Automatic Arabic Dialect Identification Systems for Written Texts: A Survey

Maha Jarallah Althobaiti *maha.j@tu.edu.sa*
Department of Computer Science
Taif University
Taif, 21944-11099, Saudi Arabia

Abstract

Arabic dialect identification is a specific task of natural language processing, aiming to automatically predict the Arabic dialect of a given text. Arabic dialect identification is the first step in various natural language processing applications such as machine translation, multilingual text-to-speech synthesis, and cross-language text generation. Therefore, in the last decade, interest has increased in addressing the problem of Arabic dialect identification. In this paper, we present a comprehensive survey of Arabic dialect identification research in written texts. We first define the problem and its challenges. Then, the survey extensively discusses in a critical manner many aspects related to Arabic dialect identification task. So, we review the traditional machine learning methods, deep learning architectures, and complex learning approaches to Arabic dialect identification. We also detail the features and techniques for feature representations used to train the proposed systems. Moreover, we illustrate the taxonomy of Arabic dialects studied in the literature, the various levels of text processing at which Arabic dialect identification is conducted (e.g., token, sentence, and document level), as well as the available annotated resources, including evaluation benchmark corpora. Open challenges and issues are discussed at the end of the survey.

Keywords: Arabic Dialect Identification, Traditional Machine Learning, Deep Learning, Feature Engineering Techniques, Benchmark Corpora, Arabic Natural Language Processing.

1. INTRODUCTION

The current era of intelligent language systems that perform many functions (e.g., machine translation, social media analysis, and marketing) creates the necessity to computationally handle texts of different domains (e.g., news, blogs, Twitter messages, and customer reviews) and topics (e.g., economy, politics, sports, and science). Arabic text processing is one of the challenges that faces the researchers and developers of computational linguistics due to many factors. The first of these factors is that Arabic language generally refers to a collection of varieties with morphological, syntactic, lexical, and phonological differences [1]. These varieties include a standardized form, Modern Standard Arabic (MSA), and many non-standardized regional dialects [2], [3].

The MSA is mostly written and not spoken while regional Arabic dialects are mainly spoken. Nevertheless, the regional dialects started to appear in a text form in the new millennium with the rise of Web 2.0, which allowed websites to have interactive contents generated by users (e.g., social media posts, blogs, emails, discussion forums). The online Arabic texts are less controlled, more speech-like, and usually written in an informal manner using colloquial dialects [4]. They are usually inconsistent, since Arabic dialects lack orthographic standards. Moreover, the Linguistic Code Switching (LCS) phenomenon appears in online Arabic content. That is, the writer of online texts sometimes switches between MSA and at least one Arabic dialect within the same utterance [5]. This makes processing the Arabic online texts computationally a challenging issue that should be addressed when building models for different Arabic Natural Language Processing (NLP) tasks. Moreover, many studies [6] have reported that tools built specifically for MSA resulted in significantly lower performance when applied to texts of Arabic dialects due to the significant linguistic differences between MSA and dialects. Therefore, more attention has recently been given to computational approaches to processing the texts written in Arabic dialects [7]–[10].

Arabic Dialect Identification (ADI) in written texts is an active NLP task aiming to automatically identify the Arabic dialect of given texts. The availability of accurate Arabic dialect identification models can be of great benefit to many Arabic NLP tasks, such as statistical machine translation, and building dialect-to-dialect lexicons. The problem of ADI in written texts received a great deal of attention from researchers in the last decade, whereby computational approaches and system architectures have been developed to enhance ADI [6], [11]–[15], as shown in Figure 1. Therefore, there is a need to shed light on the currently functioning ADI. The purpose of this paper is to provide an informative survey of studies on ADI including available lexical resources, common benchmarks for evaluation, used features, adopted learning methods, implemented system architectures, and the considered Arabic dialects. The paper also discusses the current and potential applications and tools of ADI without forgetting outstanding issues and challenges, as well as the future of the ADI in written texts. We presented a comprehensive survey of the available ADI proposed methods in the literature, focusing on their findings. We analyzed our observations on these findings in an inductive way in order to direct new ADI research efforts to close the gaps and focus on challenging cases in ADI problems.

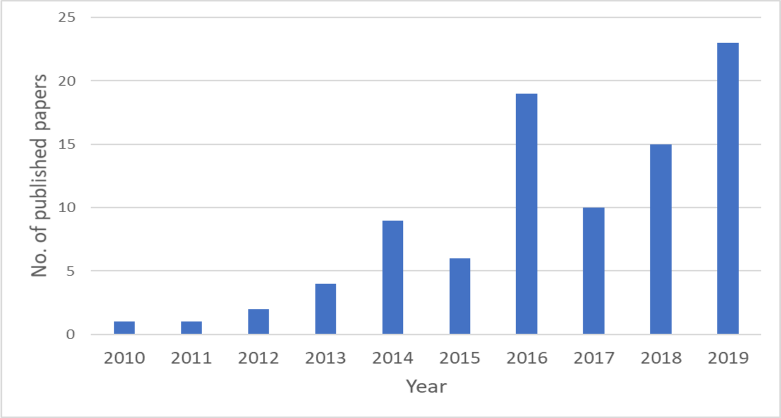

FIGURE 1: Approximate number of papers published every year on.ADI

This paper is a survey of ADI research in written texts and does not mention any work that focuses on speech alone. However, our paper references the studies conducted on speech transcripts and that utilize acoustic features in addition to lexical features. The remainder of the paper is organized as follows: Section 2 covers the Arabic dialects including and their categories. Section 3 presents the Arabic dialect identification task by defining the problem of ADI in written texts and then discussing the challenges and issues at hand. Section 4 covers the available Arabic dialect identification corpora required to build ADI models, as well as the common benchmarks used in the literature to evaluate the models. Section 5 presents the evaluation metrics utilized in the literature, shared tasks and ADI evaluation campaigns. Section 6 discusses the ADI studies, covering several aspects such as adopted features, utilized learning methods, components of the ADI systems, and the considered Arabic dialects. The section also incorporates qualitative comparison between existing works using the common benchmarks. Lastly, open issues and the future of Arabic dialect identification are presented in Section 7.

2. ARABIC LANGUAGE AND ARABIC DIALECTS

'Arabic language' refers to the collection of historically related varieties. These varieties are Modern Standard Arabic (MSA) and informal spoken dialects. MSA is the official language of the Arab World and the primary language of the culture and education system. MSA is mostly written, not spoken. The informal spoken dialects, on the other hand, are the medium of communication in daily life, even on the radio and television shows, from soap operas to music videos. These dialects are primarily spoken, not written, and seen as true native language [2], [4], [6], [16]. The sea of Arabic dialects is vast, with dialects being spoken by more than 300 million native speakers. These dialects can be categorized into various groups based on variety of factors. The most popular factor to categorize the Arabic dialects is based on geographical location. According to the Glottolog [17], a bibliographic database of the world's languages and language families, Arabic dialects are classified into five groups, composing of 38 dialects as shown in Figure 2. These dialectal groups are Arabian Peninsula Arabic, Eastern Arabic, Egyptic Arabic, Levantine Arabic, and North African Arabic. According to the annual reference on the languages of the world (Ethnologue) in its twenty-third edition [18], there are 36 Arabic dialects.

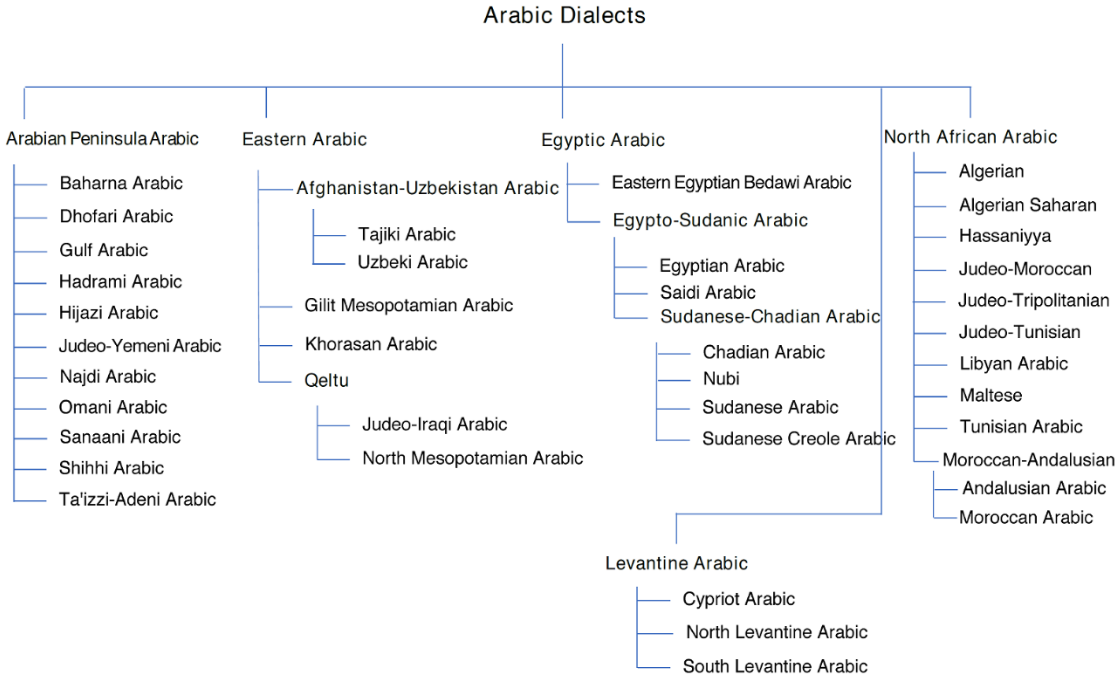

FIGURE 2: Classification tree of Arabic dialects according to the Glottolog database.

Until now the primary form of day to day communication in the Arab world has been Arabic dialects, however, the use of dialectal Arabic is evolving due to the increasing prevalence of the Web as a platform for collaborative and community-based sites such as social networking sites, blogs, forums, and reader commentary. This has created a domain where both MSA and Arabic dialects are used relatively equally in written communication [19]. Most Arabic dialects use Arabic script in their writing system. Approximately 22 out of 36 Arabic dialects mentioned in the annual reference “Ethnologue” are currently using Arabic script in the writing process. Arabic script is the second most widely used writing system worldwide after Latin script [2], [20], [21].

A set consisting of the Latin alphabet, numeric digits, digraph, and symbols like apostrophe is used sometimes informally to write Arabic texts on computers and mobile devices, especially when the keyboard does not support the Arabic script. This is known as Arabish, Araby, Arabizi, and Franco-Arabic [22]. The numeric digits, digraph, and apostrophe are used to represent Arabic phonemes that cannot be represented using Latin script. Arabizi actually gained huge popularity 15 years ago among Arab youth in instant messaging conversations and it is rarely used for lengthy communications [23]. There are many ways to represent one Arabic letter depending on local dialects [22]. Table 1 shows some examples of Arabic letters, which do not have exact equivalent sounds in Latin alphabet, and how they are written in Arabizi system.

Arabic Letter	International Phonetic Alphabet (IPA)	Arabizi Possibilities
ح	/ħ/	7, H, h
خ	/x/	kh, 7', 5
غ	/ɣ/	3', gh

TABLE 1: Some Arabic Letters with their Arabizi counterparts.

3. PROBLEM OF ARABIC DIALECT IDENTIFICATION

3.1 Definition

Arabic dialect identification in written texts can be defined as the process of building a recognizer that is able to, given an Arabic piece of text *T* (e.g., sentence, paragraph, document), determine whether or not *T* was written in dialectal Arabic and in which dialect T was written. Arabic dialect identification is a most challenging language identification task [6], [24]. That is, language identification can be considered a solved problem, especially when building a system to discriminate between languages that have little features in common. In the case of Arabic dialects, the situation is more complex since the dialects are closely related to each other and share much vocabulary [24]. The following section details the issues and challenges embedded in the task of Arabic dialect identification.

3.2 Issues and Challenges

Arabic dialect identification research has yet to overcome numerous challenges associated with discriminating between Arabic dialects; the newer investigations conducted; the more problems will be solved. Some of these difficulties are:

- Arabic speakers mix Arabic varieties in different ways. They may switch between two or more Arabic varieties within the same utterance. This phenomenon is called Linguistic Code Switching (LCS) [25]. The LCS phenomenon is present in the online Arabic content. That is, the writer of online texts sometimes alternates between MSA and at least one Arabic dialect within the same paragraph or even a sentence. It is a very challenging task to identify LCS points whether at a token or sentence level [5], [26].
- The letters of Arabic scripts and the absence of critical marks hide the vocalic and some consonantal differences across dialects. For example, The Arabic letter (ق, *qAf*) that is pronounced /q/ is pronounced differently across dialects (e.g., /g/, /q/, /ʔ/, /k/, /ʤ/). Relying on the written form of (ق, *qAf*), we run the risk of mistakenly identifying the writer's dialect, since the written letter in Arabic script does not reveal the writer's pronunciation of it. The same situation is true for the Arabic letter (ظ, Ď) which is pronounced /ðˤ/, but has a variety of pronunciations in different dialects (e.g., / ðˤ/ and /zˤ/). Simply put, the Arabic script may mask the real sounds and pronunciations of letters and words in the written texts of different dialects [27]–[30].
- Arabic dialects differ from MSA with regard to the vowel system. The vowel systems also differ substantially from one dialect to another. For example, many short vowels were deleted due to syncope, as in the pronunciation of the MSA word (جبل, *jbl*, “mountain")[1] /ʤabal/ becomes in Moroccan dialect /ʑbəl/. It is evident that Arabic texts, which are most often written without diacritical marks (i.e., short vowels), do not represent these spoken differences in the vowel system [30], [31]. Moreover, the omission of short vowels results in dialectal words that share the same spelling with MSA words, but mean something entirely different [6]. For example, the Levantine word without dialectical marks (هون, *hwn*, “here") could be mistaken for a MSA that means (هون, *hwn*, “make easy").
- There have been many attempts to adapt the Arabic script by adding new letters or dots to the existing letters to account for the new sounds and pronunciations that do not exist in its phonemic inventory [32]. For example, in the written text of the Iraqi dialect, the Persian kAf (گ) has been used to represent the /g/ sound, which is similar to the /g/ in the English word

[1] Throughout the entire paper, Arabic words are represented as follows: (Arabic word, HSB transliteration, “English gloss”)

"good". The written Levantine texts usually use the new letter (چ) to represent the pronunciation /g/. In addition, loan words from foreign languages such as English and French have introduced new sounds (e.g., /v/, /p/, /ng/) which led to many attempts to add new letters to the standard Arabic script to represent the precise pronunciation of these letters when they appear in a dialectal Arabic text. This approach of adding new letters for new phonemes to explain the real pronunciations actually creates more orthographic variations between dialects, as shown in Table 2, where the phoneme (e.g., /g/) is transliterated using different letters in different dialects. There are also cases where a letter (e.g., چ) is used to represent different sounds for different dialects [6], [33].

- The majority of the time, Arabic script is used to write dialectal Arabic. Online contents that are generated informally by users, however, are sometimes written in Arabizi, a non-standard romanization consisting of Latin characters, numeric digits, and symbols like the apostrophe. There are various ways in which Arabic letters are transliterated. That is, Arabic letters that do not have similar phonetic approximations in the Latin alphabet are often expressed using numeric digits or a combination of two Latin characters. Arabizi makes Arabic dialect identification based on written texts a challenging task; transliteration in Arabizi follows no standards or rules which creates inconsistency and ambiguity [34], [35].
- A considerable amount of previous research in Arabic NLP has focused on MSA. Therefore, a large number of annotated corpora and freely available resources are built for MSA. Recently, the dialectal Arabic has started to gain the focus of researchers and some resources have been built [1], [6], [36]. The resources available for dialectal Arabic in comparison with MSA, however, let alone resources for other languages such as English, are still severely limited in terms of size and coverage. Therefore, prior research on Arabic NLP that deals with online contents and social media texts have created their own annotated resources to fill in the gap. More efforts are required to build more annotated resources for Arabic dialects in order to develop the computational solutions for dialectal Arabic problems including Arabic dialect identification [6].
- The vocabulary varies widely between dialects. Not only the same entity can be called different names in different dialects, sometimes the same word can convey a totally different meaning in two different dialects [16]. For example, the word (ماشي, mA\^sy) means "OK" in Levantine and Egyptian dialects, but it means "not" in Moroccan dialect. The word (برّاد, brrAd) means "kettle" in Egyptian dialects, but it means "fridge" in Levantine dialects.

		Dialects		
		Iraqi	Levantine	Moroccan
phonemes	/g/	گ	چ	ڭ
	/tʃ/	چ	تش	ٿش
	/v/	ڤ	ڤ	ڥ
	/p/	پ	پ	پ

TABLE 2: Attempts to add letters to the Arabic script in dialectal

4. ARABIC DIALECT IDENTIFICATION CORPORA

4.1 Manually Annotated ADI Corpora

The Arabic Online Commentary (AOC) dataset is one of the earliest ADI corpora publicly available, which designates it a benchmark dataset for later studies. The AOC has been compiled from online commentary by readers of online versions of three Arabic newspapers: AlGhad (Jordanian newspaper), AlRiyadh (Saudi newspaper), and AlYoum AlSabe' (Egyptian newspaper). The AOC dataset contains 108K sentences, of which 63,555 sentences are MSA, and the remaining represent three dialects: Egyptian, Gulf, and Levantine. The annotation process has been conducted by Amazon's Mechanical Turk (MTurk) workers [11]. The AOC dataset has been widely used as a benchmark dataset to evaluate various ADI techniques in the literature and to compare their performances [5], [26], [37]–[43]. Following the work of [11], Cotterell and Callison-Burch [37] built Extended AOC dataset consisting of 27,239 user comments from online newspapers. However, the corpus covered two more dialects (Maghrebi, and Iraqi dialects) in addition to the three covered by Zaidan & Callison-Burch (2011) (Levantine, Gulf, and Egyptian). They also created Twitter corpus consisting of 40,229 tweets from the five aforementioned dialects. The data of the corpora was manually collected and annotated by workers from MTurk. Additionally, McNeil (2018) created the Tunisian Arabic Corpus (TAC), consisting of 895,000 words collected from three sources: (a) traditional written sources such as song lyrics and folklore (b) new written sources such as blogs and discussion forums, (c) transcriptions of audio sources such as radio broadcasts. The TAC data was collected, identified, checked, and transcribed when required by Tunisian college-level students and workers from MTurk. The corpus is only publicly available through a web-based interface in which a user can search for a word. It is not publicly available for downloading.

The Multidialectal Parallel Corpus of Arabic (MPCA) contains 2,000 parallel sentences, covering five Arabic dialects: Egyptian, Tunisian, Jordanian, Palestinian, and Syrian Arabic. The Egyptian sentences were selected from the Egyptian portion of the Egyptian-English corpus built by [10]. Then, non-professional translators hired on MTurk were asked to translate the Egyptian sentences into their native dialect [12]. The MPCA corpus is publicly available upon request. The Dial2MSA parallel corpus created by Mubarak [44] contains dialectal Arabic tweets in four main Arabic dialects (Egyptian, Maghrebi, Levantine, and Gulf) and their corresponding MSA translations. The tweets were first collected using Twitter API, and then filtered using a set of distinctive words for each dialect. The crowdsourcing platform (CrowdFlower) was then used to hire native speakers of each dialect in order to translate each tweet into its corresponding MSA. The final corpus contains 16,000 pairs for Egyptian-MSA, 8,000 pairs for Maghrebi-MSA, and 18,000 pairs for each Gulf-MSA and Levantine-MSA. Furthermore, the PADIC, a Parallel Arabic DIalect Corpus, was created from recorded conversations from everyday life, movies, and TV shows of Annaba's dialect, spoken in the east of Algeria, and Algiers's dialect. The PADIC corpus is publicly available. The sentences were manually transcribed and translated by native speakers into MSA as well as three more Arabic dialects: Sfax's dialect spoken in the south of Tunisia, Syrian and Palestinian dialects. The total number of parallel sentences in the corpus is 6,400. The MADAR travel domain corpus presented by Bouamor et al., [45] was used in the MADAR shared task on Arabic fine-grained dialect identification. It is a large-scale collection of parallel sentences covering the dialects of 25 Arab cities, in addition to English, French and MSA. The MADAR corpus is a commissioned translation of selected sentences from the Basic Traveling Expression Corpus (BTEC) [46] in English and French to the dialects of 25 Arab cities. The BTEC is a multilingual spoken language corpus containing tourism-related sentences. The MADAR corpus consists of two parts: the first called “Corpus-26'' has 2,000 sentences translated into 25 Arab city dialects in parallel as well as MSA, and the second one called “Corpus-6” consists of 10,000 additional new sentences from the BTEC corpus translated into only five selected Arab city dialects, plus MSA. The MADAR Twitter corpus was used in the MADAR Twitter user dialect identification subtask that was organized as part of the MADAR shared task on Arabic fine-grained dialect identification [15]. This corpus contains 2,980 Twitter user profiles from 21 different countries.

The LICSD'2014 and LICSD'2016 are two datasets manually annotated at token level and provided by the shared tasks for Language Identification in Code Switched Data (LICSD) in 2014 and 2016 [47], [48]. The LICSD-2014 includes code-switched data from Modern Standard Arabic-Egyptian dialect (MSA-EGY) pair. The data for the MSA-EGY variety pair was compiled from Twitter and online reader commentaries. They harvested 9,947 tweets and 6,723 commentaries (half MSA and half Egyptian) from an Egyptian newspaper provided by the Arabic Online Commentary (AOC) Dataset. The LICSD-16 dataset created by Molina et al. (2016) includes code-switched data from MSA-EGY pair. They published 11,241 tweets (8,862 tweets for the training set, and 1,117 tweets for the development set, 1,262 tweets for the test set). Many studies in ADI literature have used the LICSD-2014 and LICSD-2016 corpora to evaluate their token-level ADI systems [39], [49].

The VarDial'2016 ADI corpus was released by the organizers of the Discriminating between Similar Languages (DSL) shared task in the VarDial'2016 workshop [50]. The subtask 2 managed the Arabic dialect identification in speech transcripts. The utilized corpus is based on Arabic transcribed speeches presented by Ali et al. (2016), containing 9159 sentences and covering four Arabic dialects: Egyptian, Gulf, Levantine, and North African, as well as MSA. The VarDial'2017 ADI corpus was provided for the second edition of the ADI shared task of the 2017 VarDial Evaluation Campaign on Natural Language Processing (NLP) for Similar Languages, Varieties and Dialects [51]. It is similar to the VarDial'2016 corpus as they are both based on speech transcripts and cover the same Arabic dialects. However, the VarDial'2017 dataset provides acoustic features to the task participants [51]. The VarDial'2017 corpus contains 13,825 samples, the development set contains 1,524 samples, and the test contains another 1,492 samples. Then, organizers of the VarDial Evaluation Campaign 2018 released the data of the third edition of the ADI task VarDial'2018 ADI corpus [52]. The data for training and development were the same data released in the 2017 edition of the ADI task. Regarding testing, two datasets were prepared: an in-domain test set from broadcast news and an out-of-domain dataset from YouTube, containing 5,345 samples.

4.2 Semi and Fully Automatically Annotated ADI Corpora

Huang utilized semi-supervised learning for automatic ADI annotations [40]. The study examined two semi-supervised methods: self-training and co-training. In self-training, a strongly supervised classifier trained on a small amount of gold labeled data (AOC corpus) was used to label a large amount of data (646M words) extracted from Facebook posts. The co-training method employed two classifiers to annotate the same unlabeled data. Then, only the sentences on which the two classifiers agree (they have the same predictions) were added to the final corpus (476M words).

Some studies classify the collected data into a set of dialects using a list of distinctive regional words and phrases [53], [54]. These words are considered “seed words” in the process of automatically collecting and annotating the data for each dialect. In addition to the use of distinctive regional words, the geographical locations provided in users' profiles on many social media sites such as Facebook, Twitter, and YouTube were also used to automatically classify the collected data from these social networks [55]–[60].

5. EVALUATION METRICS

The commonly used evaluation metrics for Arabic dialect identification systems are precision, recall, and F1-score [61]. In the field of text classification, precision is the percentage of correctly classified positive instances among the total classified positive instances, and recall is the percentage of correctly classified positive instances among all positive instances examined [61], [62]. Moreover, F1 score is the weighted harmonic mean of precision and recall:

$$F_1 = 2.\frac{Precision\ .\ Recall}{Precision + Recall}$$

Many studies carried out for the ADI task have assessed the performance of their implemented models using these three common evaluation metrics: precision, recall, and F1-score [5], [11], [37], [38], [63], [64]. The accuracy is a recommended evaluation metric for classification problems in addition to precision, recall, and F1-score as it takes true negatives into accounts. Accuracy can be defined as the percentage of correctly classified instances (both true positives and true negatives) among the total number of instances examined [62], [65]. The accuracy score is used as a performance measure for the ADI task in some studies [5], [6], [38]–[40], [66]. In the first edition of the ADI subtask, which was part of the Discriminating between Similar Languages (DSL) shared task, the organizers used a macro-averaged F1-score as an official score to evaluate participating systems [50]. The macro-averaged metric calculates the metric for each

Arabic dialect independently and then takes the average. The second edition of the ADI shared task within the VarDial Evaluation Campaign 2017 used the weighted F1-score as a main evaluation metric (weighted by the number of examples in each class) [51]. The organizers of the MADAR shared task on Arabic dialect identification in 2019 ranked the participating systems based on macro-averaged F1 scores and also reported the performance in terms of macro-averaged precision, macro-averaged recall and accuracy [15].

6. ARABIC DIALECT IDENTIFICATION APPROACHES

The field of Arabic dialect identification is relatively new, having caught the attention of researchers starting in 2011. At that time, the most serious obstacles were the lack of NLP tools and annotated resources for dialectal Arabic in general, and even more so for the ADI task. Therefore, the earliest studies, employed dictionaries and rules to distinguish Arabic varieties. The language modeling approach was also extensively examined for ADI. This approach involves assigning probabilities to sentences in a language as well as assigning a probability to each sequence of words. It also assigns a probability for the likelihood of a given word to follow a sequence of words [67] where word n-grams indicates the sequences of n consecutive words and character n-grams indicates the sequences of n consecutive characters. By way of this approach, various orders of n-gram language models have been scrutinized to identify Arabic dialects. Since then, many studies have developed their own manually annotated ADI corpora, allowing for “supervised” Machine Learning (ML) methods and feature engineering to be employed to identify dialectal Arabic. Currently, deep learning is considered state-of-the-art machine learning, and research is being conducted on the use of deep learning for ADI. These experiments build several neural network architectures with embedded representations of text units used as input features. Multiple approaches are now being combined to create high performance Arabic dialect identification systems.

We can categorize Arabic dialect identification approaches as follows: (a) language modeling and minimally supervised approaches (b) feature engineering supervised approaches, and (c) deep supervised approaches. Presented below are detailed descriptions of each category's main techniques specifically designed for Arabic dialect identification.

6.1 Language Modeling and Minimally Supervised Approaches

One of the early studies carried out by Elfardy and Diab [25] addressed automatic identification of token-level dialectal words (LCS points) in Arabic social media texts in which native Arabic speakers frequently mix dialectal Arabic and MSA. Their study utilized a set of rules, dictionaries, 3-gram language models, and a MSA morphological analyzer (ALMOR [21]) to tackle the problem. They used Tharwa, a three way DA-MSA-English machine readable dictionary [36]. A set of rules was used to model the possible phonological varieties of each word in various dialects. The evaluation corpus of 1,170 forum posts was manually collected and annotated. Their system that relied on the aggregate score produced by combining methods (dictionaries, rules, MSA morphological analysis, and language modeling) achieved the best result of F1-score=72.4% in the experimental setting where context was taken into account. On the other hand, with context-insensitive setting, the use of only language modeling resulted in achieving best performance of F1-score=84.9%. Elfardy et al. [5] proposed a system to perform automatic identification of linguistic code switches in Arabic. The system relied on the use of language models and the morphological analyzer CALIMA designed for Egyptian and MSA Arabic [68]. The system achieved an F1-Score of 76.5% when taking context into account.

Moreover, other early works on ADI examined language modeling methods by investigating the use of various n-gram based features at both character-level and word-level to identify dialectal Arabic [6], [11]. The studies that used language modeling for ADI confirmed the simplicity and efficiency of this approach. Zaidan and Callison-Burch [11] built word trigram models for Levantine, Gulf, Egyptian, and MSA. The 2-way classification scenario of their method (MSA vs. dialects) achieved 77.8% accuracy on the AOC dataset. In another study, Zaidan and Callison-Burch [6] explored higher-order character language models as well as word models. They tried character unigram, trigram & 5-gram LMs as well as word unigram, bigram & trigram LMs. They concluded that a unigram word LM performs best on the AOC dataset with an accuracy of 85.7% while the character 5-gram LM fell slightly behind with an accuracy of 85.0%. The Prediction by partial matching (PPM) technique has been utilized to identify Arabic dialects in a number of studies [63], [69], [70]. Lippincott et al. (2019) participated in the MADAR shared task on fine-grained Arabic dialect identification [15], task 2 addressing Twitter user dialect identification. They created a character-based model using a PPM language modeling technique. They experimented with various values of maximal order (N) to determine the likelihood of observing a symbol following a given context of up to N characters. Their experiments indicated that N=3 was the best value for Arabic dialect identification. They achieved an F1-score of 50.43% and was ranked 6th out of the 9 participating systems.

Huang [40] tested the eligibility of semi-supervised learning techniques, self-training and co-training, for the ADI task of obtaining more annotated data in the training phase. In the self-training technique, Huang [40] employed a strongly supervised classifier trained on the AOC training dataset to automatically annotate a large unlabeled dataset collected from social media (646M words). The additional annotated data was used to train a new classifier, achieving 84.4% accuracy on the AOC test dataset and 65.5% on Facebook data. In the co-training, Huang [40] implemented two classifiers: (a) a strongly supervised classifier trained on the AOC training dataset, and (b) a weakly supervised classifier trained on automatically annotated Facebook data based on the country indicated on the author's Facebook profile. The two classifiers were applied to automatically annotate unlabeled data. Then, only sentences on which the two classifiers agreed were used to train a new classifier. The co-training classifier achieved 86.2% accuracy on the AOC test dataset and 67.7% on Facebook data.

6.2 Feature Engineering Supervised Approaches

In conventional supervised machine learning algorithms, the performance of models can be improved by extracting features from the raw data based on the domain knowledge. This process is called “feature engineering” [71]. The availability of some dialectal Arabic lexical resources as illustrated in Section 4 have resulted in enriching the conventional machine learning algorithms with feature engineering. The feature engineering supervised approaches frequently performed better than unsupervised and semi-supervised methods [6], [26], [40], [43]. The extracted features represent many aspects in the processed texts that aid in building high quality learning models. A considerable number of features have been investigated including surface features (word n-grams, character n-grams, a combination of word and character n-grams, word k-skip n-grams), grammatical features, dictionary-based features, meta features, and stylistic features. According to the research on feature engineering supervised approaches, word and character features' eligibility for the ADI task in the texts was extensively evaluated. Most of the variations between Arabic dialects are based on vocabularies and affixation. Therefore, most of the studies have investigated the use of character and word n-grams to capture lexical, sub-lexical (e.g., morphemes, affixes), and syntactic differences. The word-level and character-level n-gram features have proved in many studies to be effective for the task of ADI. Next, we will explain in detail the conventional machine learning algorithms and features utilized for ADI problem.

6.2.1 Naive Bayes

Cotterell and Callison-Burch [37] utilized word-level unigram, bigram, and trigram features along with two machine learning algorithms: Naive Bayes and linear SVM to train ADI models. The word unigram model outperformed the higher order models (bigrams and trigrams). They also reported that Naive Bayes outperformed a linear SVM, achieving an average accuracy of 87% in the pairwise classification of six Arabic varieties when tested on the Extended AOC dataset. In the study of Elaraby and Abdul-Mageed [43], similar to Cotterell and Callison-Burch [37] findings, Naive Bayes outperformed other classical machine learning algorithms in the binary classification task (MSA vs dialects) with 84.53% accuracy and in the 3-way classification task (Egyptian vs. Levantine vs. Gulf) with 87.81% accuracy. The sole exception was a linear SVM's outperformance of Naive Bayes in the 4-way classification task (Egyptian vs. Levantine vs. Gulf vs MSA) with 78.61% accuracy. Their experiments utilized a combination of word unigram, bigram and trigram features and were conducted under two different text representations, binary presence and term frequency inverse document frequency (TF-IDF). Likewise, Elfardy and Diab [26] investigated Naive Bayes by building a model to identify sentences as either MSA or Egyptian. Their approach utilized token-level dialect labels from an underlying system for token-level identification of Egyptian dialectal Arabic developed by Elfardy et al. [5], in addition to perplexity-based features and meta features to estimate a sentence's degree of informality. Their best model by far was based on Naive Bayes and trained using the aforementioned features after applying tokenization in the preprocessing step, having achieved 85.5% accuracy. Sadat et al. [72] experimented to classify Arabic dialects into 18 dialect classes representing dialects spoken in 18 countries. They experimented with Naive Bayes and character-level unigram, bigram, and trigram features. They implemented three n-gram models and found that character-based bigram and trigram features generally performed better than the character unigram features for most dialects. Their results showed that the character n-gram Naive Bayes outperformed the character n-gram Markov Language Model for most Arabic dialects. As best result, the Naive Bayes model based on character-based bigram features yielded an overall F1-score of 80% accuracy on their own data manually collected and annotated from blogs and forums. Salameh et al. [73] was one of the first studies that explored Arabic fine-grained dialect identification at city level. Their research concerned 25 city-level Arabic dialects as well as MSA. They trained a Multinomial Naive Bayes (MNB) classifier with a combination of word unigram and character 1/2/3-gram features represented as TF-IDF. The classifier achieved 93.6% accuracy on MADAR Corpus-6 and 67.5% on MADAR Corpus-26. They also investigated the length of sentences and its correlation to accurately predicting the Arabic dialect in which it was written. They reported that their classifier managed to identify the exact city of a written text at an accuracy of 67.9% for sentences with an average length of 7 words and achieved around 90% for sentences with an average length of 16 words.

6.2.2 Support Vector Machines (SVMs)

Many studies explored the use of SVM with various kernel functions and features for Arabic dialect identification. For example, the ASIREM system built by Adouane et al. [74] utilized the linear SVM with higher-order character-level n-grams where n ∈ {5,6}. Their system participated in the Discriminating between Similar Languages (DSL) shared task 2016 subtask 2, which handles Arabic dialect identification. The ASIREM system was ranked fourth in the closed track with an F1-score of 49.5% and was ranked first in the open track with 52.7% F1-score. Most researchers who participated in the 2016 DSL subtask2 investigated the use of SVM with character-based n-grams [75]–[78]. For instance, the study of Ciobanu et al. (2016) was ranked 8th out of 18 participants with an F1-score of 47.4%. They utilized an SVM with string kernels and character-level (2-7)-grams. Çöltekin and Rama [76] developed an SVM system using character (1-7)-gram features. Their system achieved an F1-score of 47.3% and was ranked 9th among 18 participants. Furthermore, Adouane et al. [63] utilized three methods: Cavnar's Text classification, linear SVM, and Prediction by Partial Matching (PPM). Adouane et al. considered the automatic classification of Arabicized Berber (i.e., Berber written in Arabic script) as well as 7 Arabic dialects: Algerian, Egyptian, Gulf, Levantine, Mesopotamian, Moroccan, and Tunisian. The binary classification for each dialect was taken into account. The SVM outperformed the Cavnar's and the PPM methods, achieving an average F1-score of 92.94%. They constructed the SVM classifier with character-level 5-grams and 6-grams, as well as dictionary-based features. The lexicon of dialectal words was compiled from online resources, namely blogs, forums, and Facebook, using a script. Lastly, they examined the problem of ADI in written texts at a document level. Their dataset has been compiled from online newspapers dedicated for dialectal Arabic, discussion forums, blogs, and Facebook. They have not divided the collected data into sentences, but considered each user comment/participation, and each paragraph in the online newspapers as a document.

6.2.3 Decision Trees

Darwish et al. [66] used a Random Forest (RF) ensemble classifier that generates many decision trees, each of which is trained on a subset of features. They created an ensemble classifier that uses word unigram, bigram, and trigram models, as well as character unigram to 5-gram models as features. Their classifier to distinguish between Egyptian and MSA achieved 83.3% accuracy.

They manually created a test set by collecting and annotating 700 tweets (350 Egyptian tweets, and 350 MSA tweets). They also experimented with two more Random Forest (RF) ensemble classifiers that possess different features. The first classifier trained on word and character n-gram features calculated from the segmented Egyptian training data based on manually predefined morphological rules. The morphological rule-based classifier reached 85.9% accuracy. The second classifier relied on dialectal Egyptian lexicon-based features and their frequencies. The lexical-based classifier achieved the highest accuracy of 94.4%. Hence, Darwish et al. concluded that the clean list of dialectal words that cover common dialectal phenomena is more efficient than the use of word and character n-grams. They also examined the use of character-level and word-level n-grams separately and reported that character-based n-gram features outperformed the word-based n-grams, because they generalized better to the new unseen test data where the lexical overlap between training and test data was low. However, the combination of character and word features resulted in better results than each one of them alone.

6.2.4 Other Methods

Elaraby and Abdul-Mageed [43] built an ADI classifier based on logistic regression to identify the dialects of four Arabic varieties, namely Egyptian, Levantine, Gulf, and MSA. They experimented with word 1/2/3-grams with two settings for feature representation: (a) presence vs. absence (1 vs. 0) vectors, and (b) TF-IDF vectors. In binary classification (dialectal Arabic vs. MSA), the model scored 83.71% and 83.24% accuracy for (1 vs. 0) and TF-IDF features representations respectively. In four-way classification (Egyptian vs. Levantine vs. Gulf vs. MSA), the model achieved 78.24% accuracy for the two different representations of features. They conducted their experiments using AOC dataset.

String kernel functions have been used in text classification to measure the pairwise similarity between text samples, simply based on character n-grams [79]. String kernels along with kernel-based learning algorithms such as Kernel Discriminant Analysis (KDA) and Kernel Ridge Regression (KRR) have also been investigated to identify Arabic dialects and proved to be effective in many studies [80]–[82]. In fact, the system, based on multiple string kernels, was submitted to the Discriminating between Similar Languages (DSL) shared task in VarDial'2016 workshop and earned second place, achieving an accuracy of 50.91% and an F1-score of 51.31% [80].

6.2.5 Ensemble Methods

Dinu et al. [83] utilized an ensemble-based system to discriminate between dialects of Arabic using the corpus made available by the organizers of the third edition of ADI at the VarDial Evaluation Campaign 2018. They used a linear SVM for the individual classifiers and employed the majority rule to combine the output of the SVM classifiers. The individual classifiers were assigned uniform weights. They experimented with a set of features represented as TF-IDF: character n-grams, where $n \in \{1, ..., 8\}$, word n-grams where $n \in \{1,2,3\}$, and word k-skip bigrams where $k \in \{1,2,3\}$. They found that the optimal feature combination was character n-grams where $n \in \{3,4,5\}$. The best performing ensemble system yielded 0.5 F1-score on the test set. Their system did not participate in the third edition of the ADI shared task, but they used the same evaluation corpus and compared their system's performance with other systems submitted to the shared task. Their system's performance outperformed the task's baseline, but did not outperform other participating systems that were earned the first five places in the competition. Ragab et al. [84] developed an ensemble model of a group of best performing classifiers on a set of features that involves character-level and word-level TF-IDF features, class probabilities of a number of linear classifiers, and language model probabilities. They used two layers of classifiers. The class (i.e., dialect) probabilities that resulted from the trained classifiers in the first layer were added to other features and used as input to train the second layer of classifiers. The classifiers in the second layer were then ensembled together by way of majority voting, which selects the most frequently detected dialect to be the final predicted dialect. The individual classifiers were built using the Multinomial Naive Bayes (MNB) technique. Their ensemble model achieved an F1-score of 67.20% and was ranked 3rd among 19 participating systems submitted to the MADAR shared task on fine-grained Arabic dialect identification 2019 task 1 about travel domain dialect identification. Malmasi et al. [41] built a higher-level classifier, or “meta-classifier” by building a single linear SVM classifier for each feature type and utilizing the class probability outputs from each of these classifiers. The Multidialectal Parallel Corpus of Arabic (MPCA) was used for training and testing. They evaluated character-level n-grams where n ∈ {1,2,3,4} and word-level unigrams and bigrams. They also tested the stacked generalization model with all feature types character and word combinations, achieving an accuracy of 74.32%. They assessed the generalization of the system and their learned features through a cross-corpus evaluation using three different corpora: the MPCA, the AOC dataset, and the manually annotated 700 tweets by Darwish et al. [66]. They suggested that character-level features generalize the most, but the word unigrams obtained the best performance with a large enough dataset. Hanani et al. [85] participated in the 2017 VarDial Evaluation Campaign on Natural Language Processing (NLP) for Similar Languages, Varieties and Dialects. They approached the shared task of ADI by combining multiple classifiers based on various machine learning algorithms. They submitted three runs. One of them combined four classifiers on the system level: SVM with a Radial Basis Function (RBF) kernel, Naive Bayes with multinomial distribution, logistic regression, and Random Forests with 300 trees. They achieved an F1-score equal to 31%. All these classifiers utilized the same features: character 1/2/3-grams presented to the system as a feature vector. Their best run, which achieved an F1-score of 62.8%, combined the text and acoustic features. That is, they used a focal multiclass model to combine the outputs of a word-based SVM multiclass model and an i-vector-based SVM multiclass model.

Generally speaking, traditional supervised learning approaches proved to be effective in addressing the problem of ADI in written texts. The most widely used ML methods in many studies have been SVM, and Naive Bayes. However, the best results in the literature were obtained by SVM. Indeed, the SVM method proved that with adequate features, it was suitable for the task of ADI in written texts [13], [33], [50], [76], [85]–[89]. The ensemble methods also demonstrated good results for ADI in written texts, providing significant increase in performance for multi-class classification [13], [87]. Although a good number of datasets have been built (see Section 4 and Appendix A), the ADI literature contains limited comparable studies. That is, a considerable number of algorithms and techniques have been investigated for the ADI problem. Nevertheless, the proposed models' performances have been tested using various dataset and different evaluation metrics. In addition, there is a large number of Arabic dialects that were considered and studied in literature either as coarse-grained dialects (i.e., at region level) or fine-grained dialects (i.e., at city-level). In our survey, we analyze the proposed ADI models in literature, aligning their findings when it is possible. Table 3 shows the results of performance comparison between Naïve Bayes and SVM across different classification tasks (i.e., binary, 3-way, 4-way, and 5-way) based on experiments conducted by various studies in literature. Each row represents one experiment conducted by one study on both methods using the same dataset and metric for evaluation. The accuracy measure is used to announce the performance results.

We observed that across the various experiments conducted by different studies using different training settings, the Naïve Bayes achieved the best accuracy on almost all binary classification task. For 4-way classification task, the SVM outperformed the Naïve Bayes by about 2.8% when the features were word 1/2/3-grams TF-IDF. For 5-way classification task, the SVM obtained the best accuracy compared to the Naïve Bayes regardless of the utilized features set. The results of the various experiments, as illustrated in Table 3, show that Naïve Bayes outperforms SVM in the case of having a small set of classes in the Arabic dialect identification problem. This can likely be explained by the large lexical overlaps between various Arabic varieties (i.e., Arabic dialects and MSA) while the Naïve Bayes treats lexical features used in the training as independent features. Therefore, in the case of multi-class identification, the degree of class overlapping is relatively large and hence Naïve Bayes would perform poorly.

6.3 Deep Supervised Approaches

Deep learning has been applied to many fields including Natural Language Processing (NLP) and has achieved great success [67], [90]. The effectiveness of deep learning techniques in many NLP problems can be attributed partially to word embedding, a distributional representation of texts that allows words with similar meaning to receive similar representations. In word embedding techniques, each word is represented as a real-valued vector with only tens or hundreds of dimensions, contrasting the millions of dimensions necessary for sparse word representation [67]. Researchers have sought out solutions in deep learning architectures, using word embedding algorithms such as GLoVe, FastText, and Word2Vec (Continuous Bag-of-Words (CBOW) model/skip-gram model) [91], in hopes of approaching the problem of ADI in written texts [92]–[96]. The DL has succeeded with NLP applications such as sentiment analysis and question classification [97]–[100]. However, the findings of some studies demonstrated a poor performance of DL models; for example, the output of participating systems in Language Variety Identification of English, Spanish, and Portuguese in the 5th Author Profiling Task at PAN 2017 [96] demonstrated that systems based on traditional machine learning algorithms [101], [102] outperformed those that relied on deep learning methods [103]–[106]. Similarly, the deep-learning based methods used by most of the previously mentioned studies for the Arabic dialect identification task have performed poorly compared to the classical machine learning techniques with adequate feature engineering [76], [84], [92], [107]–[110]. In all of the four shared tasks organized for Arabic dialect identification from 2016 to 2019, the best performing systems were those that relied on classical machine learning algorithms and ensemble approaches with feature engineering.

Ref.	Methods	Used Features	Dataset	Labels	Accuracy %
[43]	**Naïve Bayes** SVM	Word 1/2/3-grams	AOC	Binary Classification (all dialects, MSA)	**84.33** 82.41
[43]	**Naïve Bayes** SVM	Word 1/2/3-grams	AOC	3-way classification (GLF, LEV, EGY)	**87.81** 84.27
[43]	**Naïve Bayes** SVM	Word 1/2/3-grams	AOC	4-way classification (GLF, LEV, EGY, MASA)	**77.75** 75.82
[43]	Naïve Bayes **SVM**	Word 1/2/3-grams TF-IDF	AOC	Binary Classification (dialects, MSA)	82.91 **83.61**
[43]	**Naïve Bayes** SVM	Word 1/2/3-grams TF-IDF	AOC	3-way classification (GLF, LEV, EGY)	**86.87** 85.93
[43]	Naïve Bayes **SVM**	Word 1/2/3-grams TF-IDF	AOC	4-way classification (GLF, LEV, EGY, MASA)	75.81 **78.61**
[37]	**Naïve Bayes** SVM	Word unigrams	Extended AOC	Binary Classification (EGY, MSA)	**89.00** 88.00
[37]	**Naïve Bayes** SVM	Word unigrams	Extended AOC	Binary Classification (GULF, MSA)	**88.00** 85.00
[37]	**Naïve Bayes** SVM	Word unigrams	Twitter	Binary Classification (NOR, MSA)	**70.00** 65.00
[37]	**Naïve Bayes** SVM	Word bigrams	Twitter	Binary Classification (IRQ, MSA)	**70.00** 65.00
[86]	Naïve Bayes **SVM**	Word unigrams TF-IDF	VarDial'2016 ADI training data (80/20 train/dev)	5-way classification (EGY, LEV, GLF, NOR, MSA)	48.32 **59.76**
[86]	Naïve Bayes **SVM**	Word 1/2/3-grams TF-IDF	VarDial'2016 ADI training data (80/20 train/dev)	5-way classification (EGY, LEV, GLF, NOR, MSA)	47.34 **57.00**
[86]	Naïve Bayes **SVM**	Word trigrams TF-IDF	VarDial'2016 ADI training data (80/20 train/dev)	5-way classification (EGY, LEV, GLF, NOR, MSA)	35.44 **39.58**
[86]	Naïve Bayes **SVM**	Character bigrams TF-IDF	VarDial'2016 ADI training data (80/20 train/dev)	5-way classification (EGY, LEV, GLF, NOR, MSA)	50.30 **53.78**
[86]	Naïve Bayes **SVM**	Character trigrams TF-IDF	VarDial'2016 ADI training data	5-way classification (EGY, LEV, GLF,	52.73 **61.54**

			(80/20 train/dev)	NOR, MSA)	
[86]	Naïve Bayes **SVM**	Character 2/3/4/5-grams TF-IDF	VarDial’2016 ADI training data (80/20 train/dev)	5-way classification (EGY, LEV, GLF, NOR, MSA)	37.02 **65.88**

TABLE 3: Comparison between SVM and Naive Bayes algorithms on Arabic dialect identification. Dialects: Egyptian (EGY), Levantine (LEV), Gulf (GLF), North African (NOR), Iraq (IRQ), and Modern Standard Arabic (MSA).

Meanwhile, the DL classifiers achieved lower results than many classical ML classifiers [15], [50]–[52]. This can be partially attributed to the limited resources used for Arabic dialect identification in terms of size, scope, and scale in these shared tasks; whereas DL models require a tremendous amount of annotated resources to adjust all parameters and reach high performance. Moreover, the MADAR shared task on fine-grained Arabic dialect identification handled Arabic dialects at city level with only 2,000 parallel sentences of 25 city-level Arabic dialects, which created a challenge for DL models, especially those that demand a considerable number of parameters to be set.

Zirikly et al. [111] experimented with neural networks trained with a 500-neuron single hidden layer and an output layer with softmax activation function. They used binary character (2-6)-gram representations as input features for the network. They also developed an ensemble classifier based on majority voting that took the outputs of one logistic regression and two single-layer Neural Networks and produced the majority label (i.e., Arabic dialects). The ties break when the output of the best-performing individual classifier is considered. They used character (1-6)-grams for LR and character (2-6)-grams for one of the NN. The second NN was trained using character (3-5)-grams and word unigram. Both of their single-layer NNs and the ensemble classifier participated in the Discriminating between Similar Languages (DSL) shared task 2016 subtask 2 of ADI in speech transcripts. They achieved F1-scores equal to 49.17%, and 49.22% respectively and the ensemble system was ranked 5th in the ADI subtask. Guggilla [112] also presented a system in the ADI subtask of the DSL 2016 based on Convolutional Neural Network (CNN) and was ranked 13th with an F1-score of 43.29%. They used a variant of the CNN architecture with four layers: the input, convolution, max-pooling, and softmax. Belinkov and Glass [93] developed a character-level CNN for ADI to be applied on the front-end to embed the sequence of characters into vector space. The sequence was then run through multiple convolutions, similar to a character-CNN utilized in language modeling [113]. Their system participated in the Discriminating between Similar Languages (DSL) shared task 2016, subtask 2. Their system achieved an F1-score of 48.34%, ranking 6th out of 18 participating systems. Ali [94] presented their system to the Arabic Dialect Identification (ADI) shared task at the VarDial Evaluation Campaign 2018. They proposed the use of character-level CNN, a neural network classifier that uses both the transcript text as one-hot encoded sequence of characters and the corresponding dialect embedding feature vector. The character sequence passes through a series of five layers (Gated Recurrent Units (GRU), convolution, batch-normalization, max-pooling, and dropout) before finally reaching a softmax layer. Then, the embedding vector passes directly to the softmax layer. The outputs of both softmax layers are averaged to provide the final output, which represents the probability distribution over five Arabic dialects. Their proposed character-level CNN achieved an F1-score equal to 57.6% and was ranked the 2nd among 6 participants. Other deep learning architectures utilized to perform dialect identification include Long Short-Term Memory (LSTM) [109], [110], Bidirectional LSTM (BiLSTM) [14], [95], Convolutional LSTM (CLSTM) [42], [114], and Recurrent Neural Networks (RNN) [96], [114]. De Francony et al. [14] presented a DL method for Arabic fine-grained dialect identification. They implemented a hierarchical model of two levels of DNNS in which the first predicts the group of the dialects (i.e., region) and the second level predicts the exact fine-grained dialect (i.e., city) based on the region prediction. The DNN in the first level consists of three layers: a B-LSTM, followed by a fully connected layer, and then an output layer. The second level is actually a set of 7 DNNs, one for each region. The input of the model is produced using Word2Vec. The F1-score of the hierarchical DL model on the MADAR travel domain corpus is equal to 58%.

Table 4 presents a comparison of DL models on ADI. Although many ADI studies attempted to use new and sophisticated neural network architectures. We observed that NN's performance was partially affected by the input data fed to the neural network and its form such as n-grams, character, or word and whether the deep neural network is with or without embeddings and whether embeddings are pretrained or learned during training. We noticed that a simple single-layer neural network for the ADI that utilized binary character (2-6)-grams performed quite closely to an ensemble classifier based on the majority voting that took the outputs of one traditional supervised learning method (logistic regression) and two single-layer NNs. The first NN in the ensemble classifier used character (2-6)-grams while the second NN used (3-5)-grams and word unigrams. The single-layer NN obtained a macro F1-score of 49.19% while the ensemble classifier obtained a macro F1-score of 49.22% when examined them on the VarDial'2016 ADI corpus. A more sophisticated neural network using CNN architecture that initially utilized randomly generated embeddings in the range [-0.25, 0.25] actually obtained a macro F1-score equal to 43.29% when examined on the VarDial'2016 ADI corpus. Therefore, we believe that the input data, its form, the initialization of NN and its hyperparameters would affect the overall performance of the NN especially when the dataset is not large enough to properly train the NN. More investigations are required to study the effects of these factors on the NN regardless of the degree of their architecture's sophistication or the size of the dataset used in the training phase. In addition, Arabic is morphologically complex language and Arabic dialects are known for their complex cliticization system. The syntactic information (i.e., words order) cannot be easily learned using only characters or words. Therefore, more investigations are needed for potential input data forms and optimal combinations of input data forms and network structures for the ADI problem.

Ref.	**Architecture**	**Used Features**	**Dataset**	**Labels**	**Macro F1-score**
[111]	Single-layer Neural Network consisting of single hidden layer of 500 neurons and output layer with softmax activation function	Binary character (2-6)-grams representations	VarDial'2016 ADI corpus	EGY, LEV, GLF, NOR, MSA	49.17
[111]	Ensemble classifier based on majority voting that took the outputs of one logistic regression and two single-layer Neural Networks	Character (1-6)-grams for LR, character (2-6)-grams for 1st NN, character (3-5)-grams and word unigram for 2nd NN	VarDial'2016 ADI corpus	EGY, LEV, GLF, NOR, MSA	49.22
[112]	A variant of the CNN architecture (an input layer, a convolution layer, a max pooling layer, and a fully connected softmax layer)	Randomly generated embeddings in the range [−0.25, 0.25] and updated during training	VarDial'2016 ADI corpus	EGY, LEV, GLF, NOR, MSA	43.29
[93]	Character-level CNN (embedding layer followed by dropout, multiple parallel convolutional layers with different filter widths, max pooling layer, fully-connected layer, and a softmax layer)	Character embedding learned during training	VarDial'2016 ADI corpus	EGY, LEV, GLF, NOR, MSA	48.34
Ref.	**Architecture**	**Used Features**	**Dataset**	**Labels**	**Weighted F1-score**
[110]	Ensemble system: (1) LSTM + CharCNN, (2) FastText	DL models: one-hot encoded sequence of	MADAR travel domain	25 city-level Arabic dialects,	65.35

	embeddings+LSTM, (3) MNB classifier	characters and word embeddings; MNB classifier: (1-5)-grams character and unigram word TF-IDF	corpus (Corpus-26)	MSA	
[110]	Ensemble system: (1) (Character TF-IDF) + (Word TF-IDF) + NN (2) MNB classifier	NN: frequency-based features of MNB classifier's features; MNB classifier: (1-5)-grams character and unigram word TF-IDF	MADAR travel domain corpus (Corpus-26)	25 city-level Arabic dialects, MSA	65.66
[70]	Ensemble system of 3 DL models (CNN, RNN, and MLP) concatenates the hidden representations produced by the three DL models.	Character and word embeddings, language-model based features	MADAR travel domain corpus (Corpus-26)	25 city-level Arabic dialects, MSA	61.83
[14]	Hierarchical system of two levels. The 1st level is a DNN with B-LSTM, followed by fully-connected layer, and then output layer. The 2nd level is a set of 7 different DNNs each of which utilizes a RNN layer, followed by fully-connected layer, and then output layer	Word embeddings	MADAR travel domain corpus (Corpus-26)	25 city-level Arabic dialects, MSA	58.00

TABLE 4: Comparison of DL models on Arabic dialect identification. Dialects: Egyptian (EGY), Levantine (LEV), Gulf (GLF), North African (NOR), Iraq (IRQ), and Modern Standard Arabic (MSA).

Arabic dialect identification started to gain a great deal of attention in the field of Arabic NLP a decade ago with the heightened prevalence of Web 2.0 and the rapid growth of online user-generated contents. Although a considerable number of studies have been conducted so far with beneficial results for Arabic dialect identification, most of the work has been devoted to the five main groups of Arabic dialects, namely Egyptian, Gulf, Levantine, North African, and MSA. The Arabic dialect taxonomy is complex and there are many overlapping areas between Arabic dialects within the same region and even the same country. Therefore, a lot of work remains to be carried out for fine-grained Arabic dialect identification, including both approaches and large-scale annotated resources, as there are few publicly available ADI annotated corpora. Moreover, the publicly available corpora are limited in terms of size, scope, and scale, which restricts deep empirical comparisons between approaches. Much larger-scale and larger-scope annotated corpora are needed to enrich the ADI resources and to support qualitative comparisons between potential studies in this area. The main sources of dialectal Arabic in its written form are online blogs, discussion forums and social networks. These online contents are written informally with many spelling errors. They include a lot of speech effects, emojis, neologisms, elongations and are written using different scripts. Therefore, NLP tools that handle these issues and clean the texts are needed to fill the gaps and help advance the research in dialectal Arabic. In fact, more investigations are required to handle the dialectal Arabic challenges that affect dialectal Arabic processing in general, and Arabic dialect identification in particular. We noticed that few studies on ADI investigated the impact of properly pre-processing dialectal Arabic contents (e.g., tokenization, orthography normalization, stemming) and cleaning them (e.g., removing stop words, correcting spelling errors) before building and training ADI models. More investigations are also necessary to handle ADI challenges themselves. For example, there have been only a handful studies examining the Linguistic Code Switching (LCS) phenomenon where a speaker mixes two or more Arabic varieties in the same utterance. Other challenges remain such as handling Arabizi, a non-standard romanization used by some Arabic native speakers for online contents, and the rich cliticization system in many Arabic dialects. Continued intensive investigation of these challenges will help uncover new appropriate approaches to distinguish between Arabic dialects.

8. CONCLUSION

The ADI in written texts plays fundamental role in many cross-language NLP applications, as well as social media analysis. It is considered the first step in building intelligent language systems that handle online contents. The task of ADI in written texts is a complex problem, as there are a considerable number of Arabic dialects based on various levels of geographical location (e.g., classification based on region, country, city). Many other factors actually affect the appearance of Arabic dialects, even within the same geographical location such as lifestyle, education, and socioeconomic status. The limited amount of training data available for Arabic dialects, especially at the fine-grained levels, as well as the necessity to deal with online dialectal Arabic contents have attracted many researchers of Arabic NLP in the last decade and more exponentially in the last four years to investigate the problems facing ADI in written texts.

This paper presented an extensive overview of ADI studies in literature. The algorithmic learning methods examined to perform ADI were discussed starting from conventional machine learning techniques used in early studies on ADI, up until neural networks and deep learning methods. We briefly compared various proposed methods. Our survey also discussed in detail the features used in ADI studies and their efficiency in improving the overall performance of the implemented systems, as well as the techniques used to represent these features. The available Arabic dialect identification corpora required for building ADI models were also covered in the survey, along with the common benchmarks used in the literature to evaluate the models. Future work and remaining open issues were discussed to help advance the field of dialectal Arabic NLP in general and ADI in written texts specifically.

9. REFERENCES

[1] N. Habash, O. Rambow, M. Diab, and R. Kanjawi-Faraj, “Guidelines for annotation of Arabic dialectness,” in Proceedings of the LREC Workshop on HLT & NLP within the Arabic world, 2008, pp. 49–53.

[2] N. Y. Habash, “Introduction to Arabic natural language processing,” Synth. Lect. Hum. Lang. Technol., vol. 3, no. 1, pp. 1–187, 2010.

[3] C. Zhang and M. Abdul-Mageed, “No Army, No Navy: BERT Semi-Supervised Learning of Arabic Dialects,” in Proceedings of the Fourth Arabic Natural Language Processing Workshop, 2019, pp. 279–284.

[4] M. Diab, N. Habash, O. Rambow, M. Altantawy, and Y. Benajiba, “COLABA: Arabic dialect annotation and processing,” in LREC workshop on Semitic language processing, 2010, pp. 66–74.

[5] H. Elfardy, M. Al-Badrashiny, and M. Diab, “Code switch point detection in Arabic,” in International Conference on Application of Natural Language to Information Systems, 2013, pp. 412–416.

[6] O. F. Zaidan and C. Callison-Burch, “Arabic dialect identification,” Comput. Linguist., vol. 40, no. 1, pp. 171–202, 2014.

[7] N. Habash, R. Roth, O. Rambow, R. Eskander, and N. Tomeh, “Morphological analysis and disambiguation for dialectal Arabic,” in Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2013, pp. 426–432.

[8] S. Harrat, K. Meftouh, and K. Smaili, “Machine translation for Arabic dialects (survey),” Inf. Process. Manag., 2017.

[9] D. Chiang, M. Diab, N. Habash, O. Rambow, and S. Shareef, “Parsing Arabic dialects,” in 11th Conference of the European Chapter of the Association for Computational Linguistics, 2006, pp. 369–376.

[10] R. Zbib et al., “Machine translation of Arabic dialects,” in Proceedings of the 2012 conference of the north american chapter of the association for computational linguistics: Human language technologies, 2012, pp. 49–59.

[11] O. F. Zaidan and C. Callison-Burch, “The Arabic online commentary dataset: an annotated dataset of informal Arabic with high dialectal content,” in Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2, 2011, pp. 37–41.

[12] H. Bouamor, N. Habash, and K. Oflazer, “A Multidialectal Parallel Corpus of Arabic.,” in LREC, 2014, pp. 1240–1245.

[13] S. Malmasi and M. Zampieri, “Arabic dialect identification in speech transcripts,” in Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial3), 2016, pp. 106–113.

[14] G. de Francony, V. Guichard, P. Joshi, H. Afli, and A. Bouchekif, “Hierarchical Deep Learning for Arabic Dialect Identification,” in Proceedings of the Fourth Arabic Natural Language Processing Workshop, 2019, pp. 249–253.

[15] H. Bouamor, S. Hassan, and N. Habash, “The MADAR shared task on Arabic fine-grained dialect identification,” in Proceedings of the Fourth Arabic Natural Language Processing Workshop, 2019, pp. 199–207.

[16] M. Diab and N. Habash, “Arabic dialect processing tutorial,” in Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Tutorial Abstracts, 2007, pp. 5–6.

[17] H. Hammarström, R. Forkel, and M. Haspelmath, “Glottolog database 4.1,” [Online]. Available: https://glottolog.org/ (accessed Oct. 04, 2019).

[18] D. M. Eberhard, G. F. Simons, and C. D. Fennig, “Ethnologue: Languages of the World. Twenty-third edition,” [Online]. Available: http://www.ethnologue.com/ (accessed Jan. 14, 2019).

[19] T. O’reilly, What is web 2.0. “ O’Reilly Media, Inc.,” 2009.

[20] D. Testen, Semitic Languages. Encyclopedia Britannica, 2018.

[21] N. Habash, A. Soudi, and T. Buckwalter, “On arabic transliteration,” in Arabic computational morphology, Springer, 2007, pp. 15–22.

[22] M. A. Yaghan, “‘Arabizi’: A contemporary style of Arabic Slang,” Des. Issues, vol. 24, no. 2, pp. 39–52, 2008.

[23] D. Palfreyman and M. al Khalil, “A Funky Language for Teenzz to Use: Representing Gulf Arabic in Instant Messaging,” J. Comput. Commun., vol. 9, no. 1, p. JCMC917, 2003.

[24] T. S. Jauhiainen, M. Lui, M. Zampieri, T. Baldwin, and K. Lindén, “Automatic language identification in texts: A survey,” J. Artif. Intell. Res., vol. 65, pp. 675–782, 2019.

[25] H. Elfardy and M. Diab, “Token level identification of linguistic code switching,” in Proceedings of COLING 2012: Posters, 2012, pp. 287–296.

[26] H. Elfardy and M. Diab, “Sentence level dialect identification in Arabic,” in Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), 2013, vol. 2, pp. 456–461.

[27] C. Holes, Colloquial Arabic of the Gulf and Saudi Arabia. Routledge & Kegan Paul Books, 1984.

[28] C. Holes, “Bahraini dialects: sectarian differences exemplified through texts,” Zeitschrift für Arab. Linguist., no. 13, pp. 27–67, 1984.

[29] M. W. Cowell, Reference grammar of Syrian Arabic: Based on the dialect of Damascus (Richard Slade Harrell Arabic Series 6). Washington, DC: Georgetown University Press, 1964.

[30] J. Heath, “Moroccan Arabic phonology,” Phonol. Asia Africa (including Caucasus), vol. 1, pp. 205–217, 1997.

[31] D. Caubet, “Moroccan Arabic,” Encycl. Arab. Lang. Linguist., vol. 3, pp. 273–287, 2008.

[32] C. Unicode, “The Character Contents of the Unicode Standard: Technical Reports and Standards,” [Online]. Available: http://www.unicode.org/reports/ (accessed Jan. 22, 2020).

[33] R. Tachicart, K. Bouzoubaa, S. L. Aouragh, and H. Jaafa, “Automatic identification of Moroccan colloquial Arabic,” in International Conference on Arabic Language Processing, 2017, pp. 201–214.

[34] K. Darwish, “Arabizi Detection and Conversion to Arabic,” in Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP), 2014, pp. 217–224.

[35] N. Habash et al., “Unified guidelines and resources for Arabic dialect orthography,” in Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), 2018, pp. 3628–3637.

[36] M. T. Diab et al., “Tharwa: A Large Scale Dialectal Arabic-Standard Arabic-English Lexicon,” in LREC, 2014, pp. 3782–3789.

[37] R. Cotterell and C. Callison-Burch, “A Multi-Dialect, Multi-Genre Corpus of Informal Written Arabic.,” in LREC, 2014, pp. 241–245.

[38] C. Tillmann, S. Mansour, and Y. Al-Onaizan, “Improved sentence-level Arabic dialect classification,” in Proceedings of the First Workshop on Applying NLP Tools to Similar Languages, Varieties and Dialects, 2014, pp. 110–119.

[39] M. Al-Badrashiny, H. Elfardy, and M. Diab, “Aida2: A hybrid approach for token and sentence level dialect identification in Arabic,” in Proceedings of the Nineteenth Conference on Computational Natural Language Learning, 2015, pp. 42–51.

[40] F. Huang, “Improved Arabic dialect classification with social media data,” in Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, 2015, pp. 2118–2126.

[41] S. Malmasi, E. Refaee, and M. Dras, “Arabic dialect identification using a parallel multidialectal corpus,” in Conference of the Pacific Association for Computational Linguistics, 2015, pp. 35–53.

[42] L. Lulu and A. Elnagar, “Automatic Arabic Dialect Classification Using Deep Learning Models,” Procedia Comput. Sci., vol. 142, pp. 262–269, 2018.

[43] M. Elaraby and 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, pp. 263–274.

[44] H. Mubarak, “Dial2MSA: A Tweets Corpus for Converting Dialectal Arabic to Modern Standard Arabic,” in Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), OSACT2018 Workshop, May 2018, pp. 49–53.

[45] H. Bouamor et al., “The MADAR Arabic dialect corpus and lexicon,” in Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018), 2018, pp. 3387–3396.

[46] T. Takezawa, G. Kikui, M. Mizushima, and E. Sumita, “Multilingual spoken language corpus development for communication research,” in International Journal of Computational Linguistics & Chinese Language Processing, Volume 12, Number 3, September 2007: Special Issue on Invited Papers from ISCSLP 2006, 2007, pp. 303–324.

[47] T. Solorio et al., “Overview for the first shared task on language identification in code-switched data,” in Proceedings of the First Workshop on Computational Approaches to Code Switching, 2014, pp. 62–72.

[48] G. Molina et al., “Overview for the Second Shared Task on Language Identification in Code-Switched Data,” in Proceedings of The EMNLP 2016 Second Workshop on Computational Approaches to Linguistic Code Switching (CALCS), 2016, pp. 40–49.

[49] H. Elfardy, M. Al-Badrashiny, and M. Diab, “AIDA: Identifying code switching in informal Arabic text,” in Proceedings of The First Workshop on Computational Approaches to Code Switching, 2014, pp. 94–101.

[50] S. Malmasi, M. Zampieri, N. Ljubešić, P. Nakov, A. Ali, and J. Tiedemann, “Discriminating between similar languages and Arabic dialect identification: A report on the third DSL shared task,” in Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial6), 2016, pp. 1–14.

[51] M. Zampieri et al., “Findings of the VarDial evaluation campaign 2017,” in Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial 2017), 2017, pp. 1–15.

[52] M. Zampieri et al., “Language identification and morphosyntactic tagging: The second VarDial evaluation campaign,” in Proceedings of the Fifth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial 2018), 2018, pp. 1–17.

[53] K. Almeman and M. Lee, “Automatic building of arabic multi dialect text corpora by bootstrapping dialect words,” in 1st international conference on communications, signal processing, and their applications (iccspa), 2013, pp. 1–6.

[54] J. Younes and E. Souissi, “A quantitative view of Tunisian dialect electronic writing,” in 5th International Conference on Arabic Language Processing, 2014, pp. 63–72.

[55] A. Salama, H. Bouamor, B. Mohit, and K. Oflazer, “YouDACC: the Youtube Dialectal Arabic Comment Corpus,” in Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC’14), May 2014, pp. 1246–1251.

[56] J. Younes, H. Achour, and E. Souissi, “Constructing linguistic resources for the Tunisian dialect using textual user-generated contents on the social web,” in International

Conference on Web Engineering, 2015, pp. 3–14.

[57] H. Mubarak and K. Darwish, “Using Twitter to collect a multi-dialectal corpus of Arabic,” in Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP), 2014, pp. 1–7.

[58] A. O. O. Alshutayri and E. Atwell, “Exploring Twitter as a source of an Arabic dialect corpus,” Int. J. Comput. Linguist., vol. 8, no. 2, pp. 37–44, 2017.

[59] K. Abu Kwaik, M. K. Saad, S. Chatzikyriakidis, and S. Dobnik, “Shami: A Corpus of Levantine Arabic Dialects,” in Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC’18, Miyazaki, Japan, May 7-12, 2018., 2018, pp. 3645–3652.

[60] M. Abdul-Mageed, H. Alhuzali, and M. Elaraby, “You tweet what you speak: A city-level dataset of arabic dialects,” in Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC’18), 2018, pp. 3653–3659.

[61] M. Allahyari et al., “A Brief Survey of Text Mining: Classification, Clustering and Extraction Techniques.” 2017.

[62] D. M. Powers, “Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation,” J. Mach. Learn. Technol., vol. 2, no. 1, pp. 37-–63, 2011.

[63] W. Adouane, N. Semmar, R. Johansson, and V. Bobicev, “Automatic detection of arabicized berber and Arabic varieties,” in Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial3), 2016, pp. 63–72.

[64] S. Harrat, K. Meftouh, M. Abbas, S. Jamoussi, M. Saad, and K. Smaili, “Cross-dialectal Arabic processing,” in International Conference on Intelligent Text Processing and Computational Linguistics, 2015, pp. 620–632.

[65] C. E. Metz, “Basic principles of ROC analysis,” in Seminars in nuclear medicine, 1978, vol. 8, no. 4, pp. 283–298.

[66] K. Darwish, H. Sajjad, and H. Mubarak, “Verifiably effective Arabic dialect identification,” in Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2014, pp. 1465–1468.

[67] Y. Goldberg, “Neural network methods for natural language processing,” Synth. Lect. Hum. Lang. Technol., vol. 10, no. 1, pp. 1–309, 2017.

[68] N. Habash, R. Eskander, and A. Hawwari, “A morphological analyzer for Egyptian Arabic,” in Proceedings of the twelfth meeting of the special interest group on computational morphology and phonology, 2012, pp. 1–9.

[69] P. McNamee, “Language and Dialect Discrimination Using Compression-Inspired Language Models,” in Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial3), 2016, pp. 195–203.

[70] T. Lippincott, P. Shapiro, K. Duh, and P. McNamee, “JHU System Description for the MADAR Arabic Dialect Identification Shared Task,” in Proceedings of the Fourth Arabic Natural Language Processing Workshop, 2019, pp. 264–268.

[71] S. J. Russell and P. Norvig, Artificial Intelligence-A Modern Approach, Third International Edition. Pearson Education London, 2010.

[72] F. Sadat, F. Kazemi, and A. Farzindar, “Automatic identification of Arabic dialects in social

media," in Proceedings of the first international workshop on Social media retrieval and analysis, 2014, pp. 35–40.

[73] M. Salameh, H. Bouamor, and N. Habash, "Fine-grained Arabic dialect identification," in Proceedings of the 27th International Conference on Computational Linguistics, 2018, pp. 1332–1344.

[74] W. Adouane, N. Semmar, and R. Johansson, "ASIREM participation at the discriminating similar languages shared task 2016," in Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial3), 2016, pp. 163–169.

[75] A. M. Ciobanu, S. Nisioi, and L. P. Dinu, "Vanilla Classifiers for Distinguishing between Similar Languages," in Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial3), 2016, pp. 235–242.

[76] Ç. Çöltekin and T. Rama, "Discriminating similar languages with linear SVMs and neural networks," in Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial3), 2016, pp. 15–24.

[77] A. Alshutayri, E. S. Atwell, A. Alosaimy, J. Dickins, M. Ingleby, and J. Watson, "Arabic language WEKA-based dialect classifier for Arabic automatic speech recognition transcripts," in Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial 2016), 2016, pp. 204–211.

[78] A. Hanani, A. Qaroush, and S. Taylor, "Classifying ASR transcriptions according to Arabic dialect," in Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial3), 2016, pp. 126–134.

[79] H. Lodhi, C. Saunders, J. Shawe-Taylor, N. Cristianini, and C. Watkins, "Text classification using string kernels," J. Mach. Learn. Res., vol. 2, no. Feb, pp. 419–444, 2002.

[80] R. T. Ionescu and M. Popescu, "UnibucKernel: An approach for Arabic dialect identification based on multiple string kernels," in Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial3), 2016, pp. 135–144.

[81] R. T. Ionescu and A. Butnaru, "Learning to identify Arabic and German dialects using multiple kernels," in Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial), 2017, pp. 200–209.

[82] A. M. Butnaru and R. T. Ionescu, "UnibucKernel Reloaded: First place in Arabic dialect identification for the second year in a row," in Proceedings of the Fifth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial 2018), 2018, pp. 77–87.

[83] L. P. Dinu, A. M. Ciobanu, M. Zampieri, and S. Malmasi, "Classifier ensembles for dialect and language variety identification," arXiv Prepr. arXiv1808.04800, 2018.

[84] A. Ragab et al., "Mawdoo3 AI at MADAR Shared Task: Arabic Fine-Grained Dialect Identification with Ensemble Learning," in Proceedings of the Fourth Arabic Natural Language Processing Workshop, 2019, pp. 244–248.

[85] A. Hanani, A. Qaroush, and S. Taylor, "Identifying dialects with textual and acoustic cues," in Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial), 2017, pp. 93–101.

[86] M. Eldesouki, F. Dalvi, H. Sajjad, and K. Darwish, "Qcri@ DSL 2016: Spoken Arabic dialect identification using textual features," in Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial3), 2016, pp. 221–226.

[87] S. Malmasi and M. Zampieri, “Arabic dialect identification using iVectors and ASR transcripts,” in Proceedings of the Fourth Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial), 2017, pp. 178–183.

[88] D. Ghoul and G. Lejeune, “MICHAEL: Mining Character-level Patterns for Arabic Dialect Identification (MADAR Challenge),” in Proceedings of the Fourth Arabic Natural Language Processing Workshop, 2019, pp. 229–233.

[89] P. Přibáň and S. Taylor, “ZCU-NLP at MADAR 2019: Recognizing Arabic Dialects,” in Proceedings of the Fourth Arabic Natural Language Processing Workshop, 2019, pp. 208–213, doi: 10.18653/v1/w19-4623.

[90] L. Deng, D. Yu, and others, “Deep learning: methods and applications,” Found. Trends®in Signal Process., vol. 7, no. 3--4, pp. 197–387, 2014.

[91] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient estimation of word representations in vector space,” in Proceedings of the International Conference on Learning Representations (ICLR 2013), 2013, pp. 1–12.

[92] B. Talafha, W. Farhan, A. Altakrouri, and H. Al-Natsheh, “Mawdoo3 AI at MADAR Shared Task: Arabic Tweet Dialect Identification,” in Proceedings of the Fourth Arabic Natural Language Processing Workshop, 2019, pp. 239–243.

[93] Y. Belinkov and J. Glass, “A character-level convolutional neural network for distinguishing similar languages and dialects,” in Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial3), 2016, pp. 145–152.

[94] 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, pp. 122–127.

[95] Y. Samih, H. Mubarak, A. Abdelali, M. Attia, M. Eldesouki, and K. Darwish, “QC-GO Submission for MADAR Shared Task: Arabic Fine-Grained Dialect Identification,” in Proceedings of the Fourth Arabic Natural Language Processing Workshop, 2019, pp. 290–294.

[96] F. Rangel, P. Rosso, M. Potthast, and B. Stein, “Overview of the 5th author profiling task at pan 2017: Gender and language variety identification in twitter,” Work. Notes Pap. CLEF, pp. 73–1613, 2017.

[97] Y. Kim, “Convolutional neural networks for sentence classification,” in Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, 2014, pp. 1746–-1751.

[98] Y. Du, W. Wang, and L. Wang, “Hierarchical recurrent neural network for skeleton based action recognition,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 1110–1118.

[99] D. Tang, B. Qin, and T. Liu, “Document modeling with gated recurrent neural network for sentiment classification,” in Proceedings of the 2015 conference on empirical methods in natural language processing, 2015, pp. 1422–1432.

[100] R. Socher et al., “Recursive deep models for semantic compositionality over a sentiment treebank,” in Proceedings of the 2013 conference on empirical methods in natural language processing, 2013, pp. 1631–1642.

[101] A. Basile, G. Dwyer, M. Medvedeva, J. Rawee, H. Haagsma, and M. Nissim, “Is there life beyond n-grams? A simple SVM-based author profiling system,” 2017.

[102] E. S. Tellez, S. Miranda-Jiménez, M. Graff, and D. Moctezuma, “Gender and language-variety Identification with MicroTC.,” 2017.

[103] D. Kodiyan, F. Hardegger, S. Neuhaus, and M. Cieliebak, “Author Profiling with bidirectional rnns using Attention with grus: Notebook for PAN at CLEF 2017,” 2017.

[104] S. Sierra, M. Montes-y-Gómez, T. Solorio, and F. A. González, “Convolutional Neural Networks for Author Profiling,” Work. Notes CLEF 2017-Conference Labs Eval. Forum, Ireland, 11-14 Sept., 2017.

[105] N. Schaetti, “UniNE at CLEF 2017: TF-IDF and Deep-Learning for Author Profiling.,” 2017.

[106] Y. Miura, T. Taniguchi, M. Taniguchi, and T. Ohkuma, “Author Profiling with Word+ Character Neural Attention Network.,” 2017.

[107] B. Talafha, A. Fadel, M. Al-Ayyoub, Y. Jararweh, A.-S. Mohammad, and P. Juola, “Team JUST at the MADAR Shared Task on Arabic Fine-Grained Dialect Identification,” in Proceedings of the Fourth Arabic Natural Language Processing Workshop, 2019, pp. 285–289.

[108] P. Mishra and V. Mujadia, “Arabic Dialect Identification for Travel and Twitter Text,” in Proceedings of the Fourth Arabic Natural Language Processing Workshop, 2019, pp. 234–238.

[109] S. Harrat, K. Meftouh, K. Abidi, and K. Smaili, “Automatic identification methods on a corpus of twenty five fine-grained Arabic dialects,” in International Conference on Arabic Language Processing, 2019, pp. 79–92.

[110] Y. Fares et al., “Arabic Dialect Identification with Deep Learning and Hybrid Frequency Based Features,” in Proceedings of the Fourth Arabic Natural Language Processing Workshop, 2019, pp. 224–228.

[111] A. Zirikly, B. Desmet, and M. Diab, “The GW/LT3 VarDial 2016 shared task system for dialects and similar languages detection,” in COLING, 2016, pp. 33–41.

[112] C. Guggilla, “Discrimination between similar languages, varieties and dialects using cnn- and lstm-based deep neural networks,” in Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial3), 2016, pp. 185–194.

[113] Y. Kim, Y. Jernite, D. Sontag, and A. M. Rush, “Character-aware neural language models,” 2016.

[114] M. Elaraby and A. Zahran, “A Character Level Convolutional BiLSTM for Arabic Dialect Identification,” in Proceedings of the Fourth Arabic Natural Language Processing Workshop, 2019, pp. 274–278.

APPENDIX A: Summary of Arabic Dialect Identification (ADI)

Corpus	Labels	Size	Annotation Level	Annotation Method	Source
Arabic Online Commentary (AOC) dataset [11].	Levantine, Gulf, Egyptian, MSA	108K sentences	Sentence-level annotated corpus	Manually via crowdsourcing	Reader commentary from three online Arabic newspapers
Extended AOC dataset [37].	Egyptian, Gulf, Levantine, Maghrebi, Iraqi	27,239 newspapers' reader comments	Comment-level annotated corpus	Manually via crowdsourcing	Reader commentary from online five Arabic newspapers
Twitter corpus [37]	Egyptian, Gulf, Levantine, Maghrebi, Iraqi	40,229 tweets	Tweet-level annotated corpus	Manually via crowdsourcing	Twitter
Tunisian Arabic Corpus (TAC) [115].	Tunisian dialect	895,000 words	One corpus of Tunisian dialect	Manually identified, checked, and transcribed by Tunisian students and via crowdsourcing	1. Traditional written sources (e.g., folklore) 2. New written sources (e.g., blogs) 3. Transcription of audio sources (e.g., podcasts)
Levantine/ English, Egyptian/ English Parallel corpora [10].	Levantine, Egyptian, English	Levantine/English (1.1M words), Egyptian/English (380K words)	Sentence-level parallel corpus	Manually via crowdsourcing	Large data of monolingual Arabic text harvested from the Web
Multidialectal Parallel Corpus of Arabic (MPCA) [12].	Egyptian, Tunisian, Jordanian, Palestinian, Syrian, MSA, English	2,000 parallel sentences	Sentence-level parallel corpus	Manually translate 2,000 sentences written in Egyptian into their dialects, MSA, and English.	Egyptian portion of the Egyptian/ English parallel corpus built by Zbib et al. (Zbib et al., 2012)
Dial2MSA parallel corpus [44].	Egyptian, Maghrebi, Levantine, Gulf, MSA	Egyptian/MSA (16,000 pairs), Maghrebi/MSA (8,000 pairs), Levantine/MSA (18,000 pairs), Gulf/MSA (18,000 pairs)	Tweet-level parallel corpus	Manually translated Collected tweets into MSA via crowdsourcing.	Twitter
PADIC [116].	Annaba's dialect, Algiers's dialect, Sfax's dialect, Syrian, Palestinian, MSA	6,400 parallel sentences	Sentence-level parallel corpus	Manually transcribed and translated by native speakers	Recorded Conversations from everyday life, movies and TV shows

Social Media dataset [72].	Algeria, Bahrain, Egypt, Emirates, Iraq, Jordan, Kuwait, Lebanon, Libya, Mauritania, Morocco, Oman, Palestine, Qatar, Saudi Arabia, Sudan, Syria, Tunisia	61,859 sentences	Sentence-level annotated corpus	Manually collected, segmented into coherent sentences and then classified into 18 dialects	Blogs and forums of different Arabic-speaking countries
MADAR travel domain corpus, Corpus-26 & Corpus-6 [45].	MSA and 25 Arabic city dialects (Rabat, Fes, Algiers, Tunis, Sfax, Benghazi, Tripoli, Alexandria, Cairo, Aswan, Khartoum, Beirut, Damascus, Aleppo, Jerusalem, Amman, Salt, Baghdad, Mosul, Basra, Doha, Muscat, Riyadh, Jeddah, Sana'a)	14,000 parallel sentences.	Sentence-level annotated corpus	Manual translation.	Selected sentences from the Basic Traveling Expression Corpus (BTEC) created by (Takezawa et al., 2007) and written in English and French
MADAR Twitter Corpus [15].	Algeria, Bahrain, Djibouti, Egypt, Emirates, Iraq, Jordan, Kuwait, Lebanon, Libya, Mauritania, Morocco, Oman, Palestine, Qatar, Saudi Arabia, Somalia, Sudan, Syria, Tunisia, Yemen	2,980 Twitter user profiles	Document-level annotated corpus	Manual annotation by native speakers	Twitter
MSA/DA Linguistic Code Switching corpus [25].	Egyptian, Gulf, Levantine, MSA	1,170 forum posts (27,173 tokens)	Token-level annotated corpus	Manual annotation by native speakers	Egyptian and Levantine forums
Language Identification in Code Switching Data (LICSD'2014)	MSA/Egyptian corpus	9,947 tweets and 6,723 commentaries	Token-level annotated corpus	Manual annotation by native speakers	Twitter and online reader commentaries from (AOC) Dataset

[47].					
LICSD’2016 [48].	MSA/Egyptian corpus	11,241 tweets	Token-level annotated corpus	Manual annotation by native speakers	Twitter
VarDial’2016 ADI corpus [50].	Egyptian, Gulf, Levantine, North African, MSA	9,159 sentences	Sentence-level annotated corpus	Dataset containing transcribed speech in MSA and in four dialects compiled by Ali et al. (Ali et al., 2016).	Multi-dialectal speech corpus created from broadcast, and discussion programs.
VarDial’2017 ADI corpus [51].	Egyptian, Gulf, Levantine, North African, MSA	16,841 sentences	Sentence-level annotated corpus with lexical and acoustic features	Dataset containing transcribed speech in MSA and in four dialects compiled by Ali et al. (Ali et al., 2016).	Multi-dialectal speech corpus created from broadcast, and discussion programs.
VarDial’2018 ADI corpus [52].	Egyptian, Gulf, Levantine, North African, MSA	22,186 sentences	Sentence-level annotated corpus with acoustic features and phonetic inputs	Dataset containing transcribed speech in MSA and in four dialects compiled by Ali et al. (Ali et al., 2016).	Multi-dialectal speech corpus created from broadcast, and discussion programs.
Arap-Tweet corpus [119].	Morocco, Algeria, Tunisia, Libya, Egypt, Sudan, North Levant, South Levant, Iraq, Gulf, Yemen	1,100 user profiles with their 2.4M Tweets corpus	Tweet-level annotated corpus	Manual annotation by experienced annotators	Twitter
Arabic Multidialect Text corpora [53].	Gulf, Levantine, Egyptian, North African	48M words in total for the four corpora	One corpus for each dialect	Bootstrapping the Web using Bing API and 1,043 distinctive words and phrases for the four dialects	Web texts
YouDACC [55].	Egyptian, Gulf, Iraqi, Maghrebi, Levantine	630,817 sentences	Sentence-level annotated	Semiautomatically based on keywords, Youtube API, and geographical locations.	User comments on YouTube videos
Twitter multidialectal corpus of Arabic [57].	Saudi Arabia, Egypt, Kuwait, United Arab Emirate, Qatar, Other.	6.5M tweets	Tweet-level annotated corpus	Semiautomatic annotation based on keywords, and geographical locations from users’ profiles	Twitter

Shami dialects corpus [59].	Syrian, Lebanese, Jordanian, Palestinian	117,805 sentences	Sentence-level annotated corpus	Semiautomatic annotation based on Twitter API and geographical location and manual annotation for collected contents from the Web	Twitter, discussion forums, and online blogs for public Levantine figures
Twitter multidialectal corpus of Arabic [57].	Saudi Arabia, Egypt, Kuwait, United Arab Emirate, Qatar, Other.	6.5M tweets	Tweet-level annotated corpus	Semiautomatic annotation based on keywords, and geographical locations from users' profiles	Twitter
Tweets corpus [60].	29 City-level dialects: Alexandria, Cairo, Giza, Baghdad, Karbala, Zubair, Amman, Aqaba, Irbid, Ahmadi, Hawally, Kuwait City, Muscat, Salalah, Sohar, Gaza, Nablus, Ramallah, Al-Rayyan, Doha, Dammam, Jeddah, Riyadh, Abu Dhabi, Al Ain, Dubai, Aden, Sana, Taiz	1/4 billion tweets	Tweet-level annotated corpus	Automatic annotation via third-party geocoder to acquire location labels on the data.	Twitter
Semi-supervised labeled corpus [40].	Egyptian, Gulf, Levantine, MSA	476M words for co-training and 646M for self-training	Sentence-level annotated corpus	Semi-supervised learning (co-training & self-training methods)	Facebook
Weakly labeled data [40].	Egyptian, Gulf, Levantine, MSA	66M words	Sentence-level annotated corpus	Automatic annotation based on the authors' profiles	Facebook
DART [118].	Egyptian, Maghrebi, Levantine, Gulf, Iraqi	24,280 tweets	Tweet-level annotated corpus	Manual annotation via crowdsourcing	Twitter
Gumar corpus [117].	Saudi Arabia, United Arab Emirate,	1,236 documents	Document-level annotated	Manual annotation by native	MS Word documents of novels in

	Kuwait, Oman, Qatar, Bahrain, Gulf, Arabic.		corpus	speakers	an online forum
Tunisian dialect Electronic writing [54]	Tunisian dialect (Latin transcription)	43,222 messages	Message-level annotated corpus	Lexicon-based classification method.	SMS messages, Tunisian forums, sites, and Facebook
TLD, TAD [56].	Tunisian dialect (Latin and Arabic transcriptions)	31,158 messages for TLD and 7,145 messages for TAD	Message-level annotated corpus	Lexicon-based classification method.	Facebook
Twitter Corpus [58].	Gulf, Iraqi, Egyptian, Levantine, North African	210,915 tweets	Tweet-level annotated corpus	Semiautomatic annotation based on keywords, and geographical locations from authors' profiles.	Twitter