Sentiment Analysis of Arabic Tweets Using Large Language Models

Pankaj Dadure UPES Dehradun

Ananya Dixit UPES Dehradun

Kunal Tewatia UPES Dehradun

Nandini Paliwal UPES Dehradun

Anshika Malla UPES Dehradun

Abstract

In the digital era, sentiment analysis has become an indispensable tool for understanding public sentiments, optimizing market strategies, and enhancing customer engagement across diverse sectors. While significant advancements have been made in sentiment analysis for high-resource languages such as English, French, etc. This study focuses on Arabic, a low-resource language, to address its unique challenges like morphological complexity, diverse dialects, and limited linguistic resources. Existing works in Arabic sentiment analysis have utilized deep learning architectures like LSTM, BiLSTM, and CNN-LSTM, alongside embedding techniques such as Word2Vec and contextualized models like ARABERT. Building on this foundation, our research investigates sentiment classification of Arabic tweets, categorizing them as positive or negative, using embeddings derived from three large language models (LLMs): Universal Sentence Encoder (USE), XLM-RoBERTa base (XLM-R base), and MiniLM-L12-v2. Experimental results demonstrate that incorporating emojis in the dataset and using the MiniLM embeddings yield an accuracy of 85.98%. In contrast, excluding emojis and using embeddings from the XLM-R base resulted in a lower accuracy of 78.98%. These findings highlight the impact of both dataset composition and embedding techniques on Arabic sentiment analysis performance.

1 Introduction

During a time when digitalization is at its peak and platforms like Twitter (Heikal et al., 2018) are a crucial source of information for many as it is a platforms where public opinions are expressed in real-time in its most raw form, be it thoughts, opinions, personal experiences, etc. In today's world "tweets" are playing a vital role for both governments and organizations when it comes to understanding the social sentiment for them to make informed decisions it has become a critical requirement for models that can analyze and interpret sentiment from textual data. The primary objective of sentiment analysis is to study the emotional tone of textual data which is directly related to the formation of public opinion towards various services, products, brands, socio-political topics, etc. to understand the collective attitude towards a certain someone or something. The relevance or requirement of sentiment analysis is evident and has been of great help for organizations that treat public opinion with utmost importance.

Sentiment analysis is the computational study of people's opinions, attitudes and emotions toward entities, individuals, issues, events or topics (Heikal et al., 2018). Recently, deep learning has shown great success in the field of sentiment analysis but there lies a demand of accurate analysis of emotions when it comes to non–English languages, particularly Arabic because there's an immense amount of dialectal variation, lack of resources, polysemous words, other mixed languages, context etc. which increase the complexity of the data making difficult for models to parse or interpret data. Given as one of the most widely spoken languages globally there's a significant need of high-quality NLP models. NLP has undergone various changes during the development of various large language models (LLMs) such as GPT, T5, BERT etc. These models are based on deep learning methods and utilising various datasets these models have presented the world with advanced performance across numerous languages and NLP tasks.

The core objective of this paper is to work with and understand the sentiment analysis system for Arabic tweets by focusing on fine tuning models and NLP methods through various datasets, Arabic a language which is spoken by over 400 million people worldwide is know for its linguistic richness, complexity and deep historical roots, the project aims to classify the tweets into two categories posi-

Processings of the 1st Workshop on NLP for Languages Using Arabic Script (AbjadNLP), pages 88–94
January 19, 2025. ©2025 Association for Computational Linguistics

tive and negative, the system shall provide insights into the community views and collective viewpoint of the public, also this system can be utilized on various ends like e-commerce websites, political analysis, PR etc. the project will be addressing various challenges like complex linguistics, dealing with dialects, tokenization, to maximize the training process various models such as MiniLM (Aperdannier et al., 2024), XLM-R (Barbieri et al., 2021), USE (Saka and Cömert, 2024) is implemented. This paper will contribute to the advancement of sentiment analysis in Arabic which will bridge the language gap and eventually contribute to facilitating better and deeper decision-making in various domains that rely on Arabic text data.

2 Related work

As per (Al Sallab et al., 2015), several deep learning techniques are used to classify Arabic text such as Deep Neural Networks (DNN), Deep Belief Networks (DBN), Deep Auto-Encoders (DAE), and Recursive Auto-Encoders (RAE). Among all the architectures, Recursive Auto-Encoder proved to be the most effective as it could capture the context and sentence structure, addressing the shortcomings of the Bag-of-Words method used in the other models and giving an accuracy of 74.3%. As mentioned in (Duwairi et al., 2014), focused on creating a framework to classify Arabic tweets as positive, negative, or neutral. To address challenges like dialect variations, Arabizi (Arabic written in Roman characters), and the informal nature of tweets. A crowd-sourcing approach was used to collect and label a large dataset of tweets, with over 350,000 collected and 25,000 labeled for training. After applying preprocessing techniques such as tokenization, stopword removal, and stemming, they tested three classifiers: Naïve Bayes, k-nearest neighbors, and Support Vector Machines. Naïve Bayes performed the best, achieving an accuracy of 76.78. Limitations in the dialect dictionary and the dataset size impacted the overall accuracy in this research.

In (Al-Ayyoub et al., 2015), the authors present a framework that classifies Arabic tweets using a lexicon-based approach (Palanisamy et al., 2013). They constructed a sentiment lexicon consisting of over 120.000 Arabic terms and built a sentiment analysis tool that classifies tweets as positive, negative, or neutral. The tool was compared with a keyword-based approach and outperformed it, achieving an overall accuracy of 86.89. The accuracy for positive tweets was 96, for negative tweets 85.67, and neutral tweets 79.3. The work mentioned in (Heikal et al., 2018) designed an ensemble of CNN and LSTM to predict the sentiment of Arabic Tweets. Herein, the AraVec model has been used, primarily developed for Arabic with an F1 score of 64.46, the model demonstrated that the ensemble of CNN and LSTM works better and provides greater results. In (Abdul-Mageed et al., 2014) they have worked by using the SAMAR system which operates in a two-stage classification process. By combining features such as novel techniques and polarity lexicons, sentiment classification achieves an accuracy of 65.32.

3 System Architecture

3.1 Dataset

The dataset has been obtained from Kaggle[1] which is available by the name of “Arabic Sentiment Twitter Corpus". The number of instances present in the dataset is 56795, out of which 28513 tweets are labeled as positive and 28282 are labeled as negative, and its size is 5.9 MB. The coverage of the dataset began on 31st March 2019 and went on till 29th April 2019. The frequency of most used words in the tweets labeled as negative is visible in Figure 2, whereas the frequency of most used words in the tweets which have been labeled as positive can be seen in Figure 3 and the most frequent word in the overall dataset is represented in Figure 1. Many tweets in the dataset include emojis that intensify the sentiment removing or ignoring these could lead to a loss of sentiment context Figure 4, several tweets in the dataset show instances of mix-code or code switch where Twitter users have used a mix of both Arabic and English words, the dataset also includes informal writing styles.

3.2 Data Preprocessing

During this phase, first, the dataset was combined and shuffled, as the positive and negative datasets were initially separated. The next step involved cleaning the data and reducing noise. The primary goal of this phase was to help pre trained models generate better embeddings, enabling classifiers to give more accurate results. The steps included removing unwanted characters such as punctuation, special characters, dates, and times. Then, the tweets were tokenized, followed by the removal of

[1] https://www.kaggle.com/datasets/mksaad/arabic-sentiment-twitter-corpus?resource=download

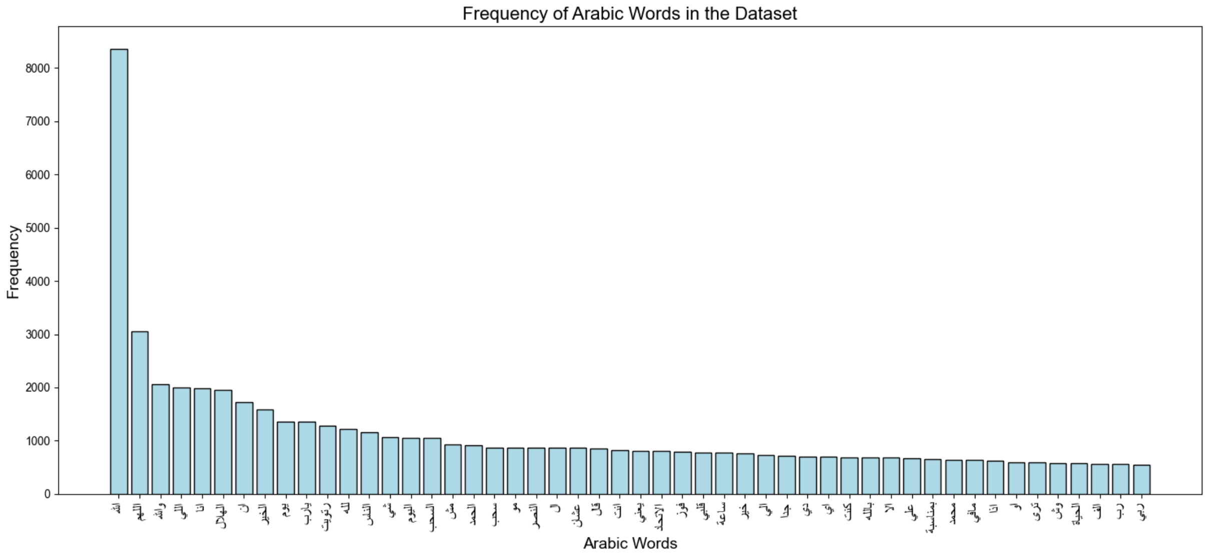


Figure 1: Frequency of Arabic words in the dataset

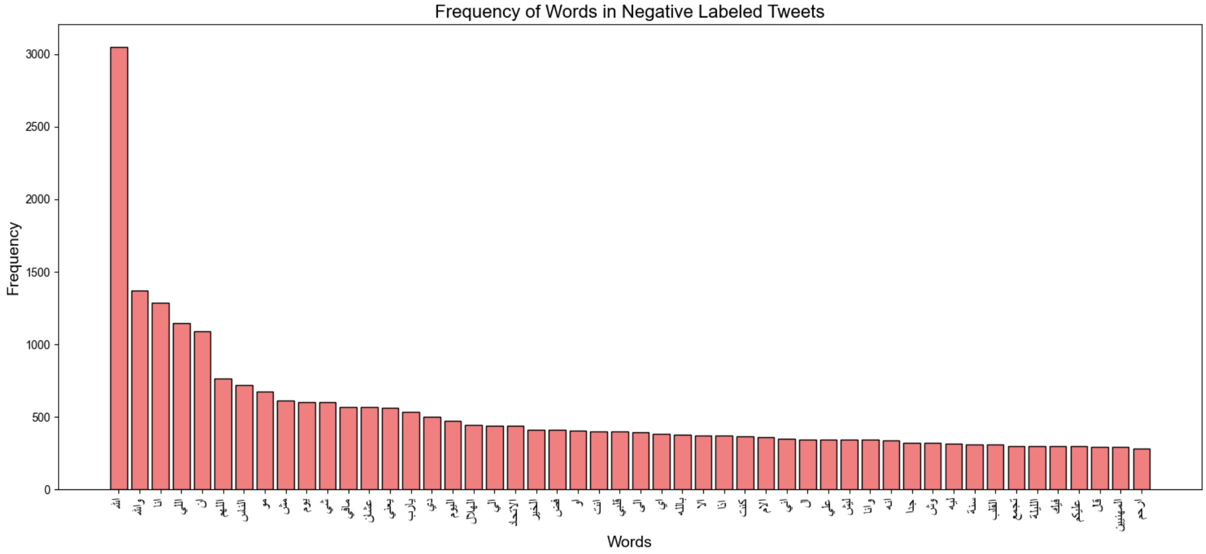


Figure 2: Frequency of Arabic words in the tweets labeled as negative

stopwords, and finally, the words were joined into a single string. Emojis were retained in the dataset to preserve the sentiment of the tweets. An excerpt of the preprocessed dataset is visible in Figure 4.

3.3 Model Training

Three models were used for the primary purpose of generating embeddings, one of the models used is MiniLM-L12-v2, this model uses transformer architecture similar to BERT the input is tokenized and passed through multiple transformer layers the final output is a 384-dimensional embedding vector for the entire input, it can also incorporate emojis as meaningful tokens it assigns embeddings based on how emojis co-occur with words and sentences in the training data. The other model which is used is USE, the transformer version is similar to other transformer models i.e. the text is tokenised and passed through different layers Each token's position and context are considered to create a 512-dimensional output vector that represents the entire input's semantics, emojis are included as part of the input sequence and are treated as tokens. Their embeddings are determined by their role in the text, similar to words. The third and final model which was incorporated is a multilingual version of RoBERTa, XLM-R trained in over 100 languages it shares the architecture of BERT with improvements in training techniques and larger-scale training data, the input text is processed by using a subword tok-

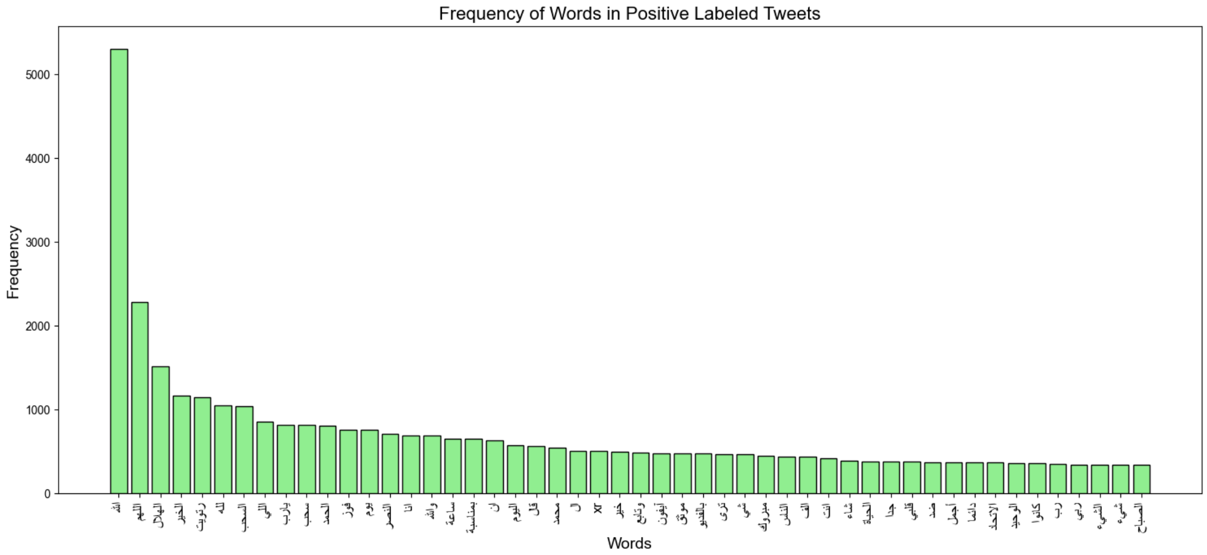
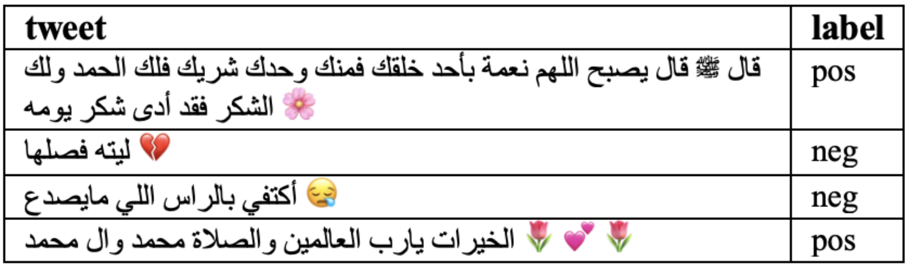


Figure 3: Frequency of Arabic words in the tweets labeled as positive



tweet	label
قال ﷺ قال يصبح اللهم نعمة بأحد خلقك فمنك وحدك شريك فلك الحمد ولك الشكر فقد أدى شكر يومه 🌸	pos
ليته فصلها 💔	neg
أكتفي بالراس اللي مايصدع 😪	neg
الخيرات يارب العالمين والصلاة محمد وال محمد 🌷 💕 🌷	pos

Figure 4: Excerpt from the preprocessed dataset

enizer that works by splitting the words into smaller and meaningful units, each token passes through a series of transformer layers like the other transformer models, final sentence embedding is generated by taking the mean of the token embeddings from the last transformer layer this pooling step summarizes the sentence's semantic content into a single vector, the model's attention mechanism captures how emojis relate to surrounding text, assigning appropriate semantic weights to them.

3.4 Classification

In this study, five different classifiers were employed to predict the correct label for a given input data: Logistic Regression, Support Vector Machine (SVM), Random Forest, Decision Tree, and K-Nearest Neighbors (KNN). Logistic Regression, a simple yet effective model, estimates probabilities for binary classification by fitting the data to a logistic curve, making it particularly useful for linearly separable datasets. Support Vector Machine, on the other hand, identifies an optimal hyperplane to separate classes and effectively handles both linear and non-linear data through the use of kernel functions. Its ability to perform well in high-dimensional spaces makes it especially suitable for text data and complex feature spaces. Random Forest employs an ensemble approach by constructing multiple decision trees and averaging their predictions, thereby improving accuracy and mitigating overfitting. This capability allows it to perform well on complex datasets by capturing intricate feature interactions. A Decision Tree, characterized by its simplicity and interpretability, uses a tree-like structure of decision rules to represent data and make predictions. Finally, K-Nearest Neighbors classifies an input by analyzing the K closest data points, making it effective for small to medium-sized datasets, though its computational intensity increases significantly with larger datasets.

4 Experimental Results and Analysis

The results indicate that MiniLM provided the highest accuracy in Arabic Tweets that included emojis, while XLM-R outperformed the others on the Arabic Tweets without emojis. A comparison of Table

Classifier	MiniLM	USE	XLM-R$_{\text{BASE}}$
Accuracy (%)			
LR	66.49	61.09	**69.46**
SVM	73.25	72.78	**74.40**
RF	78.52	74.78	**78.98**
KNN	73.30	71.04	**73.34**
DT	74.38	72.24	**74.48**
F1-Score (%)			
LR	66.48	61.08	**69.45**
SVM	73.17	72.71	**74.29**
RF	78.48	74.72	**78.92**
KNN	73.29	71.03	**73.34**
DT	74.38	72.23	**74.48**
Precision (%)			
LR	66.51	61.11	**69.52**
SVM	73.58	73.03	**74.83**
RF	78.78	75.04	**79.33**
KNN	73.35	71.04	**73.36**
DT	74.38	72.24	**74.48**
Recall (%)			
LR	66.49	61.09	**69.46**
SVM	73.25	72.78	**74.40**
RF	78.52	74.78	**78.98**
KNN	73.30	71.04	**73.34**
DT	74.38	72.24	**74.48**

Table 1: Experimental results without emojis

Classifier	MiniLM	USE	XLM-R$_{\text{BASE}}$
Accuracy (%)			
LR	**79.18**	69.62	78.08
SVM	**82.86**	81.94	80.88
RF	**85.98**	81.18	82.48
KNN	**81.65**	78.33	77.90
DT	74.38	75.25	**77.30**
F1-Score (%)			
LR	**79.18**	69.62	78.08
SVM	**82.86**	81.94	80.87
RF	**85.98**	81.18	82.46
KNN	**81.64**	78.32	77.90
DT	74.38	75.25	**77.30**
Precision (%)			
LR	**79.18**	69.64	78.09
SVM	**82.87**	81.94	80.95
RF	**86.00**	81.18	82.64
KNN	**81.73**	78.34	77.92
DT	74.38	75.25	**77.30**
Recall (%)			
LR	**79.18**	69.62	78.08
SVM	**82.86**	81.94	80.88
RF	**85.98**	81.18	82.48
KNN	**81.65**	78.33	77.90
DT	**80.02**	75.25	77.30

Table 2: Experimental results with emojis

1 and Table 2 clearly shows that the inclusion of emojis in the data set significantly improved the performance of all models evaluated.

In Table 2, MiniLM-L12-v2 demonstrated the highest accuracy, followed by XLM-R and USE. Further analysis revealed that MiniLM-L12-v2 excelled in sentiment analysis of Arabic tweets due to its transformer-based architecture and its ability to effectively utilize emojis as meaningful tokens to enrich semantic understanding. Its 12 transformer layers enable it to produce high-quality embeddings while maintaining a compact model size. This smaller size, optimized for semantic similarity tasks, allows MiniLM to capture sentiment-related nuances provided by emojis. In addition, its multilingual pre-training ensures strong performance on low-resource and multilingual datasets.

XLM-R, a multilingual transformer model, outperformed USE due to its extensive pre-training on a large corpus across 100+ languages, including Arabic. This pretraining allowed XLM-R to effectively understand Arabic and its dialects, making it particularly strong in purely textual datasets. However, its performance was comparatively weaker on datasets with emojis, as shown in Table 1 and Table 2. Being a general-purpose model, XLM-R's larger architecture may not be as fine-tuned for sentiment analysis as MiniLM, which slightly reduces its efficiency in this specific task.

USE (Cer, 2018) showed the lowest accuracy across both cases (with emoji and without emoji). This is likely because USE is designed for general-purpose sentence embeddings rather than specialized tasks like sentiment analysis. Although it supports multiple languages, its pre-trained corpus lacks sufficient Arabic-specific data, limiting its effectiveness in low-resource languages. It focuses on general sentence similarity tasks and struggles to capture the subtle details needed to identify sentiments in short Arabic tweets. However, USE demonstrated some improvement on datasets with emojis, as the emojis provided clear sentiment cues that mitigated its limitations to an extent. Despite this improvement, USE still lagged behind MiniLM and XLM-R in performance.

The accuracy achieved by each classifier varied across the embeddings (MiniLM, USE, and XLM-R BASE). For MiniLM, the Random Forest classifier achieved the highest accuracy of 85.98, followed by the SVN at 82.86, KNN at 81.65, logistic regression at 79.18, and decision tree at 74.38. With USE embeddings, Random Forest again led with 81.18 accuracy, followed closely by the support vector machine at 81.94, K-Nearest Neighbor at 78.33, Decision Tree at 75.25, and Logistic Regression at 69.62. Similarly, for XLM-R BASE, the random forest achieved the highest accuracy of 82.48, followed by the SVM at 80.88, KNN at 77.90, the decision Tree at 77.30, and Logistic Regression at 78.08. These results highlight that Random Forest consistently delivered the best performance across all embeddings in terms of accuracy.

The pre-trained models used in this research can also be applied effectively to other Abjad and Ajami languages. XLM-R, with its extensive pre-training on over 100 languages, demonstrates the ability to handle a variety of linguistic ambiguities. MiniLM-L12-v2, being lightweight yet effective, captures script-specific patterns well. While USE performs adequately, its performance in these languages could improve if fine-tuned on task-specific data.

5 Conclusions and Future Scope

This paper focuses on sentiment analysis of Arabic tweets by the use of large language models such as USE, XLM-R, and MiniLM, all three models showcased adequate accuracy with MiniLM providing the best accuracy of all, the high dimensional embeddings trained a robust model and the classifiers provided the right metrics. The results obtained are encouraging and promising keeping in mind the dialectal complexities of the Arabic language. For future work, the model can be further fine-tuned to provide even better accuracy.

References

Muhammad Abdul-Mageed, Mona Diab, and Sandra Kübler. 2014. Samar: Subjectivity and sentiment analysis for arabic social media. *Computer Speech & Language*, 28(1):20–37.

Mahmoud Al-Ayyoub, Safa Bani Essa, and Izzat Alsmadi. 2015. Lexicon-based sentiment analysis of arabic tweets. *International Journal of Social Network Mining*, 2(2):101–114.

Ahmad Al Sallab, Hazem Hajj, Gilbert Badaro, Ramy Baly, Wassim El-Hajj, and Khaled Shaban. 2015. Deep learning models for sentiment analysis in arabic. In *Proceedings of the second workshop on Arabic natural language processing*, pages 9–17.

Roman Aperdannier, Melanie Koeppel, Tamina Unger, Sigurd Schacht, and Sudarshan Kamath Barkur. 2024. Systematic evaluation of different approaches on embedding search. In *Future of Information and Communication Conference*, pages 526–536. Springer.

Francesco Barbieri, Luis Espinosa Anke, and Jose Camacho-Collados. 2021. Xlm-t: Multilingual language models in twitter for sentiment analysis and beyond. *arXiv preprint arXiv:2104.12250*.

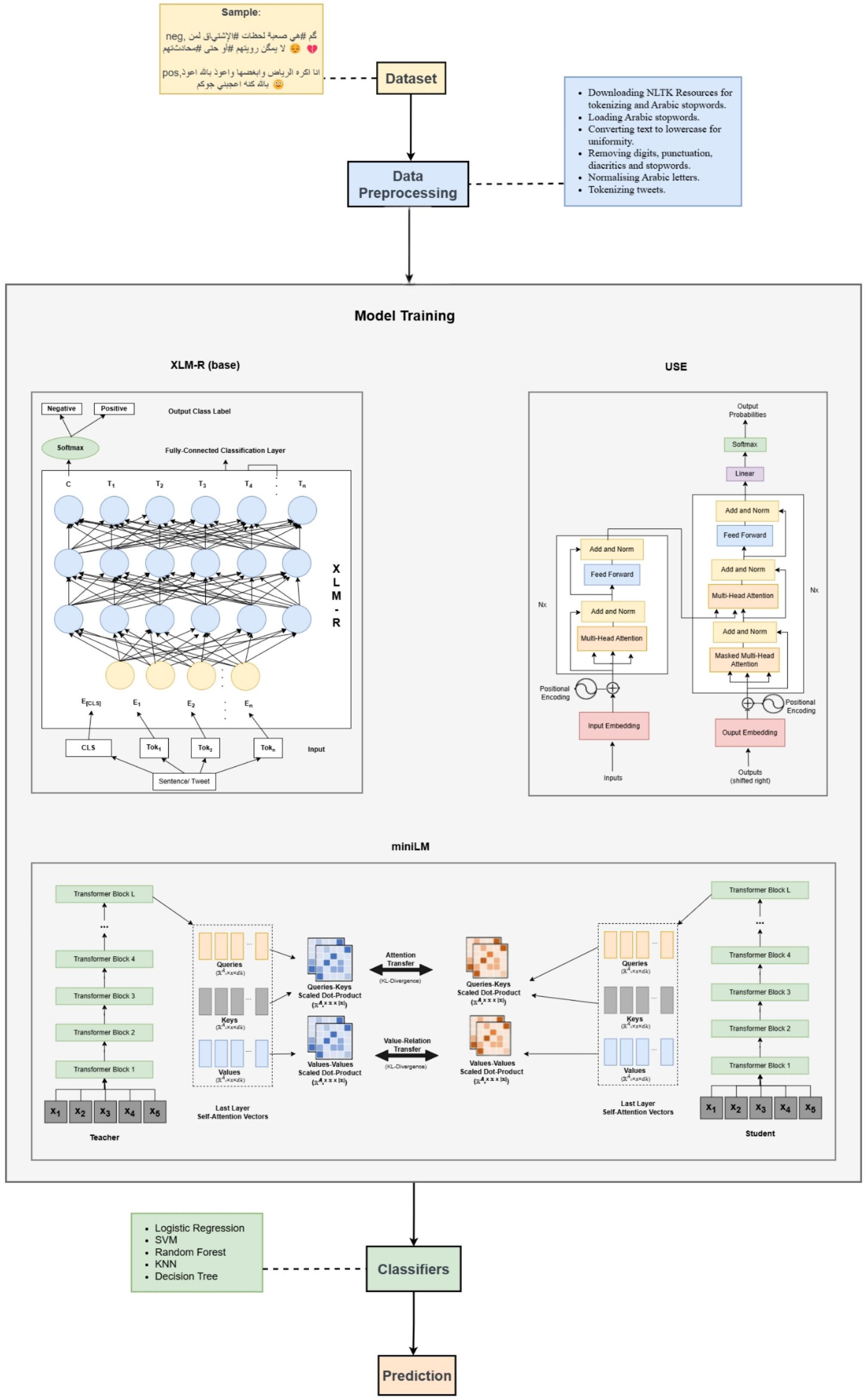
D Cer. 2018. Universal sentence encoder. *arXiv preprint arXiv:1803.11175*.

Rehab M Duwairi, Raed Marji, Narmeen Sha'ban, and Sally Rushaidat. 2014. Sentiment analysis in arabic tweets. In *2014 5th international conference on information and communication systems (ICICS)*, pages 1–6. IEEE.

Maha Heikal, Marwan Torki, and Nagwa El-Makky. 2018. Sentiment analysis of arabic tweets using deep learning. *Procedia Computer Science*, 142:114–122.

Prabu Palanisamy, Vineet Yadav, and Harsha Elchuri. 2013. Serendio: Simple and practical lexicon based approach to sentiment analysis. In *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 543–548.

Semih Osman Saka and Zafer Cömert. 2024. Sentiment analysis based on text with universal sentence encoder and cnn-lstm models. In *2024 8th International Artificial Intelligence and Data Processing Symposium (IDAP)*, pages 1–4. IEEE.



Sample:

گم #هي صعبة لحظات #الإشتياق لمن neg,
💔 😔 لا يمكن رويتهم #أو حتى #محادثتهم

انا اكره الرياض وابغضها واعوذ بالله اعوذ,pos
😳 بالله كنه اعجبني جوكم

- Downloading NLTK Resources for tokenizing and Arabic stopwords.
- Loading Arabic stopwords.
- Converting text to lowercase for uniformity.
- Removing digits, punctuation, diacritics and stopwords.
- Normalising Arabic letters.
- Tokenizing tweets.

- Logistic Regression
- SVM
- Random Forest
- KNN
- Decision Tree

Figure 5: Pictorial representation of the proposed system architecture which rely on three LLM models: XLMR-Base, USE, MiniLM for sentiment classification.