



ARTICLE

Arabic Dialect Identification in Social Media: A Comparative Study of Deep Learning and Transformer Approaches

Enas Yahya Alqulaity¹, Wael M.S. Yafooz^{1,*}, Abdullah Alourani² and Ayman Jaradat³

¹Department of Computer Science, College of Computer Science and Engineering, Taibah University, Medina, 42353, Saudi Arabia

²Department of Management Information Systems, College of Business and Economics, Qassim University, Buraydah, 51452, Saudi Arabia

³Internet of Things Department, Faculty of Science and Information Technology, Jadara University, Irbid, 21110, Jordan

*Corresponding Author: Wael M.S. Yafooz. Email: w.yafooz@taibahu.edu.sa

Received: 28 June 2024 Accepted: 15 August 2024 Published: 31 October 2024

ABSTRACT

Arabic dialect identification is essential in Natural Language Processing (NLP) and forms a critical component of applications such as machine translation, sentiment analysis, and cross-language text generation. The difficulties in differentiating between Arabic dialects have garnered more attention in the last 10 years, particularly in social media. These difficulties result from the overlapping vocabulary of the dialects, the fluidity of online language use, and the difficulties in telling apart dialects that are closely related. Managing dialects with limited resources and adjusting to the ever-changing linguistic trends on social media platforms present additional challenges. A strong dialect recognition technique is essential to improving communication technology and cross-cultural understanding in light of the increase in social media usage. To distinguish Arabic dialects on social media, this research suggests a hybrid Deep Learning (DL) approach. The Long Short-Term Memory (LSTM) and Bidirectional Long Short-Term Memory (BiLSTM) architectures make up the model. A new textual dataset that focuses on three main dialects, i.e., Levantine, Saudi, and Egyptian, is also available. Approximately 11,000 user-generated comments from Twitter are included in this dataset, which has been painstakingly annotated to guarantee accuracy in dialect classification. Transformers, DL models, and basic machine learning classifiers are used to conduct several tests to evaluate the performance of the suggested model. Various methodologies, including TF-IDF, word embedding, and self-attention mechanisms, are used. The suggested model fares better than other models in terms of accuracy, obtaining a remarkable 96.54%, according to the trial results. This study advances the discipline by presenting a new dataset and putting forth a practical model for Arabic dialect identification. This model may prove crucial for future work in sociolinguistic studies and NLP.

KEYWORDS

Dialectal Arabic; transformers; deep learning; natural language processing systems

1 Introduction

Social media usage has remarkably increased in recent years, with sites such as Facebook and Twitter—with their sizable, worldwide user bases—becoming indispensable to digital communication.



The way people connect, exchange, and express themselves—often in their original tongues—has been completely transformed by these networks [1]. This change has led to the emergence of a particular informal writing style that frequently disregards accepted orthographic and grammatical conventions [2]. Arabic is a distinctive national language with distinct structural features. Natural language processing (NLP) tasks become more complex due to their morphological richness and the existence of multiple dialects, requiring specific pre-processing [3]. Arabic is a prime example of diglossia; it comes in three main varieties: modern standard Arabic (MSA), dialectal Arabic, and classical Arabic. The Quran's original Arabic, known as classical Arabic, is the basis for MSA, a streamlined form that uses fewer diacritical markings. In formal settings, such as those involving government, education, and the media, MSA is the recommended medium [4]. Conversely, dialectal Arabic, which is distinguished by regional variances, is primarily utilized in conversational contexts and informal settings.

Dialectal Arabic differs considerably from MSA, demonstrating the wide linguistic diversity of the Arab world. Pronunciation and vocabulary are also different, as are grammatical structures, which are more simplified in spoken forms. Certain words and expressions may be added by other dialects, giving the language distinct regional characteristics. These dialects sometimes convey a more casual approach to language use and lend personality or comedy to written texts when they are translated. With the emergence of social media, dialectal Arabic has become more visible in written interactions, hence reducing the difference between formal and casual language usage. This development demonstrates how Arabic dialects are adaptable to online environments [5]. Many language processing domains, such as speech recognition, machine translation, information retrieval, and sentiment analysis, now depend on accurate dialect identification [6]. Major dialects, including Egyptian, Moroccan, Levantine, Iraqi, Gulf, and Yemeni, are more widely known in writing because dialectal Arabic is more common now [7]. These dialects have been the focus of most automatic dialect classification efforts, which have identified the following important categories: Egyptian, Gulf, Iraqi, Levantine, and Maghribi [6].

The increasing usage of Arabic dialects on social media, along with their diverse types and forms, particularly between MSA and dialectal Arabic, has presented an urgent need for their study [8], which introduces considerable challenges. The main challenge for Arabic dialects is language identification, which can be particularly difficult because all types of Arabic use the same set of letters, and much of the vocabulary is shared between the different types. As a result, differentiating between dialects and separating them from each other [9], especially the dialects of one region, are difficult. Additionally, these dialects exhibit remarkable morphological, syntactic, lexical, and phonological similarities [10].

Moreover, in online content creation, including comments and blogs, authors frequently blend MSA and one or more Arabic dialects within a single paragraph or sentence. Additionally, the Arabic script can conceal the true sounds and pronunciations of letters and words across different dialects in written form [11]. Managing Arabic dialects becomes a more considerable challenge compared with dealing with MSA given these difficulties; as a result, NLP has progressed to incorporate various techniques aimed at effectively classifying the diverse existing dialects.

Several methods are used to identify Arabic dialects, each providing a unique viewpoint on the examination of linguistic data. Sophisticated deep learning (DL) networks, advanced transformer models, and traditional machine learning (ML) algorithms are just a few of the ways that combine various approaches for thorough analysis. In addition to these approaches, using various word representation techniques is essential for transforming textual input into a format that can be analyzed by transformer, ML, and DL models. Word embeddings, such as FastText, Word2Vec, and GloVe, and word representation techniques such as TF-IDF, are essential. By capturing complex semantic

links and meanings, these approaches offer rich, context-aware word representations that improve the models' ability to distinguish and identify between Arabic dialects [12].

Whilst most Arabic NLP research in the past has been on MSA, more recent work has focused on dialectal Arabic. Currently, resources available for dialectal Arabic are considerably fewer in terms of scope and size in comparison with MSA and other languages such as English [13]. Thus, earlier Arabic NLP research that dealt with texts from social media and the internet had to fill in the gaps with its own annotated resources [14]. However, further research must be conducted to provide larger and more thorough annotated resources for Arabic dialects, especially for issues such as Arabic dialect identification.

Thus, this study endeavors to examine and compare the effectiveness of three distinct models (i.e., ML, DL, and transformers) separately for the classification of Arabic dialects on social media platforms. The comparative analysis aims to identify the strengths and weaknesses of each approach in accurately identifying Arabic dialects within the context of social media communications. The main contributions of this study can be summarised as follows:

- An innovative DL model that integrates long short-term memory (LSTM) and Bidirectional (BiLSTM) architectures for enhanced feature extraction and improved classification accuracy in identifying Arabic dialects from social media content is proposed.
- A novel dataset consisting of approximately 11,000 user comments extracted from Twitter is introduced. This dataset focuses on the Arabic language only and uses a three-class classification: Saudi, Egyptian, and Levantine Arabic dialects.
- The proposed model is evaluated using word representation techniques, such as TF-IDF, Fast-Text, Word2Vec, and GloVe, to boost performance. With data preprocessing and information extraction, these methods enable the model to discern subtle variations in Arabic dialects effectively.

The remainder of this paper is organized as follows: [Section 2](#) provides a comprehensive literature review relevant to the methodologies used in the development of the proposed models. [Section 3](#) details the methods and materials utilized in these approaches. [Section 4](#) presents the model developed for the research and elaborates on the experimental setup. [Section 5](#) discusses the findings from these models. Finally, [Section 6](#) concludes the study and [Section 7](#) suggests potential future directions for this research.

2 Related Work

This section outlines key research contributions in the area of Arabic dialect identification systems. The review is structured around three primary approaches: ML, DL, and hybrid learning.

Using of ML provides a more efficient and accurate way of identifying and classifying dialects the Arabic dialect language identification task was extensively studied by NLP researchers. Despite the numerous works on dialectal Arabic, dating back several years, we focus on the recent ones. Research work mainly addressed building dialectal corpora, in addition to language classification between MSA and the varieties of dialectal Arabic. Obeid et al. [15] and Ghoual et al.'s [16] research in their works applied the MNB classifier on the MADAR dataset which contains, besides those 25 dialects, sentences in MSA, English, and French. In the work of Obeid et al. [15], they applied MNB with N-gram and TF-IDF to build an Automatic Dialect Identification for Arabic (ADIDA) which is an online interface that uses a system to identify Arabic dialects. It creates a probability vector for 25 cities and MSA and displays the results on a map to help researchers and language learners

better understand the variations in Arabic dialects and their geographic distribution. The findings indicate that the model was able to correctly determine a speaker's city with a 67.9% accuracy rate for sentences averaging 7 words in length. On the other hand, in [16], they presented MICHAEL which uses character-level features to perform classification without pre-processing, and trains MNB classifier on character N-grams from the original sentences. the accuracy score of 53.25% with $1 \leq N \leq 3$, but a higher accuracy of 62.17% was obtained using character 4-grams. The Nuanced Arabic Dialect Identification (NADI) shared task aims to encourage research on Arabic dialect processing by providing datasets and evaluation setups. NADI 2020 focused on province-level dialects, while NADI 2021 focused on identifying both MSA and DA based on their geographical origin at the country level. NADI 2022 offers two subtasks—dialect identification and dialectal sentiment analysis. Teams were encouraged to submit systems for both subtasks, with the hope of receiving diverse machine learning and other methods and architectures [17]. In NADI 2021 [18], the data used in this task covers a total of 100 provinces from all 21 Arab countries and comes from the Twitter domain. Ali et al. [19] used five machine-learning approaches, including Complement Naive Bayes (CNB), Support Vector Machine (SVM), Decision Tree (DT), Logistic Regression (LR), and Random Forest (RF) Classifiers with TF-IDF for feature extraction after the pre-processing process. The dataset is comprised of two levels of classification, country-level with 21 classes and province-level with 100 provinces. Each province related to one of the 21 countries, the results show that the NB classifier outperformed all other classifiers with the highest F1 macro-averaged score for both development and test data. In NADI 2022 shared task [17]. Jauhiainen et al. [20] used a NB-based language identifier that employs character N-grams ranging from one to four. They also developed a new version of the system that optimizes its parameters automatically. The team experimented with clustering the training data based on different topics. The system achieved a macro F1 score of 0.1963 on test set A and 0.1058 on test set B. The study of El Chrif et al. [21] explored the application of machine learning algorithms to classify Arabic dialect texts, specifically focusing on the Mauritanian dialect. The research utilizes a dataset comprising comments in the Mauritanian dialect from Facebook, testing the effectiveness of stemming methods combined with machine learning models such as RF, NBM, and LR. The study finds that RF and NBM algorithms achieve improved performance when used with the ArabicStemmerKhoja, with accuracies of 96.37% and 71.40%, respectively. Conversely, Logistic Regression shows the best results with Null Stemmer, reaching an accuracy of 81.65%.

DL is a subfield of machine learning that has caught great interest from researchers in the past few years. It's known for working well with big amounts of data and can handle complex, unstructured information, automatically identifying important features [22]. In the NADI shared task Talafha et al. [23] and Bayrak et al. [24] used a pre-trained BERT model. Talafha et al. [23] presented a model on NADI 2021 shared task which are an ensemble of different iterations of the pre-trained BERT model called Multi-dialect-Arabic-BERT. The model was pre-trained on 10 million tweets and trained multiple times on labeled data with varying sentence lengths and learning rates. The top 4 performing iterations were aggregated using an averaging function to produce the final prediction for a given tweet, achieving a micro-averaged F1 score of 26.78%. on the other hand, in the NADI 2022 shared task, Bayrak et al. [24] developed a deep learning model using AraBERT and MARBERT for classifying Arabic dialects and sentiment analysis of tweets. The model achieved high scores for both tasks using test datasets supplied by NADI. MARBERTv2 with pre-processing performed best for identifying dialects with a Macro F1 score of 33.89%, while MARBERT without pre-processing achieved the highest Macro-F1-PN score of 74.29% for sentiment analysis.

The study by Lulu et al. [25] used the AOC dataset to classify the most frequent dialects in the dataset, including Egyptian, Levantine, and Gulf, and explored four deep neural network

models LSTM, CNN, BiLSTM, and CLSTM. They conducted four experiments for binary and ternary classification and found that BiLSTM performed the best for the EGP-GLF pair, while LSTM performed the best for the remaining pairs and the ternary classification task. However, they reported classification errors for the EGP dialect as LEV and recommended further examination and refinement of the AOC annotated sentences. In Arabic Dialect Identification (ADI) shared task, Ali [26] investigated a CNN architecture that operates on the character level to classify five Arabic dialects, namely GLF, MSA, EGY, MAG, and LEV. The architecture consists of five sequential layers, starting with an input layer that maps each character in the input into a vector, followed by a convolutional layer, a max pooling layer, and two sequential fully connected softmax layers. The system achieved a classification F1 score of 57.6%. The proposed approach of using character-level convolution neural networks and dialect embedding vectors is a valuable contribution to the existing literature on Arabic dialect identification.

The study of Fares et al. [27] used data pre-processing and data augmentation techniques to expand the training set and evaluate different models based on various word and document representations like TF-IDF and FastText. The results show that improving traditional systems using frequency-based features and non-deep learning classifiers is challenging. The top model proposed in the paper achieves an F1 macro-averaged score of 65.66 on MADAR's small-scale parallel corpus of 25 dialects and MSA. In the field of detecting a specific dialect, Younes et al. [28] identifying Tunisian dialect on social media, the proposed method uses a deep learning approach based on BiLSTM with CRF decoding (BiLSTM-CRF), which utilizes word-char BiLSTM embeddings combined with FastText embeddings to capture each word's morphology and character N-gram features. The overall accuracy achieved by the method is 98.65%, Table 1 gives a summarizing these studies based on Deep Learning approaches. The hybrid learning approach combines the strengths of both traditional ML methods and DL techniques to increase the accuracy and efficiency of Arabic dialect identification. In the MADAR dataset, De Francony et al. [29] proposed two methods for Arabic Dialect Identification with TF-IDF and Word2Vec features, the first method utilizes recurrent neural networks (BiLSTM, BGRU) with hierarchical classification. The approach involves dividing the sentence classification process into two phases: a higher-level classification with 8 classes, followed by a finer-grained classification with 26 classes. The second method involves a voting system based on NB and RF. The method achieves an F1 score of 63.02% on the subtask evaluation dataset. On the same corpus, Mishra et al. [30] implemented pre-processing steps and use of different features such as character and word N-grams, TF-IDF, language model probabilities, and different ML classifiers (Linear SVM, MNB, LR) with multi-layer perception (MLP) neural network, the texts were either tourist help guides (subtask 1) or social media texts (subtask 2). The data set size was 46,800 Sentence and 26,7554 Tweets, an accuracy of 66.31% and 67.20 % on both sub-tasks, ultimately finding that traditional machine learning classifiers performed better than neural network models in low-resource settings.

In NADI 2021 shared task, the authors [31] presented models to distinguish between 21 country-level Arabic dialects. They investigated several machine learning approaches (SVM, LR, RF) and deep learning models (GRU, BiLSTM) with various features (CBOW, Word2Vec, TFIDF, N-gram) using small datasets. The authors also created dictionaries based on Pointwise Mutual Information and labeled datasets to enrich the feature space. The SVM with dictionary-based features and PMI values was found to be the winning model, achieving an 18.94% macros-average F1 score. In NADI 2022 shared task, Abdul-Mageed et al. [32] created model has several machine learning techniques, as well as deep learning transformer-based models, were utilized. An ensemble classifier of various machine learning models was built such as LR, SVM, and MNB. The features were extracted using TF-IDF vectors with N-grams. For the deep learning approach, a BiLSTM model and AraBERT

pre-trained model were considered with word embeddings such as Word2Vec and The CBOW. It was demonstrated that the deep learning approach performed noticeably better than the other machine learning approaches with 68.7% accuracy on the development set. In the NADI 2023 shared task, Hatekar et al. [33] explored the use of both traditional machine learning models and advanced deep learning techniques for Arabic dialect classification in tweets. Their methodology combined LR, SVM, and MNB with deep learning approaches like BiLSTM and AraBERT pre-trained models. The models were trained on a primary dataset of tweets from 18 Arab countries, enhanced with TF-IDF vectors incorporating N-grams. Notably, the deep learning models, particularly those utilizing word embeddings such as Word2Vec and CBOW, significantly outperformed the traditional methods, achieving a remarkable 68.7% accuracy on the development set.

Table 1: A summary of recent studies using different methods

Ref.	Approaches	Methods/ classifier	Word rep.	Dataset size	Classes	Evaluation results
[15]	ML	MNB	N-gram	N/A	25 Arabic countries	Acc. = 67.90%
[16]	ML	MNB	TF-IDF N-grams	46,800 sentence	MSA 25 Arabic countries	Acc. = 62.17%
[23]	DL	BERT	N/A	MSA 31,000 tweets	21Arabic countries	F1 = 26.78%
[28]	DL	BLSTM BLSTM-CRF	Fast text N-gram	13,656 messages	LTD NLTD PUNCT EMO SYMB EMO	Acc. = 98.65%
[30]	Hybrid learning	SVM MNB LR MLP	TF-IDF N-grams	46,800 sentence 26,7554 tweets	26 Arabic countries	Acc. = 66.31% Acc. = 67.20%
[32]	Hybrid learning	LR SVM MNB BiLSTM AraBERT	TF-IDF N-grams CBOW BOW Word2Vec Fast text	25,269 tweets	18 Arabic countries	Acc. = 68.7%

3 Methods and Materials

This section presents a model that clarifies the necessary phase to reach the final goal. The proposed model involves steps as shown in Fig. 1. These steps include data collection, data cleaning, data pre-processing, feature extraction, model construction, and model performance evaluation. The initial phase involves gathering data and constructing a dataset, it includes key factors for implementing the proposed concept. next, the collected data is cleaned and readied for processing. The core of the proposed concept is formed by three phases: feature extraction, model construction, and model performance evaluation.

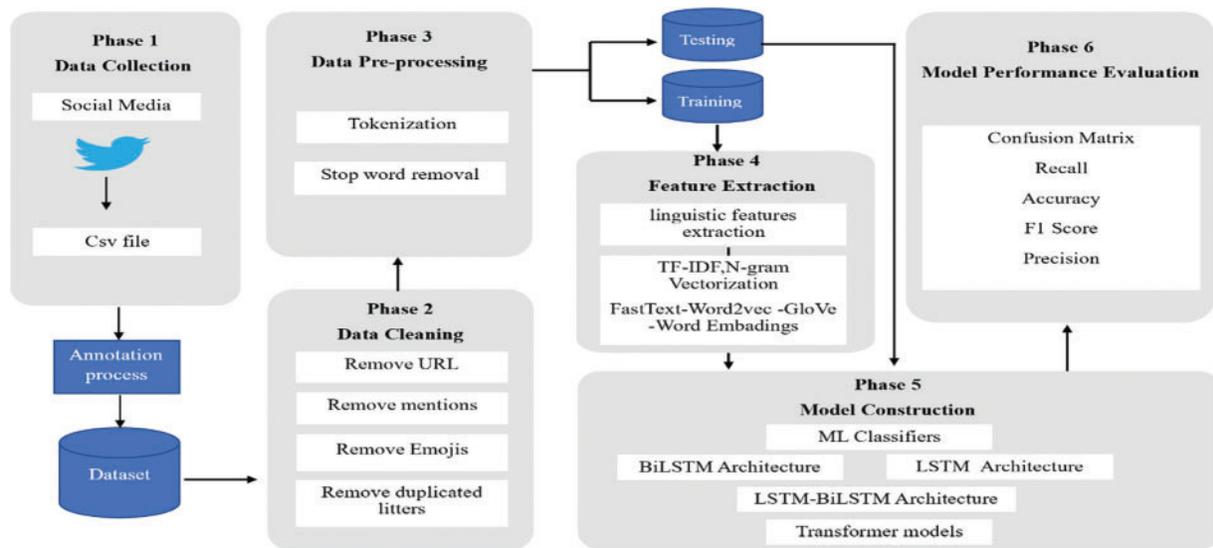


Figure 1: The proposed approach

3.1 Data Collection

In this research, the first step involves collecting data, this is crucial to get an accurate and in-depth analysis. This process, primarily conducted through data scraping from social networking sites like Twitter, aims to gather a diverse range of tweets that reflect various Arabic dialects due to Twitter's broad usage and the real-time nature of its content. Using Twitter's APIs, data is collected based on specific keywords related to Saudi, Egyptian, or Levantine dialects, ensuring the dataset's quality and relevance, as illustrated in Table 2. The selection of Saudi, Egyptian, and Levantine dialects for this study was based on several key considerations. Firstly, these dialects are highly prevalent and widely used across different regions. Saudi Arabia, for instance, is spoken in Saudi Arabia and is one of the most influential dialects in the Gulf region. A significant portion of Arabic content on social media is represented by this dialect due to the country's high internet penetration and active social media usage. Egyptian Arabic is the most widely understood dialect across the Arab world, primarily because of Egypt's prominent media industry, including films, television shows, and music, which have a vast audience. This makes Egyptian Arabic highly recognizable and prevalent. The Levantine dialect, spoken in countries such as Lebanon, Jordan, Syria, and Palestine, has a significant presence on social media and is known for its distinct phonological and lexical features. In addition, these dialects were chosen to represent three distinct regions within the Arab world, providing a diverse sample that captures significant linguistic variation. This diversity is crucial for understanding the broader spectrum of Arabic dialects and enhances the model's generalizability. Furthermore, these dialects have been the focus of previous studies, providing a foundation of existing resources and annotated data. Leveraging these resources allows for more accurate comparisons and evaluations of new models, contributing to the robustness of the study.

Table 2: Examples of keywords for Saudi, Egyptian, and Levantine dialects

Arabic	English transliteration	Meaning (Contextual)
قهوجي	Qahwaji	Coffee server/maker
علومك	'Ulumak	What's new with you?
بناعتك	Beta'atak	Yours (informal possessive)
محدش	Mahadesh	No one (negation)
لمي حالك	Lami halak	Pull yourself together/Get a grip
بلكي	Balke	Maybe/Perhaps

The collection aims to capture a wide spectrum of linguistic expressions, providing a solid basis for studying Arabic dialect nuances within social media discourse. This phase results in 11,000 comments and saves as a CSV file following the deletion of some comments due to uncertainty about their dialect classification, the revised totals for each dialect, along with the overall count, are as follows (Table 3).

Table 3: Number of tweets for each dialect

Dialect	Number of tweets
Saudi	2866
Egyptian	3065
Levantine	4749
Total	10,680

Finally, the CSV file, which encapsulates all the extracted comments, will be used for the annotation process. The dataset annotation is a crucial phase where each comment is carefully labeled according to its corresponding Arabic dialect. As outlined in Table 4, the labels are assigned as follows: 1 for Saudi, 2 for Egyptian, and 3 for Levantine dialects. To ensure the highest accuracy in classification, this process involves three annotators. Each annotator independently reviews and classifies every tweet. This multiple-annotator approach helps minimize biases and errors, ensuring that each comment is accurately classified into the appropriate dialect category.

Table 4: Class for each dialect

Dialect	Class
Saudi	1
Egyptian	2
Levantine	3

As previously mentioned, the analysis of unique word usage is crucial for distinguishing among Saudi, Egyptian, and Levantine dialects within our dataset. This analysis has led to the identification of distinct linguistic markers that characterize each dialect, as a result of this enriching the dataset's linguistic profile. This differentiation is fundamental for developing more sophisticated NLP technologies capable of accurately recognizing and interpreting these diverse Arabic dialects. Table 5 contains

a sentence from Saudi, Egyptian, and Levant Arabic dialects, each with its English translation. The sentences provide insight into the versatile style of each dialect.

Table 5: Example of dialects

Dialect	Sentence	Translation
Saudi	يكفي إنه لو سألني كيف حالك بيبتسم حالي ولو ماني بطيب	It's enough that if you ask me how I am, my condition smiles even if I'm not fine.
Egypt	يا عم انت فين؟ الدنيا مستنياك	Hey man, where are you? The world is waiting for you.
Levant	كيف الحال؟ شو أخبارك؟	How are you? What's up?

The second phase involves the cleaning of the collected social media data by removing elements like usernames, URLs, and emojis, and standardizing the text to lowercase with single spaces, making it ready for analysis.

3.2 Pre-Processing

In the data pre-processing phase, cleaning is crucial to prepare text for accurate insight extraction and is especially important for text classification in noisy online environments. This involves removing unimportant content like HTML tags and ads. Removing words that hardly impact classification can reduce complexity and improve classifier performance. The main goal of efficient data preprocessing is to enhance performance by eliminating noise from the text [34]. Within this phase, three critical pre-processing steps are implemented in Arabic dialect identification: Tokenization, Stop Word Removal, and Lemmatization:

- **Tokenization:** This process breaks down the text into its basic units or tokens, such as words or phrases. For example, Given the sentence “بنتور الرياض وبتنور السعودية كلها”, tokenization would break it down into individual tokens as follows: [“بتنور”, “الرياض”, “و”, “بتنور”, “السعودية”, “كلها”]. Each component of the sentence is separated, allowing for detailed analysis and processing.
- **Stop Word Removal:** This step involves eliminating commonly used Arabic words that typically do not add significant meaning to the sentence, such as “في” (in), “على” (on), and “و” (and). For example, after removing stop words, the phrase “جمال اللهجة في تعبيراتها” (the beauty of the Egyptian dialect in its expressions) might be reduced to [“جمال”, “اللهجة”, “تعبيراتها”].
- **Lemmatization:** For the purpose of this study, lemmatization, which generally transforms words to their base form, is not applied. This is done to preserve the distinctive word variations crucial for dialect identification. For example, the word for “writing” could appear as “يكتبون” in MSA. However, in the Levantine dialect, it might be “عم يكتبوا” and in Egyptian dialect, it could be “بيكتبوا”. Lemmatizing these verbs to their root “كتب” would erase the specific features that indicate a particular dialect. The decision not to use lemmatization is important for dialect identification because it allows the model to capture the nuances and specific variations in word forms that are characteristic of different dialects. Oftentimes, dialects differ in their use of prefixes, suffixes, and infixes, which are critical markers of regional linguistic identity. For example, the prefix “عم” in Levantine Arabic and “بي” in Egyptian Arabic are indicative of continuous tense in these dialects, which would be lost if lemmatization were applied. Moreover, lemmatization might lead to the loss of contextual information that is essential for understanding the syntactic and

semantic differences between dialects. Each dialect has its own set of morphological rules and patterns, and retaining these variations helps in building a more accurate and robust model for dialect identification. By keeping the words in their full, dialect-specific forms, the model can better learn and recognize the unique usage patterns that distinguish one dialect from another.

By applying these steps, the data is streamlined, retaining meaningful content that aids in the effective identification of regional dialects within the study's corpus.

3.3 Feature Extraction

In the Feature Extraction phase, the focus shifts to transforming the meticulously pre-processed data into a format that's interpretable by the models developed for dialect identification. This phase is crucial in teasing out the linguistic nuances necessary for the algorithms to recognize and differentiate between the various Arabic dialects, this transformation is achieved through a combination of techniques designed in order to encapsulate the richness of the Arabic language:

- **TF-IDF with N-gram Vectorization:** Leveraging the TF-IDF technique in combination with N-gram analysis allows the model to evaluate not just the frequency of words but also their contextual relevance in sequences of 'n' words. This approach is adept at capturing the complex word usage patterns that are emblematic of Arabic dialects, offering a deeper insight into the syntactic and semantic intricacies.
- **Advanced Word Embeddings:** The study employs state-of-the-art word embedding methods including FastText, Word2Vec, and GloVe. These methods are necessary in translating textual information into a numerical format by generating dense vector representations for words. Such embeddings are capable of capturing and representing the subtleness of the semantic and syntactic relationships between words, which is important for distinguishing between the subtle differences in dialectal Arabic.
- **To create an embedding matrix for the vocabulary in the dataset,** Arabic words are converted into vectors using the FastText model "cc.ar.300.bin". The vocabulary size and maximum sequence length are established once the model has been loaded. Vocabulary terms have their vectors collected and used to fill the embedding matrix; words that are not in the vocabulary are given zero vectors. The approach yields an embedding matrix that is included into the model, enhancing its capacity to recognize and classify subtleties in Arabic dialect. FastText works better than other embedding methods because it takes into account subword information, which is crucial for Arabic because of its complex morphology. FastText handles the inflectional characteristics of Arabic dialects better by capturing subtler differences and similarities between words by breaking words down into character N-grams. Obeid et al. [15] demonstrated that character-level information significantly enhances the performance of dialect classification models.
- **To improve model accuracy,** the study uses pre-trained GloVe embeddings to translate text into vector representations. The 300-dimensional GloVe vectors are loaded into an embedding_index using a custom function. An embedding matrix is generated to match the vocabulary of the dataset, because of the matrix's integration into the model's design, text may be effectively classified by converting it to vectors utilizing GloVe embeddings' semantic depth. Although GloVe is more skilled at gathering broad statistical data regarding word co-occurrences in a corpus, its inability to include subword information hinders its capacity to manage the intricate morphology of Arabic dialects to the same extent as FastText. Models that rely on GloVe

embeddings frequently have difficulty in handling the diversity and richness of Arabic dialects, according to the research done by Ali et al. [19].

- The text in this study is transformed into vectors using Arabic Word2Vec embeddings. Afterwards, a Word2Vec model is trained on the dataset, and the vectors are then integrated into a neural network to classify Arabic text. To ensure that the model catches the semantic nuances of the text for effective classification, the method consists of tokenization, sequence padding, and one-hot encoding of class labels. By predicting words based on their surrounding context, Word2Vec focuses on local context, which is advantageous for capturing syntactic and semantic links. However, Word2Vec lacks subword handling, which is a crucial component for efficiently handling Arabic dialects, just like GloVe. Because of this restriction, it is less efficient than FastText for this particular activity. Studies by Jauhiainen et al. [20] found that while Word2Vec is effective at capturing local word relationships, it struggles with morphologically rich languages where subword information is crucial.

4 Proposed Model

This section presents the proposed model for dialect classification. As the study progresses, word frequency and contextual analysis are used in classical ML to establish a baseline. It moves on to a suggested model that incorporates word embeddings, improving its capacity to capture the subtleness of Arabic dialects. Next, the model moves on to transformers, where it improves dialect identification by using self-attention techniques. This method progresses from basic ML techniques to advanced DL models, providing a basis for comprehending the intricacies of the Arabic dialect and outlining their methods for thorough dialect analysis. The following is a detailed description of the approaches, including their main elements and methods.

4.1 Machine Learning

ML algorithms have two types: supervised and unsupervised. For tasks such as text classification in NLP, which classifies texts into specified categories, supervised learning uses labeled data for training, requiring the improvement of retrieval systems. In this study, Arabic dialect classification was accomplished specifically through the use of supervised ML, which was trained on labeled datasets and then tested on fresh data, to assess a model's capacity for classification. Several classifiers, including LR, RF, SVM, DT, and BNB, were used in conjunction with text analysis methods, including TF-IDF and N-grams [35]. In this first stage, a performance baseline was established, how well conventional techniques captured the subtleties of Arabic dialects was evaluated, and a framework for introducing more refined models into the research was created [36].

4.2 Deep Learning

With the goal of modeling high-level abstractions in data, DL uses algorithms to learn representations at several layers. The capacity of DL to perform feature engineering automatically is a major benefit over standard ML, especially when it comes to producing concise, effective representations for text by creating its parts [37]. This study investigates Arabic dialect identification using DL methods, particularly LSTM and BiLSTM models. LSTM models are preferred over conventional recurrent neural network structures due to their better contextual information processing capabilities and better memory management. These characteristics are critical for efficiently handling long-term dependencies that are needed for difficult tasks, such as time-series analysis and language modeling. The intention is to use LSTMs' strong context retention and sequential data processing

capabilities. Three primary architectural techniques were used in the experiments: using LSTM models freestanding, using BiLSTM models, and combining them to enhance contextual analysis in the suggested architecture. LSTM and BiLSTM networks were used in the extensive studies, which were focused on optimizing the topologies of the networks by adjusting parameters, including layer count, neuron count per layer, batch sizes, epochs, and dropout rates.

4.2.1 LSTM

As shown in Fig. 2, an LSTM neural network starts with input text that is converted into numerical vectors by the embedding layer. This neural network type is especially good at learning from sequences, such as sentences or time-series data, because it can recall prior knowledge and use it to influence the current processing task [37]. Subsequently, an LSTM layer analyses these vectors, with consideration to the word order and context. The network then classifies the text into one of the dialects—Saudi, Egyptian, or Levantine—after a dense layer analyses the LSTM’s output.

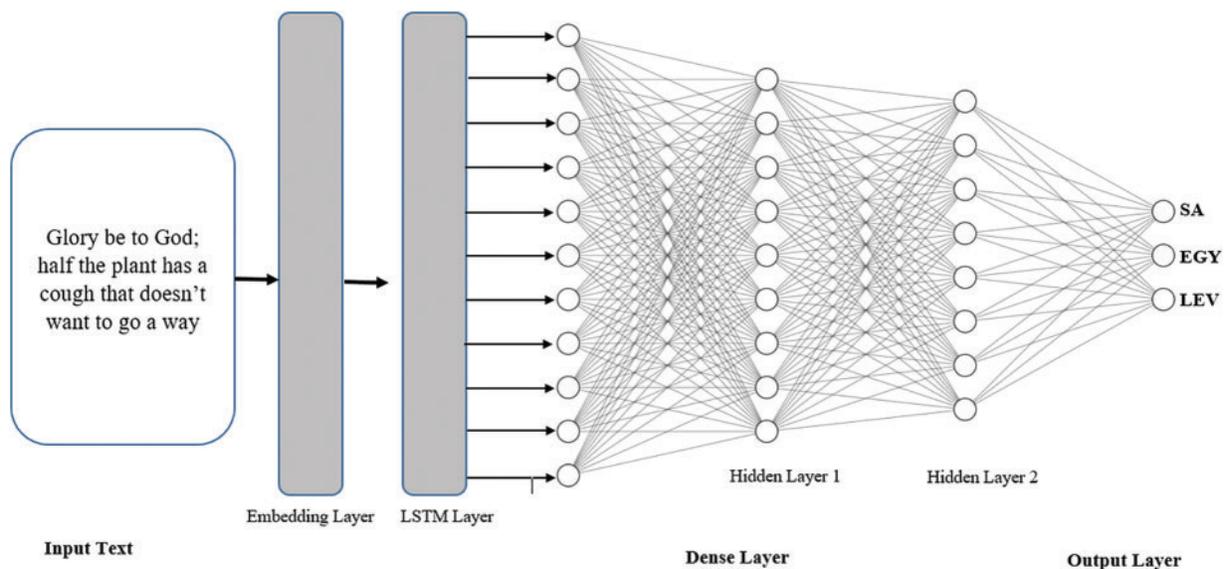


Figure 2: LSTM architecture

4.2.2 BiLSTM

An LSTM version called BiLSTM has the potential to enhance model performance in sequence classification issues. BiLSTM analyses data in both forward and backward directions, in contrast to conventional LSTMs, which only process data in one direction. As a result, one can fully understand the context by capturing context from both the past and the future in the sequence [38]. An input text is fed through an embedding layer, as shown in Fig. 3, before being transformed into vectors. Afterward, a BiLSTM layer processes them, reading the data forward and backward to understand the context. Ultimately, this data is analyzed by a dense layer, which assigns a particular dialect to the statement.

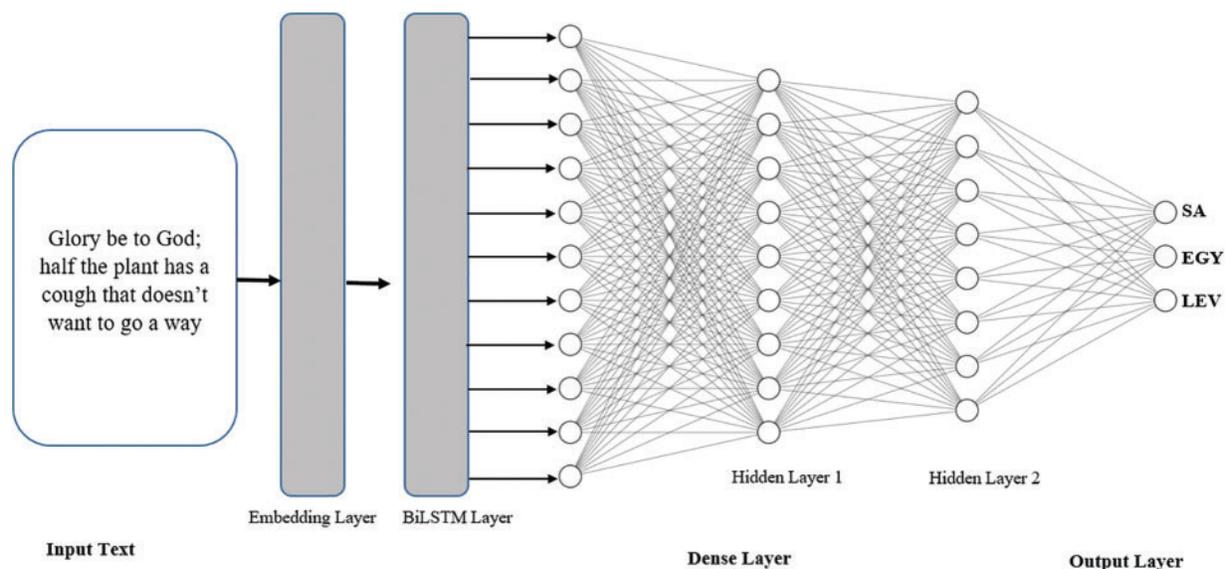


Figure 3: BiLSTM architecture

4.2.3 LSTM-BiLSTM

By utilizing the advantages of both architectures, combining LSTM and BiLSTM layers in a neural network enables the model to comprehend the sequence context more thoroughly [39]. This method works especially well for dialect categorization, where accuracy can be greatly increased by knowing the meaning of words from both directions. As shown in Fig. 4, the suggested neural network for dialect classification starts with an input sentence that is converted by an embedding layer into numerical vectors. This layer captures the semantic associations between words by converting each word into a denser vector of a fixed size. Subsequently, the embedding sequence is fed into an LSTM layer, which proceeds to process the sequence forward, taking in prior words to understand the context. The output of the LSTM layer is then fed into two BiLSTM layers that follow one another. These levels evaluate the sequence both forward and backward, evaluating the context from both perspectives and generating hidden states that combine data from words that come before and after. The final step involves passing the combined hidden states from the BiLSTM layers via a dense layer, which performs the final classification after analyzing all the processed data. The most likely dialect of the input sentence is determined by the dense layer, which produces probabilities for each dialect category: Saudi, Egyptian or Levantine. Tuning the hyperparameters was a crucial aspect of building the model. A grid search was conducted to determine the ideal learning rate, batch size, number of units in the LSTM and BiLSTM layers, and dropout and recurrent dropout rates. A range of learning rates, including 0.001, 0.0005, and the Adam optimizer's default rate. A comparison of 32, 64, and 128 batch sizes was conducted to determine the optimal performance. The number of units in the LSTM and BiLSTM layers varied; nonetheless, 64, 128, and 256 units were frequently selected. To improve the model's generalization and avoid overfitting, additional dropout rates of 0.2, 0.3, 0.4, and 0.5 were applied to the LSTM and BiLSTM layers.

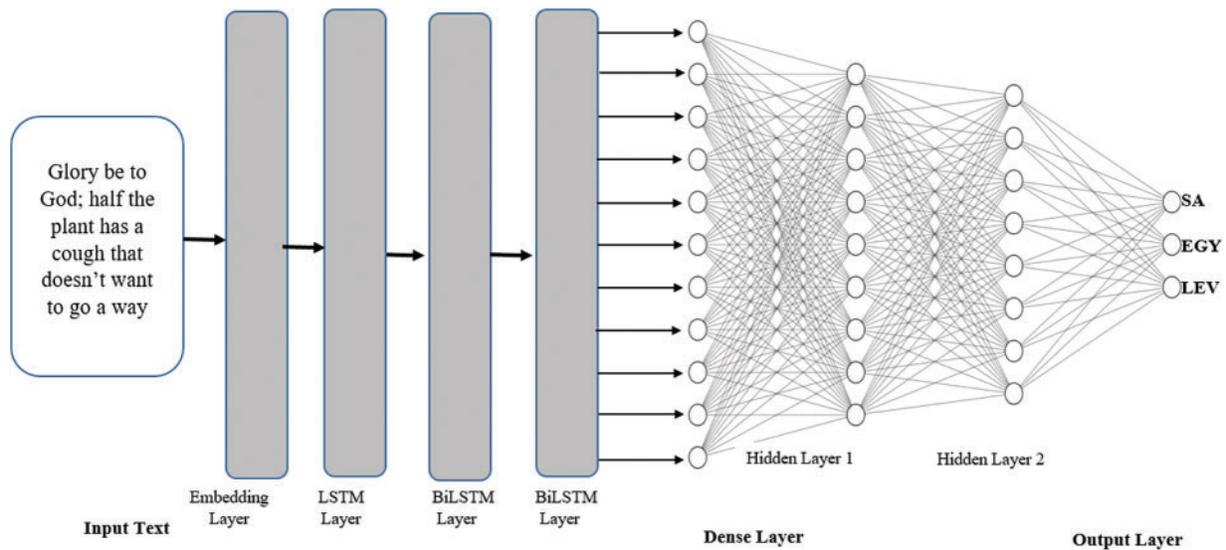


Figure 4: The proposed model

4.3 Transformer Models

Transformer models are an important part for progressing Arabic NLP; by providing tailored architectures and undergoing training on different types of datasets, their capability in functions such as sentiment analysis, text creation, and beyond [40,41]. Models like ARABERT, MARBERT, CAMELBERT, QARIB, and ARAELECTRA, known for their self-attention mechanisms and great performance at managing intricate language structures in the experiments which were thoroughly evaluated.

5 Results and Discussions

This section explains the experimental settings utilized to explore the identification of Arabic dialects through ML, DL, and transformer models.

5.1 Experimental Setup

The experiments leveraged various libraries for data handling, model building, and data visualization, which were all conducted using Python. ‘pandas’ and ‘openpyxl’ facilitated data manipulation and storage. ‘scikit-learn’ was used to implement ML models. Advanced DL models were implemented by using ‘PyTorch’ and ‘TensorFlow’. For fine-tuning and optimization, ‘Hugging Face’ was used to implement transformer models with libraries. Visualization tools and text data preprocessing were managed with standard Python libraries. The dataset was divided into training, validation, and test sets to ensure thorough model evaluation. The dataset was split into two parts, with 80% of the data used for training and the remaining 20% used as a temporary set. This temporary set was then split into validation and test sets, each comprising 10% of the total data. Consistency and reproducibility in data shuffling were performed by setting a random state of 42 for both splits. This structured data dividing method allowed for effective tuning and evaluation of the ML classifiers, DL models and transformer models at various stages of the testing process.

5.2 Model Evaluation

This section shows the results of the proposed model in this study after experimentation. Outcomes were compared at various stages of the testing process to evaluate the performance of the model. These comparisons provided a detailed understanding of how each model performed. The findings that primarily focus on the accuracy metric to assess effectiveness are comprehensively summarized in [Table 6](#). This summary facilitates the understanding of the models' capabilities in terms of how well they perform in identifying Arabic dialects across different scenarios.

Table 6: Results of model accuracy on the existing Arabic dataset

	Model	Acc.
Machine learning	LG	95.55%
	RF	95.13%
	SVM	95.74%
	DT	93.40%
	LSVM	95.46%
	BNB	93.87%
	MNB	93.63%
Deep learning	LSTM	95.74%
	BiLSTM	95.79%
	LSTM+BiLSTM	96.54%
Transformer models	ARABERT	94.85%
	ARBERT & MARBERT:	95.41%
	CAMELBERT	96.25%
	QARIB	96.54%
	ARAELECTRA	95.60%

6 Discussion

This section explains the outcomes of the three distinct experiments conducted in this study. Interpretation and discussion were offered on the basis of the comprehensive experiments and analyses. The experimentation on ML classifiers utilizing TF-IDF with N-gram demonstrated remarkable findings. Amongst the classifiers, LG and SVM stood out as the top performers, with SVM achieving an accuracy of 95.74% and LG an accuracy of 95.55%. This performance proves the effectiveness of linear models in handling high-dimensional data characteristics of text classification tasks. The accuracy metrics in [Fig. 5](#) also indicate a strong linear relationship between TF-IDF features and target classes, suggesting that these models are well equipped at capturing the smallest of differences of Arabic dialects. To identify Arabic dialects, the research explored DL techniques, particularly LSTM and BiLSTM models. This demonstrates their efficiency in managing contextual information and long-term dependencies. Various architectures were put through a series of experiments (including combinations of LSTM and BiLSTM) to optimize performance. A rigorous evaluation process, employing a validation set and utilizing the 'adam' optimizer and 'categorical_crossentropy' loss function, facilitated a precise assessment of the models' capabilities. The findings proved that slight

modifications in learning rates and the introduction of regularization considerably affected model accuracy. Notably, hybrid models incorporating LSTM and BiLSTM architectures, achieved the highest accuracy of 96.54%, when trained with FastText word embeddings. This result is further shown in Fig. 6, which details the model's precision through a confusion matrix. This finding proves the importance of model architecture, parameter optimization, and the use of advanced embedding techniques in accurately identifying Arabic dialects. Additionally, the investigation assessed the effect of pre-trained word embeddings, i.e., FastText, GloVe, and Word2Vec, on LSTM+BLSTM models for identifying Arabic dialects. The top-performing model, which utilizes FastText embeddings, achieved an impressive accuracy of 96.54%. This result highlights the remarkable enhancement in model performance attributed to FastText embeddings, proving their effectiveness in identifying Arabic dialects. Models incorporating Word2Vec embeddings demonstrated improved performance over those with GloVe, achieving an accuracy of 95.41%. Whilst enhancement was notable, these models still performed more poorly in terms of accuracy compared with the FastText-embedded models. Models utilizing GloVe embeddings exhibited lower performance compared with FastText, with the highest accuracy reaching 92.98%. Despite employing the same model architecture as the FastText model, GloVe-embedded models could not achieve the same level of accuracy, as depicted in Fig. 7. Overall, FastText embeddings remarkably enhanced model accuracy, followed by Word2Vec, whilst GloVe embeddings yielded comparatively lower performance in accurately identifying Arabic dialects. Transformer models, known for their advanced capabilities in handling complex language patterns, offered groundbreaking insights. QARiB, with its impressive accuracy of 96.54%, stood out as the most effective model for Arabic dialect identification. This model's success can be attributed to its innovative training approaches and the incorporation of dialectical variations.

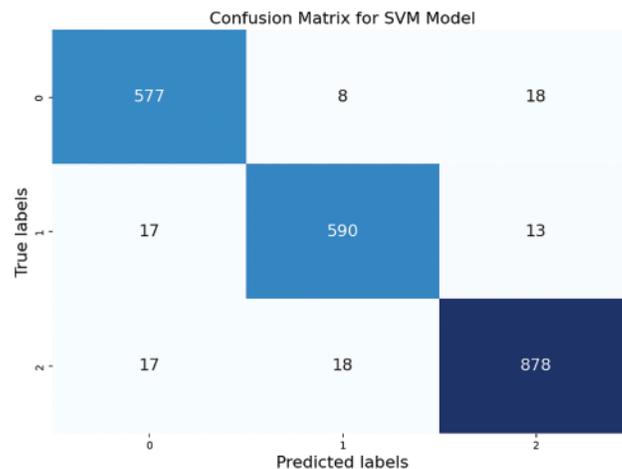


Figure 5: Confusion matrix of SVM

The detailed examination of the confusion matrix in Fig. 8 for transformer models provided a comprehensive understanding of their classification capabilities and areas for improvement. As evidenced by the data presented in Fig. 9, it's both DL approaches, specifically those utilizing LSTM and BiLSTM architectures, along with transformer-based models led by QARiB, demonstrate exceptional proficiency in accurately identifying Arabic dialects. The remarkable performance of these models emphasizes their potential to advance the field of Arabic dialect recognition. Specifically, the success of QARiB proves the transformative role of transformer architectures in navigating the

complexities of language patterns and dialect variations, marking a pivotal advancement in Arabic dialect studies.

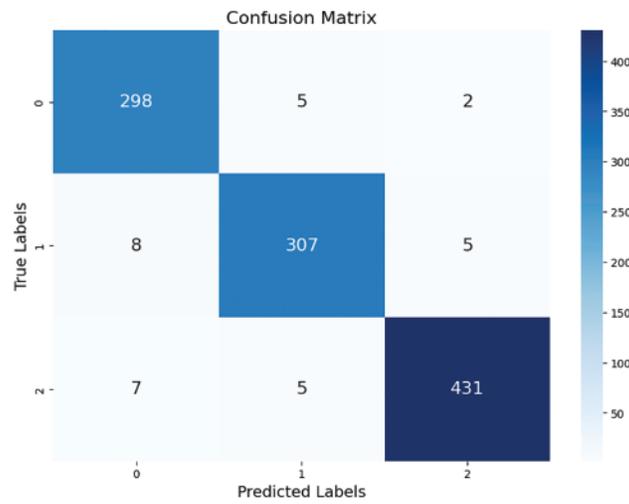


Figure 6: Confusion matrix of LSTM And BiLSTM

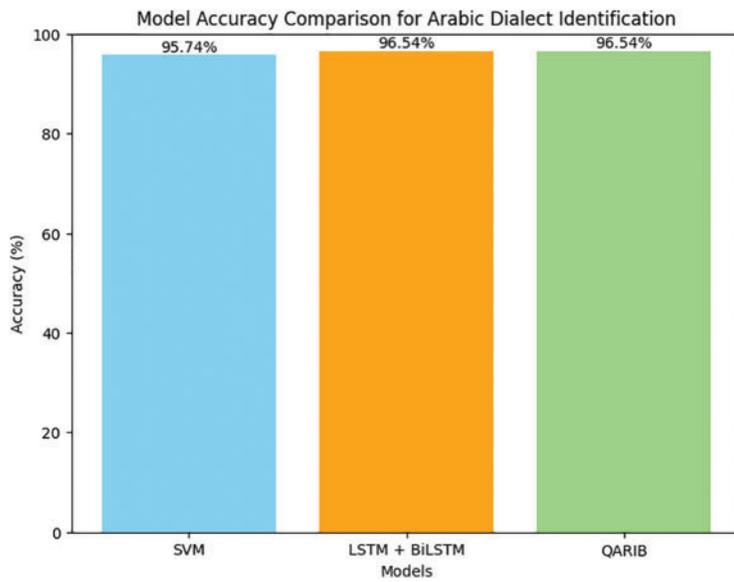


Figure 7: Results of word embeddings

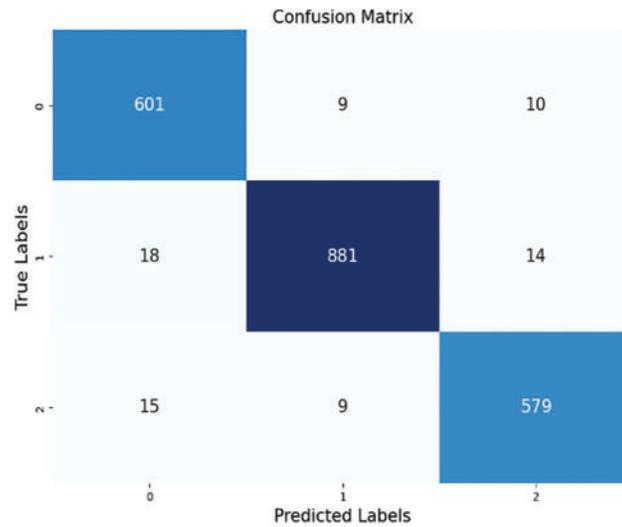


Figure 8: Confusion matrix of Qarib

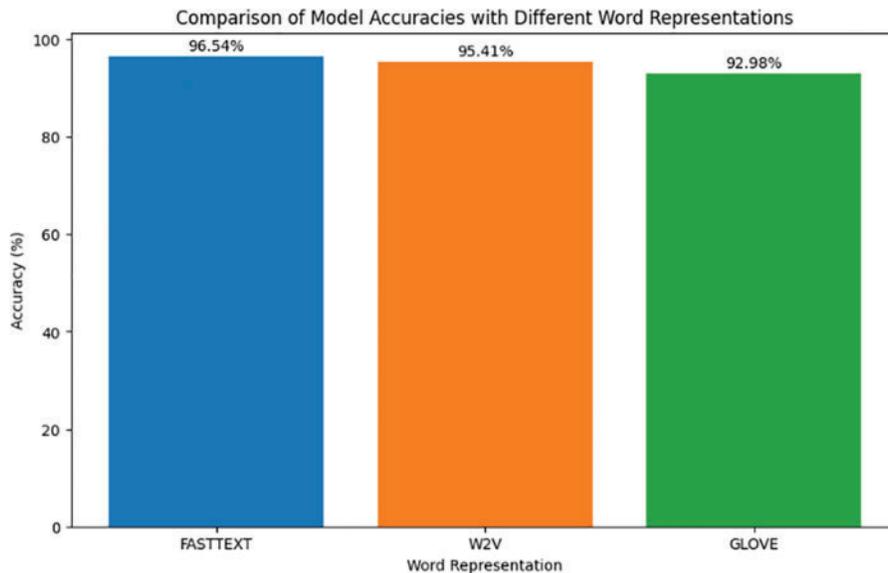


Figure 9: Comparison of models' accuracy

7 Conclusion

This study remarkably advances the field of Arabic dialect identification by leveraging DL models, and transformer-based models across various social media platforms. It highlights the efficiency of these methods in NLP applications specifically tailored to Arabic dialects. ML models such as SVM and LG achieved accuracies above 95% which proves their remarkable capabilities in text data analysis. These models captured the unique characteristics of Arabic dialects and established a robust baseline for dialect identification by effectively using TF-IDF and N-gram features. The hybrid DL model showed the importance of processing sequential data to grasp contextual language nuances.

The highest accuracy in the study was 96.54%, it was achieved by combining LSTM and BiLSTM, further proving the potential of neural networks to reflect the complex dynamics of spoken language in text. Transformer models, particularly QARiB, set new benchmarks for dialect identification with their advanced self-attention mechanisms, performing greatly at identifying subtle linguistic variations. In comparison with the results of the proposed model, no considerable difference in its impressive accuracy of 96.54%, which stood out as the most effective model for Arabic dialect identification. The dataset which was gathered from Twitter proved to be a crucial resource in training and testing, which will support ongoing and future research in this area. The practical implications of this research are extensive. Enhanced dialect identification improves sentiment analysis, opinion mining, and social media monitoring, leading to enriched insights into public opinion. In machine translation, it allows for more precise systems that account for regional language variations. Dialect-aware chatbots offer more natural and effective interactions, enhancing user experience. Improved dialect recognition also benefits educational tools by providing more tailored resources. Additionally, it enhances the ability of voice assistants to understand and respond to different Arabic dialects, making them more versatile and user-friendly. Future research could expand on the dataset to cover more dialects overall and integrate multimodal data to enhance the models' understanding and accuracy. Additionally, exploring advanced model training techniques and unsupervised learning algorithms could and can make improvements in automatic dialect recognition. Future work could also focus on developing capabilities for handling cross-dialectal understanding and code-switching which are prevalent in multilingual and dialectal communications. These directions not only promise to enhance the robustness and applicability of the models but also pave the way for their evolution in real-world applications.

Acknowledgement: The researchers would like to thank the Deanship of Graduate Studies and Scientific Research at Qassim University for financial support (QU-APC-2024-9/1) and express their gratitude for the valuable feedback and suggestions provided by all the anonymous reviewers and the editorial team.

Funding Statement: The researchers would like to thank the Deanship of Graduate Studies and Scientific Research at Qassim University for financial support (QU-APC-2024-9/1).

Author Contributions: Conceptualization, Enas Yahya Alqulaity and Wael M.S. Yafooz; Data curation, Enas Yahya Alqulaity and Wael M.S. Yafooz; Formal analysis, Enas Yahya Alqulaity and Wael M.S. Yafooz; Investigation, Wael M.S. Yafooz; Methodology, Enas Yahya Alqulaity and Wael M.S. Yafooz; Validation, Abdullah Assaedi; Visualization, Abdullah Alourani and Ayman Jaradat; Writing—original draft, Enas Yahya Alqulaity and Wael M.S. Yafooz. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The datasets can be accessed at <https://www.kaggle.com/datasets/waelshaher/arabic-tweet-dataset> (accessed on 9 June 2024)

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

- [1] T. Kanan *et al.*, “A review of natural language processing and machine learning tools used to analyze Arabic social media,” in *IEEE Int. Joi. Conf. Elec. Eng. Info. Tech.*, Amman, Jordan, May 2019, pp. 622–628. doi: [10.1109/JEEIT.2019.8717369](https://doi.org/10.1109/JEEIT.2019.8717369).
- [2] A. Ferchichi, E. Souissi, J. Younes, and H. Achour, “Contributions to the automatic processing of the user-generated Tunisian dialect on the social web,” *Int. J. Comput. Intel. Stud.*, vol. 9, no. 1/2, pp. 33–51, 2020. doi: [10.1504/IJCISTUDIES.2020.10028139](https://doi.org/10.1504/IJCISTUDIES.2020.10028139).
- [3] H. Alamoudi *et al.*, “Arabic sentiment analysis for student evaluation using machine learning and the AraBERT transformer,” *Eng. Tech. Appl. Sci. Res.*, vol. 13, no. 5, pp. 11945–11952, Oct. 2023. doi: [10.48084/etasr.6347](https://doi.org/10.48084/etasr.6347).
- [4] W. Alghibab, N. Alawwad, A. Aldawish, and S. AlHumoud, “Arabic speech recognition with deep learning: A review,” in *Social Computing and Social Media. Design, Human Behavior and Analytics*. Orlando, FL, USA: Springer Verlag, 2019, pp. 15–31. doi: [10.1007/978-3-030-21902-4_2](https://doi.org/10.1007/978-3-030-21902-4_2).
- [5] A. H. A. Alnosairee and N. W. Sartini, “A sociolinguistics study in Arabic dialects,” *Prasasti: J. Linguist.*, vol. 6, no. 1, pp. 1–17, 2020. doi: [10.20961/prasasti.v6i1.43127](https://doi.org/10.20961/prasasti.v6i1.43127).
- [6] A. Elnagar, S. M. Yagi, A. B. Nassif, I. Shahin, and S. A. Salloum, “Systematic literature review of dialectal Arabic: Identification and detection,” *IEEE Access*, vol. 9, pp. 31010–31042, 2021. doi: [10.1109/ACCESS.2021.3059504](https://doi.org/10.1109/ACCESS.2021.3059504).
- [7] E. A. Abozinadah and J. H. Jones Jr, “Improved micro-blog classification for detecting abusive Arabic twitter accounts,” *Int. J. Data Min. Knowl. Manage. Process.*, vol. 6, no. 6, pp. 17–28, Nov. 2016. doi: [10.5121/ijdkp.2016.6602](https://doi.org/10.5121/ijdkp.2016.6602).
- [8] F. Sadat, F. Kazemi, and A. Farzindar, “Automatic identification of Arabic language varieties and dialects in social media,” in *Proc. Second Workshop Nat. Lang. Proc. Soc. Media (SocialNLP)*, Dublin, Ireland, 2014, pp. 22–27.
- [9] O. F. Zaidan and C. Callison-Burch, “Arabic dialect identification,” *Computational Linguist. J.*, vol. 40, no. 1, 2014. doi: [10.1162/COLI_a_00169](https://doi.org/10.1162/COLI_a_00169).
- [10] A. Omar and M. Aldawsari, “Lexical ambiguity in Arabic information retrieval: The case of six web-based search engines,” *Int. J. Engl. Linguist.*, vol. 10, no. 3, Apr. 2020, Art. no. 219. doi: [10.5539/ijel.v10n3p219](https://doi.org/10.5539/ijel.v10n3p219).
- [11] M. J. Althobaiti, “Automatic Arabic dialect identification systems for written texts: A survey,” 2020. doi: [10.48550/arXiv.2009.12622](https://doi.org/10.48550/arXiv.2009.12622).
- [12] B. Singh, R. Desai, H. Ashar, P. Tank, and N. Katre, “A trade-off between ML and DL techniques in natural language processing,” *J. Phys.: Conf. Ser.*, vol. 1831, pp. 1–11, Mar. 2021. doi: [10.1088/1742-6596/1831/1/012025](https://doi.org/10.1088/1742-6596/1831/1/012025).
- [13] J. Grieve, C. Montgomery, A. Nini, A. Murakami, and D. Guo, “Mapping lexical dialect variation in British English using Twitter,” *Front. Artif. Intell.*, vol. 2, Jul. 2019, Art. no. 11. doi: [10.3389/frai.2019.00011](https://doi.org/10.3389/frai.2019.00011).
- [14] K. A. Kwaik, M. Saad, S. Chatzikiyriakidis, and S. Dobnik, “Shami: A corpus of levantine Arabic dialects,” in *Proc. Elev. Int. Conf. Lang. Reso. Eval. (LREC 2018)*, 2018.
- [15] O. Obeid, M. Salameh, H. Bouamor, and N. Habash, “ADIDA: Automatic dialect identification for Arabic,” in *Conf. Nor. Amer. Chap. Assoc. Comput. Ling.*, Minneapolis, MN, USA, 2019, pp. 6–11.
- [16] D. Ghoual and G. Lejeune, “MICHAEL: Mining character-level patterns for Arabic dialect identification (MADAR Challenge),” in *Proc. Fourth Arab. Nat. Lang. Process. Workshop*, Florence, Italy, 2019, pp. 229–233.
- [17] M. Abdul-Mageed, C. Zhang, A. Elmadany, H. Bouamor, and N. Habash, “NADI 2022: The third nuanced Arabic dialect identification shared task,” presented at the Proc. 2022 Int. Conf. Comput. Ling. and Nat. Lang. Proces. Oct. 2022. Accessed: Aug. 1, 2024. [Online]. Available: <http://arxiv.org/abs/2210.09582>
- [18] M. Abdul-Mageed, C. Zhang, A. Elmadany, H. Bouamor, and N. Habash, “NADI 2021: The second nuanced Arabic dialect identification shared task,” 2021. Accessed: Aug. 1, 2024. [Online]. Available: <https://codalab.org/>
- [19] M. S. Ali, A. H. Ali, A. A. El-Sawy, and H. A. Nayel, “Machine learning-based approach for Arabic dialect identification,” in *Proc. Six. Ara. Nat. Lang. Proc. Work*, Kyiv, Ukraine, Apr. 2021, pp. 287–290.

- [20] T. Jauhiainen, H. Jauhiainen, and K. Lindén, “Optimizing Naive Bayes for Arabic dialect identification,” in *Arabic Nat. Lang. Proc. Workshop, The Associat. Comput. Linguistics*, Abu Dhabi, The United Arab Emirates, Dec. 2022, pp. 409–414.
- [21] M. El Chrif *et al.*, “Investigate the impact of stemming on mauritanian dialect classification using machine learning techniques,” *Int. J. Adv. Comp. Sci. App.* vol. 14, no. 10, pp. 1013–1019, 2023.
- [22] A. Salau, N. A. Nwojo, M. M. Boukar, and O. Usen, “Advancing preauthorization task in healthcare: An application of deep active incremental learning for medical text classification,” *Eng. Technol. Appl. Sci. Res.*, vol. 13, no. 6, pp. 12205–12210, Dec. 2023. doi: [10.48084/etasr.6332](https://doi.org/10.48084/etasr.6332).
- [23] B. Talafha *et al.*, “Multi-dialect Arabic BERT for country-level dialect identification,” 2020, *arXiv:2007.05612*.
- [24] G. Bayrak and A. M. Issifu, “Domain-adapted BERT-based models for nuanced Arabic dialect identification and tweet sentiment analysis,” in *Proc. Sev. Ara. Nat. Lang. Proces. Work. (WANLP)*, pp. 425–430, 2022. Accessed: Aug. 1, 2024. [Online]. Available: <https://paperswithcode.com/sota/sentiment-analysis-on->
- [25] L. Lulu and A. Elnagar, “Automatic Arabic dialect classification using deep learning models,” in *Procedia Computer Science*. Dubai, The United Arab Emirates: Elsevier B.V, 2018, pp. 262–269. doi: [10.1016/j.procs.2018.10.489](https://doi.org/10.1016/j.procs.2018.10.489).
- [26] M. Ali, “Character level convolutional neural network for Arabic dialect identification,” in *Proc. Fif. Work. NLP Sim. Lang. Var. Dia., VarDial 2018*, pp. 122–127, Aug. 2018. Accessed: Aug. 1, 2024. [Online]. Available: <https://github.com/bigoooh/adi>
- [27] Y. Fares, Z. E. -Z. Kareem, A. -S. Muhammed, E. Aliaa, M. K. El-Awaad and M. Torki, “Arabic dialect identification with deep learning and hybrid frequency based features,” in *Proc. Fourth Arab. Nat. Lang. Process. Workshop, Associat. Computational Linguistics*, Florence, Italy, Aug. 2019, pp. 224–228. doi: [10.18653/v1/W19-4626](https://doi.org/10.18653/v1/W19-4626).
- [28] J. Younes, H. Achour, E. Souissi, and A. Ferchichi, “A deep learning approach for the romanized tunisian dialect identification,” *Int. Arab J. Inform. Technol.*, vol. 17, no. 6, pp. 935–946, Nov. 2020. doi: [10.34028/iajit/17/6/12](https://doi.org/10.34028/iajit/17/6/12).
- [29] G. De Francony, V. Guichard, P. Joshi, and H. Afli, “Hierarchical deep learning for Arabic dialect identification,” in *Proc. Fourth Arab. Nat. Lang. Process. Workshop, Associat. Computational Linguistics*, Florence, Italy, Aug. 2019, pp. 249–253. doi: [10.18653/v1/W19-4631](https://doi.org/10.18653/v1/W19-4631).
- [30] P. Mishra and V. Mujadia, “Arabic dialect identification for travel and twitter text,” in *Proc. Fourth Arabic Nat. Lang. Process. Workshop*, Florence, Italy, 2019, pp. 234–238. doi: [10.18653/v1/W19-4628](https://doi.org/10.18653/v1/W19-4628).
- [31] M. J. Althobaiti, “Country-level Arabic dialect identification using small datasets with integrated machine learning techniques and deep learning models,” in *Proc. Sixth Arab. Nat. Lang. Process. Workshop, Association for Computational Linguistics*, 2021.
- [32] M. Abdul-Mageed, C. Zhang, H. Bouamor, and N. Habash, “NADI 2020: The first nuanced Arabic dialect identification shared task,” Oct. 2020. doi: [10.48550/arXiv.2010.11334](https://doi.org/10.48550/arXiv.2010.11334).
- [33] Y. A. Hatekar and M. S. Abdo, “IUNADI at NADI 2023 shared task: Country-level Arabic dialect classification in tweets for the shared task NADI 2023,” in *Proc. ArabicNLP 2023*, 2023.
- [34] M. R. R. Rana, A. Nawaz, T. Ali, A. M. El-Sherbeeney, and W. Ali, “A BiLSTM-CF and BiGRU-based deep sentiment analysis model to explore customer reviews for effective recommendations,” *Eng. Technol. Appl. Sci. Res.*, vol. 13, no. 5, pp. 11739–11746, Oct. 01, 2023. doi: [10.48084/etasr.6278](https://doi.org/10.48084/etasr.6278).
- [35] W. M. Yafooz, A. H. M. Emar, and M. Lahby, “Detecting fake news on COVID-19 vaccine from YouTube videos using advanced machine learning approaches,” *Combating Fake News Comput. Intell. Tech.*, vol. 1001, pp. 421–435, 2022. doi: [10.1007/978-3-030-90087-8](https://doi.org/10.1007/978-3-030-90087-8).
- [36] D. Bouchiha, A. Bouziane, and N. Doumi, “Machine learning for Arabic text classification: A comparative study,” *Malaysian J. Sci. Adv. Technol.*, pp. 163–173, Oct. 2022. doi: [10.56532/mjsat.v2i4.83](https://doi.org/10.56532/mjsat.v2i4.83).
- [37] H. Elfaik and E. H. Nfaoui, “Deep bidirectional LSTM network learning-based sentiment analysis for Arabic text,” *J. Intell. Syst.*, vol. 30, no. 1, pp. 395–412, Jan. 2021. doi: [10.1515/jisys-2020-0021](https://doi.org/10.1515/jisys-2020-0021).

- [38] R. L. Abduljabbar, H. Dia, and P. W. Tsai, "Unidirectional and bidirectional LSTM models for short-term traffic prediction," *J. Adv. Transp.*, vol. 2021, pp. 1–16, 2021. doi: [10.1155/2021/5589075](https://doi.org/10.1155/2021/5589075).
- [39] W. Yafooz and A. Alsaeedi, "Leveraging user-generated comments and fused BiLSTM models to detect and predict issues with mobile apps," *Comput. Mater. Contin.*, vol. 79, no. 1, pp. 735–759, 2024. doi: [10.32604/cmc.2024.048270](https://doi.org/10.32604/cmc.2024.048270).
- [40] I. El Karfi and S. El Fkihi, "An ensemble of Arabic transformer-based models for Arabic sentiment analysis," *Int. J. Adv. Comp. Sci. App.*, vol. 13, no. 8, pp. 561–567, 2022.
- [41] W. M. Yafooz, A. Al-Dhaqm, and A. Alsaeedi, "Detecting kids cyberbullying using transfer learning approach: Transformer fine-tuning models," in *Kids Cybersecurity Using Computational Intelligence Techniques*. Cham: Springer International Publishing, 2023, pp. 255–267.