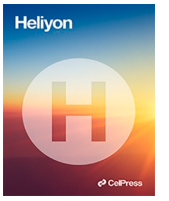
Heliyon 10 (2024) e36280

Contents lists available at ScienceDirect

Heliyon

journal homepage: www.cell.com/heliyon



Research article

Arabic dialect identification in social media: A hybrid model with transformer models and BiLSTM

Amjad A. Alsuwaylimi

Department of Information Technology, Faculty of Computing and Information Technology, Northern Border University, Rafha, 91911, Saudi Arabia

ARTICLE INFO

Keywords:
Transformers
Arabic dialect Identification
Machine learning
Deep learning

ABSTRACT

Arabic Dialect Identification (ADI) is a challenging task in natural language processing applications due to its diversity and regional variations. Despite previous efforts, this task is still difficult. Therefore, this study aims to use transformers to address the issue of ADI on social media. A combination of two hybrid models is proposed in this study: one that combines Bidirectional Long Short-Term Memory (BiLSTM) with CAMeLBERT, and the second model that combines the BiLSTM model with AlBERT. In addition, a novel dataset comprising 121,289 user-generated comments from various social media network platforms and four major Arabic dialects (Egyptian, Jordanian, Gulf and Yemeni) was introduced. Several experiments have been conducted using conventional Machine Learning Classifiers (MLCs) and Deep Learning Models (DLMs) as baselines to measure the performance and effectiveness of the proposed models. In addition, binary classification is performed between two dialects to determine which are closest to each other. The performance of the model is measured using common metrics such as precision, recall, F-score and F-measure. Experiment results demonstrate the superior efficiency of the proposed hybrid models in ADI, CAMeLBERT with BiLSTM and ALBERT with BiLSTM, which both recorded an accuracy of 87.67 % and 86.51 %, respectively.

1. Introduction

With the enormous number of Arabic speakers using the Internet in recent years, the Arabic Dialect Identification (ADI) task has gained considerable attention [1–3]. This task belongs to Natural Language Processing (NLP). A substantial increase in social media users, driven in part by the extensive use of the Internet and the popularity of smartphones, can be attributed to the widespread adoption of technologies and social media. Social media platforms, such as Facebook and YouTube, have become one of the main sources of communication between users on the Internet. In the Arab world, three main types of communication exist: Classical Arabic Language (CAL), Modern Standard Arabic (MSA) and Arabic dialects [4–6]. CAL is a language of early literature. MSA, which is used in TV, official news, magazines and formal letters, is understood by nearly all Arabic speakers and is primarily used in writing and formal contexts. Meanwhile, Arabic dialects, or slang, are commonly used in everyday interactions amongst the public and are used on social media networks.

ADI can be challenging due to the remarkable linguistic variations across different regions. Moreover, ADI varies by country and differs from one city to another, despite belonging to the same country. Accurately classifying or identifying Arabic dialects is difficult for automated systems due to their diversity in vocabulary and pronunciation. Approximately 22 Arab countries exist, and these countries all speak in different types of dialects [7]. The different types of Arabic can be divided into the six predominant dialects:

E-mail address: amjad.alsuwaylimi@nbu.edu.sa.

https://doi.org/10.1016/j.heliyon.2024.e36280
Received 1 May 2024; Received in revised form 10 August 2024; Accepted 13 August 2024
Available online 13 August 2024

Maghreb (Morocco, Algeria, Tunisia, Libya and Mauritania), Levantine (Syria, Lebanon, Jordan and Palestine), Peninsular (Saudi Arabia, United Arab Emirates, Qatar, Bahrain, Kuwait and Yemen), Egyptian, Sudanese and Iraqi. Speakers of the spoken regional dialects of Arabic (the real 'native languages' of Arabic) substantially differ from MSA.

The Arabic language possesses unique characteristics compared with other languages. Thus, automatic processing of Arabic text is challenging. One of the main challenges lies in the manner of writing, given that Arabic is written from right to left, which is different from the left-to-right writing direction of most other languages [8]. This fundamental difference requires specific handling in text processing algorithms to ensure the correct interpretation and processing of text. Another challenge is that the Arabic script itself is non-Roman. Different from languages that use the Roman alphabet, Arabic has its own set of characters and rules of writing. Thus, specialised tools and algorithms are required for tasks such as character recognition, word segmentation and text normalisation. Additionally, Arabic has rich morphology and complex grammar, increasing its complexity for NLP tasks. For example, Arabic words can have multiple forms based on their grammatical context, and verbs can have various conjugations and forms.

Arab social media users communicate in different dialects. Thus, understanding them can be difficult for a large audience and the entire Arab world. For example, a person from Syria speaking the Levantine dialect of Arabic might encounter difficulties in attempting to understand a person from Morocco speaking the Maghreb dialect of Arabic. ADI is a common issue tackled by researchers using different approaches. Scholars have attempted to identify Arabic dialects using various NLP methods [9–14], Machine Learning Classifiers (MLCs) [15–23] and highly accurate Deep Learning Models (DLMs) [24–30]. Transformer-based pre-training models such as BERT [31–38] and GPT [39,40] have also been recently utilised. In addition, some datasets, such as NADI [41,42], MADAR [43], QADI [44] and ADI17 [45], have been proposed to perform ADI. Moreover, various word representations, including bags of words, TF-IDF and embeddings (such as word2vec, GloVe and fastText), have been employed to enhance the performance of the models.

These approaches have demonstrated some success. However, these existing approaches often encounter several challenges. MLCs may struggle with the complex morphology and syntax of Arabic dialects. Meanwhile, despite their power, DLMs suffer from issues that require a high computational cost, present difficulty in generalisation across different Arabic dialects and affect the performance of the model in terms of accuracy. Thus, ADI remains a major challenge due to the complexity and variability of the dialect. Arabic dialects considerably vary across different regions, making mutual understanding difficult even amongst native speakers. The research questions in this study are focused on improving the identification of Arabic dialects using advanced NLP techniques. This study helps improve technologies such as translation and speech recognition, increasing their accuracy for different dialects. The results of this study also contribute to the extensive goals of NLP research and practical applications in multilingual and multicultural contexts in the Arab world. Overall, improved dialect identification supports business, education and diplomacy.

Therefore, this study proposes to combine two hybrid models: one is CAMeLBERT with BiLSTM, and the other is ALBERT with BiLSTM. The combination of these models was utilised to enhance ADI performance in terms of accuracy. The knowledge of the pertained models of CAMeLBERT and ALBERT is utilised because these models are well trained on Arabic text. Meanwhile, BiLSTM is used because it can capture the data sequences of user comments in the forward and backward directions. In addition, a new dataset, which comprises of 198,233 user-generated comments in the initial stage from various social media platforms, after the filtering process which includes the removal of noise and the removal of duplicate user comments, the amount of user-generated comments decreased to 121,289. Furthermore, the dataset of the user-generated comments underwent a final filtering procedure of our annotation process by three Arabic native speakers which then reduced the total number of user-generated comments to 38,394.

This dataset covers four major Arabic dialects: Egyptian, Jordanian, Gulf and Yemeni. Some NLP methods, such as pre-processing and annotation steps, are applied. Several experiments were conducted using the proposed dataset to assess the performance of the proposed hybrid models. In these experiments, the MLCs and DLMs were used as baselines. Additionally, several experiments were performed using MLCs to determine which Arabic dialects are close to each other. The experimental results show that the proposed models outperform conventional MLCs, LSTM and BiLSTM models in terms of accuracy. CAMeLBERT with BiLSTM and ALBERT with BiLSTM achieve 87.67 % and 86.51 % accuracy, respectively. The contributions of this study are listed as follows.

- A novel Arabic dataset, which comprises 38,394 user comments from four different dialects (Egyptian, Jordanian, Gulf and Yemeni), is introduced. These comments were collected from social media.
- Two models using BiLSTM with CAMeLBERT and BiLSTM with ALBERT are proposed for ADI.
- Traditional MLCs are used for comparison using the new dataset to assess their effectiveness.
- Arabic dialects that are most similar to each other are identified.

The remaining part of this paper is structured as follows: Section 2 describes the related studies, methods, and techniques used in detecting and classifying Arabic dialects. In Section 3, it describes the methods of the proposed models for ADI. Section 4 explains the results and discussion of the experiments conducted to evaluate the proposed models. Finally, Section 5 concludes this paper, summarizing the key findings and contributions, and suggesting directions for future research in the field of Arabic dialect identification.

2. Related studies

ADI is emerging as a new research area. In previous studies, researchers have utilised various methods, techniques, and approaches to distinguish between Arabic dialects. These approaches include conventional MLCs, DLMs and transformer-based methods, which demonstrate the breadth of the strategies used to tackle this challenge.

In the conventional methods with MLCs, the pre-processing steps include feature extraction, feature selection and noise reduction. These methods use classifiers such RF, DT, SVM and KNN. Aliwy et al. [16] used DT, LR and NB along with two other techniques and an

integrated voting mechanism for categorisation on the NADI dataset to classify tweets from 21 Arabic-speaking countries after removing noise. The investigation yields positive results with f-measure values of 27.17 %, 41.34 % and 52.38 % for different approaches. The classification accuracy is increased by minimizing noise from MSA tweets in the dataset through the use of clustering techniques. Similarly, Sobhy et al. [17] concentrated on the word representation methods TF-IDF and Word2Vec using the 2022 NADI dataset. They examined several MLCs, including SVM, KNN, DT, RF, MNB, CNB, RF and MLP. The experiments show that Word Embeddings outperform TF-IDF. The accuracy of MLP (30 H) with Word Embeddings is 38.63 %; CNB with TF-IDF comes in second with 39.05 %, while MLP (20 H) + Word Embeddings comes in third with 38.97 %. Conversely, using the Word2Vec model leads to lower accuracy levels. In the same way, Nahar et al. [18] highlighted the importance of identifying dialects and accents in speech recognition, which can provide useful information about the nationality and cultural background of the speaker. They focused on Arabic dialects using the (ADI7) dataset. They proposed techniques using previously extracted features as input, such as KNN, RF, MLP and ANN. The experimental results based on Mel-Frequency Cepstral Coefficient features using ANN record a best accuracy of 62 %.

DLMs have demonstrated high accuracy in model performance. The most common architectures are RNN, LSTM, CNN-LSTM, BiLSTM and GriGate. Lulu et al. [25] investigated the use of CNN and LSTM DLMs for automatic Arabic dialect classification. Using the Arabic Online Commentary dataset, they concentrated on categorising dialects of Arabic that are spoken in the Gulf, Levantine and Egypt. For most binary classification tasks, the accuracy is usually between 70 % and 80 %. In the case of 3-way multi-class classification on these dialects, the accuracy is approximately 71 %. Using the most recent AOC dataset with 110,000 sentences, 30,000 sentences are allocated for each of the three main dialects (EGP, GLF and LEV). LSTM performs best overall. However, analysis of the dataset shows some conflicting annotations, particularly between Gulf and Levantine dialects. The difficulties in manual annotation and the overlap in dialect vocabulary are major challenges. Using AOC, El Araby et al. [26] found that the DLMs obtain an accuracy of 84.41 % on blinded test data. Among these models, the attention-based bidirectional recurrent neural network variations outperform all competing baselines by a wide margin. While the conventional models including SVM, LR and MNB are employed, traditional classifiers outperform the majority of DLMs in 3-way classification when the size of the training dataset is decreased by excluding the MSA comments. Similarly, in 4-way classification, the outcomes decrease when the data are sparsely distributed throughout the four categories.

Elnagar et al. [27] presented two brand-new, sizable corpora for Arabic text classification: NADiA (480,000 articles) for multi-label classification and SANAD (200,000 articles) for single-label classification. Prior studies on the categorisation of Arabic text have utilised tiny datasets and traditional MLCs, such as SVM and NB, which need extensive preprocessing. This study eliminates the preprocessing step by using DLMs such as CNN, RNN and attention networks. On the SANAD corpus, the greatest single-label accuracy obtained with HANGRU is 95.81 %. Using HANGRU, the best multi-label accuracy on the NADiA dataset is 88.68 %. They also investigated the effects of Word Embeddings, which demonstrates enhanced performance. Using an attention layer, Elaraby et al. [26] applied several attention-based bidirectional recurrent neural networks on the AOC dataset. These networks are effective in reliably identifying Arabic dialects in MSA vs. dialects. Their performance was compared with those of standard MLCs. The attention-based BiLSTM model surpasses all with a slightly higher accuracy on the test set, but it still achieves the best accuracy of 87.81 % on the development set, according to the data for binary classification. Similarly, Elnagar et al. [27] used attention with DLMs such as CNN and RNN and proposed two datasets, namely, NADiA (480,000 articles) and SANAD corpus. The greatest accuracy of 96.94 % is achieved by attention–GRU.

Abdelali et al. [2] offered a special method for generating the highly accurate dialectal dataset of Arabic obtained from Twitter, which is known as the QCRI Arabic Dialects Identification (QADI) dataset. The dataset consists of about 540,000 tweets from 2525 Twitter accounts across 18 Arab countries. Methodologies for machine learning and transfer learning were covered in the study. SVMs and modified transformer models were the instruments utilised in the classification. At the national level, 91.5 % of the dialectal tags of the dataset are correct. The study produces a macro-averaged F1-score of 60.6 % over 18 separate classes using AraBERT on the QADI dataset. These results outperform those from a publicly accessible dataset and prove the usefulness of the dataset. Alsemaree et al. [46] and Baniata & Kang [47] concentrated on the sentiment analysis based on Arabic dialect. Alsemaree et al. [46] proposed the LSAnArTe framework, initially for sentiment evaluation of Arabic text specializing in customer perceptions concerning coffee products on Twitter. It depends on lexicon-based mostly sentiment analysis to grasp shopper perception toward coffee products in Arabic tweets from social media notably Twitter. Similarly focusing on sentimental analysis, Baniata & Kang [47]., proposed a switch-transformer sentiment analysis (ST-SA) model for sentiment analysis of Arabic dialects. The model used multi-task learning with cross-entropy loss, and a multi-headed attention mechanism and a Mixture of Experts Mechanism to improve textual sequences' representations. Three different datasets (HARD, BRAD, LABR) were collected from multiple resources and rated in five scales. The model achieved an accuracy rate of 83.91 % which is better than SVM, MNP,and AraBERT.

Transformers have given considerable attention in recent studies due to their performance in classification. Based on the BERT model, Talafha et al. [32] proposed a multi-dialect-Arabic-BERT model using 10 million tweets from the NADI competition organisers. They pre-trained a BERT model, which was then refined using the labelled NADI dataset. Their strategy is effective, as evidenced by the micro-averaged F1-score of 26.78 % obtained from the ensemble of the best performing iterations. Using the NADI dataset, AlShenaifi et al. [33} used a pre-trained AraBert model with fine-tuning and BilSTM. Both models outperform conventional machine learning techniques. The BiLSTM model and AraBert achieve notable accuracy by reaching 68.7 % on the development set. Abdul-mageed et al. [34] demonstrated the results of the third NADI 2020. The participating teams employed various techniques based on BERT, RNN, CNN and traditional machine learning. The databases used for the study contained data on 100 provinces located throughout 21 Arab countries. The best average macro-F1 score of 27.06 is recorded using MARBERT, AraBERT and AraGPT2. Based on BERT and LSTM architecture, Mohammed et al. [35] suggested a three-phase neural network model for the detection of Arabic dialects. To filter out terms that are shared by different dialects, the model firstly gathers dialect-relevant data and then models dialectal vector

representations using those data. In this way, less noisy data will be transmitted to the fully linked layer. Two versions are presented: one utilising BERT contextual representations and the other LSTM-based and Transformer-based architecture. Several Arabic dialect datasets, such as MADAR, NADI and QADI, were used to assess the model. The AraBERT transformer obtains the best accuracy of 98.37 % on the MADAR2 dataset.

Humayun et al., 2022 [48]., presented a comprehensive analysis based on transformer fine-tuning for Arabic text dialect identification task. The authors used NADI 2021 dataset for country level dialect identification among Arabic. The reported accuracy of 53.96 % shows the proof of concept for this approach and stresses on not just hyper-parameter tuning but also factoring in what information to keep when adapting to specific tasks and improving performance on text classification tasks. In the same way focusing on analyzing Arabic dialects, Al-Deaibes et al., 2021 [49]., investigated how gender and dialect of speakers affect the production of emphasis on the labio-velar/w/in Urban and Rural Jordanian Arabic. Based on a dataset of 24 native speakers producing instances of/w/with an emphasis contrast, acoustic correlates such as F1, F2, F3, and vowel duration are measured. The results show that there are differences in the production of emphasis by male and female speakers as well as urban vs. rural dialects. Stress affects mainly consonantal realizations.

3. Problem formulation

ADI is considered as a multiclass problem**.** The objective of this study is to improve ADI performance by combining BiLSTM with transformer-based models (CAMeLBERT and ALBERT). The problem of ADI can be formulated as a classification task.

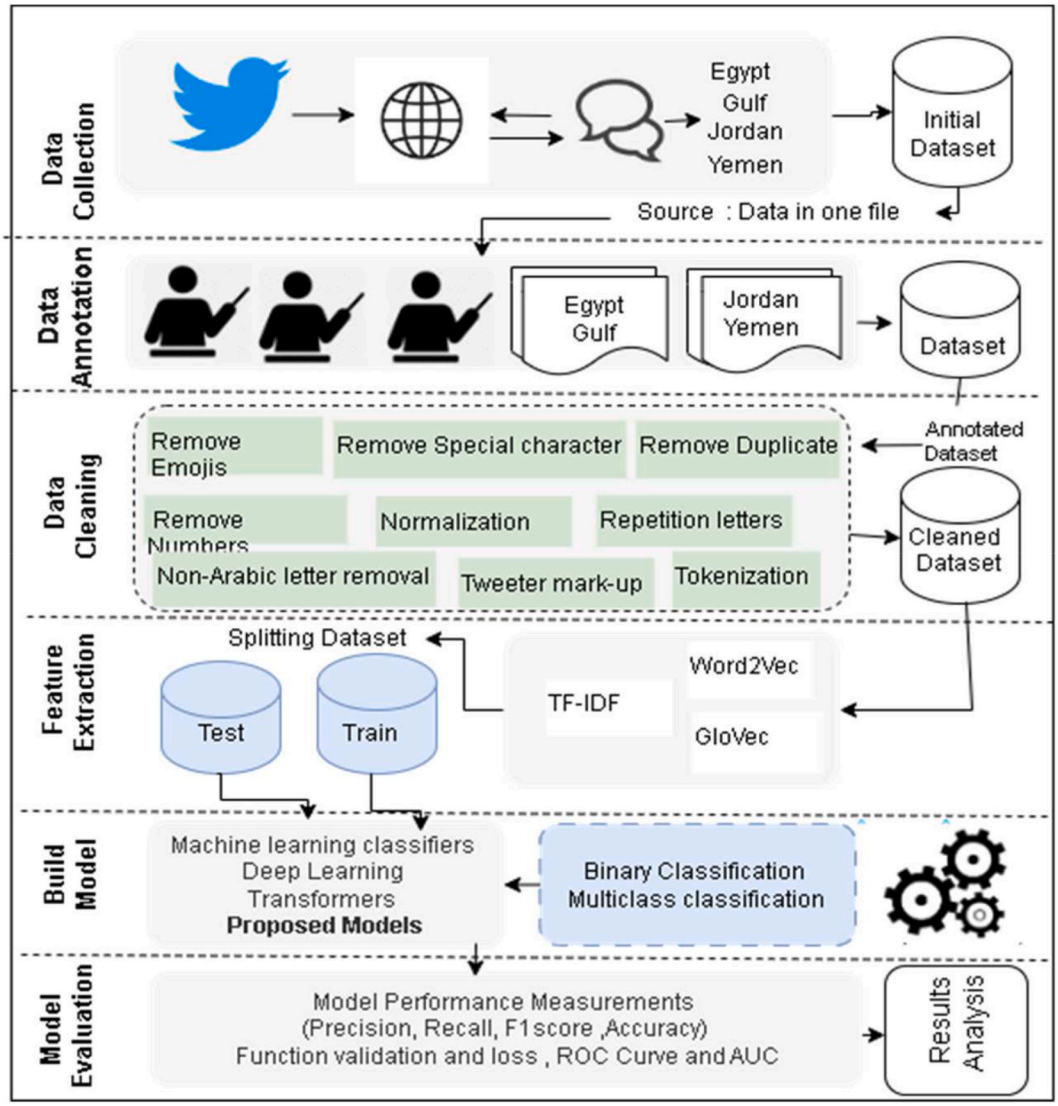


Fig. 1. Phases and methods of this research.

Let's denote the dataset as $= \{(x^{(i)}, y^{(i)})\}$, of user-generated comments, $x^{(i)}$ is the input of Arabic user comments and $y^{(i)}$ is the corresponding dialect label, which can be one of the four dialects: Egyptian, Jordanian, Gulf, or Yemeni. The goal is to learn a function f that maps each input comment x to its corresponding dialect label y. The function f is typically implemented as a neural network model, such as a BiLSTM with CAMeLBERT or BiLSTM with ALBERT, which takes the input comment x and outputs a probability distribution over the dialect labels. For a given input comment x, the function f assigns it to a dialect label $\hat{y}$ based on the model's prediction. Mathematically, this can be represented as in mathematical formula (1):

$$\hat{y} = f(x) \tag{1}$$

The function f is learned by training the model on the dataset D to minimize a certain loss function L, which measures the difference between the predicted label $\hat{y}$ and the true label y. The overall objective is to find the parameters of the model that minimize the expected value of the loss function over the dataset D:

$min_\theta \frac{1}{N} \sum_{1=0}^{N} L\left(f(x^i;\ \theta, y^i\right)$, where θ represents the parameters of the model and N number of the comments in the D.

Once the model is trained, it can be used to predict the dialect label for new comments by computing the output probabilities and selecting the label with the highest probability. Algorithm 1 shows the steps of the proposed model to Identify the Arabic Dialect into Four Classes.

Algorithm 1. Identify Arabic Dialect into Four Classes

1. **Input:** User generated comments D represented in dataset X and classes represented in y
2. **Output:** Each x belong to y
3. $X_i \rightarrow$tokenize(x_i)
4. Embedding representation $x_i \rightarrow$Model(x_i)
5. Model → $BiLSTM_{CAMeLBERT}$/$BiSLTM_{ALBERT}$
5. Initialization Step
$BiLSTM_{CAMeLBERT}$
BiLSTMALBERT
6. Training Process
$\hat{y} = BiLSTM_{CAMeLBERT}(BERT(x^i)$
$\hat{y} = BiLSTM_{ALBERT}(BERT(x^i)$
7. Compute the loss using a suitable loss function L
8. Model evaluation using the confusion matrix (Accuracy, precision, recall, and F1

4. Methods and materials

This section provide a detailed description of the methods and tools used in each phase of the study for detecting and classifying Arabic dialects. The study follows five phases: data collection, data annotation, data cleaning, model building, and model evaluation as shown in Fig. 1.

4.1. Data collection

The data was gathered from user-generated comments on social media, specifically Twitter, using Python utilising "ntscraper "library. Comments were selected based on criteria related to four dialects: Egyptian, Jordanian, Gulf, and Yemeni, identified through general conversations on Twitter. The data includes comments from users of these nationalities. Approximately 198,233 user comments were downloaded into separate CSV files, covering the period from July 2023 to January 2024. Keywords or common words representing each dialect were selected based on input from individuals from the respective countries as shown in Tables 1 and 2. For

Table 1
Sample of Egyptian and Jordanian dialect.

Egyptian dialect	Pronunciation in English	Translated to English	Jordanian dialect	Pronunciation in English	Translated to English
مشوار	Meshwar	Outing	على راسي	Ala rasi	On my head
معلش	Ma'alesh	Never mind	شو بدك؟	Shu biddak?	What do you want?
مافيش	Mafeesh	There isn't	شو صار؟	Shu sar?	What happened?
حاجة	Haga	thin	خلصنا	Khlasna	We're done
جامد	Gamid	Cool	على قلبي	Ala albi	On my heart
زي	Zay	Like	منيح	Mneeh	Good
عامل ايه؟	Aamal eh?	How are you?	بعدك هون؟	Ba'dak hawn?	Are you still here
بتاع	Beta'	Belonging to	بدك شي؟	Biddak shi?	Do you want anything
كده	Keda	This way	ايمتى بترجع؟	Emta btrja'?	When are you coming back?
أزيك	Ezayek	How are You ?	يازلمة	ya zalameh	oh friend
عايز	3ayez	I want	ما بعرف	ma ba3rif	I don't know

example, Table 1 displays Egyptian and Jordanian words or phrases, while Table 2 presents the Gulf and Yemeni words used to extract user comments. Following this, several pre-processing steps were applied to the data.

After filtering the data to remove duplicate entries, HTML tags, English comments, and other noise, the total number of user comments was reduced to 121,289. This filtered dataset serves as the input for the next phase, which involves annotating the data by labelling it with the appropriate dialect.

4.2. Data annotation

In the data annotation process, also known as data labelling, three native Arabic speakers assist in this process. The result of this process was obtained based on two criteria. The first criteria, is if two of the annotators (raters/judges) agreed on the annotation then the decision will be taken. And the second criteria is using the Cohen's kappa agreement which was applied, and the agreement reached to 81 %. Then after this process the total number of user-generated comments was reduced to 38,394. The final dataset is described in Table 3. The Word Cloud of the four dialects is shown in Fig. 2(A–D).

4.3. Data cleaning

In the data cleaning phase, several steps were performed using regular expressions and a Python package called "ar_corrector" to improve the accuracy of the models. These steps included.

- Removing repeated characters in Arabic words. For example, the word "مرحباااا" (which means "welcome") was corrected to its original form, "مرحبا".
- Removing English characters from user comments.
- Removing special characters such as @, #, $, and %.
- Removing unnecessary spaces between words.

4.4. Feature extraction

In this sections, the three widely recognized approaches for text representation into numerical data were employed: TF-IDF, Word2Vec, and GloVe. These representations are used to provide input to the model.

4.5. TF-IDF

Term Frequency-Inverse Document Frequency (TF-IDF) is a numerical statistic used in word representation to input into an MLC. It is used to show the importance of a term in comments relative to a collection of dataset. The TF component calculates the frequency of a term in a comment as shown in mathematical formulas (2), (3), and (4) for TF, IDF, and TF-IDF respectively.

$$TF = \frac{number\ of\ times\ the\ words\ appears\ in\ a\ comments}{total\ number\ of\ words\ in\ the\ comments.} \tag{2}$$

$$IDF = Log\left(\frac{number\ of\ comments\ in\ the\ dataset}{number\ of\ comments\ in\ the\ dataset\ contain\ the\ word}\right) \tag{3}$$

$$\mathrm{TF} - \mathrm{IDF} = \mathrm{TF} * \mathrm{IDF} \tag{4}$$

Table 2
Sample of Gulf and Yemeni Dialect.

Gulf dialect	Pronunciation in English	Translated to English	Yemeni dialect	Pronunciation in English	Translated to English
كيف حالك؟	Kaif halak?	How are you	jadeedak? (What's new with you?) Eesh - ايش جديدك؟	Esh jadeedak?	What's new with you?
شنو الأخبار؟	Shno al akhbar?	Whats the news?	ما اشتيش	Ma ashteesh	I don't want
وشلونك؟	Washlounak?	How are you	وايش فيك	wesh feek	what's in you?
هلا و غلا	Hala w'ghala	Hello and welcome	ماحصلتكش	Ma 7saltiksh	I didn't find you
هذا شيء جميل	Hathaa shay' jameel	This is something nice	وينك	waynak	where are you
حمدلله	Hamdulilah	Thank god	نسير مع بعض	nseer ma' ba'dh	we'll walk together
راح اروح	Ra7 aru7	I will go	ضابح	dhabi7	Tired
تكفى	Takfa	Please	نصطبح	nstab7	we'll eat breakfast
هذا شيء حلو	Hathaa shayy heloo	This is something nice	بكلمك بعدين	bklmk b3den	I'll talk to you later
ما في مشكلة	Mafi mushkla	No problem	وحشتنا	wa7shtna	we've missed you
إيش عندك؟	Esh 'andak?	Whats wrong?	شخبرك	sha khabrik	I want to tell you

Table 3
Dataset description.

Dialect	No. of User comment	Max. length	Min length
Egyptian	9891	86	4
Jordanian	12,493	36	6
Gulf	8470	22	5
Yemeni	7540	31	2
Total	38,394		

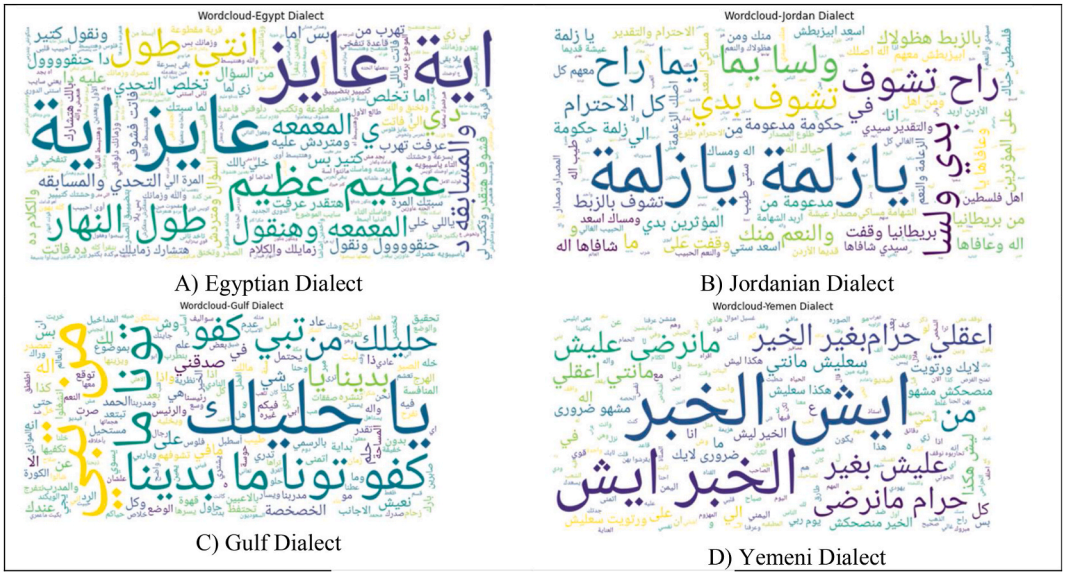


Fig. 2. Word Cloud for the four Dialects.

4.6. Word2Vec

Word2Vec is a technique used to represent words as vectors in a high-dimensional space, where words with similar meanings are closer to each other. The Skip-gram model in Word2Vec aims to maximize the average log probability of predicting context words given a target word as shown in mathematical formula (5). This is done using the softmax function to convert the output of the model into a probability distribution over the entire vocabulary.

$$maximize = \frac{1}{T}\sum_{t}^{T}\sum_{-C\le j\le c, j\neq 0} \log p\left(w_{t+j} \mid w_t\right) \tag{5}$$

where T is the total number of words in the corpus, as well as w_t being the target word at position t, w_{t+j} is the context word at position $t+j$ within the context window of size c, $p(w_t + j\, w_t)$ is the conditional probability of observing context word $w_t + j$ given target word w_t.

The conditional probability $p(w_t + j\, w_t)$ is computed using the softmax function as presented in mathematical formula (6).

$$p\left(w_{t+j} \mid w_t\right) = \frac{e^{v'_{wt+j} \cdot v_{wt}}}{\sum_{j=1}^{V} e^{v'_i \cdot v_{wt}}} \tag{6}$$

where, v_w is the input vector (word embedding) for target word w_t , v'_{wt+i} is the output vector for context word $w_t + j$, V is the size of the vocabulary. The goal of the Skip-gram model is to learn the word embeddings v_w and v'_{wt+i} that maximize this objective function, effectively capturing the semantic relationships between words in the corpus.

4.7. GloVe

Global Vectors (GloVe) are used for Word Representation, it is an unsupervised learning algorithm for obtaining vector representations for words. Unlike Word2Vec, which focuses on predicting context given to a word (or vice versa), GloVe learns word embeddings by considering the global word co-occurrence statistics in the corpus.

GloVe uses the co-occurrence matrix X, where X_{ij} represents how often word j appears near word i. The objective of GloVe is to derive word vectors w_i and 'w'j' (for input and output contexts, respectively) so that their dot product is equivalent to the algorithm of the co-occurrence probability of words i and j, adjusted by a bias term as presented in mathematical formula (7).

$$\sum_{i,j} f(x_{i,j}) \left(w_i^T w_j' + b_j + b_j' - \log\left(X_{i,j}\right) \right)^2 \tag{7}$$

Where w_i *and* w_j' represent the vectors of words i and j, b_i and b_j^i are bias for both words while $f(w_{ij})$ is a weighting function.

By minimizing this objective function, GloVe learns word embeddings by minimizing the objective function. Therefore, it captures both syntactic and semantic relationships between words based on their co-occurrence patterns in the dataset.

4.8. Model building

This section introduces two hybrid models that combine the strengths of CAMeLBERT and ALBERT with BiLSTM networks for ADI, as shown in Fig. 3. CAMeLBERT and ALBERT are specialised variants of the BERT model, and each model is designed for specific purposes. These models aim to improve performance by leveraging the pre-trained representations of transformer models and the sequential learning capabilities of BiLSTM. BiLSTM is selected due to its performance in dealing with sequential data, particularly its effectiveness at capturing dependencies and relationships in sequences in the Arabic dialect. Therefore, BiLSTM is well-suited for understanding the structure and nuances of Arabic sentences. As a transformer-based model, CAMeLBERT excels at providing rich contextual embeddings by considering the entire sentence at once due to its attention mechanisms. Therefore, combining the outputs of BiLSTM and CAMeLBERT leads to rich feature representations that improve the model performance in terms of accuracy. BiLSTM can also capture short-to medium-range dependencies in sequences, whilst CAMeLBERT captures long-range dependencies between Arabic dialects. ALBERT efficiently captures semantic nuances across entire sentences, reduces the parameter size and requires less memory compared to CAMeLBERT, thereby accelerating the training process.

The architecture of the CAMeLBERT–BiLSTM and ALBERT–BiLSTM hybrid models involves encoding input text into contextual

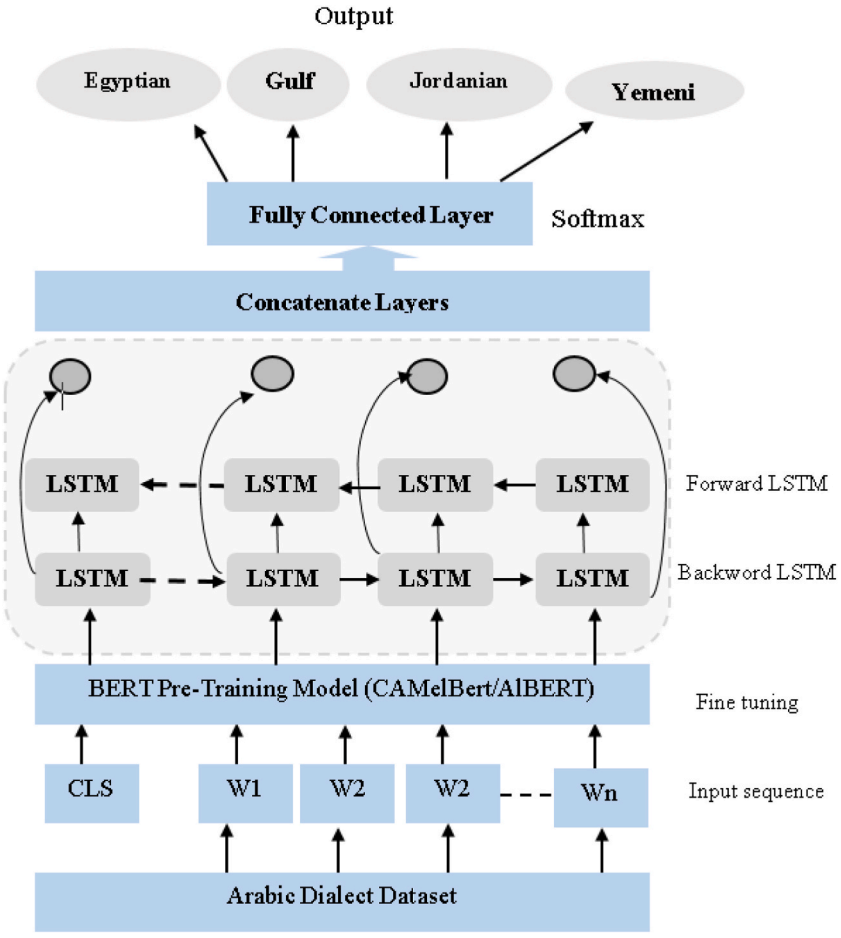


Fig. 3. Architecture of proposed model.

embeddings using CAMeLBERT or ALBERT. These embeddings capture the semantic meaning of words in context. The output embeddings are denoted as X = {x_1,x_2, …,x_n }, where n is the number of tokens in the input text. The output embeddings are then passed through a BiLSTM layer to capture sequential dependencies in the data in forward and backward directions. This method produces a sequence of hidden states H = {h1,h2, …,hn }. The forward pass of the BiLSTM is defined in mathematical formulas (8) and (9).

$$\vec{h_t} = \mathrm{LSTM}\left(x, \vec{h_{t-1}}\right), \tag{8}$$

$$\overleftarrow{h_t} = \mathrm{LSTM}\backslash\left(x, \overleftarrow{h_{t+1}}\right), \tag{9}$$

where $\vec{h}$ and $\overleftarrow{h}$ are the hidden states at t timestamp. The output of BiLSTM passed through a SoftMax function to obtain the probabilities of each class (Arabic dialect) as seen in mathematical formula (10).

$$\hat{y} = softmax(Wh_n + b), \tag{10}$$

where W is the weight and b is the bias.

The CAMeLBERT is pre-trained on a large corpus of Arabic texts, enabling learning of general language aspects. Therefore, the fine-tuning of CAMeLBERT adapts it to the specific task of ADI. Meanwhile, ALBERT is trained on general text corpora in multiple languages to learn universal language representations. The models can benefit from contextual understanding when CAMeLBERT or ALBERT are combined with BiLSTM. Additionally, BiLSTM captures fine-grained sequential patterns. This hybrid approach may improve performance and robustness in identifying Arabic dialects.

The proposed models were validated using conventional MLCs such as DT, SVM, RF, LR, NB, XGB, SGD and KNN [50–52]. Furthermore, BiLSTM can be used with a variety of word representations, such as TF-IDF, Word2Vec and GloVe, which are commonly utilised for text classification and sequence dependence problems.

4.9. Model performance evaluation

In all of the experiments, the performance of each model was assessed using widely used metrics which include; accuracy, precision, recall, and F1-score [53,54]. Each calculated based on the confusion matrix.

Accuracy is a metric used to evaluate the performance of a classification model. It is calculated as the ratio of the number of correct predictions to the total number of predictions made by the model. The formula for accuracy is shown in mathematical formulas (11).

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total number of predications}} \text{X100} \tag{11}$$

Precision is the ratio of true positive predictions to the total number of positive predictions made by the model as calculated in mathematical formula (12).

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \tag{12}$$

Recall, also known as sensitivity or true positive rate, is the ratio of true positive predictions to the total number of actual positive instances in the dataset, as presented in mathematical formula (13)

$$Recall = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \tag{13}$$

The F1-score is the harmonic mean of precision and recall. It provides a balance between precision and recall, giving equal weight to both metrics as seen in mathematical formula (14)

$$\text{F1} - \text{score} = 2X\frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{14}$$

5. Results and discussion

The experiment settings are explained in this section, and the model performance on ADI is validated using conventional MLCs, DLMs, transformers and proposed models. The performance of the model was measured using the recall, precision, F-Score, accuracy and confusion matrix.

5.1. Experiment settings

The settings for ML, DL and transformer experiments are presented in this section. All experiments were conducted using Python 3 Google Compute Engine backend (GPU) which known as google Colab. The dataset was divided into the following: 30 % for testing and 70 % for training in ML experiments and 80 % for training and 20 % for testing in DL and transformers experiments. In the ML experiments, the sklearn package was employed for splitting datasets, extracting features, classifying ML and evaluating confusion

matrices and models. Meanwhile, the NLTK package was used for tokenisation and the removal of stop words. The TensorFlow framework was utilised in the deep learning experiment. All the transformers were used on the Hugging Face platform and transformers[torch] has installed. Table 4 demonstrates the hyperparameters of MLCs, whilst Table 5 shows the hyperparameters of DL experiments. The hyperparameters for Transformers and proposed models is presented in Table 6.

5.2. Machine learning experiments

Several MLCs are utilised on the proposed dataset to validate the performance of the proposed models in determining which two Arabic dialects are similar to each other. Therefore, two types of experiments are conducted. The first experiment is conducted to reveal which two dialects are close to each other in terms of binary classification, whilst the second experiment is performed to determine the performance of the model for the four classes. Table 7 presents a comparative analysis between the accuracies of the aforementioned MLCs.

LR and SVM outperformed the other models across all the two dialect pairs to demonstrate which pairs are the closest to each other and it shows that LR slightly reached higher scores than SVM further showing that LR is the best at differentiating between Arabic dialects and also shows how similar the pair of dialect are to one-another. LR was overall the highest performing classifier, therefore LR went on to achieve the highest accuracy in several pairs. For example, LR, SVM attained high accuracies of 91.44 % and 91.11 %, respectively, when tested on the Egyptian–Gulf dialect pair. RF, SGD, and XGB also performed well, falling a little behind the models which received the highest score (LR and SVM), especially in pairs such as Egyptian–Yemeni. In the Egyptian-Yemeni pair the three models; RF, SGD, and XGB obtained accuracy rates of 94.70 %, 94.71 %, and 94.34 % respectively. KNN and DT demonstrated mixed performance, with DT having the lowest average accuracy amongst the models. For example, the DT model attained the lowest overall accuracy of 65.89 % when tested using the Egyptian–Jordanian dialect pair. Under the testing of KNN, the Gulf–Jordanian dialect pair attained an accuracy of 76.59 %, the Egyptian–Jordanian pair obtained a score of 76.59 %, and the Egyptian–Gulf pair attained a score of 77.86 %, demonstrating one of the lowest accuracies on the aforementioned list. NB also revealed moderate performance, in dialect pairs such as Egyptian–Jordanian. Specifically, NB achieves a score of 78.42 %, but it also reaches high scores for Gulf-Yemeni dialect pairs, it displayed a score of 95.09 %.

Table 7 also shows the average accuracy of closeness obtained by all the aforementioned models between all the dialect pairs to identify the closest dialect pairs to each other. The two average closest dialect pairs are the Gulf-Yemeni pair and the Egyptian–Yemeni pair, with the Gulf-Yemeni dialect pair receiving an rate of relatedness of 94.16 % and The Egyptian-Yemeni dialect pair slightly falls behind at an average of 93.62 %, further demonstrating that the two dialect pairs (Gulf–Yemeni and Egyptian–Yemeni) are the closest pairs to each other, with the Gulf-Yemeni being slightly closer than the Egyptian-Yemeni dialect pair on average, also as presented through confusion matrix using the LR classifier in Fig. 4. An average of 79.07 % is obtained for the Egyptian–Jordanian dialect pair, which is the two dialect pairs which proved to be the furthest apart from each other. Similarly, an average of 92.19 % is attained for the Jordanian–Yemeni dialect pair on average.

Generally, the highest accuracies were achieved by the LR and SVM classifiers. With the LR classifier showing the best performance in the Gulf-Yemeni. And the SVM classifier similarly displaying the best score in the Gulf-Yemeni dialect pair 95.03 %. as portrayed through the confusion matrix in Fig. 4(A and B) and Fig. 5(A and B) which shows the AUC-ROC which indicates the model performance of the model. the poorest accuracies were achieved by the DT and KNN classifiers. With the DT classifier showing the poorest accuracy for the Egypt-Gulf pair and the Egyptian-Jordanian pair. Additionally, the KNN classifier presenting the poorest accuracy for the Gulf-Jordanian pair. Both are as depicted through the confusion matrix in Fig. 6(Aand B) and AUC-ROC is shown in Fig. 7(A and B).

For the experiments using MLCs on the multiclass datasets, the model selection impacts performance across different dialects. For instance, the SVC consistently achieves high precision across all dialects, particularly for the Egyptian dialect, which has the highest precision of 77 %. This result demonstrates its effectiveness in accurately classifying instances of these dialects. Meanwhile, the DT model struggles with low precision and recall values, especially for the Jordanian and Gulf dialects. The model maintains a precision of 66 % for the Gulf dialect, as well as a 45 % precision score for the Jordanian dialect. This finding shows the limitations of this model for accurately classifying these dialects. Fig. 8 presents the comparison between the MLCs in terms of recall, precision, F-score and Accuracy.

The results reveal the weak points of the models when tested with certain dialects (such as the Gulf dialect). These results consistently show low precision and recall values across multiple models. This finding could be due to the complexity and diversity of the Gulf dialect compared with other dialects, which causes difficulty in its accurate classification. Furthermore, the performance

Table 4
Hyperparameters of MLCs.

Algorithm	Parameter 1	Value	Parameter 2	Value	Parameter 3	Value
KNN	n_neighbors	5	weights	'uniform'	algorithm	'auto'
RF	n_estimators	100	criterion	'gini'	max_depth	None
DT	criterion	'gini'	splitter	'best'	max_depth	None
LR	penalty	'l2'	C	1.0	solver	'lbfgs'
SVM	C	1.0	kernel	'rbf'	gamma	'scale'
NB	alpha	1.0 (MultinomialNB)	var_smoothing	1e-9 (GaussianNB)	–	–
XGB	n_estimators	100	learning_rate	0.3	max_depth	6
SGD	loss	'hinge'	penalty	'l2'	alpha	0.0001

Table 5
Hyperparameter of DL experiments (BilSTM).

Item(s)	Values
Embedding Dimension	256
LSTM Units	128
Dropout	0.5
Batch Size	32
Sequence Length	100
Optimizer	"Adam"
Loss Function	"binary_crossentropy"
Metrics	["accuracy"]
Number of Epochs	30
Activation function (Hidden)	relu
Activation function (output)	softmax

Table 6
Hyperparameters transformers.

Item(s)	Values
Batch Size	16
Number of Epochs	5
weight_decay	0.01
logging_steps	100
learning_rate	2e-5

Table 7
Accuracy of MLCs in determining the similarity between Arabic dialects.

Dialect/Accuracy	KNN	RF	DT	LR	SVM	NB	XGB	SGD	Average
Egyptian–Gulf	77.86 %	89.91 %	**69.71 %**	91.44 %	91.11 %	81.88 %	87.68 %	88.25 %	84.73 %
Egyptian–Jordanian	76.95 %	82.45 %	**65.89 %**	82.41 %	82.21 %	78.42 %	81.88 %	82.32 %	79.07 %
Egyptian–Yemeni	90.76 %	94.70 %	91.09 %	**94.76 %**	94.76 %	93.75 %	94.34 %	94.71 %	**93.62 %**
Gulf–Jordanian	76.59 %	82.03 %	73.43 %	81.92 %	81.87 %	80.71 %	81.43 %	81.83 %	79.98 %
Gulf –Yemeni	91.31 %	95.04 %	92.31 %	**95.02 %**	95.09 %	94.28 %	94.93 %	95.01 %	**94.16 %**
Jordanian–Yemeni	87.59 %	93.28 %	90.01 %	93.82 %	93.38 %	92.86 %	93.32 %	93.32 %	92.19 %

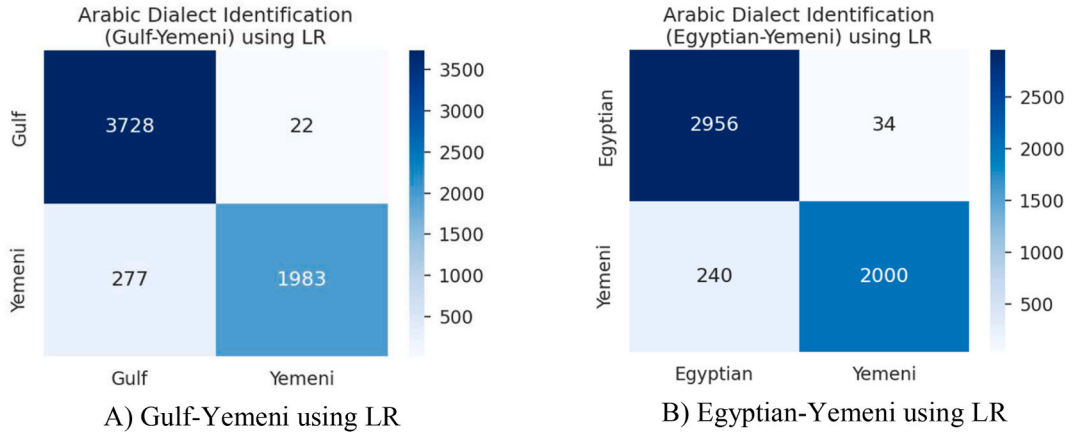


Fig. 4. Confusion matrix using the LR classifier (Gulf-Yemeni and Egyptian-Yemeni).

variation amongst models highlights the importance of considering model complexity and training data quality. Models such as RF and LR show relatively balanced performance across all dialects, indicating the robustness and generalisation capabilities of these models. By contrast, some models, such as NB and DT, exhibit more inconsistent performance, indicating their sensitivity to variations in the training data or requiring additional fine-tuning to achieve superior results.

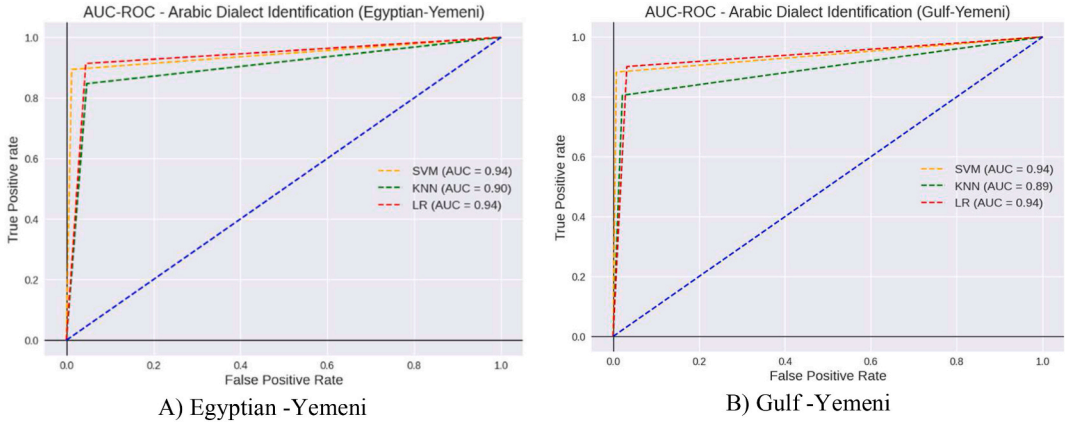


Fig. 5. AUC-ROC (Gulf-Yemeni and Egyptian-Yemeni).

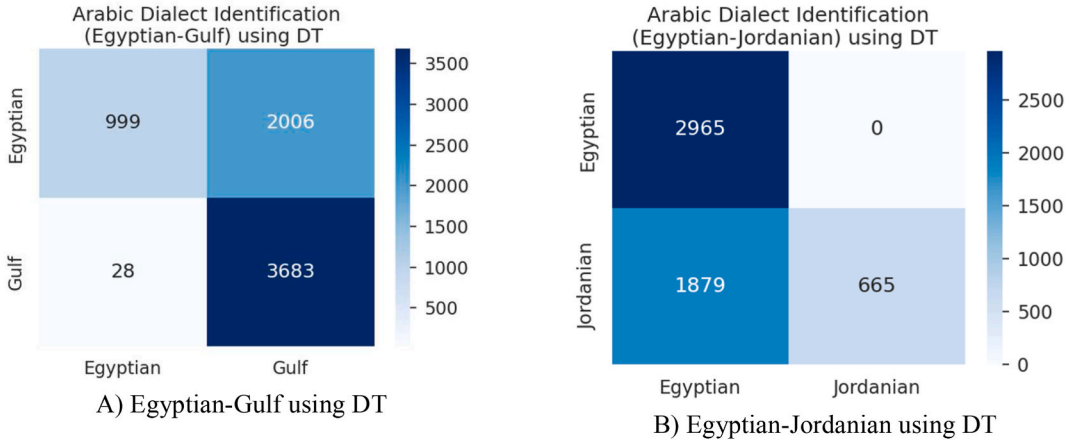


Fig. 6. The confusion matrix using DT as poor accuracy (Egyptian -Jordanian and Egyptian and Gulf).

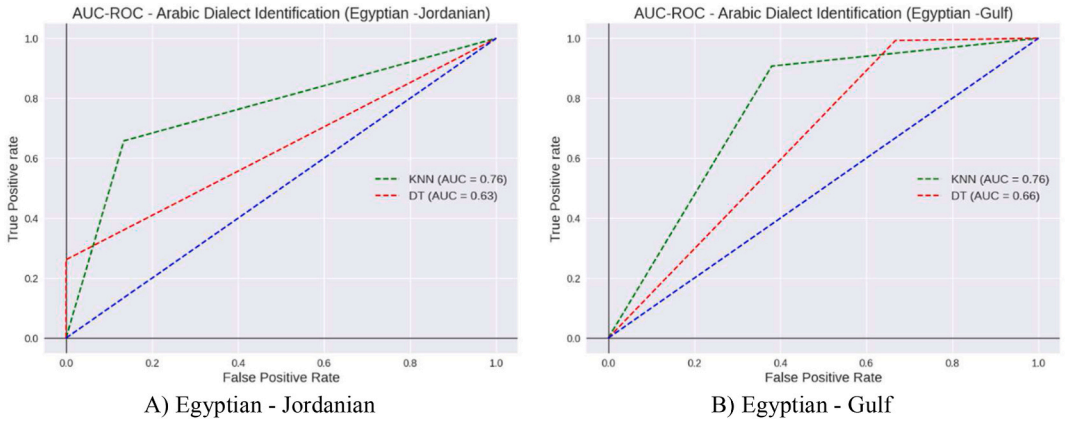


Fig. 7. AUC-ROC(Egyptian -Jordanian and Egyptian and Gulf).

5.3. Deep learning experiments

The experimental results based on DLMs are presented in this section to validate the performance of the two proposed models for detecting and classifying the Arabic dialects amongst the four aforementioned dialect groups. This experiment can be categorised into

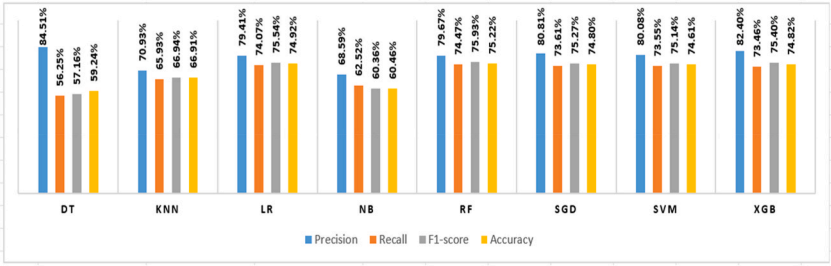


Fig. 8. Comparison between MLCs for multiclass classification (Four Dialects).

three types using the BiLSTM model, with three different word representation methods such as TF-IDF, Word2Vec and Glove. Transformer models such as ALBERT and CAMELBERT are used in the second experiment. Table 8 provides a detailed comparison of various models for ADI, emphasising their performance across key metrics.

In the first experiment, the BiLSTM model using TF-IDF embeddings demonstrated a moderate performance of 54.46 % in precision, 56.76 % in recall, 54.16 % in F1-score and 54.46 % in accuracy. This result shows that the BiLSTM model with the TF-IDF word representation method may ineffectively capture the unique features of Arabic dialects, which leads to an under-average performance as shown in Fig. 9(A and B) and in confusion matrix in Fig. 10By contrast, the BiLSTM model using GloVe embeddings achieves significantly higher scores, revealing a precision score of 80.16 %, recall of 81.46 %, F1-score of 80.62 % and accuracy of 80.74. This result indicates that using the GloVe word representation method along with the BiLSTM model is highly effective in detecting and classifying between dialects as presented in Fig. 10(A and B) and the confusion matrix as shown in Fig. 11 (A, B, C).

The BiLSTM model using Word2Vec embeddings performs the poorest amongst the three, revealing a precision score of 57.18 %, recall of 60.13 %, F1-score of 57.81 % and accuracy of 58.28 %. This result indicates that the Word2Vec embeddings may not be well suited for capturing the dialect-specific features of Arabic as shown in Fig. 12(A and B). Meanwhile, the CAMeLBERT and ALBERT models demonstrate high performance across all categories of testing, revealing approximately 85 % and 80 % precision, recall, F1-score and accuracy, respectively. This result indicates that embeddings such as CAMeLBERT and ALBERT are highly effective for ADI and are well suited for capturing the complex linguistic patterns of Arabic dialects, as presented in Fig. 13 (A,B) for confusion matrix.

In the last experiment, the proposed combinations of CAMeLBERT with BiLSTM and ALBERT with BiLSTM showed improved performance in detecting Arabic dialects over individual models, revealing accuracy scores of 87.67 % and 86.51 %, respectively. This result shows that the idea of combining different models can enhance performance in ADI. Overall, the results indicate the importance of embedding choice and model combination as shown in Fig. 14(A and B) for confusion matrix.

Through the combination of two transformer models (CAMeLBERT and ALBERT) with BiLSTM, the proposed models are improved by utilising the strengths of both architectures. Transformers are effective and efficient at capturing long-range dependencies and contextual information, whilst BiLSTM is efficient at learning sequential patterns. The fusion of these models likely results in a highly robust and comprehensive representation of the input data. In addition, the proposed models were fine-tuned more effectively than the other models. Fine-tuning allows models to adapt to the specific characteristics of the dataset, potentially leading to improved performance. The results of the experiments reveal that transfer learning can leverage pre-trained knowledge and adapt it to the specific task of. Combining BiLSTM and CAMeLBERT provides a powerful approach for identifying Arabic dialects. Leveraging the deep contextual embeddings of CAMeLBERT, which is trained on diverse Arabic text, and the sequential learning capabilities of BiLSTM, can help effectively obtain and distinguish the nuances of different Arabic dialects by capturing dependencies in forward and backward directions in Arabic user comments and refining the sequence level of words in the comments. In addition, CAMeLBERT effectively captures nuanced language features and contextual word representations extracted from Arabic comments. Additionally, generalised pre-trained embeddings of CAMeLBERT prevent the model from reducing the risk of overfitting. Pre-trained knowledge of CAMeLBERT on Arabic helps in initial dialect differentiation, and the capability of BiLSTM to learn from sequence data fine-tunes this differentiation. Meanwhile, ALBERT is used to reduce model size and training time. Table 9 demonstrates the accuracy of exiting models that used a multiclass classification problem, particularly on text by focus on Arabic dialect.

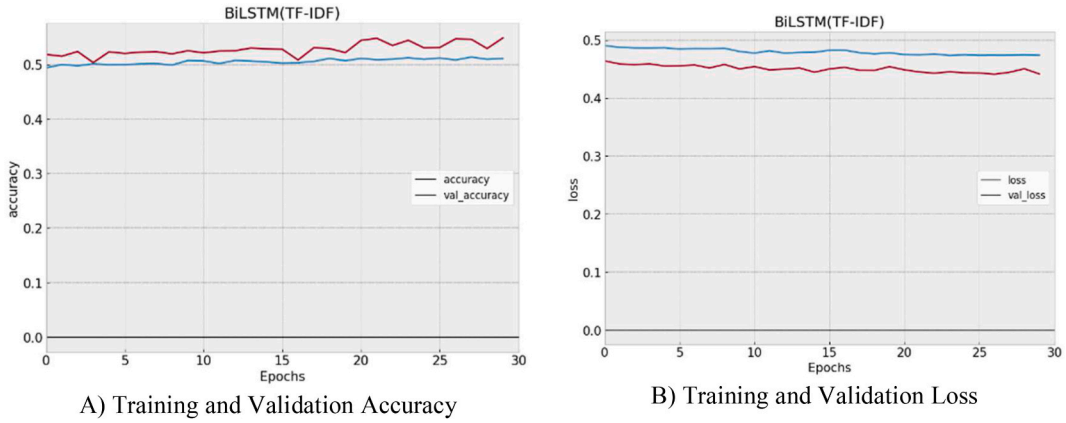
6. Conclusion

This study proposed two hybrid models for ADI, namely BiLSTM with CAMeLBERT and ALBERT. In addition, a novel dataset containing comments created by users on numerous social media platforms spanning four major Arabic dialects is presented. Extensive trials demonstrate that the introduced models produce high accuracy rates of 86.51 % for ALBERT with BiLSTM and 87.67 % for CAMeLBERT with BiLSTM, outperforming MLCs and DLMs. The experimental results demonstrate the importance of utilising several model architectures for this task because the combination of transformer models and BiLSTM produces excellent results in ADI. Furthermore, comparing GloVe embeddings with BiLSTM shows notable gains over alternative word representation techniques. The dataset comprises 121,289 user-generated comments but may fail to capture the full diversity of Arabic dialects because it only focuses on four major dialects (Egyptian, Jordanian, Gulf and Yemeni). Thus, increasing the dataset size is required.

Numerous directions could be considered in the future to improve ADI performance and applicability. One important field involves investigating sophisticated fine-tuning techniques for transformer models and BiLSTM to optimise learning rates, batch sizes and

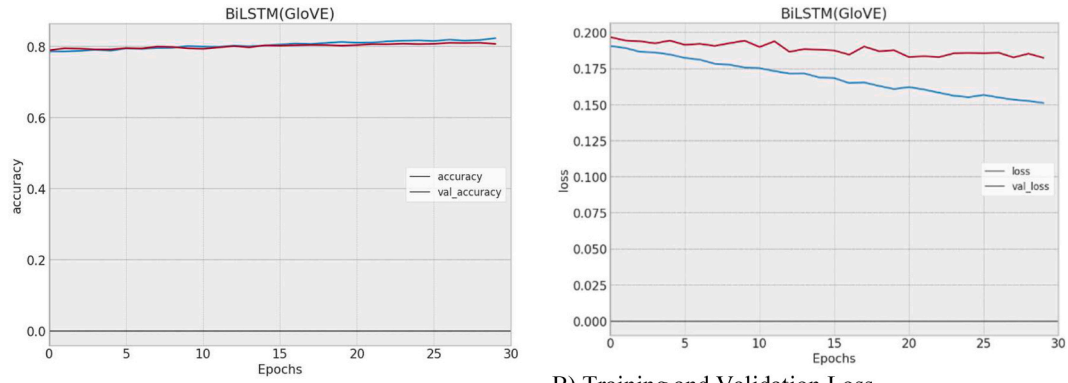
Table 8
Comparison between accuracies of the DLMs and the proposed models.

Models	Precision	Recall	F1	Accuracy
BiLSTM (TF-IDF)	54.46 %	56.76 %	54.16 %	54.46 %
BiLSTM (Glove)	80.16 %	81.46 %	80.62 %	80.74 %
BiLSTM (Word2Vec)	57.84 %	60.13 %	57.81 %	58.28 %
CAMeLBERT	84.12 %	85.45 %	85.12 %	85.16 %
ALBERT	81.12 %	77.66 %	78.79 %	78.49 %
CAMeLBERT with BiLSTM (proposed)	87.96 %	87.59 %	87.76 %	**87.67 %**
ALBERT with BiLSTM (proposed)	87.06 %	86.38 %	86.69 %	**86.51 %**



A) Training and Validation Accuracy B) Training and Validation Loss

Fig. 9. BiLSTM using TF-IDF.



A) Training and Validation Accuracy B) Training and Validation Loss

Fig. 10. BiLSTM using GloVe.

optimisation algorithms. Furthermore, additional investigation into model architecture, including transformer model variants such as BERT, RoBERTa and XLNet, may help elucidate the best methods for ADI. The diversity and volume of the data could be increased by employing data augmentation techniques to supplement the dataset, which could strengthen the resilience of the model. Investigating multilingual and cross-lingual ADI techniques using training models in several languages or transferring knowledge between languages could also be advantageous.

Data availability statement

The dataset can be accessed at https://www.kaggle.com/datasets/amjadalsuwaylimi/arabic-dialect-dataset. Click or tap if you trust this link.">https://www.kaggle.com/datasets/amjadalsuwaylimi/arabic-dialect-dataset.

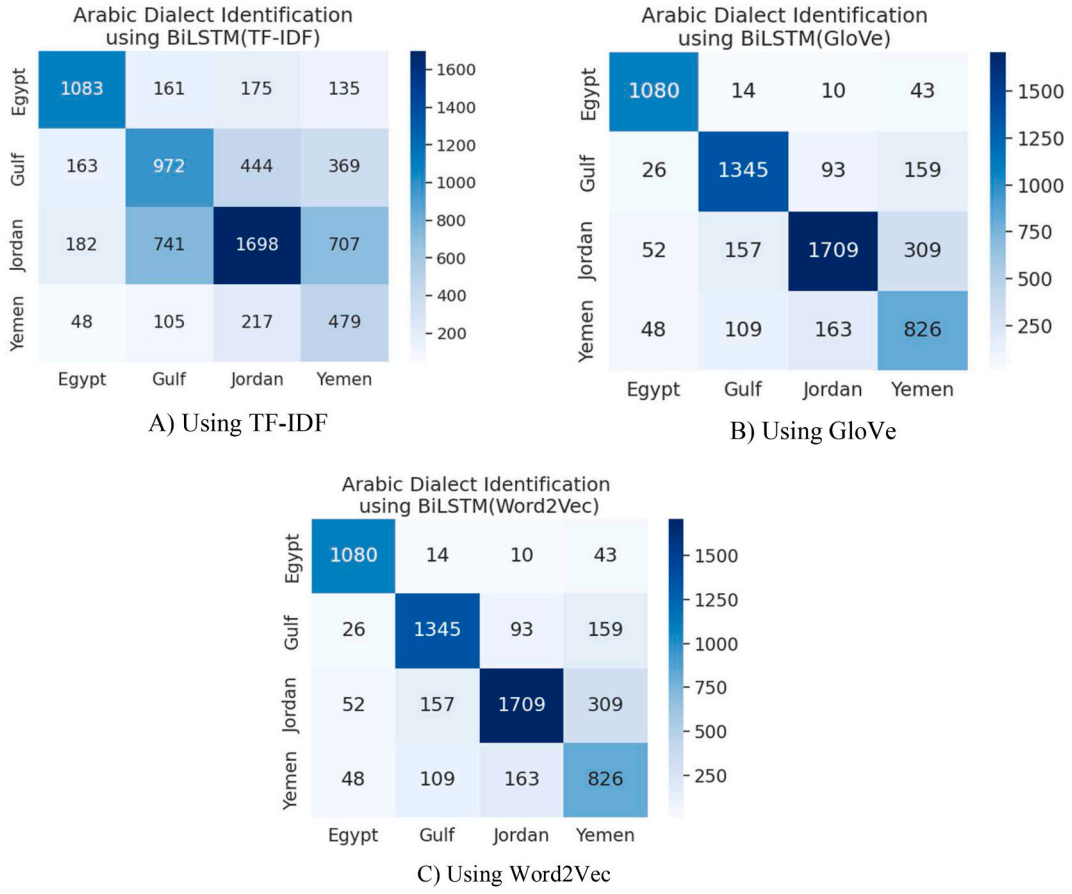


Fig. 11. Confusion Matrix BiLSTM using TF-IDF,GloVe, and Word2Vec.

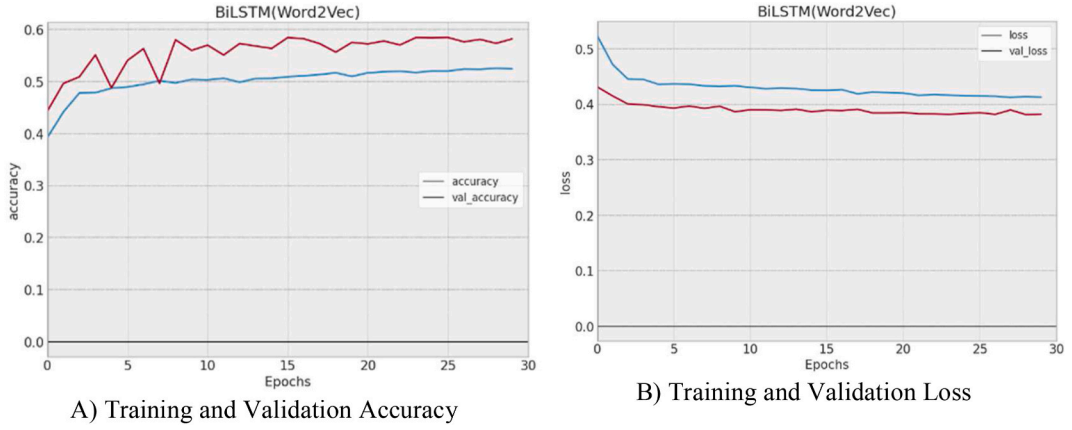


Fig. 12. BiLSTM using Word2Vec.

CRediT authorship contribution statement

Amjad A. Alsuwaylimi: Writing – review & editing, Writing – original draft, Validation, Supervision, Methodology, Investigation, Funding acquisition, Data curation, Conceptualization.

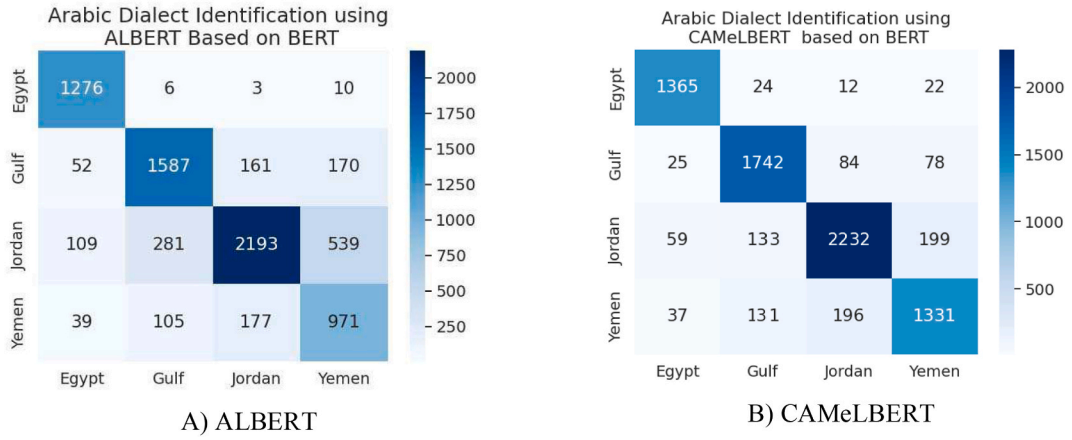


Fig. 13. Confusion matrix of ALBERT and CAMeLBERT based on BERT.

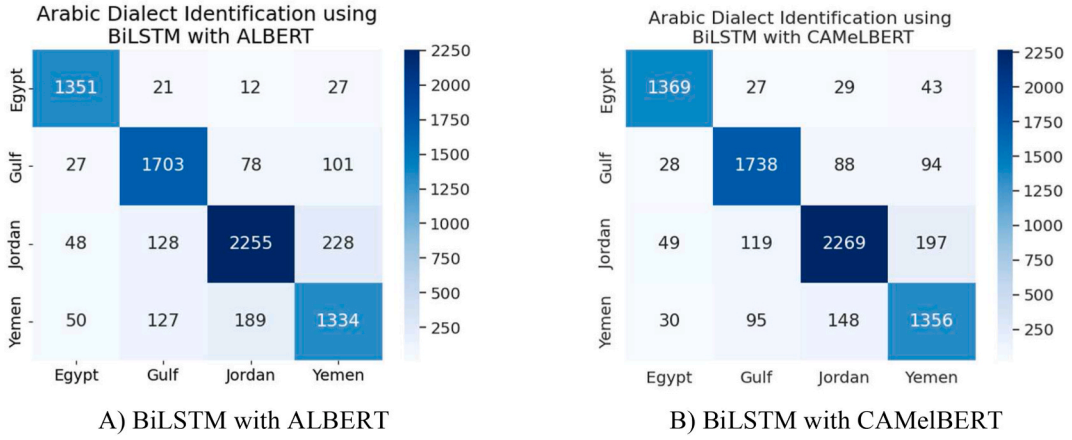


Fig. 14. Confusion matrix of proposed models based BiLSTM with ALBERT and CAMeLBERT.

Table 9
Compare between the accuracy produced using of the pre-train model and the proposed model.

References	Model	dataset	Size	Accuracy
[55]	Based on BERT	Multiple sources	9312	77.61 %
[56]	Multi-dialect-Arabic-BERT-Ensemble-Diff-Lenwith rules	NADI (21 Arabic dialect)	31,000	45.07 %
[57]	M_BERT	NADI	31,000	40.95 %
[58]	CNN with a GRU recurrent layer	AOC	14591 utterances for training and 1566	57.59 %
[59]	MARBERT	NADI	31,000	53.890 %
[60]	Arabic-BERT	tweet	9,999,978	23.45 %
[61]	BERT-Linear-pair	HAAD	2838	73.23 %
[62]	Approached based ArabicBERT	ArSenTD-Lev (Multi-classes)	4000	75 %
[63]	BERT	Public dataset	5000	67 %
[64]	ArSenTD-LEV		4000	53.7 %

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Amjad Alsuwaylimi reports financial support was provided by Northern Border University. Amjad Alsuwaylimi reports a relationship with Northern Border University that includes: employment. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The author extend their appreciation to the Deanship of Scientific Research at Northern Border University, Arar, Kingdom of Saudi Arabia, for funding this research work through project number "NBU-FFR-2024-1197-03".

References

[1] M. Salameh, H. Bouamor, N. Habash, Fine-grained Arabic dialect identification, in: Proceedings of the 27th International Conference on Computational Linguistics, 2018, August, pp. 1332–1344.
[2] A. Abdelali, H. Mubarak, Y. Samih, S. Hassan, K. Darwish, QADI: Arabic dialect identification in the wild, in: Proceedings of the Sixth Arabic Natural Language Processing Workshop, 2021, April, pp. 1–10.
[3] W.M. Yafooz, E.A. Hizam, W.A. Alromema, Arabic sentiment analysis on chewing Khat leaves using machine learning and ensemble methods, Eng. Technol. Appl. Sci. Res. 11 (2) (2021) 6845–6848.
[4] M.A. Sghaier, M. Zrigui, Rule-based machine translation from Tunisian dialect to modern standard Arabic, Procedia Comput. Sci. 176 (2020) 310–319.
[5] Ahmed Abdelali, Hamdy Mubarak, Younes Samih, Sabit Hassan, Kareem Darwish, "Arabic dialect identification in the wild.", arXiv preprint arXiv:2005.06557 (2020).
[6] R. Alhejaili, A. Alsaeedi, W.M. Yafooz, Detecting hate speech in Arabic tweets during COVID-19 using machine learning approaches, in: Proceedings of Third Doctoral Symposium on Computational Intelligence: DoSCI 2022, Springer Nature Singapore, Singapore, 2022, November, pp. 467–475.
[7] M. Abdul-Mageed, C. Zhang, H. Bouamor, N. Habash, NADI 2020: the First Nuanced Arabic Dialect Identification Shared Task, 2020 arXiv preprint arXiv: 2010.11334.
[8] M.J. Althobaiti, Automatic Arabic dialect identification systems for written texts: a survey, arXiv preprint arXiv:2009.12622 (2020).
[9] A.A. Al Shamsi, S. Abdallah, Text mining techniques for sentiment analysis of Arabic dialects: literature review, Adv. Sci. Technol. Eng. Syst. J. 6 (1) (2021) 1012–1023.
[10] A. Alsudais, W. Alotaibi, F. Alomary, Similarities between Arabic dialects: investigating geographical proximity, Inf. Process. Manag. 59 (1) (2022) 102770.
[11] M. Zampieri, P. Nakov, Y. Scherrer, Natural language processing for similar languages, varieties, and dialects: a survey, Nat. Lang. Eng. 26 (6) (2020) 595–612.
[12] S. Hajbi, Y. Chihab, R. Ed-Dali, R. Korchiyne, Natural Language processing based approach to overcome arabizi and code switching in social media Moroccan dialect, in: Advances in Information, Communication and Cybersecurity: Proceedings of ICI2C'21, Springer International Publishing, 2022, pp. 57–66.
[13] D. Demszky, D. Sharma, J.H. Clark, V. Prabhakaran, J. Eisenstein, Learning to Recognize Dialect Features, 2020 arXiv preprint arXiv:2010.12707.
[14] O. Kuparinen, A. Miletić, Y. Scherrer, Dialect-to-Standard normalization: a large-scale multilingual evaluation, in: Findings of the Association for Computational Linguistics: EMNLP 2023, 2023, December, pp. 13814–13828.
[15] N.A. Alhassan, A.S. Albarrak, S. Bhatia, P. Agarwal, A novel framework for Arabic dialect chatbot using machine learning, Comput. Intell. Neurosci. 2022 (2022).
[16] A. Aliwy, H. Taher, Z. AboAltaheen, Arabic dialects identification for all Arabic countries, in: Proceedings of the Fifth Arabic Natural Language Processing Workshop, 2020, December, pp. 302–307.
[17] M. Sobhy, A.H.A. El-Atta, A.A. El-Sawy, H. Nayel, Word representation models for Arabic dialect identification, in: Proceedings of the the Seventh Arabic Natural Language Processing Workshop (WANLP), 2022, December, pp. 474–478.
[18] K.M. Nahar, O.M. Al-Hazaimeh, A. Abu-Ein, M.A. Al-Betar, Arabic Dialect Identification Using Different Machine Learning Methods, 2022.
[19] S. Touileb, Ltg-st at nadi shared task 1: Arabic dialect identification using a stacking classifier, in: Proceedings of the Fifth Arabic Natural Language Processing Workshop, 2020, December, pp. 313–319.
[20] N.A. Alhassan, A.S. Albarrak, S. Bhatia, P. Agarwal, A novel framework for Arabic dialect chatbot using machine learning, Comput. Intell. Neurosci. 2022 (2022).
[21] T. Jauhiainen, H. Jauhiainen, K. Lindén, Optimizing naive Bayes for Arabic dialect identification, in: Proceedings of the the Seventh Arabic Natural Language Processing Workshop (WANLP), 2022, December, pp. 409–414.
[22] H. Nayel, A. Hassan, M. Sobhi, A. El-Sawy, Machine learning-based approach for Arabic dialect identification, in: Proceedings of the Sixth Arabic Natural Language Processing Workshop, 2021, April, pp. 287–290.
[23] S. Hussein, M. Farouk, E. Hemayed, Gender identification of egyptian dialect in twitter, Egyptian Informatics Journal 20 (2) (2019) 109–116.
[24] K.M. Nahar, O.M. Al-Hazaimeh, A. Abu-Ein, M.A. Al-Betar, Arabic Dialect Identification Using Different Machine Learning Methods, 2022.
[25] M.J. Althobaiti, Country-level Arabic dialect identification using small datasets with integrated machine learning techniques and deep learning models, in: Proceedings of the Sixth Arabic Natural Language Processing Workshop, 2021, April, pp. 265–270.
[26] L. Lulu, A. Elnagar, Automatic Arabic dialect classification using deep learning models, Procedia Comput. Sci. 142 (2018) 262–269.
[27] M. Elaraby, M. Abdul-Mageed, Deep models for Arabic dialect identification on benchmarked data, in: Proceedings of the Fifth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial 2018), 2018, August, pp. 263–274.
[28] A. Elnagar, R. Al-Debsi, O. Einea, Arabic text classification using deep learning models, Inf. Process. Manag. 57 (1) (2020) 102121.
[29] S. ElSayed, M. Farouk, Gender identification for Egyptian Arabic dialect in twitter using deep learning models, Egyptian Informatics Journal 21 (3) (2020) 159–167.
[30] A.A. Alvarez, E.S.A. Issa, Learning Intonation Pattern Embeddings for Arabic Dialect Identification, 2020 arXiv preprint arXiv:2008.00667.
[31] B. Talafha, M. Ali, M.E. Za'ter, H. Seelawi, I. Tuffaha, M. Samir, H.T. Al-Natsheh, Multi-dialect Arabic bert for country-level dialect identification, arXiv preprint arXiv:2007.05612 (2020).
[32] B. Talafha, M. Ali, M.E. Za'ter, H. Seelawi, I. Tuffaha, M. Samir, H.T. Al-Natsheh, Multi-dialect Arabic bert for country-level dialect identification, arXiv preprint arXiv:2007.05612 (2020).
[33] N. AlShenaifi, A. Azmi, Arabic dialect identification using machine learning and transformer-based models: submission to the NADI 2022 Shared Task, in: Proceedings of the the Seventh Arabic Natural Language Processing Workshop (WANLP), 2022, December, pp. 464–467.
[34] M. Abdul-Mageed, C. Zhang, H. Bouamor, N. Habash, NADI 2020: the First Nuanced Arabic Dialect Identification Shared Task, 2020 arXiv preprint arXiv: 2010.11334.
[35] A. Mohammed, Z. Jiangbin, A. Murtadha, A three-stage neural model for Arabic Dialect Identification, Comput. Speech Lang 80 (2023) 101488.
[36] A. El Mekki, A. Alami, H. Alami, A. Khoumsi, I. Berrada, Weighted combination of BERT and n-GRAM features for nuanced Arabic dialect identification, in: Proceedings of the Fifth Arabic Natural Language Processing Workshop, 2020, December, pp. 268–274.
[37] A. El Mekki, A. El Mahdaouy, K. Essefar, N. El Mamoun, I. Berrada, A. Khoumsi, BERT-based multi-task model for country and province level MSA and dialectal Arabic identification, in: Proceedings of the Sixth Arabic Natural Language Processing Workshop, 2021, April, pp. 271–275.
[38] K. Gaanoun, I. Benelallam, Arabic dialect identification: an Arabic-BERT model with data augmentation and ensembling strategy, in: Proceedings of the Fifth Arabic Natural Language Processing Workshop, 2020, December, pp. 275–281.
[39] Malik Sallam, Dhia Mousa, Evaluating ChatGPT performance in Arabic dialects: a comparative study showing defects in responding to Jordanian and Tunisian general health prompts, Mesopotamian Journal of Artificial Intelligence in Healthcare 2024 (2024) 1–7.
[40] K. Khaled, T. Wael, S. Khaled, W. Medhat, Arabic dialect identification: experimenting pre-trained models and tools on country-level datasets, in: 2023 20th ACS/IEEE International Conference on Computer Systems and Applications (AICCSA), IEEE, 2023, December, pp. 1–7.
[41] M. Abdul-Mageed, C. Zhang, H. Bouamor, N. Habash, NADI 2020: the First Nuanced Arabic Dialect Identification Shared Task, 2020 arXiv preprint arXiv: 2010.11334.

[42] M. Abdul-Mageed, C. Zhang, A. Elmadany, H. Bouamor, N. Habash, NADI 2021: the Second Nuanced Arabic Dialect Identification Shared Task, 2021 arXiv preprint arXiv:2103.08466.
[43] H. Bouamor, N. Habash, M. Salameh, W. Zaghouani, O. Rambow, D. Abdulrahim, K. Oflazer, The madar Arabic dialect corpus and lexicon, in: Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), 2018, May.
[44] A. Abdelali, H. Mubarak, Y. Samih, S. Hassan, K. Darwish, QADI: Arabic dialect identification in the wild, in: Proceedings of the Sixth Arabic Natural Language Processing Workshop, 2021, April, pp. 1–10.
[45] S. Shon, A. Ali, Y. Samih, H. Mubarak, J. Glass, ADI17: a fine-grained Arabic dialect identification dataset, in: ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2020, May, pp. 8244–8248.
[46] O. Alsemaree, A.S. Alam, S.S. Gill, S. Uhlig, An analysis of customer perception using lexicon-based sentiment analysis of Arabic Texts framework, Heliyon 10 (11) (2024) e30320.
[47] L.H. Baniata, S. Kang, Switch-transformer sentiment analysis model for Arabic dialects that utilizes a mixture of Experts mechanism, Mathematics 12 (2) (2024) 242.
[48] M.A. Humayun, H. Yassin, J. Shuja, A. Alourani, P.E. Abas, A transformer fine-tuning strategy for text dialect identification, Neural Comput. Appl. 35 (8) (2023) 6115–6124.
[49] M. Al-Deaibes, E. Al-Shawashreh, M. Jarrah, Emphatic variation of the labio-velar/w/in two Jordanian Arabic dialects, Heliyon 7 (11) (2021) e08295.
[50] R. Alfred, J.H. Obit, The roles of machine learning methods in limiting the spread of deadly diseases: a systematic review, Heliyon 7 (6) (2021) e07371.
[51] R.A. Alsaidi, W.M. Yafooz, H. Alolofi, G.A.M. Taufiq-Hail, A.H.M. Emara, A. Abdel-Wahab, Ransomware detection using machine and deep learning approaches, Int. J. Adv. Comput. Sci. Appl. 13 (11) (2022).
[52] S.N.H. Bukhari, E. Elshiekh, M. Abbas, Physicochemical properties-based hybrid machine learning technique for the prediction of SARS-CoV-2 T-cell epitopes as vaccine targets, PeerJ Computer Science 10 (2024) e1980.
[53] S.N.H. Bukhari, J. Webber, A. Mehbodniya, Decision tree based ensemble machine learning model for the prediction of Zika virus T-cell epitopes as potential vaccine candidates, Sci. Rep. 12 (1) (2022) e1980.
[54] A.E. Yahya, A. Gharbi, W.M. Yafooz, A. Al-Dhaqm, A novel hybrid deep learning model for detecting and classifying non-functional requirements of mobile apps issues, Electronics 12 (5) (2023) 1258.
[55] M. Alruily, A. Manaf Fazal, A.M. Mostafa, M. Ezz, Automated Arabic long-tweet classification using transfer learning with BERT, Appl. Sci. 13 (6) (2023) 3482.
[56] B. Talafha, M. Ali, M.E. Za'ter, H. Seelawi, I. Tuffaha, M. Samir, H.T. Al-Natsheh, Multi-dialect Arabic bert for country-level dialect identification, arXiv preprint arXiv:2007.05612 (2020).
[57] A. El Mekki, A. Alami, H. Alami, A. Khoumsi, I. Berrada, Weighted combination of BERT and n-GRAM features for nuanced Arabic dialect identification, in: Proceedings of the Fifth Arabic Natural Language Processing Workshop, 2020, December, pp. 268–274.
[58] M. Ali, Character level convolutional neural network for Arabic dialect identification, in: Proceedings of the Fifth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial 2018), 2018, August, pp. 122–127.
[59] A.S. Khered, I.Y.H.A. Abdelhalim, R.T. Batista-Navarro, Building an ensemble of transformer models for Arabic dialect classification and sentiment analysis, in: Proceedings of the the Seventh Arabic Natural Language Processing Workshop (WANLP), 2022, December, pp. 479–484.
[60] K. Gaanoun, I. Benelallam, Arabic dialect identification: an Arabic-BERT model with data augmentation and ensembling strategy, in: Proceedings of the Fifth Arabic Natural Language Processing Workshop, 2020, December, pp. 275–281.
[61] M.M. Abdelgwad, T.H.A. Soliman, A.I. Taloba, Arabic aspect sentiment polarity classification using BERT, Journal of Big Data 9 (1) (2022) 1–15.
[62] H. Chouikhi, H. Chniter, F. Jarray, Arabic sentiment analysis using BERT model, in: Advances in Computational Collective Intelligence: 13th International Conference, ICCCI 2021, Kallithea, Rhodes, Greece, September 29–October 1, 2021, Proceedings, vol. 13, Springer International Publishing, 2021, pp. 621–632.
[63] E. Fsih, S. Kchaou, R. Boujelbane, L.H. Belguith, Benchmarking transfer learning approaches for sentiment analysis of Arabic dialect, in: Proceedings of the the Seventh Arabic Natural Language Processing Workshop (WANLP), 2022, December, pp. 431–435.
[64] M. Alruily, A. Manaf Fazal, A.M. Mostafa, M. Ezz, Automated Arabic long-tweet classification using transfer learning with BERT, Appl. Sci. 13 (6) (2023) 3482.