10.48047/jocaaa.2025.34.05.37

Balancing Social Influence: Integrating Positive and Negative Opinions in Online Networks

B. Sridhar[1*], K. Chandu[2], L. Bhanu Prakash[2], M. Santhosh Reddy[2], M. Harsha Vardhan Reddy[2], M. Raju[2]

[1]Assistant Professor, [2]UG Student, [1,2]Department of Computer Science and Engineering-Data Science, [1,2]Malla Reddy Engineering College and Management Sciences, Kistapur, Medchal, 501401, Telangana, India

*Corresponding author: sridhar@mrem.ac.in

Received: 02.03.2025 Revised: 15.04.2025 Accepted: 25.05.2025

Abstract

This research presents a Tkinter-based application for classifying tweets as fake or authentic using natural language processing (NLP) and polarity score–based supervised learning. The system allows users to upload tweet datasets in CSV format and performs text preprocessing by removing punctuation, stopwords, and non-alphabetic characters. Sentiment polarity scores are computed using TextBlob, and tweets are classified as *AUTHENTICATED* when the polarity exceeds 0.51, or *FAKE* otherwise. The classification results are displayed through an intuitive graphical user interface and visualized using pie charts generated with matplotlib. The application employs pandas for efficient data handling and NLTK for text preprocessing, providing a smooth end-to-end workflow from data ingestion to result interpretation. Although effective as a prototype for identifying potentially misleading content, the use of a fixed polarity threshold and limited error handling restricts its robustness. Future enhancements may include the integration of advanced machine learning models, multilingual support, and real-time tweet analysis, positioning the system as a scalable tool for mitigating misinformation on social media platforms.

Keywords: Fake News Detection, Tweet Classification, Natural Language Processing, Sentiment Analysis, Polarity Score, Tkinter GUI, Social Media Analytics

1. Introduction

Online networks have become the central platform for the exchange of information, emotions, and public sentiments. Social media platforms like Twitter, Facebook, Reddit, and YouTube receive millions of posts, comments, and interactions daily. For instance, Facebook reported over 2.91 billion monthly active users as of late 2024, while Twitter sees more than 500 million tweets per day. Amid this enormous influx of content, the ability to understand, monitor, and respond to public opinion has become critical, especially in the fields of marketing, politics, and public health. These platforms serve as echo chambers where both positive and negative opinions can go viral within hours, significantly influencing public perception and decision-making.

Studies show that negative opinions tend to have a stronger psychological impact than positive ones. According to research by Baumeister et al. (2001), negative information is processed more thoroughly than positive information, resulting in a greater influence on public opinion. This phenomenon plays out in online reviews, social media backlash, or cancel culture, where a single negative comment can outweigh multiple positive ones. Thus, analyzing how opinions propagate and interact in online networks is essential to maintaining balance in digital discourse. Without proper analysis and moderation, online spaces can become biased, polarizing, or toxic, leading to misinformation or unwarranted damage to reputations.

10.48047/jocaaa.2025.34.05.37

2. Literature Survey

While considerable effort has been devoted to sentiment analysis for structured languages, relatively few studies have efficiently explored sentiment classification for unstructured and dialectal languages, such as the Moroccan dialect [2]. In previous research [3], we conducted a comprehensive review and comparative analysis of recent studies in sentiment analysis. Our findings highlighted that many studies translate comments into structured languages, such as English, to facilitate sentiment analysis, driven by the greater availability of standardized resources for structured languages compared to the limited resources for dialect-heavy languages.

Moreover, we identified key challenges specific to the Moroccan dialect (MD) that affect automated sentiment analysis. For instance, MD comments may appear in Arabic script, Latin characters, or a combination of both, which creates significant obstacles for automated text processing [4]. Additionally, the frequent use of character repetition, such as 'خاااااااايب' ('Baaaaad') and 'مزياااااان' ('Gooood'), poses substantial normalization challenges. These repetitions are not merely typographic anomalies but often convey strong emotional intensity, highlighting the lack of consistent syntactic or grammatical conventions in dialectal languages [5].

The historical development of the Moroccan dialect adds another layer of complexity. MD emerged from interactions between the first Arab horsemen from the East, who introduced classical Arabic, and the native Berber populations, further influenced by colonial languages such as French and Spanish [6]. This blend of influences has resulted in a distinctive linguistic landscape for MD, setting it apart from other colloquial languages [7]. Unlike other dialects, which often evolve from a single root language or a combination of two, the Moroccan dialect is recognized as one of the most complex Arabic dialects to comprehend. This complexity stems from several factors, including: Complex normalization and standardization processes compared to Modern Standard Arabic (MSA) [8]. Vocabulary influenced by Arabic sources, with modifications to original words due to phonological changes [9].

Limited writing resources, which contribute to its status as an under-resourced language. Additionally, colonial influences, particularly from the French language, have significantly shaped the Moroccan dialect (MD) [10].

MD has incorporated elements from a wide range of languages, including Italian, Spanish, and French, alongside MSA and other forms of dialectal Arabic, resulting in a highly intricate linguistic framework. This assimilation of languages has made MD one of the most complex dialects to analyze linguistically. The current literature reveals a significant gap in addressing text balancing for the Moroccan dialect and Arabic dialects in general, particularly in the context of sentiment analysis, a rapidly growing field for these dialects. Notably, there is a scarcity of studies specifically tackling the challenges of balancing and sentiment analysis in dialects [2,10].

YAFOOZ et al. [11] utilized machine learning classifiers to analyze user comments on YouTube related to herbal remedies in films. They developed a new Arabic dataset focused on herbal diabetes treatments (ADHTD) and applied preprocessing techniques such as N-grams with Arabic words, stemming, stop-word removal, and managing character extensions (TATAWEEL or MAD). Using the Synthetic Minority Oversampling Technique (SMOTE), they balanced the dataset, achieving a model performance with 95%-word identification accuracy, primarily for the Emirati dialect.

Similarly, Al Shamsi et al. [12] conducted a detailed sentiment analysis by manually annotating comments for polarity (positive, negative, and neutral) and categorizing them by dialect type, including Emirati, other Arabic dialects, and Modern Standard Arabic (MSA). Their 70,000-comment dataset, predominantly in the Emirati dialect, underwent TF-IDF feature extraction and various preprocessing

10.48047/jocaaa.2025.34.05.37

techniques. The study evaluated eight machine learning classifiers, including Logistic Regression (LR), Random Forest (RF), Decision Tree (DT), and Support Vector Machine (SVM), achieving an accuracy of 80%. Additionally, they explored deep learning models such as Gated Recurrent Units (GRU), Long Short-Term Memory (LSTM), and Bidirectional LSTM (BiLSTM), which consistently delivered accuracy rates above 70%.

Habbat et al. [13] developed an ensemble model for Arabic sentiment analysis, leveraging basic deep learning models as classifiers and AraBERT for meaningful embeddings. By combining predictions from stacked deep learning models with SVM as a meta-classifier and employing data augmentation with AraGPT, their approach demonstrated high accuracy across datasets such as HARD, BRAD, and Twitter, achieving 88%, 90.88%, and 87.23%, respectively.

Similarly, Al-Hashedi et al. [14] utilized Naïve Bayes (NB) as a baseline classifier for Arabic tweet classification, employing Word2Vec for word embedding. They evaluated multiple machine learning classifiers with and without SMOTE, finding that combining word embeddings with ensemble methods and SMOTE significantly improved average F1 scores. Albahli [15] proposed a method for identifying reliable COVID-19-related news in Arabic for Gulf countries, using sentiment-based Twitter data. By applying SMOTE alongside Multinomial Naïve Bayes, they achieved the highest accuracy of 91% among evaluated techniques, effectively addressing dataset imbalances.

Mountassir et al. [16] examined four under-sampling methods on imbalanced Arabic and English datasets, including random removal, clustering removal, and removing similar or distant instances. Their results highlighted random under-sampling (RU) as the most effective method based on the g-performance metric when applied to classifiers such as NB, SVM, and KNN. Al-Azani et al. [17] explored approaches to handling imbalanced datasets in dialectal Arabic tweets. Their 2017 study employed SMOTE with various classifiers, resulting in a 15% performance improvement compared to baseline experiments. In 2018, they combined SMOTE with bootstrap aggregating (bagging) to balance concatenated datasets, demonstrating that balanced bagging classifiers produced the best outcomes.

Similarly, El-Alfy et al. [18] evaluated nine machine learning classifiers on imbalanced Arabic tweet datasets using neural word embeddings and over-sampling techniques. Metrics such as AUC, G-Mean, and F1 score identified the Stochastic Gradient Descent (SGD) classifier with over-sampling as the top performer. Khalifa et al. [19] analyzed Twitter datasets in both balanced and imbalanced forms, employing feature representations like TF-IDF and word embeddings. Logistic Regression (LR) with TF-IDF achieved the highest F1 score of 87.71% on imbalanced training data. Matrane et al. [20] used the MSTD dataset to evaluate models, with Bidirectional LSTM (BiLSTM) achieving the highest accuracy of 83.24%. Among traditional algorithms, SVM performed well, attaining 80.82% accuracy.

3. Proposed System

The advantage of this architecture lies in its modular and application-specific workflow as shown in Figure 4.1. By structuring the pipeline from data acquisition to sentiment classification and performance analysis, the method ensures clarity, scalability, and efficiency. The input is domain-specific, i.e., Twitter text data, which allows the pipeline to be tightly integrated with natural language processing requirements. This system offers easy adaptability for real-time applications like fake news detection, public opinion mining, and sentiment trend analysis, particularly where trust in textual content is critical.

Step 1: Tweets Dataset Upload: The process begins with uploading a tweets dataset, typically in CSV format. This dataset includes tweet text data, possibly with associated metadata. The input is chosen specifically to match the context of sentiment or authenticity analysis, ensuring relevance to the classification task.

10.48047/jocaaa.2025.34.05.37

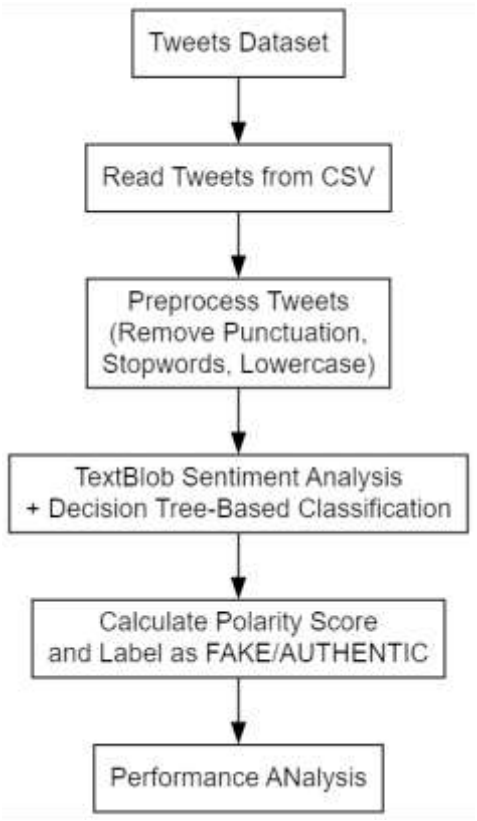

Fig. 1: Proposed System Architecture.

Step 2: Reading Tweets from CSV: Once the dataset is uploaded, the system reads the relevant column (usually the 'Text' column) from the CSV file. The system iteratively loads each tweet into memory and prepares it for preprocessing. This forms the basis for constructing the raw input corpus.

Step 3: Preprocess Tweets: In this stage, tweets undergo preprocessing to remove noise and irrelevant components. Steps include converting text to lowercase, removing punctuation marks, filtering out stopwords, and excluding short or non-alphabetic tokens. This cleaned dataset becomes the refined corpus used for accurate feature extraction and sentiment evaluation.

Step 4: TextBlob Sentiment Analysis + Decision Tree-Based Classification: The cleaned tweets are analyzed using TextBlob, a sentiment analysis library that returns a polarity score for each tweet. These scores range from -1 to 1, indicating negative to positive sentiment. After polarity computation, the sentiment information is integrated with a rule-based or supervised classifier, such as a Decision Tree, which helps enhance interpretability and decision logic in classifying tweets as fake or authentic.

Step 5: Calculate Polarity Score and Label Tweets: Based on predefined thresholds (e.g., polarity > 0.51 indicating authentic tweets), each tweet is labeled as FAKE or AUTHENTIC. This classification is driven by both the polarity score from the NLP engine and the decision rules structured in the classifier, effectively separating credible content from misleading or low-trust content.

Step 6: Performance Analysis: Finally, the results are compiled and displayed using analytical tools such as pie charts to show the distribution of fake vs authentic tweets. This visualization helps users understand the effectiveness of the classification and the sentiment trends present in the input dataset. It also provides a quantitative basis for further decisions or actions based on content authenticity.

4. Results Analysis

10.48047/jocaaa.2025.34.05.37

Figure 2 illustrates the text area output after the user clicks the "Data Preprocessing" button, which calls the clean function. This function processes each tweet in tweets_list using the tweetCleaning function, which removes punctuation, non-alphabetic characters, stopwords (using NLTK's English stopwords), and words shorter than two characters, then joins the tokens into a cleaned string stored in clean_list. The text area displays these cleaned tweets, one per line, followed by the total count (e.g., "Total tweets found in dataset is: 4" for four tweets). For the sample tweets from Figure 3, the cleaned output might include strings like "rightsatbirth manipulative dishonest coercive coerced sex consensual sex sexual assault" transformed to "rightsatbirth manipulative dishonest coercive coerced sex consensual sexual assault" (after removing punctuation and stopwords like "and") and "happy angelalansbury bedknobs amp broomsticks still one favorite moviesnnheres" transformed to "happy angelalansbury bedknobs broomsticks still favorite movies" (removing "amp," "one," and other stopwords). This figure highlights the preprocessing step, showing the cleaned text ready for sentiment analysis.

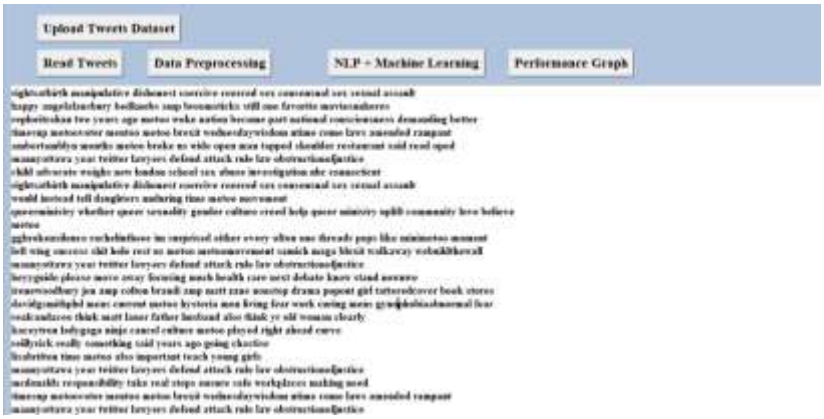

Fig. 2: Dataset after Preprocessing.

Figure 3 displays the text area output after the user clicks the "NLP + Machine Learning" button, triggering the machineLearning function. This function analyzes each tweet in clean_list using TextBlob to compute a polarity score and classifies tweets as "FAKE" (polarity ≤ 0.5) or "AUTHENTICATED" (polarity > 0.51). The text area shows each tweet, its predicted sentiment, polarity score, and a delimiter ("===="). The provided sample output includes four tweets: (1) "rightsatbirth manipulative dishonest coercive coerced sex consensual sex sexual assault" with a polarity score of 0.1 (FAKE), (2) "happy angelalansbury bedknobs amp broomsticks still one favorite moviesnnheres" with a polarity score of 0.65 (AUTHENTICATED), (3) "reploritrahan two years ago metoo woke nation become part national consciousness demanding better" with a polarity score of 0.5 (FAKE), and (4) "timesup metoovoter mentoo metoo brexit wednesdaywisdom ntime come laws amended rampant" with a polarity score of 0.0 (FAKE). This figure represents the core sentiment analysis output, showing the classification results and their underlying polarity scores.

10.48047/jocaaa.2025.34.05.37

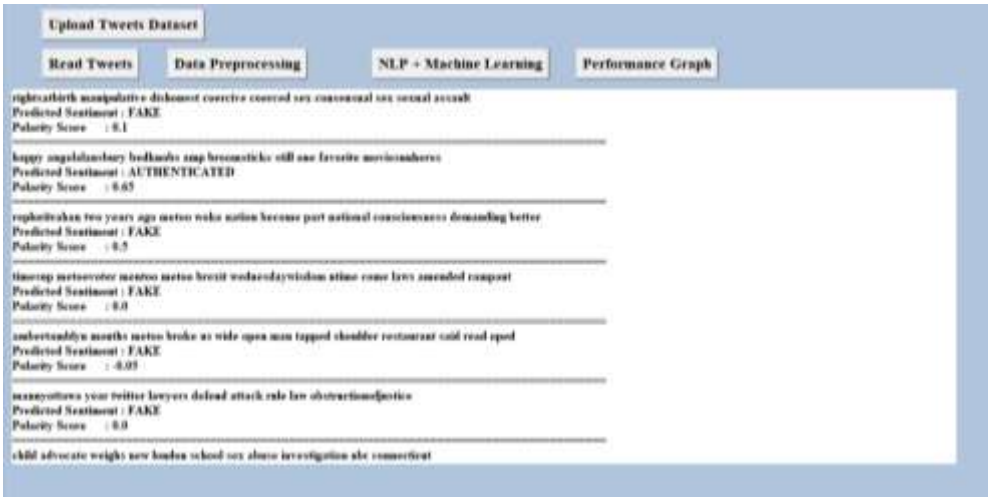

Fig. 3: Polarity Score based Prediction Output.

Figure 4 depicts a pie chart generated by the graph function, invoked when the user clicks the "Performance Graph" button. The chart visualizes the distribution of tweets classified as AUTHENTICATED and FAKE, based on the global counters pos (AUTHENTICATED) and neg (FAKE). Using matplotlib.pyplot.pie, the chart displays two segments labeled "Authenticated" and "Fake," with percentages formatted to one decimal place (e.g., 25.0% for AUTHENTICATED and 75.0% for FAKE if one tweet is AUTHENTICATED and three are FAKE, as per the sample in Figure 3). The text area also shows the counts (e.g., "Authenticated: 1," "Fake: 3"), total tweets (e.g., "Length of tweets: 4"), and percentages (e.g., "Authenticated: 1 / 4 = 0.25%," "Fake: 3 / 4 = 0.75%"). The pie chart has equal axes and a title "Sentiment Graph," providing a visual summary of the sentiment analysis results.

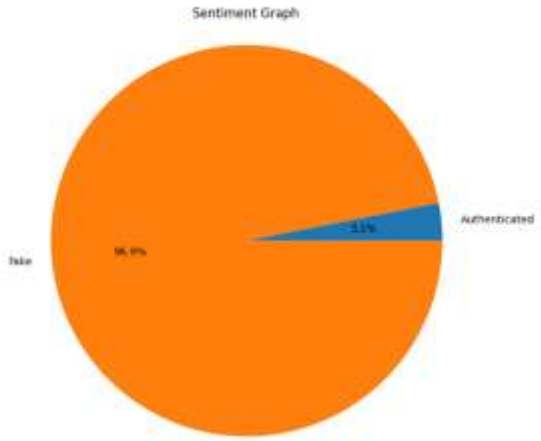

Fig. 4: Sentiment Pie Chart.

5. Conclusion

The provided Tkinter-based application effectively demonstrates a polarity score-based supervised learning approach for classifying tweets as fake or authentic using natural language processing (NLP) techniques. By leveraging libraries such as pandas, NLTK, TextBlob, and matplotlib, the application provides a user-friendly graphical interface to upload a tweet dataset, preprocess the text, perform sentiment analysis, and visualize the results. The workflow begins with loading a CSV file containing tweets, followed by cleaning the text to remove noise such as punctuation, stopwords, and non-

10.48047/jocaaa.2025.34.05.37

alphabetic characters, ensuring the data is suitable for analysis. The core functionality lies in the sentiment analysis, where TextBlob's polarity scoring is used to classify tweets based on a predefined threshold, offering a straightforward method to distinguish between authentic and potentially misleading content. The application's ability to display intermediate outputs in a text area and summarize results through a pie chart enhances its usability, making it accessible for users to interpret the classification outcomes. While the application successfully meets its objective of tweet classification, its reliance on a fixed polarity threshold and limited error handling highlights areas for improvement. Overall, it serves as a functional prototype for analyzing social media content, providing a foundation for further development in automated text classification tasks.

References

[1] Malviya, S.; Tiwari, A.K.; Srivastava, R.; Tiwari, V. Machine learning techniques for sentiment analysis: A review. SAMRIDDHI A J. Phys. Sci. Eng. Technol. 2020, 12, 72–78.

[2] Nassr, Z.; Sael, N.; Benabbou, F. A comparative study of sentiment analysis approaches. In Proceedings of the 4th International Conference on Smart City Applications, Casablanca, Morocco, 2–4 October 2019; pp. 1–8.

[3] Mavridis, N. A review of verbal and non-verbal human–robot interactive communication. Robot. Auton. Syst. 2015, 63, 22–35.

[4] Nazir, F.; Ghazanfar, M.A.; Maqsood, M.; Aadil, F.; Rho, S.; Mehmood, I. Social media signal detection using tweets volume, hashtag, and sentiment analysis. Multimed. Tools Appl. 2019, 78, 3553–3586.

[5] Wang, H.; Castanon, J.A. Sentiment expression via emoticons on social media. In Proceedings of the 2015 IEEE International Conference on Big Data (Big Data), Santa Clara, CA, USA, 29 October–1 November 2015; IEEE: New York, NY, USA, 2015; pp. 2404–2408.

[6] Liu, B. Sentiment analysis and opinion mining. Synth. Lect. Hum. Lang. Technol. 2012, 5, 1–167.

[7] Nassr, Z.; Sael, N.; Benabbou, F. Machine Learning for Sentiment Analysis: A Survey. In Proceedings of the Third International Conference on Smart City Applications, Tétouan, Morocco, 10–11 October 2018; Springer: Cham, Switzerland, 2019; pp. 63–72.

[8] Nassr, Z.; Nawal, S.A.E.L.; Benabbou, F. Generate a list of stop words in moroccan dialect from social network data using word embedding. In Proceedings of the 2021 International Conference on Digital Age & Technological Advances for Sustainable Development (ICDATA), Marrakech, Morocco, 29–30 June 2021; IEEE: New York, NY, USA, 2021; pp. 66–73.

[9] Nassr, Z.; Sael, N.; Benabbou, F. Preprocessing arabic dialect for sentiment mining: State of art. Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci. 2020, 44, 323–330.

[10] Matrane, Y.; Benabbou, F.; Sael, N. A Systematic Literature Review of Arabic Dialect Sentiment Analysis. J. King Saud Univ.-Comput. Inf. Sci. 2023, 35, 101570.

[11] Yafooz, W.M.; Alsaeedi, A. Sentimental analysis on health-related information with improving model performance using machine learning. J. Comput. Sci. 2021, 17, 112–122.

[12] Al Shamsi, A.; Abdallah, S. Sentiment Analysis of Emirati Dialect. Big Data Cogn. Comput. 2022, 6, 57.

[13] Habbat, N.; Nouri, H.; Anoun, H.; Hassouni, L. Using AraGPT and ensemble deep learning model for sentiment analysis on Arabic imbalanced dataset. ITM Web Conf. 2023, 52, 02008.

[14] Al-Hashedi, A.; Al-Fuhaidi, B.; Mohsen, A.M.; Ali, Y.; Gamal Al-Kaf, H.A.; Al-Sorori, W.; Maqtary, N. Ensemble classifiers for Arabic sentiment analysis of social network (Twitter data) towards COVID-19-related conspiracy theories. Appl. Comput. Intell. Soft Comput. 2022, 2022, 1–10.

10.48047/jocaaa.2025.34.05.37

[15] Mountassir, A.; Benbrahim, H.; Berrada, I. An empirical study to address the problem of unbalanced data sets in sentiment classification. In Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, Seoul, Republic of Korea, 14–17 October 2012; IEEE: New York, NY, USA, 2012; pp. 3298–3303.

[16] Al-Azani, S.; El-Alfy, E.S.M. Using Word Embedding and Ensemble Learning for Highly Imbalanced Data Sentiment Analysis in Short Arabic Text. Procedia Comput. Sci. 2017, 109, 359–366.

[17] El-Alfy, E.S.M.; Al-Azani, S. Empirical study on imbalanced learning of Arabic sentiment polarity with neural word embedding. J. Intell. Fuzzy Syst. 2020, 38, 6211–6222.

[18] Khalifa, Y.; Elnagar, A. Colloquial Arabic Tweets: Collection, Automatic Annotation, and Classification. In Proceedings of the 2020 International Conference on Asian Language Processing, IALP 2020, Kuala Lumpur, Malaysia, 4–6 December 2020; IEEE: New York, NY, USA, 2020; pp. 163–168.

[19] Addi, H.A.; Ezzahir, R. Sampling techniques for Arabic Sentiment Classification: A Comparative Study. In ACM International Conference Proceeding Series; ACM: New York, NY, USA, 2020; pp. 1–6.

[20] Matrane, Y.; Benabbou, F.; Sael, N. Sentiment analysis through word embedding using AraBERT: Moroccan dialect use case. In Proceedings of the 2021 International Conference on Digital Age & Technological Advances for Sustainable Development (ICDATA), Marrakech, Morocco, 29–30 June 2021; IEEE: New York, NY, USA, 2021; pp. 80–87.