Unraveling the Linguistic Complexity: A Comprehensive Study on Language Identification of Moroccan Darija Text

JBEL MOUAD, TAOUSSI CHAIMAE, JABRANE MOURAD,HAFIDI IMAD

***Abstract*—This paper presents a language identification study on Moroccan dialect text, also called Darija, which is commonly written in various encodings, making other NLP tasks complicated and significantly affecting performance. To address this, we first created a large and diverse dataset, which constitutes a major contribution to Moroccan dialect NLP tasks. Following this, we constructed multiple word representations and embeddings for the Moroccan dialect across different encodings, encompassing three languages: Standard Arabic, Darija in Roman encodings, and Darija in Arabic encodings. Specifically, we employed several approaches, including TF-IDF and Word2Vec, utilizing both continuous bag-of-words and skip-grams, as well as FastText-CBOW and FastText-SkipGram. The performance evaluation of these techniques indicated that the Word2Vec-SkipGrams model, applied to a BLSTM classifier, achieved the best results in terms of language identification, reaching an accuracy of 93.46%. Furthermore, to find the best model for our dataset, we implemented several transfer learning models. The highest accuracy, 97.56%, was achieved by fine-tuning the DarijaBert-mix model on our dataset. For a more comprehensive evaluation, we compared these results with other available datasets.**

***Index Terms*—Moroccan Darija, Dialectic Arabic, Word Embeddings, Machine learning, Transfer Learning, Code Switching.**

I. Introduction

MOROCCAN Dialect (MD) is the mother tongue of a significant portion of the Moroccan population. Although it is an Arabic dialect, MD incorporates elements from various other languages, particularly French, Berber, and Spanish. Consequently, many MD speakers opt to represent their language using the characters of different languages, a practice commonly referred to as transliteration.

Transliteration, or the process of encoding a language in the characters of another language, is a technique utilized to represent words and phrases of one language using characters from another language. Its primary goal is to increase the accessibility of text written in one language to speakers of another. An example is representing the Arabic script in Roman characters to enhance readability for individuals accustomed to the Roman alphabet. It is essential to differentiate transliteration from translation, which refers to changing the text's whole language. Transliteration is commonly employed as an educational tool for language acquisition and in the development of text-to-speech systems. To achieve this, various methods can be used, such as rule-based transliteration, which utilizes predetermined rules to convert text, or machine learning-based transliteration, which applies models such as Word2Vec or other neural networks to learn the mapping between characters from different scripts.

Transliteration has become prevalent in the Arab world due to a combination of factors, including colonialism, globalization, increased education levels, and greater social media use in the MENA region. Today, many Arabs use multiple languages in their day-to-day communication. While French is common in North Africa, Arabic speakers in the Gulf countries and Egypt often incorporate English words into their language. In Morocco, the use of English and French in social media is widespread, with many Moroccans using a combination of both languages in their online communications. This is partly because French is widely spoken in Morocco and is often used in education, while English is seen as the global language of business and technology. However, many Moroccans also use transliteration, which involves writing words and phrases in one language using the characters of another language. This can make it challenging to detect the language used in social media posts, as words may be written in a mixture of English and French characters or a blend of Arabic and Roman characters. For instance, a Moroccan may write "Ahlan" in Roman characters instead of "اهلا" in Arabic characters, even though they are using the same word. Transliteration poses difficulties for automated tools to detect language and makes it harder for non-native speakers to understand the text. Therefore, it is not recommended for formal settings.

The study of Arabic and its dialectical variations through Natural Language Processing (NLP) has prompted many researchers to seek ways to address transliteration, as evidenced by several studies [1,2,3,4]. Nevertheless, various obstacles impede their progress, including the need for transliteration data for these languages and the limited size and high noise levels in the available datasets. Researchers have focused on developing large corpora of transliterated or code-mixed text to overcome these challenges by leveraging online resources, such as social media platforms [2,3]. Similarly, others have endeavored to create computational resources for dialectical Arabic by mining data from online sources [5,6].

In recent years, word embedding or distributed representa-

Manuscript received March 13, 2024; revised December 29, 2024.

Mouad JBEL is an undergraduate student of University Sultan Moulay Slimane, Beni-Mellal 23000, Morocco.(e-mail: mouad.jbel@gmail.com, phone: +212 611054748)

CHAIMAE TAOUSSI is an undergraduate student of University Sultan Moulay Slimane, Beni-Mellal 23000, Morocco.(e-mail : chaimae-taoussi27@gmail.com)

JABRANE MOURAD is an undergraduate student of University Sultan Moulay Slimane, Beni-Mellal 23000, Morocco.(e-mail : mourad-jabrane1998@gmail.com)

HAFIDI IMAD is an associate professor of University Sultan Moulay Slimane, Beni-Mellal 23000, Morocco.(e-mail: i.hafidi@usms.ma)

tions of words have become one of the most popular methods for word representation in various text mining and NLP tasks, as evidenced by several studies [1,4]. Compared to traditional one-hot representations of words, word embedding offers several advantages. Instead of encoding every word in a large vector, word-embedding projects all words into a low-dimensional continuous space. Additionally, word embedding provides meaningful syntactic and semantic information about words by learning from textual data instances. As such, they are an essential component of state-of-the-art NLP models, including those developed for Arabic NLP. For example, Soliman et al. [4] developed AraVec, a collection of efficient distributed word representations for the Arabic language. Hamed et al. [1] focused on building word embeddings for Egyptian Arabic-Arabic code-mixed data.

Our work aims to build a transliteration dataset with a focus on different encodings of MD data and Arabic data. We then aim to develop models that can effectively capture the nuances and subtleties of MD for accurate LI results. This requires an extensive exploration of various ML models, including deep learning (DL) approaches such as Long Short-Term Memory (LSTM) and Convolutional Neural Networks (CNN), as well as language models such as BERT pretrained on MD text. In the process, we build the best model for LI of MD encodings to differentiate between those encodings and Modern Standard Arabic (MSA) too, in addition to the first transliteration dataset that includes both scripting types of MD to date, to the knowledge of the authors.

The rest of the paper is organized as follows: Section 2 discusses the related works regarding the Language identification (LI) task. Section 3 explores the various materials we relied on and the dataset we created. Section 4 represents the DL models and approaches used to develop the comparative study. While section 5 represents the transformers based approaches we used. In section6, we highlight the results we achieved through our models. In addition, in section 7, we present a conclusion and aspirations for future work.

II. Related works

NLP researchers have studied the LI task in detail over the years, but we will only focus on the most relevant works in the field for this paper. Our studies primarily focus on texts containing transliteration and code-switching phenomena, particularly informal short texts. Our research identified three main approaches for the LI task: language modeling, machine learning, and deep learning-based algorithms.

A. Language Modelling Approaches

Numerous studies have employed language models to tackle the LI task in code-switched texts. This approach is grounded on the belief that each language possesses unique phonology, morphology, and character behavior [7]. Some studies have focused on MSA and Arabic dialects. For instance, Zaidan and Callison-Burch [8] proposed an approach based on a smoothed n-gram model. Elfardy et al. [9] developed a code-switched MS Egyptian word identification system that relies on an MSA morphological analyzer. To identify the Romanized form of Arabic dialects (Arabizi), Eskander et al. [10] used a language model for name tagging and introduced a set of features. Shreshta [12], a participant in the first shared task on LI on code-switched data [11], focused on the LI of code-switched Spanish-English and Nepali-English texts and used an incremental n-gram approach. In the second shared task on LI on code-switched data [13], several researchers relied on language modeling approaches. For example, Chanda et al. [14] developed an algorithm for code-switched English-Spanish tweets that generates a word's n-gram and checks its presence in dictionaries. Shirvani et al. [15] focused on Swahili-English code-switched texts. They introduced several features, including character n-grams, which were also used by Piergallini et al. [43] for code-switched English-Spanish with 17 new features, including Part of Speech (POS) tags. Jhamtani et al. [16] have devised a model for identifying code-switched Hindi-English texts at the word level in various settings. This model utilizes multiple features, including character n-grams and the POS tags of neighboring words. They also experimented with several classifiers: Decision Trees, Support Vector Machines (SVM), and Random Forests. Nguyen and Cornips [17] conducted a study to detect Dutch Limburgish tweets that contained code-switching originating from a province in the Netherlands by utilizing word probabilities. On the other hand, Guellil and Azouaou [18] directed their attention towards the LI of the Algerian dialect. They employed an Algerian lexicon and an enhanced Levenstein distance to achieve more accurate outcomes.

B. Machine Learning and Deep Learning Approaches

As the volume of textual content, particularly on social media, continues to grow, research on LI has shifted towards machine learning approaches. For example, Giwa and Davel [19] employed Naïve Bayes (NB) and Support Machine Vector (SVM) algorithms to detect code-switched South African words. Sadat et al. [20] focused on identifying 18 Arabic dialects using a character-based n-gram Markov Model and NB. Bar and Dershowitz [22] contributed to the first shared task on LI on code-switched data, using SVM with different features to identify code-switched English-Spanish tweets. In their study, Barman et al. [23] concentrated on identifying code-switched words in Bengali, English, and Hindi textual material found on social media. They employed SVM and CRF algorithms for this purpose. Barman et al. [24] introduced a classification method that utilized K-Nearest Neighbors (KNN) and SVM to detect code-switched tweets in Nepali-English and Spanish-English. Chittaranjan et al. [25] used CRF with several features to identify the LI for code-switched English-Spanish, English-Nepali, English-Mandarin, and MSA-AD texts. Similarly, King et al. [26] extended a Markov Model to detect the same languages. In the same context, Lin et al. [27] used a baseline CRF model with labeled data and a CRF autoencoder with word embeddings and word lists as features with unlabeled data.

Multiple contributors in the second shared task on LI in code-switched data utilized machine learning models. For instance, Chanda et al. [21] employed J48, KNN, Random Forest algorithms, and various features to identify code-switched Bengali-English language productions. Similarly, Shrestha [29] used Conditional Random Fields (CRF) with features to identify code-switched English-Spanish and

MSA-AD language productions. Sikdar and Gambäk [47] also utilized CRF to LI code-switched English-Spanish texts. Other researchers addressed the LI task for different languages, including MSA-Moroccan texts using CRF by Samih and Maier [30], Latin-Middle English contents with CRF by Schulz and Keller [31], and several other language pairs using CRF by Al-Badrashiny and Diab [32]. Dongen [33] employed SVM, Decision Trees, and CRF to identify code-switched Dutch-English language productions on social media. Rijhwani et al. [34] worked on the LI task for a variety of languages, including Dutch, English, French, German, Portuguese, Spanish, and Turkish, using a Hidden Markov Model (HMM)-based Generalized Word-level Language Detection system. Aridhi et al. [35] utilized N-Gram Cumulative Frequency Addition and SVM for the LI of the Romanized Tunisian Dialect. Lastly, Salameh et al. [36] used Multinomial Naïve Bayes (MNB) and trained a 5-gram character-level language model using KenLM [38] to identify several AD and MSA dialects. Meanwhile, Lichouri et al. [37] used Linear SVM, Bernoulli Naïve Bayes (BNB), and MNB for the LI of AD and Algerian dialects.

Deep learning, a subset of machine learning, has recently gained significant attention from researchers. However, there are few works on the LI task compared to machine learning and language modeling approaches. Chang and Lin [39] used Elman-type and Jordan-type recurrent neural networks (RNN) with optional pre-trained Word2Vec and character n-gram features to detect the language of code-switching Twitter corpus for English-Spanish, English-Nepali, Mandarin-English, and MSA-Egyptian in the first shared task on LI in code-switched data [11].

Jaech et al. [40] focused on the LI of code-switched English-Spanish and MSA-AD tweets in the second shared task on LI in code-switched data [13]. Their LI system includes a convolutional neural network (CNN) component that provides word embeddings and a bidirectional long short-term memory (BLSTM) component for labeling. Samih et al. [41] employed a Long Short-Term Memory (LSTM) neural network with a Conditional Random Field (CRF) layer and a feature-rich template to detect code-switched English-Spanish and MSA-Egyptian Dialect textual content.

Mave et al. [42] used BLSTM, word-character LSTM, and CRF with different features (n-grams, POS tags...) to identify code-switched Hindi-English and Spanish-English textual content. In another study, Mager et al. [43] introduced a segmental model based on RNN to detect code-switching content in Spanish-Wixarika and German-Turkish languages. Elaraby and Abdul-Mageed [44] used six deep learning models, including CNN, LSTM, Contextual LSTM (CLSTM), BLSTM, Bidirectional Gated Recurrent Units (BiGRU), and BLSTM with attention mechanisms, as well as three machine learning classifiers (Logistic Regression, MNB, and SVM) to identify MSA and Arabic dialects.

Sayadi et al. [45] focused on the Tunisian Dialect, mainly using an LSTM RNN to identify TD-MSA and a set of five AD-MSA. The TD language election dataset and the AD languages' multidialect parallel corpus of Arabic [46] were utilized.

This research [28] paper introduces a framework for identifying languages at the word level in mixed script text, specifically English Roman and Hindi Devanagari. It employs word embedding techniques such as word2vec, TF-IDF, skip-gram, and CBOW to detect language patterns and ambiguous words. The approach uses character-based embedding to handle spelling variations, effectively addressing LI challenges and normalizing spelling variations in mixed-script text. Another work [57] examines LI in Hindi-English code-mixed text using Multilingual Meta Embeddings (MME). Comparing classifiers like CNN, GRU, LSTM, BiLSTM, and BiGRU on the LinCE Benchmark corpus, BiLSTM was the most effective. The study highlights the effectiveness of MME in handling language mixing within sentences or words.

Meanwhile, Raazia et al.[58] presents a Bi-LSTM CNN model for LI and Localization in Code-Mixed Urdu-English text, achieving 90.40% accuracy and a 90.39% F1 score. Trained on social media data with variant spellings of Roman Urdu words, the study demonstrates the effectiveness of neural networks in managing multilingual data. This research [59] develops a word-level LI model for code-mixed Indonesian, Javanese, and English tweets, introducing the IJELID corpus. It compares various strategies, including fine-tuning BERT, BLSTM-based, and CRF models. The results show that fine-tuned IndoBERTweet models perform the best due to BERT's contextual understanding and effective sub-word language representation. Another work[60] studies Arabic dialect identification, precisely Saudi dialects, using a character-level model. It highlights the importance of classifying dialects for document retrieval and language modeling applications. The study uses classical machine learning algorithms and a character convolutional neural network, achieving better performance with term frequency-inverse document frequency and character n-grams.

Amal et al. [61] address Arabic dialect identification using deep bidirectional transformers, focusing on Gulf, Iraqi, Egyptian, Levantine, and North African dialects. It evaluates MARBERT and ARBERT models on Arabic text datasets, finding that MARBERT achieves higher F1 scores. The study concludes that deep bidirectional transformers like MARBERT are effective for accurately classifying Arabic dialects in natural language processing tasks.

Another work [62] introduces the Unsupervised Deep Language and Dialect Identification (UDLDI) method for short texts, focusing on closely related languages or dialects. By leveraging attention to character relations, the UDLDI model learns sentence embeddings and optimizes language clustering based on sentence structures. This method significantly enhances performance in unsupervised scenarios with minimal training data, proving more effective than supervised systems.

C. Language Identification Datasets

Over the years, numerous datasets have been developed to facilitate research in LI, particularly addressing challenges such as transliteration, code-switching, and the informal nature of short texts. We provide an overview of the most relevant datasets created for the LI task, highlighting their unique contributions and methodologies. By examining these works, we aim to contextualize our study within the broader landscape of LI research and underscore the advancements made in this domain.

The authors of the MSDA dialect detection dataset [63] built a dataset of approximately 50k social media posts in different Arabic dialects, including Algerian, Egyptian, Tunisian, and Moroccan, all of which were written in Arabic characters.

The Moroccan YouTube Corpus(MYC) [64] comprises manually annotated comments collected from the widely used website YouTube. Through the efforts of multiple annotators and the application of a voting approach, the dataset was curated to include 20,000 comments labeled as positive or negative and enriched with additional metadata such as topic, likes, and dislikes. As the most extensive subjectivity corpus for the MD, MYC presents a valuable resource for developing dialect-specific NLP applications.

MADAR[65] is another multilingual dataset with approximately 111k sequences in 25 Arabic dialects, and all the text is scripted in Arabic characters. Shazal et al.[66] used a sequence-to-sequence deep learning model to transliterate SMS/chat written in Arabizi. For this task, they used a dataset with a size of about 60,000.

Abdelali et al. [67] proposed a method developed to rapidly construct a tweet dataset encompassing a wide range of country-level Arabic dialects spanning 18 countries in the Middle East and North Africa. This approach applies multiple filters to ascertain users' country of origin based on their account descriptions and to exclude tweets predominantly written in MSA or containing vulgar language. The resulting dataset comprises 540,000 tweets from 2,525 users, evenly distributed across 18 Arab countries, with all collected texts scripted in Arabic characters.

In summary, previous research on language detection for MD and other dialects has employed various approaches, including deep learning, word embeddings, and specialized feature sets. However, there is still a need to develop more accurate and robust language detection systems for MD. Among these methodologies, using word embeddings has demonstrated encouraging outcomes. The author's research aims to ascertain the most effective model for the LI task of MD, encompassing both MD written in Roman characters (MDR) and MD written in Arabic characters (MDA). Additionally, we aim to create a dataset that includes MD written in Arabizi (Roman characters), which is a significant contribution to this research field. It is noteworthy that prior research have solely concentrated on MDA.

III. Materials and methods

A. Data Retrieval and Processing

The procedure for gathering the MD raw text for our dataset is described in this section. In contrast to MSA, which predominates in written resources like news media, education, science, and books, MD is only utilized in informal circumstances like dialogues in TV series and movies. Recently, written MD has begun to develop on social media platforms (Facebook, Twitter, blogs). MD delivers socially motivated commentary on various areas and issues, from personal narratives to traditional folk literature, even though it is employed in informal settings (stories, songs, chat). It was incredibly challenging to locate and gather resources for MD. The lack of standardized orthography, the existence of various subdialects, and the widespread use of different writing scripts (Arabic vs. Arabizi) make MD resources susceptible to significant noise and inconsistency, which makes it difficult for techniques using query matching to identify dialectal text in the particular dialect of interest. Table I gives an example of each scripting type of MD.

TABLE I
Examples of Moroccan dialect texts in Arabic letters and roman letters

Text in English	MDA	MDR
this is beautiful	هادشي زوين	Hadchi zuin
I really like this product	عجبني هاد البرودوي	3jbni had lproduit

In our dataset, we focus on variety, too, rather than on mere size. We chose YouTube videos with different topics to extract their comments. Our approach to collecting MD comments is described as follows:

- Creating a script using YouTube API to extract comments from YouTube channels.
- Manually choosing 50 famous Moroccan YouTube channels of different topics as targets to ensure the variety of resources, then select and determine suitable MD content. Native MD speakers performed the selection and reviewing of resources.
- Manual annotation of comments into three classes, MSA, MDR (Moroccan dialect written in Roman characters), and MDA (Moroccan dialect written in Arabic characters), was performed by a native MD speaker.
- The dataset [68] is built of fifteen thousand annotated comments, five thousand of each class, to keep the dataset balanced since MD's resources were limited to 7,000 of each representation type and 144,912 tokens. Table II describes the content of the dataset more clearly.

TABLE II
Dataset components

	Sentence			
Language	MSA	MDA	MDR	Total
Global	7,000	7,000	7,000	21,000
Train	5,600	5,600	5,600	16,800
Test	1,400	1,400	1,400	4,200

B. Word Embedding algorithms

In order to build word embedding for transliterated MD, we opted for Word2Vec, using both CBOW and Continuous Skip-Gram, among the most commonly used architectures for training word embeddings [49]. We also experiment with a Word2Vec extension, namely FastText [50]—the Continuous Bag of Words (CBOW) architecture introduced by Mikolov et. Al [49] takes a set of context words as input to the left and the right and tries to predict the target (current) word based on that. While the context window size is a hyperparameter that can be tuned, experimentation has shown that a window size of 8 (4 words to the left and 4 to the right) performs better for many tasks [49]. Usually, the loss function used is the logarithmic loss function. As the predicted word is one of the many words in the vocabulary, the activation function is usually the SoftMax activation function. In [48], an alternative architecture called Continuous Skip-gram was

proposed. It operates similarly to CBOW, but instead of predicting the target word using the context words, it focuses on a single word and attempts to predict the group of words located at a specific distance from that word. The researchers have noticed that the performance of the SG model improves the further the window; however, this comes at the cost of increased computational complexity. However, FastText [50] is essentially a Word2Vec extension that aims to improve the modeling of languages with complex morphology. FastText achieves this by learning character n-grams and representing words based on the accumulation of these n-grams. Consequently, in contrast to Word2Vec, FastText exhibits a greater computational complexity, necessitating a lengthier training period. Additionally, we employed TF-IDF algorithms to generate comment vectors and compare them with traditional vectorization techniques.

IV. Adopted approaches for deep learning models

A. MDA-MDR-MSA word embedding

We utilized the Genism library developed by Rehurek et al. [48] for our experiments, as it offers a convenient and reliable implementation for the toolkits we selected to conduct our experiments, specifically Word2Vec and FastText. We constructed two models for each toolkit, utilizing the CBOW and SG architecture. Upon conducting experiments with various dimensions for word vectors, we observed that vectors with a size of 110 provided slightly superior representations. To account for spelling errors, we discarded words that appeared less than 50 times in the vocabulary. As a result, the models were all trained on a vocabulary of almost 120,000 tokens.

B. Classifiers

To assess the quality of word embedding models in language detection tasks, we trained deep learning models on each of the embedding models. Then, we compared all the models results. We looked into DL models, such as CNN, LSTM, and BLSTM.

1) Long short-term memory (LSTM): This type of recurrent neural network can learn long-term dependencies. Unlike standard RNNs, LSTMs are specifically designed to overcome the issue of long-term dependencies. In this study, a bidirectional LSTM is employed, where one LSTM retains the context of the previous words, and the other retains the context of the following words. The LSTMs process the comments individually, and the resulting outputs from each LSTM are combined to create a vector of length two h. This vector is then passed to a fully connected layer that utilizes the ReLU activation function. To enhance the model's robustness, a dropout layer is introduced after the LSTM layer, followed by another layer after the fully connected layer. Lastly, a SoftMax layer is incorporated to classify the sentiment category of the comment. The hyperparameters of LSTM model were tuned as follows:

- Number of hidden layers: 400
- Number of epochs: 100
- Batch size: 32

2) Bidirectional Long-Term Memory (BLSTM): BLSTM is a recurrent neural network (RNN) network that improves upon traditional LSTM models by processing data in both forward and backward directions. This bidirectional approach enables the model to capture context from past and future states, providing a more comprehensive understanding of the sequence data. BLSTMs are particularly effective for tasks involving sequential information, such as language modeling, speech recognition, and text classification, where understanding the context from both directions can lead to more accurate predictions. The hyperparameters of BLSTM model were tuned as follows:

- Number of hidden layers: 400
- Number of epochs: 40
- Batch size: 32
- LSTM units: 64

V. Adopted approaches for transfer learning

Transfer learning is a method that focuses on transferring the knowledge across domains, which is a promising machine learning methodology for solving NLP tasks problems. In our study, we fine tune previously created transformer-based language models of MD using our dataset to create a more accurate LI model for this dialect.

A. Moroccan dialect language models

BERT's architecture utilizes a multi-layer bidirectional transformer encoder. To perform transfer-learning experiments, we considered the following models:

- DarijaBert [69] is a BERT model specifically designed for the Moroccan Arabic dialect, developed by AIOX Labs, a Moroccan AI company. It has a vocabulary size of 80k and is trained on 691MB of MD text sourced from web stories, tweets, and YouTube comments. Utilizing the BERT base configuration, it employs a masked language modeling (MLM) task and comprises 147 million parameters.
- DarijaBert-mix [69] is an advanced BERT model for the Moroccan Arabic dialect developed by the same research team. This model is trained on a more extensive dataset with a vocabulary size of 160k and a corpus of 1.7GB that incorporates both Arabic and Latin script.
- MARBERT [70] is an Arabic multi-dialect model constructed using the BERT architecture and has been trained on 128GB of Arabic tweets. This model utilizes a masked language modeling (MLM) objective and follows the BERT base configuration, incorporating 163 million parameters.
- MorrBERT [71] is also a BERT model designed for the MD. It followed the exact configuration of BERTBASE with 12 layers, 12 attention heads, and a batch size of 64 and was trained for 565,980 steps. It employs a vocabulary size of approximately 52k.
- MorRoBERTa is a model conducted by the same authors of MorrBERT and is a compact variant of the RoBERTa-base model [72], featuring six layers, 12 attention heads, 768 hidden dimensions, and a maximum sequence length of 512. The training process utilized a batch size of 128 and extended over 565,980 steps.

B. Experiment

To evaluate the effectiveness of transferring contextual embeddings from Morrocan text models to the LI task and explore our dataset further, we fine-tune each model using our MD dataset and additional public datasets to perform a comparative study between them. The dataset we used are:

1) MSDA dialect detection dataset [63]: approximately 50k social media posts in different Arabic dialects. To be compatible to our study we only used the Arabic and MD text included in it, which gave about 9,964 input text.
2) MAC: Moroccan Arabic corpus [73] is a dataset made from Facebook comments expressed in modern standard or MD Arabic. As a result a dataset was created under the name of MAC that contains 8,360 MD text.

This approach aims to highlight the value of our dataset and identify the most suitable LI model for it. All models were fine-tuned using identical hyperparameters for all evaluation tasks: a maximum sequence length of 128, 5 epochs, a batch size of 64, "AdamW" optimizer with a learning rate of 5e-5, and the mixed precision data type "FP16" for gradient computations. In all experiments, the models were evaluated using F1-score and Accuracy metrics. Additionally, each experiment was conducted three times to ensure reliable results, and we reported the highest F1 score achieved by each model on every evaluation task. For each task, the dataset was divided into 80% for training and the remaining 20% reserved for validation.

VI. Results and discussion

A. Comparative study of deep Learning created models

In this section, we present the experiment for LI of Moroccan texts. To determine the most precise ML algorithms for LI tasks, we focused on the accuracy as an evaluation metric. Table III provides details about each model using each of the proposed vectorization methods.

TABLE III
ACCURACY RESULTS OF SUGGESTED MODELS

	CNN	LTSM	BLSTM
TF-IDF (%)	87.09	86.77	89.31
Word2vec-CBOW (%)	90.57	91.1	91.86
Word2vec-SG(%)	91.65	92.32	**93.46**
FastText-CBOW (%)	91.16	91.98	92.69
FastText-SG (%)	90.8	90.45	91.44

As we can see from Table III, all the word-embedding models have been able to identify which language every sentence belongs to with high accuracy compared to the TF-IDF model. SG models performed slightly better than their CBOW counterparts on most of the classifiers; Word2Vec-SG is the best architecture for this experiment, as it has a 93.46% accuracy using the BLSTM classifier, which is more than all the models in the accuracy metric. We detail the results this model gave in terms of accuracy and F1-score for each tag in Table IV.

We can conclude from Table 4 that the model we generated for the LI task yields promising results, especially for MDR, where it has an accuracy of 94.24% and an F1 score of 94.22%. In contrast, it lacks slightly on the side of MDA, which showed low results compared to others writing scripts.

TABLE IV
DETAILED RESULTS OF BLSTM MODEL TRAINED USING WORD2VEC-SG EMBEDDING

Accuracy	Languages	Accuracy	F1-score
93,46%	Modern Standard Arabic	94.03	94.03
	Moroccan dialect (Roman characters)	94.24	94.22
	Moroccan dialect (Arabic characters)	92.11	92.11

However, still, we believe these results are the current state-of-art as they classify both encoding types of the dialect where we scored the highest f1-score for both encoding types in MD, and we can conclude that Word2vec-SG is the best word embedding for the MD for the task of LI. The error rate is higher in MSA and MDA compared to MDR, and we believe it is due to the complexity of the Arabic language and Arabic characters, and the high number of words in this language extends the error rate. However, for cases of errors that occurred in the MDR pairs, we notice that most of the errors result of tagging an MDR sentence as a foreign sentence are due to similarity in words in the sentence or words that have a root from a foreign language (English and French); we summarize and categorize the observed errors as follows:

1) *MDR text containing words with foreign roots:* we notice several cases of erroneous identification with the words having foreign roots and MD affixes. We show some examples in Table V.

TABLE V
EXAMPLES OF MDR TEXT CONTAINING WORDS WITH FOREIGN ROOTS

Text	MDR words	meaning	root
Instagramek zuin	Instagramek	Your Instagram	Instagram
Nshari lik lconection	Nshari	Share with you	Share
	lconexion	The connection	Connection

2) *MDR text containing words with foreign origins:* This case differs from the previous one since the foreign word's morphology is fully altered to match an MD conjugation (verb, plural...) Table VI shows some examples.

TABLE VI
EXAMPLES OF MDR TEXT CONTAINING WORDS WITH FOREIGN ORIGINS

Words	meaning	Source word	Source language
Videowat	videos	video	English
nprepariw	We prepare	Prepare / preparer	French / English

3) *MDR text containing words from two or more languages:* The error mostly happens when the number of words from each language is almost equal. Table VII shows some examples.

TABLE VII
EXAMPLES OF MDR TEXT CONTAINING WORDS FROM DIFFERENT LANGUAGES

Text	meaning	MDR words	Foreign words
Bghit like	I want a like	Bghit	like
3jbatni bzaf la video	I really loved the video	3jbtani / bzaf	La video

Based on the error analysis, we notice that part of the observed inaccuracies is related to the rarity of words. We can remedy this issue by enlarging the corpus size to cover maximum vocabulary. A larger corpus will certainly incorporate different kinds and styles of written MDR texts, and

consequently help reduce the errors related to oddly written words. The errors related to conjunctions and determiners that are occasionally attached to foreign words could be avoided if we conduct a character-level study and adopt a segmentation method to consider morphological information, such as affixes and stop words.

B. Comparative study of Transformers-based models

As a baseline, we considered the model based on a combination of CNN and Word2vec-SG using the same parameters as those used for our CNN model.

We cannot use the previous work of the other datasets as a reference, as they use the cross-validation method. The results of the experiment are more detailed in Tables VIII, IX, X and XI.

The tables illustrate that the DarijaBert-mix model attained the highest F1 Score and Accuracy score across all datasets, enhancing our dataset's baseline accuracy by 4.1%. Additionally, it improved the baseline accuracy and F1 Score of the MSAC dataset by approximately 5%. Conversely, DarijaBert and MARABERT demonstrated inferior metrics and it's due to theier training phase was limited to text scripted using Arabic characters. Overall, DarijaBert-mix has significantly outperformed all other models, with an average Accuracy of 94.96% and an average F1 Score of 94.6%. It is followed by MorrBERT, which has average scores of 93.58% for Accuracy and 93.3% for F1 Score, respectively.

TABLE VIII
F1-SCORE AND ACCURACY OBTAINED USING DIFFERENT TRANSFORMER-BASED MODELS ON OUR DATASET

	Our Dataset	
	Accuracy	F1
Baseline	93.46	93.12
DarijaBert	74.12	64.23
DarijaBert-mix	**97.56**	**97.67**
MARBERT	59.12	59.14
MorrBERT	97.13	95.13
MorRoBERTa	94.96	96.2

TABLE IX
F1-SCORE AND ACCURACY OBTAINED USING DIFFERENT TRANSFORMER-BASED MODELS ON MSDA DATASET

	MSDA	
	Accuracy	F1
Baseline	87.46	88.16
DarijaBert	69.65	64.83
DarijaBert-mix	**94.64**	**94.97**
MARBERT	61.34	60.95
MorrBERT	92.89	92.11
MorRoBERTa	91.62	91.05

TABLE X
F1-SCORE AND ACCURACY OBTAINED USING DIFFERENT TRANSFORMER-BASED MODELS ON MAC DATASET

	MAC	
	Accuracy	F1
Baseline	91.27	90.48
DarijaBert	64.03	60.24
DarijaBert-mix	**92.70**	**91.18**
MARBERT	56.12	57.63
MorrBERT	90.72	90.67
MorRoBERTa	91.51	90.71

TABLE XI
AVERAGE METRICS OBTAINED BY EACH MODEL

	Average	
	Accuracy	F1
Baseline	90.73	90.58
DarijaBert	69.26	63.1
DarijaBert-mix	**94.96**	**94.6**
MARBERT	58.86	59.24
MorrBERT	93.58	93.3
MorRoBERTa	92.69	92.65

The bar chart in Fig. 1 reveals the F1 scores of each model across "Our Dataset," "MSDA," and "MAC," as well as the average across datasets. DarijaBert-mix continues to dominate, achieving the highest F1 scores across all datasets and an impressive average of 94.6%. MorrBERT performs consistently well, particularly on "Our Dataset" (95.13%), but slightly trails behind DarijaBert-mix in overall performance. In contrast, DarijaBert shows significant drops in F1 scores, with the lowest average of 63.1%. This suggests a potential decline in performance for specific datasets. MARBERT also struggles, with its average F1 score falling below 60%. The chart underscores the dominance of models like DarijaBert-mix and MorrBERT in precision-driven tasks.

The accuracy bar chart in Fig. 2 highlights a similar trend to the F1 scores. DarijaBert-mix achieves the highest accuracy across all datasets, peaking at 97.56% on "Our Dataset" and maintaining a strong average of 94.96%. MorrBERT follows closely, with consistent accuracy scores across all datasets, averaging at 93.58%. Meanwhile, MARBERT and DarijaBert exhibit much lower accuracies, with averages of 58.86% and 69.26%, respectively. Notably, MARBERT's performance appears the weakest on "MAC" (56.12%), indicating dataset-specific challenges. This chart reinforces the superiority of pre-trained models fine-tuned on relevant datasets. The poor performance of DarijaBert and MARBERT can be attributed to their training data, which predominantly consists of text scripted in Arabic letters. This limits their effectiveness in the current task of language identification, which requires distinguishing between different scripts.

In conclusion, our findings indicate that transfer-learning models consistently outperform both traditional machine learning and deep learning models in the task of language identification. This is evidenced by the notable increase in accuracy across almost all models tested on various datasets. The superior performance of these models highlights the effectiveness of transfer learning techniques in capturing the complexities of the Moroccan dialect, further validating their utility in language identification tasks.

VII. CONCLUSION

In this paper, we present a comparison study for LI of MD written on the social web using Roman and Arabic scripts. We resorted to different embedding algorithms, such as Word2vec and FastText, to create data vectors, and we trained different deep learning learners and transformers-based language models.

We utilized a corpus comprising comments from Moroccan YouTube videos, which we subsequently segmented and annotated into three categories: Latin Moroccan, Arabic Moroccan, and MSA. In addition to ensuring the corpus was

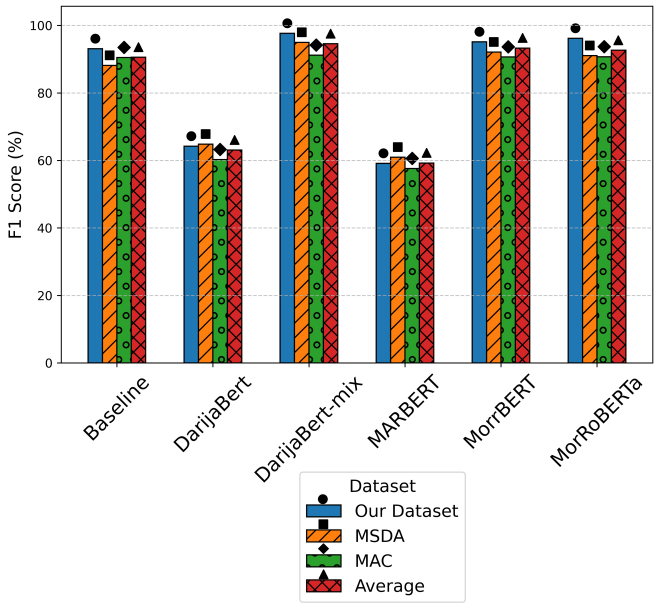

Fig. 1. F1-score across datasets

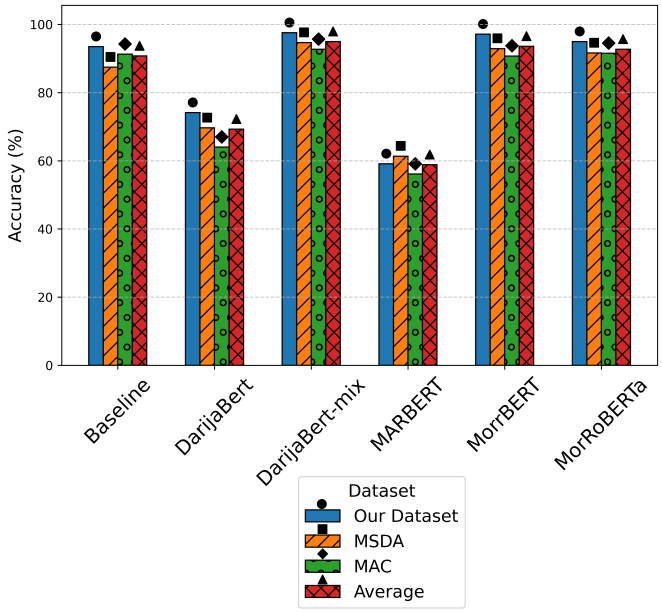

Fig. 2. Accuracy across datasets

well-balanced across each class for reliable classification, we aimed to contribute by creating a comprehensive dataset for the LI task of the MD, encompassing various writing scripts and styles.

The identification results were promising, with an overall accuracy of 97.56% using the DarijaBert-mix language model fine-tuned to our dataset. Some of the observed errors were related to the rarity of the words; others were due to the adopted segmentation method. The inaccuracy cases can be reduced by enlarging the corpus size to include a maximum vocabulary and improving the segmentation process, which will require a character-level study.

Recognizing and processing MD written in two scripting types—Arabic and Roman characters—is crucial for advancing NLP tasks in this dialect. Each scripting type carries unique linguistic nuances and cultural expressions that significantly affect text interpretation and sentiment analysis. By addressing both scripting types, NLP systems can better capture the diversity and richness of MD across digital platforms. This comprehensive approach enhances the accuracy of LI and sentiment analysis tasks. It fosters the development of more inclusive and practical NLP applications tailored to diverse user interactions and online content.

We aim to expand our future studies by exploring advanced identification approaches to generate substantial MD corpora enriched with dialectal content. Our goal includes developing lexicons and dictionaries tailored to this form of MD, enabling comprehensive research and the development of robust NLP tools. This effort will extend beyond different scripting types and Arabic, encompassing a broader range of languages.

References

[1] Injy Hamed, Moritz Zhu, Mohamed Elmahdy, Slim Abdennadher, and Ngoc Thang Vu (2019) Codeswitching Language Modeling With Bilingual Word Embeddings: A Case Study for Egyptian Arabic English, Proceedings of the 21st International Conference on Speech and Computer (SPECOM'19), Istanbul, Turkey, August 20-25, 2019

[2] Prajwol Shrestha. (2014). Incremental N-gram Approach for Language Identification in Code-Switched Text. In Proceedings of The First Workshop on Computational Approaches to Code Switching, pages 133–138, Doha, Qatar, October. Association for Computational Linguistics.

[3] Shruti Rijhwani, Royal Sequiera, Monojit Choudhury, Kalika Bali and Chandra Sekhar Maddila (2017). "Estimating Code-Switching on Twitter with a Novel Generalized Word-Level Language Detection Technique". The 55th Annual Meeting of the Association for Computational Linguistics (ACL).

[4] Soliman, A. B., Eissa, K., & El-Beltagy, S. R. (2017). AraVec: A set of Arabic Word Embedding Models for use in Arabic NLP. Procedia Computer Science, 117, 256–265. https://doi.org/10.1016/j.procs.2017.10.117

[5] Laoudi, J., Bonial, C., Donatelli, L., Tratz, S., & Voss, C. (2018). Towards a Computational Lexicon for Moroccan Darija: Words, Idioms, and Constructions. ACL Anthology, 74–85. Retrieved from https://www.aclweb.org/anthology/W18-4910/

[6] Samih, Y., & Maier, W. (2016). An Arabic-Moroccan Darija Code Switched Corpus. LREC 2016.

[7] Al-Badrashiny M. and Diab M., "LILI: A Simple Language Independent Approach for Language Identification," in Proceedings of COLING 26th International Conference on Computational Linguistics: Technical Papers, Osaka, pp. 1211-1219, 2016.

[8] Zaidan O. and Callison-Burch C., "Arabic Dialect Identification," Computational Linguistics, vol. 40, no. 1, pp. 171-202, 2014.

[9] Elfardy H., Al-Badrashiny M., and Diab M., "AIDA: Identifying Code Switching in Informal Arabic Text," in Proceedings of 1st Workshop on Computational Approaches to Code Switching, Doha, pp. 94-101, 2014.

[10] Eskander R., Al-Badrashiny M., Habash N., and Rambow O., "Foreign Words and the Automatic Processing of Arabic Social Media Text Written in Roman Script," in Proceedings of 1st Workshop on Computational Approaches to Code Switching, Doha, pp. 1-12, 2014.

[11] Solorio T., Blair E., Maharjan S., Bethard S., Diab M., Gohneim M., Hawwari A., AlGhamdi F., Hirschberg J., Chang A., and Fung P., "Overview for the First Shared Task on Language Identification in Code-Switched Data," in Proceedings of 1st Workshop on Computational Approaches to Code Switching, Doha, pp. 62-72, 2014.

[12] Shrestha P., "Incremental N-gram Approach for Language Identification in Code-Switched Text," in Proceedings of 1st Workshop on Computational Approaches to Code Switching, Doha, pp. 13-138, 2014.

[13] Molina G., Rey-Villamizar N., Solorio T., AlGhamdi F., Ghoneim M., Hawwari A., and Diab M., "Overview for the Second Shared Task on Language Identification in Code-Switched Data," in Proceedings of 2nd Workshop on Computational Approaches to Code Switching, Austin, pp. 40-49, 2016.

[14] Chanda A., Das D., and Mazumdar C., "Columbia-Jadavpur submission for EMNLP 2016 Code-Switching Workshop Shared Task: System Description," in Proceedings of the 2nd Workshop on Computational Approaches to Code Switching, Austin, pp.112-115, 2016.

[15] Shirvani R., Piergallini M., Gautam G., and Chouikha M., "The Howard University System Submission for the Shared Task in Language Identification in Spanish-English Codeswitching," in Proceedings of 2 nd Workshop on Computational Approaches to Code Switching, Austin, pp. 116-120, 2016.

[16] Jhamtani H., Kumar B., and Raychoudhury V., "Word-level Language Identification in Bilingual Code-switched Texts," in Proceedings of 28th Pacific Asia Conference on Language, Information and Computation, Phuket, pp. 348- 357, 2014.

[17] Nguyen D. and Cornips L., "Automatic Detection of Intra-Word Code-Switching," in Proceedings of 14th Annual SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology, Berlin, pp. 82-86, 2016.

[18] Guellil I. and Azouaou F., "Arabic Dialect Identification with an Unsupervised Learning (based on a lexicon) Application case: ALGERIAN Dialect," in Proceedings of IEEE International Conference on Computational Science and Engineering, IEEE International Conference on Embedded and Ubiquitous Computing, and International Symposium on Distributed Computing and Applications to Business, Engineering and Science, AnYang, pp. 724-731, 2016.

[19] Giwa O. and Davel M., "N-Gram based Language Identification of Individual Words," in Proceedings of Conference: Pattern Recognition Association of South Africa, Johannesburg, pp. 1- 22, 2013.

[20] Sadat F., Kazemi F., and Farzindar A., "Automatic Identification of Arabic Language Varieties and Dialects in Social Media," in Proceedings of 2 nd Workshop on Natural Language Processing for social media, Dublin, pp. 22-27, 2014

[21] Chanda A., Das D., and Mazumdar C.,"Unraveling the English-Bengali Code Mixing Phenomenon," in Proceedings of 2nd Workshop on Computational Approaches to Code Switching, Austin, pp. 80-89, 2016.

[22] Bar K. and Dershowitz N., "The Tel Aviv University System for the Code-Switching Workshop Shared Task," in Proceedings of 1st Workshop on Computational Approaches to Code Switching, Doha, pp. 139-143, 2014.

[23] Barman U., Das A., Wagner J., and Foster J., "Code Mixing: A Challenge for Language Identification in the Language of Social Media,"in Proceedings of 1st Workshop on Computational Approaches to Code Switching, Doha, pp. 13-23, 2014.

[24] Barman U., Wagner J., Chrupala G., and Foster J., "DCU-UVT: Word-Level Language Classification with Code-Mixed Data," in Proceedings of 1st Workshop on Computational Approaches to Code Switching, Doha, pp. 127- 132, 2014.

[25] Chittaranjan G., Vyas Y., Bali K., and Choudhury M., "Word-level Language Identification using CRF: Code-switching Shared Task Report of MSR India System," in Proceedings of 1st Workshop on Computational Approaches to Code Switching, Doha, pp. 73-79, 2014.

[26] King L., Baucom E., Gilmanov T., Kübler S.,Whyatt D., Maier W., and Rodrigues P., "The IUCL+ System: Word-Level Language Identification via Extended Markov Models," in Proceedings of 1st Workshop on Computational Approaches to Code Switching, Doha, pp. 102-106, 2014.

[27] Lin C., Ammar W., Levin L., and Dyer C., "The CMU Submission for the Shared Task on Language Identification in Code-Switched Data,"in Proceedings of 1st Workshop on Computational Approaches to Code Switching, Doha, pp. 80-86, 2014.

[28] D. Singh and S. Shekhar, "An Architectural Framework for Word level Language Identification in Mixed Script Text," 2023 6th International Conference on Information Systems and Computer Networks (ISCON), Mathura, India, 2023, pp. 1-5, doi: 10.1109/ISCON57294.2023.10112136.

[29] Shrestha P., "Codeswitching Detection via Lexical Features using Conditional Random Fields," in Proceedings of 2nd Workshop on Computational Approaches to Code Switching, Austin, pp. 121-126, 2016.

[30] Samih Y. and Maier W., "Detecting CodeSwitching in Moroccan Arabic Social Media," in Proceedings of 4th International Workshop on Natural Language Processing for social media Social NLP, New York, 2016.

[31] Schulz S. and Keller M., "Code-Switching Ubique Est - Language Identification and Part-ofSpeech Tagging for Historical Mixed Text," in Proceedings of 10th SIGHUM Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities, Berlin, pp. 43-51, 2016.

[32] Al-Badrashiny M. and Diab M., "LILI: A Simple Language Independent Approach for Language Identification," in Proceedings of COLING 26th International Conference on Computational Linguistics: Technical Papers, Osaka, pp. 1211- 1219, 2016.

[33] Dongen N., Analysis and Prediction of DutchEnglish Code-switching in Dutch Social Media Messages, Master's Thesis, Universiteit van Amsterdam, 2017.

[34] Rijhwani S., Sequeira R., Choudhury M., Bali K., and Maddila C., "Estimating Code-Switching on Twitter with a Novel Generalized Word-Level Language Detection Technique," in Proceedings of 55th Annual Meeting of the Association for Computational Linguistics, Vancouver, pp. 1971-1982, 2017.

[35] Aridhi C., Achour H., Souissi E., and Younes J., "Word-Level Identification of Romanized Tunisian Dialect," in Proceedings of International Conference on Applications of Natural Language to Information Systems, Liège, pp. 170-175, 2017.

[36] Salameh M., Bouamor H., and Habash N., "FineGrained Arabic Dialect Identification," in Proceedings of 27th International Conference Computational Linguistics, Santa Fe, pp. 1332- 1344, 2018.

[37] Lichouri M., Abbasa M., Freihatc A., and Megtoufa D., "Word-Level vs Sentence-Level Language Identification: Application to Algerian and Arabic Dialects," Procedia Computer Science, vol. 142, pp. 246-253, 2018.

[38] Heafield K., "KenLM: Faster and Smaller Language Model Queries," in Proceedings of 6th Workshop on Statistical Machine Translation, Edinburgh, pp. 187-197, 2011.

[39] Chang J. and Lin C., "Recurrent-neural-network for Language Detection on Twitter CodeSwitching Corpus," arXiv preprint, arXiv:1412.4314, pp. 1-9, 2014.

[40] Jaech A., Mulcaire G., Hathi S., Ostendorf M., and Smith N., "A Neural Model for Language Identification in Code-Switched Tweets," in Proceedings of 2nd Workshop on Computational Approaches to Code Switching, Austin, pp. 60-64, 2016.

[41] Samih Y., Maharjan S., Attia M., Kallmeyer L., and Solorio T., "Multilingual Codeswitching Identification via LSTM Recurrent Neural Networks," in Proceedings of 2nd Workshop on Computational Approaches to Code Switching, Austin, pp. 50-59, 2016.

[42] Mave D., Maharjan S., and Solorio T., "Language Identification and Analysis of CodeSwitched Social Media Text," in Proceedings of 3rd Workshop on Computational Approaches to Code-Switching, Melbourne, pp. 51-61, 2018.

[43] Mager M., Çetinoğlu Ö., and Kann K., "Subword-Level Language Identification for Intra-Word Code-Switching," Ground AI, vol. 1, 2019.

[44] Elaraby M. and Abdul-Mageed M., "Deep Models for Arabic Dialect Identification on Benchmarked Data," in Proceedings of 5th Workshop on NLP for Similar Languages, Varieties and Dialects, Santa Fe, pp. 263-274,2018.

[45] Sayadi K., Hamidi M., Bui M., Liwicki M., and Fischer A., "Character-Level Dialect Identification in Arabic Using Long Short-Term Memory," in Proceedings of International Conference on Computational Linguistics and Intelligent Text Processing, Budapest, pp. 324-337, 2017.

[46] Bouamor H., Habash N., and Oflazer K., "A Multidialectal Parallel Corpus of Arabic," in Proceedings of 9th International Conference on Language Resources and Evaluation, Reykjavik, pp. 1240-1245, 2014.

[47] Sikdar U. and Gambäck B., "Language Identification in Code-Switched Text Using Conditional Random Fields and Babelnet," in Proceedings of 2nd Workshop on Computational Approaches to Code Switching, Austin, pp. 127-131, 2016.

[48] Radim Rehurek, Petr Sojka (2010). Software Framework for Topic Modelling with Large Corpora. In Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks (pp. 45–50). ELRA.

[49] Mikolov, T., Chen, K., Corrado, G., Dean, J. (2013). Efficient Estimation of Word Representations in Vector SpacearXiv e-prints, arXiv: 1301.3781.

[50] Bojanowski, P., Grave, E., Joulin, A., Mikolov, T. (2016). Enriching Word Vectors with Subword Information arXiv preprint arXiv: 1607.04606.

[51] V. Vapnik, The nature of statistical learning theory. New York: Springer-Verlag, 1995. https://doi.org/10.1007/978-1-4757-3264-1

[52] N. Cristianini, J. Shawe-Taylor. An Introduction to Support Vector Machines, Cambridge: Cambridge University Press, 2000.

[53] J. A. K. Suykens, "Nonlinear modeling and support vector machines", Proceeding of IEEE Instrumentation and measurement technology, Budapest, 2001.

[54] J. A. K.Suykens, L. Lukas, J. Vandewalle, "Sparse approximation using least squares support vector machines". IEEE International symposium on circuits and systems. Geneva 2000.

[55] Y. LeCun, J. Denker, D. Henderson, R. Howard, W. Hubbard, and L. Jackel. Handwritten Digit Recognition with a Back-Propagation Network. Advances in Neural Information Processing Systems, 1990.

[56] A. Krizhevsky, I. Sutskever, and G. Hinton. ImageNet Classification with Deep Convolutional NeuralNetworks. Advances in Neural Information Processing Systems, 2012.

[57] Teja, T.R., Shilpa, S., Joseph, N. (2023). Meta Embeddings for LinCE Dataset. In: Shakya, S., Balas, V.E., Haoxiang, W. (eds) Proceedings of Third International Conference on Sustainable Expert Systems . Lecture Notes in Networks and Systems, vol 587. Springer, Singapore. https://doi.org/10.1007/978-981-19-7874-6_26

[58] E. Raazia, A. Bibi and M. U. Arshad, "Word-level Language Identification and Localization in Code-Mixed Urdu-English Text," 2022 16th International Conference on Open Source Systems and Technologies (ICOSST), Lahore, Pakistan, 2022, pp. 1-6, doi: 10.1109/ICOSST57195.2022.10016848.

[59] Hidayatullah, Ahmad Fathan, et al. "Corpus creation and language identification for code-mixed Indonesian-Javanese-English Tweets." PeerJ Computer Science 9 (2023): e1312.

[60] Alqurashi, T. Applying a Character-Level Model to a Short Arabic Dialect Sentence: A Saudi Dialect as a Case Study. Appl. Sci. 2022, 12, 12435. https://doi.org/10.3390/app122312435

[61] Amal Alghamdi, Areej Alshutayri, and Basma Alharbi. 2023. Deep Bidirectional Transformers for Arabic Dialect Identification. In Proceedings of the 6th International Conference on Future Networks & Distributed Systems (ICFNDS '22). Association for Computing Machinery, New York, NY, USA, 265–272. https://doi.org/10.1145/3584202.3584243

[62] Goswami, Koustava, et al. "Unsupervised deep language and dialect identification for short texts." Proceedings of the 28th International Conference on Computational Linguistics. 2020.

[63] Boujou, E., Chataoui, H., Mekki, A.E., Benjelloun, S., Chairi, I., Berrada, I.: An open access nlp dataset for arabic dialects: Data collection, labeling, and model construction. preprint arXiv:2102.11000 (2021)

[64] Jbel, M., Hafidi, I., Metrane, A. (2023). MYC: A Moroccan Corpus for Sentiment Analysis. In: Aboutabit, N., Lazaar, M., Hafidi, I. (eds) Advances in Machine Intelligence and Computer Science Applications. ICMICSA 2022. Lecture Notes in Networks and Systems, vol 656. Springer, Cham. https://doi.org/10.1007/978-3-031-29313-9_6

[65] Bouamor, H., Habash, N., Salameh, M.,Zaghouani, W., Rambow, O., Abdulrahim, D.,Obeid, O., Khalifa, S., Eryani, F., Erdmann,A., et al.: The madar arabic dialect corpus and lexicon. In: LREC (2018)

[66] Shazal, Ali, Aiza Usman, and Nizar Habash. "A unified model for Arabizi detection and transliteration using sequence-to-sequence models." Proceedings of the fifth arabic natural language processing workshop. 2020.

[67] Abdelali, A., Mubarak, H., Samih, Y., Hassan, S. and Darwish, K., 2021, April. QADI: Arabic dialect identification in the wild. In Proceedings of the sixth Arabic natural language processing workshop (pp. 1-10). https://aclanthology.org/2021.wanlp-1.1

[68] Language identification dataset for Moroccan dialect https://github.com/MouadJb/LID

[69] Gaanoun, K., Naira, A. M., Allak, A., & Benelallam, I. (2024). Darijabert: a step forward in nlp for the written moroccan dialect. International Journal of Data Science and Analytics, 1-13. https://doi.org/10.1007/s41060-023-00498-2

[70] Abdul-Mageed, M., Elmadany, A., & Nagoudi, E. M. B. (2020). ARBERT & MARBERT: Deep bidirectional transformers for Arabic. arXiv preprint arXiv:2101.01785.

[71] Moussaoui, O., & El Younnoussi, Y. (2023, June). Pre-training Two BERT-Like Models for Moroccan Dialect: MorRoBERTa and MorrBERT. In MENDEL (Vol. 29, No. 1, pp. 55-61). https://doi.org/10.13164/mendel.2023.1.055

[72] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... & Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692.

[73] Garouani, Moncef, and Jamal Kharroubi. "MAC: an open and free Moroccan Arabic Corpus for sentiment analysis." The Proceedings of the International Conference on Smart City Applications. Cham: Springer International Publishing, 2021. https://doi.org/10.1007/978-3-030-94191-8_68.