Automatic Detection of Arabicized Berber and Arabic Varieties

Wafia Adouane[1], Nasredine Semmar[2], Richard Johansson[3], Victoria Bobicev[4]
Department of FLoV, University of Gothenburg, Sweden[1]
CEA Saclay – Nano-INNOV, Institut CARNOT CEA LIST, France[2]
Department of CSE, University of Gothenburg, Sweden[3]
Technical University of Moldova[4]
wafia.gu@gmail.com, nasredine.semmar@cea.fr
richard.johansson@gu.se, vika@rol.md

Abstract

Automatic Language Identification (ALI) is the detection of the natural language of an input text by a machine. It is the first necessary step to do any language-dependent natural language processing task. Various methods have been successfully applied to a wide range of languages, and the state-of-the-art automatic language identifiers are mainly based on character n-gram models trained on huge corpora. However, there are many languages which are not yet automatically processed, for instance minority and informal languages. Many of these languages are only spoken and do not exist in a written format. Social media platforms and new technologies have facilitated the emergence of written format for these spoken languages based on pronunciation. The latter are not well represented on the Web, commonly referred to as under-resourced languages, and the current available ALI tools fail to properly recognize them. In this paper, we revisit the problem of ALI with the focus on Arabicized Berber and dialectal Arabic short texts. We introduce new resources and evaluate the existing methods. The results show that machine learning models combined with lexicons are well suited for detecting Arabicized Berber and different Arabic varieties and distinguishing between them, giving a macro-average F-score of 92.94%.

1 Introduction

Automatic Language Identification (ALI) is a well-studied field in computational linguistics, since early 1960's, where various methods achieved successful results for many languages. ALI is commonly framed as a categorization.[1] problem. However, the rapid growth and wide dissemination of social media platforms and new technologies have contributed to the emergence of written forms of some varieties which are either minority or colloquial languages. These languages were not written before social media and mobile phone messaging services, and they are typically under-resourced. The state-of-the-art available ALI tools fail to recognize them and represent them by a unique category; standard language. For instance, whatever is written in Arabic script, and is clearly not Persian, Pashto or Urdu, is considered as Arabic, Modern Standard Arabic (MSA) precisely, even though there are many Arabic varieties which are considerably different from each other.

There are also other less known languages written in Arabic script but which are completely different from all Arabic varieties. In North Africa, for instance, Berber or Tamazight[2], which is widely used, is also written in Arabic script mainly in Algeria, Libya and Morocco. Arabicized Berber (BER) or Berber written in Arabic script is an under-resourced language and unknown to all available ALI tools which misclassify it as Arabic (MSA).[3] Arabicized Berber does not use special characters and it coexists with Maghrebi Arabic where the dialectal contact has made it hard for non-Maghrebi people to distinguish

[1] Assigning a predefined category to a given text based on the presence or absence of some features.

[2] An Afro-Asiatic language widely spoken in North Africa and different from Arabic. It has 13 varieties and each has formal and informal forms. It has its unique script called Tifinagh but for convenience Latin and Arabic scripts are also used. Using Arabic script to transliterate Berber has existed since the beginning of the Islamic Era (L. Souag, 2004).

[3] Among the freely available language identification tools, we tried Google Translator, Open Xerox language and Translated labs at http://labs.translated.net.

Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects,
pages 63–72, Osaka, Japan, December 12 2016.

it from local Arabic dialects.[4] For instance each word in the Arabicized Berber sentence 'AHml sAqwl mA$y dwl kAn' [5] which means 'love is from heart and not just a word' has a false friend in MSA and all Arabic dialects. In MSA, the sentence means literally 'I carry I will say going countries was' which does not mean anything.

In this study, we deal with the automatic detection of Arabicized Berber and distinguishing it from the most popular Arabic varieties. We consider only the seven most popular Arabic dialects, based on the geographical classification, plus MSA. There are many local dialects due to the linguistic richness of the Arab world, but it is hard to deal with all of them for two reasons: it is hard to get enough data, and it is hard to find reliable linguistic features as these local dialects are very hard to describe and full of unpredictability and hybridization (Hassan R.S., 1992). We start the paper by a brief overview about the related work done for Arabicized Berber and dialectal Arabic ALI in Section 2. We then describe the process of building the linguistic resources (dataset and lexicons) used in this paper and motivate the adopted classification in Section 3. We next describe the experiments and analyze the results in Sections 4 and 5, and finally conclude with the findings and future plans.

2 Related Work

Current available automatic language identifiers rely on character n-gram models and statistics using large training corpora to identify the language of an input text (Zampieri and Gebre, 2012). They are mainly trained on standard languages and not on the varieties of each language, for instance available language identification tools can easily distinguish Arabic from Persian, Pashto and Urdu based on character sets and topology. However, they fail to properly distinguish between languages which use the same character set. Goutte et al., (2016) and Malmasi et al., (2016) give a comprehensive bibliography of the recently published work dealing with discriminating between similar languages and language varieties for different languages. There is some work done to identify spoken Berber. For instance Halimouche et al., (2014) discriminated between affirmative and interrogative Berber sentences using prosodic information, and Chelali et al., (2015) used speech signal information to automatically identify Berber speaker. We are not aware of any work which deals with automatic identification of written Arabicized Berber.

Recently, there is an increasing interest in processing Arabic informal varieties (Arabic dialects) using various methods. The main challenge is the lack of freely available data (Benajiba and Diab, 2010). Most of the work focuses on distinguishing between Modern Standard Arabic (MSA) and dialectal Arabic (DA) where the latter is regarded as one class which consists mainly of Egyptian Arabic (Elfardy and Diab 2013). Further, Zaidan and Callison-Burch (2014) distinguished between four Arabic varieties (MSA, Egyptian, Gulf and Levantine dialects) using n-gram models. The system is trained on a large dataset and achieved an accuracy of 85.7%. However, the performance of the system can not be generalized to other domains and topics, especially that the data comes from the same domain (users' comments on selected newspapers websites). Sadat et al., (2014) distinguished between eighteen[6] Arabic varieties using probabilistic models (character n-gram Markov language model and Naive Bayes classifiers) across social media datasets. The system was tested on 1,800 sentences (100 sentences for each Arabic variety) and the authors reported an overall accuracy of 98%. The small size of the used test dataset makes it hard to generalize the performance of the system to all dialectal Arabic content. Also Saâdane (2015) in her PhD classified Maghrebi Arabic (Algerian, Moroccan and Tunisian dialects) using morpho-syntactic information. Furthermore, Malmasi et al., (2015) distinguished between six Arabic varieties, namely MSA, Egyptian, Tunisian, Syrian, Jordanian and Palestinian, on sentence-level, using a Parallel Multidialectal Corpus (Bouamor et al., 2014).

It is hard to compare the performance of the proposed systems, among others, namely that all of them were trained and tested on different datasets (different domains, topics and sizes). To the best of our

[4] In all polls about the hardest Arabic dialect to learn, Arabic speakers mention Maghrebi Arabic which has Berber, French and words of unknown origins unlike other Arabic dialects.

[5] We use Buckwalter Arabic transliteration scheme. For the complete chart see: `http://www.qamus.org/transliteration.htm`.

[6] Egypt; Iraq; Gulf including Bahrein, Emirates, Kuwait, Qatar, Oman and Saudi Arabia; Maghrebi including Algeria, Tunisia, Morocco, Libya, Mauritania; Levantine including Jordan, Lebanon, Palestine, Syria; and Sudan.

knowledge, there is no single work done to evaluate the systems on one large multi-domain dataset. Hence, it is wrong to consider the automatic identification of Arabic varieties as a solved task, especially that there is no available tool which can be used to deal with further NLP tasks for dialectal Arabic.

In this paper, we propose an automatic language identifier which distinguishes between Arabicized Berber and the eight most popular high level Arabic variants (Algerian, Egyptian, Gulf, Levantine, Iraqi (Mesopotamian), Moroccan, Tunisian dialects and MSA). We also present the dataset and the lexicons which were newly built as part of a Masters thesis project in Language Technology (Adouane, 2016). Both the dataset and the lexicons are freely available for research from the first author.

3 Building Linguistic Resources

Arabicized Berber (BER) has been officially used only in online newspapers and official institutions in North African countries like Algeria and Libya. It has been also used recently on social media by people who do not master the Berber script or Tifinagh and by those who do not master French.[7] An important question to answer when dealing with Arabic varieties is whether these variants are dialects or languages. There is no linguistically well-motivated answer since these varieties are different with their own regional/local varieties and are spoken in different countries. However, modern Arabic dialectology considers each Arabic variety as a stand-alone language (Hassan R.S., 1992). In this paper, we use the terms variety, dialect and language interchangeably.

It is necessary to decide how to cluster Arabic variants in order to be able to properly analyze and process them automatically. Nonetheless, it is not easy to distinguish each variant from another, particularly for short texts, because of the considerable lexical overlap and similarities between them. Moreover, it is very hard and expensive to collect data for each single variant given that some are rarely used on the Web. Based on the fact that people of the same region tend to use the same vocabulary and have the same pronunciation, Habash (2010) suggested to group Arabic dialects in six main groups, namely Egyptian (which includes Egyptian, Libyan and Sudanese), Levantine (which includes Lebanese, Jordanian, Palestinian and Syrian), Gulf (including Gulf Cooperation Council Countries), Iraqi, Maghrebi (which includes Algerian, Moroccan and Tunisian) and the rest is grouped in one class called 'Other'.

We use slightly different division where we count each Maghrebi variant as a stand-alone language. Moreover, we differently cluster Gulf/Mesopotamian[8] dialect group. We base our dialect clustering on common linguistic features, for instance the use of 'ch' instead of 'k' (Palva, 2006). So for the Mesopotamian Arabic, we include many local variants of Iraqi, Kuwaiti, Qatari and Emirati spoken Arabic. We group the rest of regions in the Gulf Arabic.[9] Our motivation is that these two broad regional dialectal groups (Maghrebi and Gulf/Mesopotamian) include a wide variety of languages which are easily distinguished by humans. Therefore, machines should be also able to discriminate between these varieties. In this study, we consider eight high level dialectal groups which are: Algerian (ALG), Egyptian (EGY), Gulf (GUL), Levantine (LEV), Mesopotamian (KUI), Moroccan (MOR), Tunisian (TUN) dialects plus MSA. In all cases, we focus on the language of the indigenous populations and not on the Pidgin Arabic.[10]

The use of Arabic dialects (in written format) on the Web is a quite recent phenomenon which started with the emergence of social media platforms and new technology devices. These Arabic variants, which use non-standardized orthography based on pronunciation or what is called 'write as you speak' principle, are still not well represented on the Web. This makes it hard to automatically process and analyze them (Diab et al., 2010). To overcome the deficiency of linguistic resources,[11] we built from scratch

[7] It is wrong to assume that all people from North Africa master French and use it in social media instead of Berber.

[8] There is no clear-cut dialectal borderlines between the Arabic varieties spoken in the Arabian Peninsula, namely between Gulf Arabic and Mesopotamian Arabic. Qafisheh (1977) gave a thorough morpho-syntactic analysis of the Gulf Arabic including Bahraini, Emirati, Qatari, Kuwaiti and regions of Saudi Arabia and excluding the Arabic dialects spoken in the rest of the Gulf countries. However, we do not have any morpho-syntactic parser, if it exists at all, to take all the grammars into account.

[9] Recent works consider all spoken Arabic in Gulf Cooperation Council Countries as Gulf Arabic.

[10] Simplified language varieties created by foreigners living in Arabic-speaking countries to make communication easier.

[11] There are collections by individuals but unfortunately not digitalized or which do not respect corpus linguistics annotation conventions.

linguistic resources consisting of dataset and lexicon for each Arabic variety considered in this study and Arabicized Berber.

3.1 Dataset

For Arabicized Berber, two Berber native speakers collected 503 documents (5,801 words) from north African countries mainly from forums, blogs and Facebook. For more data, we have selected varied texts from Algerian newspapers and segmented them. Originally the news texts are short, around 1,500 words each, so we considered each paragraph as a document (maximum 178 words). The selected newspapers use various Berber standard varieties written in Arabic script.

For each Arabic variety, two native speakers have manually collected content from various social media platforms (forums, blogs and micro-blogs) where each user's comment is counted as a single document/text. We gave instructions, for instance 'Collect only what is clearly written in your dialect, i.e. texts containing at least one clear dialectal word and you can easily understand it and reproduce the same in your daily interactions'. We have also compiled a list of dialectal words for each Arabic variety based on our knowledge. We then used a script with the compiled words as keywords to collect more data. Likewise, we collected 1,000 documents (around 54,150 words) for each dialect, roughly published between 2012-2016 in various platforms (micro-blogs, forums, blogs and online newspapers) from all over the Arab world. The same native speakers have been asked to clean the data following the same set of instructions.

We ended up with an unbalanced corpus of between 2,430 documents (64,027 words) and 6,000 documents or (170,000 words) for each dialect. In total, the collected dataset contains 579,285 words. In terms of data source distribution, the majority of the content comes from blogs and forums where users are trying to promote their dialects; roughly 50%, around 30% of the data comes from popular YouTube channels and the rest is collected from micro-blogs. The selection of the data sources is based on the quality of the dialectal content, i.e. we know that the content of the selected forums and blogs is dialectal which is used to teach or promote dialects between users. Ideally we would have looked at just some data resources and harvest content as much as possible either manually or using a script. But given the fact that data depends on the platform it is used in[12] and our goal that is to build a general system which will be able to handle various domain/topic independent data, we have used various data domains dealing with quite varied topics like cartoons, cooking, health/body care, movies, music, politics and social issues. We labeled each document with the corresponding Arabic variety.

We introduced necessary pre-processing rules such as tokenization, normalization and removal of non-discriminative words including punctuation, emoticons, any word occurring in the MSA data more than 100 times (prepositions, verbs, common nouns, proper nouns, adverbs, etc.) and Named Entities (NE). Removing non-discriminative words is motivated by the fact that these words are either prevalent in all Arabic varieties or they do not carry any important linguistic information like emoticons and punctuation. The choice of removing NE is motivated by the fact that NE are either dialect (region) specific or prevalent; i.e. they exist in many regions, so they are weak discriminants. Moreover, we want the system to be robust and effective by learning the language variety and not heuristics about a given region. The pre-processing step was done manually because of the absence of the appropriate tools.

To assess the reliability of the annotated data, we have conducted a human evaluation. As a sample, we have picked up randomly 100 documents for each language from the collection, removed the labels, shuffled and put all in one file (900 unlabeled documents in total). We asked two native speakers for each language, not the same ones who collected the original data, to pick out what s/he thinks is written in his/her dialect, i.e. can understand easily and can produce the same in his/her daily life. All the annotators are educated, either have already finished their university or are still students. This means that all of them are expected to properly distinguish between MSA and dialectal Arabic. To interpret the results, we computed the inter-annotator agreement for each language to see how often the annotators agree. Since we have two annotators per language, we computed the Cohen's kappa coefficient which is

[12]For instance the use of special markers in some platforms and the allowed length of the texts where shorter text means more abbreviations.

a standard metric used to evaluate the quality of a set of annotations in classification tasks by assessing the annotators' agreement (Carletta, 1996). Overall, the data quality is 'satisfactory' for Algerian, Gulf and Tunisian dialects by interpreting the kappa metric which is between 0.6–0.8. The quality of the rest of the dialectal data is 'really good', kappa 0.8–1.

3.2 Lexicons

We removed 18,000 documents (2,000 documents, between 60,000 and 170,000 words, for each Arabic variety and Arabicized Berber) to be used for training and evaluation. We extracted from the rest of the data all the unique vocabulary, using a script, to build lexicons. We have also added dialectal words collected from exchange forums where users were trying to promote their culture and dialects. The reason we have done so is the desperate lack of digitalized dialectal lexicons[13] and the few available ones are outdated word lists in paper format. For MSA, we have used the content of two freely available books. We would have also used an MSA dictionary, but this would need more effort as the freely available dictionaries are not designed to be easily used for any computational purpose.

In order to have even more refined lexicons, we used Term Frequency-Inverse document Frequency (TF-IDF)[14] to measure the importance of each word to each dialect. Table 1 shows the number of unique words (types) of the compiled lexicons for each language after applying TF-IDF and removing non-informative words. The specific vocabulary of each Arabicized Berber and Arabic variety is stored in a separate .txt file, one word per line.

Language	ALG	BER	EGY	GUL	KUI	LEV	MSA	MOR	TUN
#Types	9 172	21 786	5 979	10 349	10 272	9 969	88 361	11 879	13 101

Table 1: The size (total number of unique vocabulary) of the compiled lexicons.

4 Methods and Experiments

We use supervised machine learning, namely Cavnar's Text classification, support vector machines (SVM) and Prediction by Partial Matching (PPM) methods. For features, we use both character-based n-gram[15] and word-based n-gram[16] models, then we combine them. We also use the words of the compiled lexicons as features. We focus more on social media short texts, so we limit the text maximum length to 140 characters (which is the maximum length of a tweet) assuming that if a method works for short texts, it should work better for longer texts as there will be access to more information. We use a balanced dataset containing 18,000 documents (2,000 documents, between 60,000 and 170,000 words, for each language) where we used 80% (total of 14,400 documents or 1,600 for each language) for training and 20%, total of 3,600 documents or 131,412 words (400 documents for each language), for evaluation.

4.1 Cavnar's Text Classification Method

Cavnar's Text Classification Method is one of the automatic language identification (ALI) statistical standard methods. It is a ranked collection of the most common character-based n-grams for each language used as its profile (Cavnar and Trenkle, 1994). The distance between language profiles is defined as the sum of all distances between the ranking of the n-gram profiles, and the language with the minimum distance from the source text will be returned. We experimented with different character-based n-grams and combinations and found that 3-grams performed the best with a macro-average F-score of 52.41%. Table 2 shows the performance of the Cavnar's method per language.

[13] "For many regions, no substantial dictionaries are available. We have reasonable dictionaries for Levantine, Algerian and Iraqi, but these are sometimes outdated and need to be replaced or updated" (Behnstdt and Woidich, 2013).

[14] A weighting scheme used to measure the importance of each word in a document and a other documents based on its frequency.

[15] A sequence of n characters from a given sequence of text where n is an integer.

[16] A sequence of n words from a given sequence of text where n is an integer.

Language	Precision (%)	Recall (%)	F-score (%)
ALG	41.34	37.00	39.05
BER	98.43	94.00	**96.16**
EGY	56.20	38.50	45.70
GUL	32.69	50.50	39.69
KUI	47.05	53.75	50.18
LEV	46.23	36.75	40.95
MOR	57.14	48.00	52.17
MSA	63.28	81.00	**71.05**
TUN	39.71	34.25	36.78

Table 2: Cavnar's method performance using character 3-grams.

The results show that except for Arabicized Berber (BER) which is properly identified, Cavnar's classifier finds it hard to distinguish Arabic varieties from each other even though it performs better in distinguishing MSA from dialectal Arabic. Our main purpose in using Cavnar's method is to set its performance as our baseline.

4.2 Support Vector Machines (SVM)

We use the LinearSVC classifier (method) as implemented in Scikit-learn package (Pedregosa et al., 2011)[17] with the default parameters.[18] Furthermore, we use the binary classification setting as opposed to the 9-class classification, for instance 'is a document written in BER or something else (Arabic varieties)' as opposed to 'is a document written in BER, MSA, ALG, EGY, GUL, LEV, KUI, MOR or TUN.' Both classification settings return only one label or category as an output because each classifier is implemented as a group of classifiers, and the label with the highest prediction score is returned. We experimented with various features (character and word based n-grams of different lengths and combinations) and found that combining character-based 5-grams and 6-grams with the words of the compiled lexicons performed the best with a macro-average F-score of 92.94%. Table 3 shows the performance of the SVM method per language.

Language	Precision (%)	Recall (%)	F-score (%)
ALG	91.79	92.25	92.02
BER	100	100	**100**
EGY	95.63	82.00	88.29
GUL	86.92	89.75	88.31
KUI	91.20	93.25	92.21
LEV	91.71	88.50	90.08
MOR	93.84	95.25	94.54
MSA	93.46	100	**96.62**
TUN	92.98	96.00	94.46

Table 3: SVM performance combining character-based 5-grams and 6-grams with lexicons.

SVM classifier performs very well for BER and even better than the Cavnar's classifier. It also performs very well in distinguishing Arabic varieties. It identifiers MOR and TUN better than ALG. Likewise, it recognizes KUI better than GUL. MSA is also well distinguished from other varieties.

[17]For more information see: `http://scikit-learn.org/stable/`.

[18]The default parameters for each classifier are detailed in `http://scikit-learn.org/stable/`.

4.3 Prediction by Partial Matching (PPM)

A lossless compression algorithm which has been successfully applied to language identification (Bobicev, 2015) as well as other tasks. PPM encodes all the symbols (characters or words) of a training data within their context where a context of each symbol is a sequence of preceding symbols of different lengths.[19] PPM is a simple method which does not require feature selection as it considers the entire text as a single string and computes the probability distribution for each symbol using a blending mechanism. We implemented a simple version of the PPM method as explained in (Moffat, 1990; Bobicev, 2015) where we used the context of 5 characters for each symbol and the benchmark escape method called C. Hence, we implemented the PPMC5 version of PPM. Here, we use the entire text length. The method reaches a macro-average F-score of 87.55%.

At the end, we validated our three models using the 10-fold cross-validation technique. Each time, we preserve one fold for validation and train on the rest 9 folds. This gives us an idea on how a model is dataset independent. For each method, we used the same settings above and found that the accuracy values are close to each other for all cross-validation folds, and close to the overall accuracy. This means that the models are not an overfit.

It is unfair to compare the results of the three methods as we limited the maximum text length to 140 characters for both SVM and Cavnar's methods and used full-length text for the PPM method. Now, we use the full-length text for all methods using the same experimental setups. The results are shown in Table 4 where 'DV' is short for 'dialectal vocabulary' and it refers to the words of the compiled lexicons.

Method	**Features**	**Maximum Text Length**	**Macro-average F-score (%)**
Cavnar	Character 3-grams	140 characters	52.41
Cavnar	Character 3-grams	Full length	81.57
SVM	Character 5-6-grams + DV	140 characters	**92.94**
SVM	Character 5-6-grams + DV	Full length	**93.40**
PPMC5	No features	Full length	87.55

Table 4: Performance of the three methods with full-length text.

The results show that increasing the length of the text improves the performance of both Cavnar's and SVM methods. Cavnar's method performs poorly for short texts (maximum length of 140 characters). It is true that SVM outperforms the Cavnar's method because it has access to extra data (lexicons). However, even with the same experimental setup (using character-based 3-grams as features with text maximum length of 140 characters), SVM still outperforms the Cavnar's method which is taken as our baseline.

5 Error Analysis

Analyzing the confusion matrix of each method shows that the confusions are of the same type with different frequencies. For illustration, we show in Table 5 the confusion matrix of the SVM method using the combination of character-based 5-6-grams and the dialectal vocabulary as features and text maximum length of 140 characters.

Most confusions are between very close Arabic varieties, namely Maghrebi dialects (ALG, MOR, TUN) and between GUL and KUI dialects. This is expected and accepted because, as mentioned above, there are no dialectal clear-cut borderlines between neighboring dialects. In more details, there are more MOR and TUN documents confused with ALG ones compared to the ALG documents confused with MOR or TUN documents. The same is applicable for KUI documents confused with GUL ones. This may be related to the fact that in practice it is impossible to draw the dialectal borderlines, especially for very short texts as in our case. Moreover, there are confusions between Maghrebi, Egyptian and Levantine varieties. This is explained by the fact that some Levantine dialects (southern Syria and some

[19] Previous works reported that taking the context of 5 characters is the best maximum context length. This makes a perfect sense because long matches are less frequent to occur by chance.

		Misclassified languages								
Correct languages		ALG	BER	EGY	GUL	KUI	LEV	MSA	MOR	TUN
	ALG	369	0	1	0	0	1	2	12	15
	BER	0	400	0	0	0	0	0	0	0
	EGY	6	0	328	15	6	19	10	8	8
	GUL	1	0	5	359	24	7	3	0	1
	KUI	1	0	1	21	373	4	0	0	0
	LEV	7	0	7	16	3	354	10	1	2
	MSA	0	0	0	0	0	0	400	0	0
	MOR	9	0	0	2	1	1	3	381	3
	TUN	9	0	1	0	2	0	0	4	384

Table 5: SVM confusion matrix using character-based 5-6-grams and dialectal vocabulary.

parts of Lebanon, including Beirut) share the use of split-morpheme negations with Egyptian and north African dialects (Palva, 2006). It is also important to notice that while BER is rarely confused, MSA is often confused with the rest of Arabic varieties.

6 Conclusion and Future Directions

In this study, we dealt with both tasks of identifying Arabicized Berber and different Arabic varieties as well as discriminating between all of them. For Arabic, we considered eight high level varieties (Algerian (ALG), Egyptian (EGY), Gulf (GUL), Levantine (LEV), Mesopotamian (KUI), Moroccan (MOR), Tunisian (TUN) dialects plus Modern Standard Arabic (MSA)) which are the most popular Arabic variants. The task is challenging at many levels. First, Arabicized Berber and Arabic varieties, except MSA, are under-resourced and undocumented. Second, dialectal Arabic is mostly used in social media and mobile phone messages. This makes the task harder since this genre allows only short texts.

To overcome these challenges, we created the necessary linguistic resources (dataset and lexicons). We framed the task as a categorization problem for short texts written in very similar languages. We applied one of the automatic language identification standard methods, namely supervised machine learning including Cavnar's text classification, support vector machines (SVM) and the Prediction by Partial Matching methods. We set the performance of the Cavnar's method as our baseline. All in all, for short texts of 140 characters or less, Cavnar's character-based method is not efficient in distinguishing Arabic varieties from each other, particularly the very close ones like Maghrebi dialects. The reason is that all the varieties use the same character set with almost the same distribution. Nevertheless, it performs better in discriminating between MSA and dialectal Arabic. Also, it distinguishes Arabicized Berber fairly well from Arabic. SVM combining the character-based 5-6-grams with the words of the compiled lexicons performs fairly well for short texts, and increasing the text length performs even better. Likewise, the PPM (precisely PPMC5) method is good at distinguishing Arabicized Berber from Arabic and MSA from dialectal Arabic. Error analysis shows that all the errors, whatever the method, are of the same type; confusion between very similar languages.

So far, we have applied the automatic language identification standard methods to discriminate between Arabicized Berber and Arabic varieties which are under-resourced languages, and we found that supervised machine learning using character-based n-gram models are well suited for our task to a large extent. This should be a good start to automatically process dialectal Arabic. For now, we find it hard to compare our system to other reported results of related work because the datasets used in evaluation are different. We would like to test our system on larger and multi-domain/topic datasets to see how it performs as well as test it on some newly collected corpora, for instance (Salama et al., 2014). This will allow us to improve the system and generalize the results.

Still, there are other points we want to explore further in future work like distinguishing between

varieties of Arabicized Berber, and applying the two step classification process which consists in first identifying the regional dialectal group, for instance Maghrebi Arabic, then apply some different feature weighting to identify the dialect itself. It would be also possible to analyze the misspellings which seem to be consistent within the same variant because the orthography is based on the pronunciation. This could help improving the dialectal Arabic identification. Another way worth exploring is to include user metadata (extralinguistic information) like the location.

Acknowledgments

The authors would like to thank all anonymous reviewers for their comments that over time have substantially improved the paper.

References

Ahmed Salama, Houda Bouamor, Behrang Mohit and Kemal Oflazer. 2014. YouDACC: the Youtube Dialectal Arabic Comment Corpus. *In the Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 1246–1251, Reykjavik, Iceland.

Alistair Moffat. 1990. Implementing the PPM data compression scheme. *IEEE Transactions on Communications*, 38(11), pages 1917–1921.

Abd-El-Jawad, Hassan R.S. 1992. Is Arabic a pluricentric language?. *In Clyne, Michael G. Pluricentric Languages: Differing Norms in Different Nations. Contributions to the sociology of language 62*. Berlin & New York: Mouton de Gruyter. pages 261–303.

Cyril Goutte , Serge Léger, Shervin Malmasi and Marcos Zampieri. 2016. Discriminating Similar Languages: Evaluations and Explorations. *In the Proceedings of Language Resources and Evaluation (LREC)*. Portoroz, Slovenia.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake VanderPlas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot and Edouard Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Machine learning in Python. Journal of Machine Learning Research*, 12, pages 2825–2830.

Fatma Zohra Chelali, Khadidja Sadeddine and Amar Djeradi. 2015. Speaker identification system using LPC-Application on Berber language. *HDSKD journal*, 1(2):29–46.

Fatiha Sadat, Farnazeh Kazemi and Atefeh Farzindar. 2014. Automatic Identification of Arabic Language Varieties and Dialects in Social Media. *In the Proceedings of the Second Workshop on Natural Language Processing for Social Media (SocialNLP)*, pages 22–27, Dublin, Ireland.

Hamdi A. Qafisheh. 1977. A short reference grammar of Gulf Arabic. *Tucson: University of Arizona Press*.

Heba Elfardy and Mona Diab. 2013. Sentence-Level Dialect Identification in Arabic. *In the Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, ACL 2013*, Sofia, Bulgaria.

Heikki Palva. 2006. *Encyclopedia of Arabic languages and linguistics, v.1, A-Ed.*. Leiden: Brill, pages 604–613.

Houda Bouamor, Nizar Habash and Kemal Oflazer. 2014. A Multidialectal Parallel Corpus of Arabic. *In the Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 1240–1245, Reykjavik, Iceland.

Houda Saâdane. 2015. Le traitement automatique de l'arabe dialectalise: aspects methodologiques et algorithmiques. *PhD thesis*, Université Grenoble Alpes.

Jean Carletta. 1996. Assessing agreement on classification tasks: The kappa statistic. *Computational Linguistics, 22(2)*, pages 249–254.

Lameen Souag. 2004. Writing Berber Languages: a quick summary. *L. Souag. Archived from* `http://goo.gl/ooA4uZ`, Retrieved on April 8th, 2016.

Marcos Zampieri and Binyam Gebrekidan Gebre. 2012. Automatic Identification of Language Varieties: The Case of Portuguese. *In the Proceedings of KONVENS 2012 (Main track: poster presentations)*, Vienna.

Mona Diab, Nizar Habash, Owen Rambow, Mohamed Altantawy and Yassine Benajiba 2010. COLABA: Arabic dialect annotation and processing. *In the Proceedings of the LREC Workshop on Semitic Language Processing*, pages 66–74.

Nizar Habash. 2010. Introduction to Arabic Natural Language Processing. *Morgan & Claypool Publishers*.

Omar F. Zaidan. 2012. Crowdsourcing Annotation for Machine Learning in Natural Language Processing Tasks. *PhD thesis*, Johns Hopkins University.

Omar F. Zaidan and Chris Callison-Burch. 2014. Arabic dialect identification. *Computational Linguistics, 40(1)*, pages 171–202.

Peter Behnstadt and Manfred Woidich. 2013. Dialectology. *In the Oxford Handbook of Arabic Linguistics, Dialectology*, pages 300–323.

Ramzi Halimouche, Hocine Teffahi and Leila Falek. 2014. Detecting Sentences Types in Berber Language. *International Conference on Multimedia Computing and Systems (ICMCS)*, pages 197–200.

Shervin Malmasi, Eshrag Refaee and Mark Dras. 2015. Arabic Dialect Identification using a Parallel Multidialectal Corpus *In the Proceedings of the 14th Conference of the Pacific Association for Computational Linguistics (PACLING 2015)*, pages 209–217, Bali, Indonesia.

Shervin Malmasi, Marcos Zampieri, Nikola Ljubešić, Preslav Nakov , Ahmed Ali and Jörg Tiedemann. 2016. Discriminating between Similar Languages and Arabic Dialect Identification: A Report on the Third DSL Shared Task. *In the Proceedings of the 3rd Workshop on Language Technology for Closely Related Languages, Varieties and Dialects (VarDial)*, Osaka, Japan.

Victoria Bobicev. 2015. Discriminating between similar languages using ppm. *In the Proceedings of the LT4VarDial Workshop*, Hissar, Bulgaria.

Wafia Adouane. 2016. *Automatic Detection of Under-resourced Languages: The case of Arabic Short Texts*. Master's thesis, University of Gothenburg.

William B. Cavnar and John M. Trenkle. 1994. N-gram-based text categorization. *In the Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval*, pages 161–175, Las Vegas, US.

Yassine Benajiba and Mona Diab. 2010. A web application for dialectal Arabic text annotation. *In the Proceedings of the LREC Workshop for Language Resources (LRs) and Human Language Technologies (HLT) for Semitic Languages: Status, Up-dates, and Prospects*.