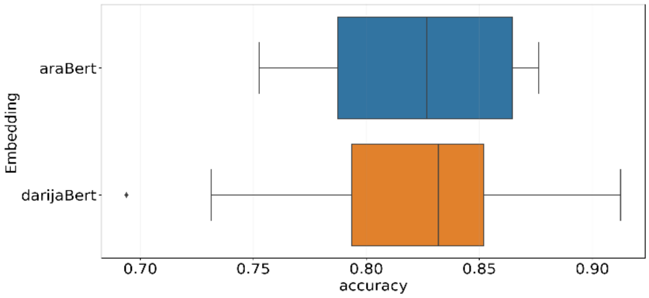

FIGURE 14. Accuracy variation depending on embedding

Figure 14 shows how AraBERT has more performance stability than DarijaBERT, thus the classifiers are more impacted by the larger vocabulary of DarijaBERT, which captures and identifies more tokens. In this context, DarijaBERT has an average OOV of 0.41 of all the datasets, while AraBERT has 0.59. However, this statement cannot be considered valid for every dataset, as denoted by Figure 1

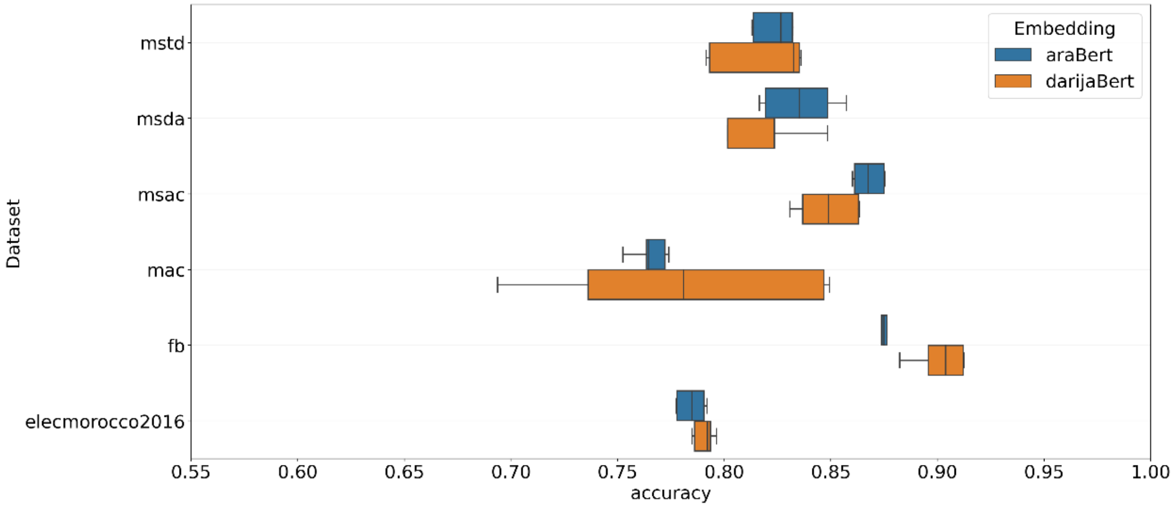

FIGURE 15. Accuracy variation depending on embedding per dataset

We can perceive how AraBERT yields a more stable performance compared to DarijaBERT as per Figure 15. The reason behind this stability is that the preprocessing approach is less effective with that specific embedding, as shown in Figure 13, thus not altering datasets to the same extent in the case of DarijaBERT.

Depending on the used embedding, data sources can also have a significant role in determining a model's final performance. Twitter-based datasets are more likely to be in MSA, whereas Facebook-based datasets are more likely to be in 'Darija', given that the majority of Moroccans use Facebook as their primary social media platform. Therefore, when evaluating the quality of a dataset, it is essential to consider the informal nature of Facebook posts and their propensity to contain spelling errors. In comparison to more formal sources such as HESPRESS, Facebook datasets may be more likely to reflect subjective opinions rather than objective facts. This can introduce a bias toward a particular viewpoint on a given topic, which can impact the overall polarity of the dataset and, in turn, the labeling process. Figure 16 illustrates fluctuations in performance based on the origins of the dataset.

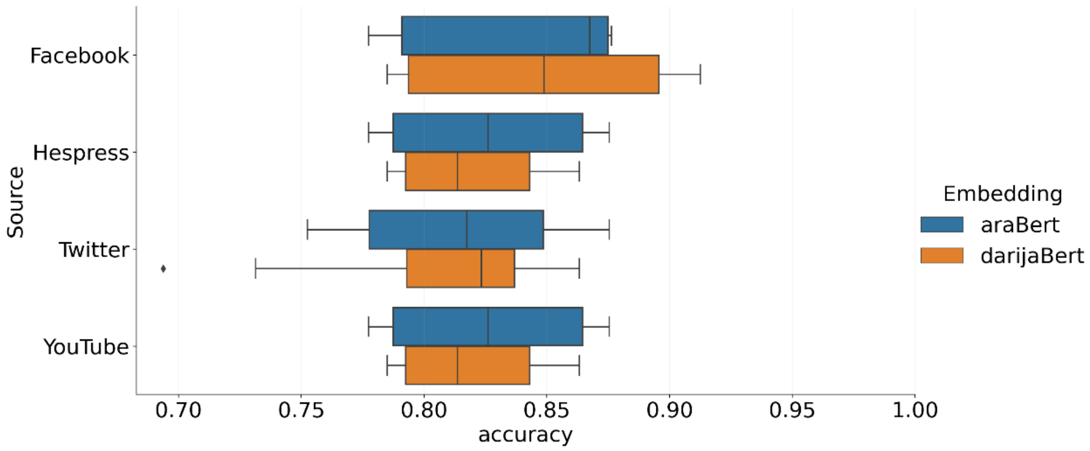

FIGURE 16. **Accuracy variation for each data source**

Figure 16 demonstrates that AraBERT performs better in every scenario except for Facebook data, which can be explained by the fact that content collected from Facebook is predominantly in colloquial Arabic, in contrast to Twitter, YouTube, and Hespress. Because only datasets containing Hespress comments also contain YouTube comments, the results are comparable. In contrast, Twitter data is prevalent among the investigated data. Facebook, on the other hand, is one of multiple data sources for two datasets, but the only source for the FB dataset. Due to the nature of the Facebook user base in Morocco, who tend to use it as their primary social media platform, the majority of its content is written in 'Darija'. As a result, DarijaBERT is able to achieve a higher accuracy score than AraBERT, whose performance is limited to 87%.

V. Discussions

In conclusion of this investigation, it is found that the automated labeling methodology proposed exhibits considerable potential in labeling tasks of the Moroccan dialect, despite its restricted vocabulary. The core innovation of our methodology is anchored in the adoption of a groundbreaking stacking technique. Additionally, to the best of our knowledge, this is the first initiative to develop such an automated annotation approach for Moroccan dialect. By incorporating a wider range of datasets to produce more baseline models. This expansion will enable our system to cover a wider array of themes and subtle linguistic elements inherent in the dialect, thus significantly improving both the practicality and effectiveness of our sentiment analysis annotation system. Furthermore, compared to the majority of works in this context, we evaluated our approach on multiple Moroccan datasets via the agreement rate. Although our method is applied to Darija, it can be applied to other structured or unstructured languages while adapting the preprocessing and feature extraction parts suitable for the used language. This suggests that its effectiveness could be amplified in languages with more varied vocabulary. An interesting observation is that a dataset with greater topic variation may lead to more accurate annotation, but the most influential factor remains the balance between positive and negative comments. Hypothetically, a balanced dataset should deliver superior annotation outcomes.

Nevertheless, the achievement of this method in text classification tasks does not imply similar success in annotating sequence-to-sequence tasks like Part-of-Speech tagging, Named Entity Recognition, or Dependency Parsing. Future endeavors will look into the viability of using this approach for such intricate tasks and extending its application to different areas of AI, such as computer vision or time-series forecasting. Incorporating graph-based embedding and attention layers could be another research direction to explore, with a view to enhancing the robustness of the proposed approach.

Our proposed method addresses several drawbacks commonly associated with traditional sentiment analysis techniques. Firstly, unlike the lexicon-based technique that requires constructing a lexicon, our approach benefits from the use of multiple models generated from diverse datasets, eliminating the need for lexicon construction. This enhances the method's flexibility and applicability to various contexts. Secondly, we avoid the pitfalls of the user rates technique, which is prone to human bias and manipulation, such as trolling. Our method operates independently of human intervention, relying solely on machine learning algorithms and data-driven insights. Thirdly, our approach circumvents the issue of misinterpretation of satirical content, which can skew results in reader-based methods. Instead, it bases its analysis on dataset knowledge, ensuring objective and consistent sentiment analysis. Finally, our method outperforms manual annotation in terms of speed and scalability. While automatic annotation may sometimes be less accurate than manual annotation, it offers significant advantages in processing large volumes of data rapidly and efficiently, making it more suited for handling extensive datasets and real-time analysis.

Furthermore, the proposed methodology is not devoid of limitations. It is not universally applicable, implying potential shortcomings when applied to datasets encompassing diverse Arabic dialects. Additionally, its efficacy might be compromised when utilized in tasks other than text classification. To encapsulate the vast array of words spanning various topics and dialects, the proposed approach necessitates training on an even broader range of datasets.

The following measures can address the noted challenges and further enhance the fully automated annotation process:

- Negation is a common challenge in sentiment analysis, leading to potential errors. Therefore, advancements in this domain would directly augment the quality of annotations produced by this approach.
- Exploring more advanced neural network architectures could improve baseline model performance.
- Investigating new strategies to handle imbalanced datasets.
- Implementing a continuous learning approach to label new comments and manually verifying the annotation when the model's confidence is low could be beneficial. This active learning process would enable the model to adapt in real time to incoming data streams.
- Certain dialects possess specific words that provide immediate insight into the overall polarity of a comment, justifying why lexicon-based approaches sometimes yield high performance.
- Employing a reinforcement learning approach could mitigate annotation challenges to a certain degree.
- Using larger datasets for the Moroccan dialect could enhance the coverage of Moroccan words.

VI. Conclusion and future work

In summarizing this study, we have navigated the cutting-edge of annotation process within sentiment analysis, especially when it comes to its application to the Moroccan dialect, in addition to the associated difficulties with annotating sizable datasets, which can be an arduous and time-intensive undertaking. Furthermore, we proposed the first automatic annotation technique for the Moroccan dialect, which may be used as a pre-trained model for future researchers. A collection of the most effective neural network models was conditioned on six individual datasets constrained to positive and negative labels. These baseline models were subsequently deployed to annotate a dataset employing a weighted voting scheme, which leverages predictions from the neural network models as input data. This model harnesses the mode of these predictions to feed into a meta-classifier. The purpose of this is twofold: firstly, to generate specific coefficients, and secondly, to multiply these coefficients with the original neural network model predictions to ascertain final outputs. In the final phase, we evaluated the efficiency of the proposed method in annotating each of the six datasets. The outcomes demonstrated concurrence rates of 87.34%, 91%, 85.72%, 83.16%, 84.93%, and 79.65% for MSAC, FB, MSDA, MSTD, MAC, and ElecMorocco2016 respectively, underlining the potential and proficiency of the proposed automated annotation system for sentiment analysis in the Moroccan dialect. In forthcoming research, our team is committed to exploring cutting-edge methodologies to address the challenges posed by imbalanced datasets. This involves harnessing the potential of continuous learning to facilitate the adaptation of models in real-time, particularly when there is diminished confidence in the generated annotations. Moreover, the assimilation of reinforcement learning techniques offers a promising avenue for dealing with prevalent issues related to annotation. Additionally, by amplifying the datasets specific to the Moroccan dialect, we anticipate achieving a more comprehensive coverage of lexical items, further enhancing the efficacy of the sentiment analysis models.

VII. References

[1] E. B. Adam, 2020 “Sentiment Analysis for Moroccan Dialect”.

[2] M. Errami, M. A. Ouassil, R. Rachidi, B. Cherradi, S. Hamida, and A. Raihani, 2023 “Sentiment Analysis on Moroccan Dialect based on ML and Social Media Content Detection,” Int. J. Adv. Comput. Sci. Appl., vol. 14, no. 3.

[3] Jbel, M., Hafidi, I., & Metrane, A. (2023). An Experimental Study on Sentiment Classification of Moroccan dialect texts in the web. arXiv preprint arXiv:2303.15987.

[4] F. Mallek, B. Belainine, and F. Sadat, 2017 “Arabic Social Media Analysis and Translation,” Procedia Comput. Sci., vol. 117, pp. 298–303, doi: 10.1016/j.procs.2017.10.121.

[5] A. A. Khrisat and Z. A. Alharthy, 2015 “Arabic Dialects and Classical Arabic Language,” Adv. Soc. Sci. Res. J., vol. 2, no. 4, Apr. 2015, doi: 10.14738/assrj.24.1048.

[6] A. Alsayat and N. Elmitwally, 2020 “A comprehensive study for Arabic Sentiment Analysis (Challenges and Applications),” Egypt. Inform. J., vol. 21, no. 1, pp. 7–12, Mar, doi: 10.1016/j.eij.2019.06.001.

[7] A. Farghaly and K. Shaalan, 2009 “Arabic Natural Language Processing: Challenges and Solutions,” ACM Trans. Asian Lang. Inf. Process., vol. 8, no. 4, pp. 1–22, Dec. doi: 10.1145/1644879.1644881.

[8] J. Cohen, 1960 “A Coefficient of Agreement for Nominal Scales,” Educ. Psychol. Meas., vol. 20, no. 1, pp. 37–46, Apr, doi: 10.1177/001316446002000104.

[9] F. Weeber, F. Hamborg, K. Donnay, and B. Gipp, 2022 “Assisted Text Annotation Using Active Learning to Achieve High Quality with Little Effort.” arXiv, Dec. 15, 2021. Accessed: Oct. 16, [Online]. Available: http://arxiv.org/abs/2112.11914

[10] G. Badaro, R. Baly, H. Hajj, N. Habash, and W. El-Hajj, 2014 “A Large Scale Arabic Sentiment Lexicon for Arabic Opinion Mining,” in Proceedings of the EMNLP 2014 Workshop on Arabic Natural

Language Processing (ANLP), Doha, Qatar: Association for Computational Linguistics, pp. 165–173. doi: 10.3115/v1/W14-3623.

[11] M. Taboada, 2016 “Sentiment Analysis: An Overview from Linguistics,” Annu. Rev. Linguist., vol. 2, no. 1, pp. 325–347, Jan. doi: 10.1146/annurev-linguistics-011415-040518.

[12] H. Abdellaoui and M. Zrigui, 2018 “Using Tweets and Emojis to Build TEAD: an Arabic Dataset for Sentiment Analysis,” Comput. Sist., vol. 22, no. 3, Sep. 2018, doi: 10.13053/cys-22-3-3031.

[13] A. E. Abdouli, L. Hassouni, and H. Anoun, 2017 “Sentiment Analysis of Moroccan Tweets using Naive Bayes Algorithm,” vol. 15, no. 12, p. 10.

[14] A. Baccouche, B. Garcia-Zapirain, and A. Elmaghraby, 2018 “Annotation Technique for Health-Related Tweets Sentiment Analysis,” in 2018 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT), Louisville, KY, USA: IEEE, Dec, pp. 382–387. doi: 10.1109/ISSPIT.2018.8642685.

[15] I. Guellil, F. Azouaou, and F. Chiclana, 2020 “ArAutoSenti: automatic annotation and new tendencies for sentiment classification of Arabic messages,” Soc. Netw. Anal. Min., vol. 10, no. 1, p. 75, Dec, doi: 10.1007/s13278-020-00688-x.

[16] M. El-Masri, N. Altrabsheh, H. Mansour, and A. Ramsay, 2017 “A web-based tool for Arabic sentiment analysis,” Procedia Comput. Sci., vol. 117, pp. 38–45, doi: 10.1016/j.procs.2017.10.092.

[17] Kathrein Abu Kwaik, Stergios Chatzikyriakidis, and Simon Dobnik. 2019. "Can Modern Standard Arabic Approaches be used for Arabic Dialects? Sentiment Analysis as a Case Study." In Proceedings of the third Workshop on Arabic Corpus Linguistics, pages 40–50, Cardiff, United Kingdom. Association for Computational Linguistics.

[18] Kathrein Abu Kwaik, Motaz Saad, Stergios Chatzikyriakidis, and Simon Dobnik. 2018. Shami: A Corpus of Levantine Arabic Dialects. In Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), Miyazaki, Japan. European Language Resources Association (ELRA).

[19] Mohamed Aly and Amir Atiya. 2013. LABR: A Large Scale Arabic Book Reviews Dataset. In Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pages 494–498, Sofia, Bulgaria. Association for Computational Linguistics.

[20] M. Youssef and S. R. El-Beltagy, 2018 “MoArLex: An Arabic Sentiment Lexicon Built Through Automatic Lexicon Expansion,” Procedia Comput. Sci., vol. 142, pp. 94–103, doi: 10.1016/j.procs.2018.10.464.

[21] Tamer, M., Khamis, M.A., Yahia, A. et al. 2023 "Arab reactions towards Russo-Ukrainian war." EPJ Data Sci. 12, 36. https://doi.org/10.1140/epjds/s13688-023-00415-4.

[22] R. Alahmary and H. Al-Dossari, 2021 “A semiautomatic annotation approach for sentiment analysis,” J. Inf. Sci., p. 016555152110065, Apr, doi: 10.1177/01655515211006594.

[23] A. Al-Laith, M. Shahbaz, H. F. Alaskar, and A. Rehmat, 2021 “AraSenCorpus: A Semi-Supervised Approach for Sentiment Annotation of a Large Arabic Text Corpus,” Appl. Sci., vol. 11, no. 5, p. 2434, Mar, doi: 10.3390/app11052434.

[24] Sara Rosenthal, Noura Farra, and Preslav Nakov. 2017. "SemEval-2017 Task 4: Sentiment Analysis in Twitter." In Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017), pages 502–518, Vancouver, Canada. Association for Computational Linguistics.

[25] M. Nabil, M. Aly, and A. Atiya, 2015 “ASTD: Arabic Sentiment Tweets Dataset,” in Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal: Association for Computational Linguistics, pp. 2515–2519. doi: 10.18653/v1/D15-1299.

[26] A. Omar, T. M. Mahmoud, T. Abd-El-Hafeez, and A. Mahfouz, 2021 “Multi-label Arabic text classification in Online Social Networks,” Inf. Syst., vol. 100, p. 101785, Sep, doi: 10.1016/j.is.2021.101785.

[27] N. A. Abdulla, N. A. Ahmed, M. A. Shehab, and M. Al-Ayyoub, 2013 “Arabic sentiment analysis: Lexicon-based and corpus-based,” in IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies (AEECT), Amman, Jordan: IEEE, Dec. 2013, pp. 1–6. doi: 10.1109/AEECT.2013.6716448.

[28] Elmadany, A. A., Mubarak, H., & Magdy, W. 2018. "An Arabic Speech-Act and Sentiment Corpus of Tweets." In Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018) European Language Resources Association (ELRA). Advance online publication. http://lrec-conf.org/workshops/lrec2018/W30/summaries/22_W30.html

[29] A. A. Altowayan and L. Tao, 2016 “Word embeddings for Arabic sentiment analysis,” in 2016 IEEE International Conference on Big Data (Big Data), Washington DC,USA: IEEE, Dec, pp. 3820–3825. doi: 10.1109/BigData.2016.7841054.

[30] K. A. Kwaik, M. Saad, S. Chatzikyriakidis, S. Dobnik, and R. Johansson, 2020 “An Arabic Tweets Sentiment Analysis Dataset (ATSAD) using Distant Supervision and Self Training,” Proc. 4th Workshop Open-Source Arab. Corpora Process. Tools, p. 9.

[31] I. Guellil, A. Adeel, F. Azouaou, and A. Hussain, “SentiALG: Automated Corpus Annotation for Algerian Sentiment Analysis,” in *Advances in Brain Inspired Cognitive Systems*, 9th International Conference on Brain Inspired Cognitive Systems (BICS 2018)., vol. 10989, J. Ren, A. Hussain, J. Zheng, C.-L. Liu, B. Luo, H. Zhao, and X. Zhao, Eds. Cham: Springer International Publishing, 2018, pp. 557–567. doi: 10.1007/978-3-030-00563-4_54.

[32] A. Elnagar and O. Einea, 2016 "BRAD 1.0: Book reviews in Arabic dataset," IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA), Agadir, Morocco, 2016, pp. 1-8, doi: 10.1109/AICCSA.7945800.

[33] A. Elnagar, Y. S. Khalifa, and A. Einea, 2018 “Hotel Arabic-Reviews Dataset Construction for Sentiment Analysis Applications,” in Intelligent Natural Language Processing: Trends and Applications, vol. 740, K. Shaalan, A. E. Hassanien, and F. Tolba, Eds., in Studies in Computational Intelligence, vol. 740. , Cham: Springer International Publishing, pp. 35–52. doi: 10.1007/978-3-319-67056-0_3.

[34] G. Imane, D. Kareem, and A. Faical, 2019 “A set of parameters for automatically annotating a Sentiment Arabic Corpus,” Int. J. Web Inf. Syst., vol. 15, no. 5, pp. 594–615, Dec, doi: 10.1108/IJWIS-03-2019-0008.

[35] A. Oussous, A. A. Lahcen, and S. Belfkih, 2018 “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., in Communications in Computer and Information

Science, vol. 872. , Cham: Springer International Publishing, pp. 91–104. doi: 10.1007/978-3-319-96292-4_8.

[36] A. Elouardighi, M. Maghfour, and H. Hammia, 2017 "Collecting and Processing Arabic Facebook Comments for Sentiment Analysis," in Model and Data Engineering, vol. 10563, Y. Ouhammou, M. Ivanovic, A. Abelló, and L. Bellatreche, Eds., in Lecture Notes in Computer Science, vol. 10563. , Cham: Springer International Publishing, pp. 262–274. doi: 10.1007/978-3-319-66854-3_20.

[37] S. Mihi, B. Ait, I. El, S. Arezki, and N. Laachfoubi, 2020 "MSTD: Moroccan Sentiment Twitter Dataset," Int. J. Adv. Comput. Sci. Appl., vol. 11, no. 10, doi: 10.14569/IJACSA.2020.0111045.

[38] M. Maghfour and A. Elouardighi, 2018 "Standard and Dialectal Arabic Text Classification for Sentiment Analysis," in Model and Data Engineering, International Conference on Model and Data Engineering., vol. 11163, E. H. Abdelwahed, L. Bellatreche, M. Golfarelli, D. Méry, and C. Ordonez, Eds., in Lecture Notes in Computer Science, vol. 11163., Cham: Springer International Publishing, pp. 282–291. doi: 10.1007/978-3-030-00856-7_18.

[39] M. Garouani and J. Kharroubi, 2022 "MAC: An Open and Free Moroccan Arabic Corpus for Sentiment Analysis," in Innovations in Smart Cities Applications Volume 5, vol. 393, M. Ben Ahmed, A. A. Boudhir, İ. R. Karaș, V. Jain, and S. Mellouli, Eds., in Lecture Notes in Networks and Systems, vol. 393. , Cham: Springer International Publishing, pp. 849–858. doi: 10.1007/978-3-030-94191-8_68.

[40] E. Boujou, H. Chataoui, A. E. Mekki, S. Benjelloun, I. Chairi, and I. Berrada, 2022 "An open access NLP dataset for Arabic dialects : Data collection, labeling, and model construction." arXiv, Feb. 06, 2021. Accessed: Dec. 21. [Online]. Available: http://arxiv.org/abs/2102.11000

[41] A. Abdelali, K. Darwish, N. Durrani, and H. Mubarak, 2016 "Farasa: A Fast and Furious Segmenter for Arabic," in Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations, San Diego, California: Association for Computational Linguistics, pp. 11–16. doi: 10.18653/v1/N16-3003.

[42] Y. Matrane, F. Benabbou, and N. Sael, 2023 "A systematic literature review of Arabic dialect sentiment analysis," J. King Saud Univ. - Comput. Inf. Sci., vol. 35, no. 6, p. 101570, Jun, doi: 10.1016/j.jksuci.2023.101570.

[43] Wissam Antoun, Fady Baly, and Hazem Hajj. 2020. AraBERT: Transformer-based Model for Arabic Language Understanding. In Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection, pages 9–15, Marseille, France. European Language Resource Association.

[44] K. Gaanoun, A. M. Naira, A. Allak, and I. Benelallam, 2023 "Darijabert: a Step Forward in Nlp for the Written Moroccan Dialect," In Review, preprint, Feb. doi: 10.21203/rs.3.rs-2560653/v1.

[45] Y. Matrane, F. Benabbou, and N. Sael, 2021 "Sentiment analysis through word embedding using AraBERT: Moroccan dialect use case," in International Conference on Digital Age & Technological Advances for Sustainable Development (ICDATA), Marrakech, Morocco: IEEE, Jun. 2021, pp. 80–87. doi: 10.1109/ICDATA52997.2021.00024.

YASSIR MATRANE received the B.Sc. degree in mathematical sciences and computer science from the Faculty of Sciences Ben M'Sick, Hassan II University of Casablanca, Morocco, in 2018, and the M.Sc. degree in data science and big data from Hassan II University of Casablanca, in 2020, where he is currently pursuing the Ph.D. degree in computer science. His research interests includes sentiment analysis of Arabic dialect, precisely Moroccan dialect, machine learning, deep learning, knowledge graph embedding, and reinforcement learning.

FAOUZIA BENABBOU has been a Teacher Researcher, since 1994, an Authorized Professor, since 2008, and a Professor of higher education with the Department of Mathematics and Computer Science, Ben M'Sick Faculty of Sciences, Hassan II University of Casablanca, Casablanca, since 2015. She is a member of the Information Technology and Modeling Laboratory and the Leader of the Cloud Computing, Network and Systems Engineering (ICCNSE) Team. Her research interests include cloud computing, data mining, machine learning, and natural language processing.

ZOUHEIR BANOU received the B.Sc. degree in mathematical sciences and computer science from the Faculty of Sciences Ben M'Sick, Hassan II University of Casablanca, Morocco, in 2018, and the M.Sc. degree in data science and big data from Hassan II University of Casablanca, in 2020, where he is currently pursuing the Ph.D. degree in computer science. His research interests includes emotion analysis of Modern Standard Arabic, machine learning, deep learning, knowledge graph embedding, and reinforcement learning.