Building Dialectal Arabic Corpora

Hani A. Elgabou
Dept. of Computer Science
University of York
York YO10 5DD, UK
he583@york.ac.uk

Dimitar Kazakov
Dept. of Computer Science
University of York
York YO10 5DD, UK
dimitar.kazakov@york.ac.uk

Abstract

The aim of this research is to identify local Arabic dialects in texts from social media (Twitter) and link them to specific geographic areas. Dialect identification is studied as a subset of the task of language identification. The proposed method is based on unsupervised learning using simultaneously lexical and geographic distance. While this study focusses on Libyan dialects, the approach is general, and could produce resources to support human translators and interpreters when dealing with vernaculars rather than standard Arabic.

1 Introduction

The Arabic content on social media is increasingly becoming a mix of modern standard Arabic (MSA) and a collection of different dialects of the Arabic language. It is common to find a degree of this mixture even in Arabic news broadcasts, political dialogues and public events. While almost all mainstream Arabic NLP tools and research focus on MSA, the majority of Arabic dialects are barely touched. With more than 27 spoken varieties of Arabic dialects with a variable degree of intelligibility between them, the need for tools dedicated to dialect processing is essential.

Dialectal corpora would be of an interest to different applications of NLP and information retrieval in general. They would also be useful in building tools and resources, such as dictionaries and terminology databases (Trados, 2017), to aid human translators and interpreters adapt to the local variations of the Arabic language, with partial machine translation and automatic subtitling systems only becoming viable when a substantial body of resources is gathered. Dialect can be used to switch register, and it is not uncommon for Arabic speakers to alternate seamlessly between MSA and their dialects.

All this favours MSA translators and interpreters who have knowledge of the relevant dialects of Arabic, and an automated, large scale effort to provide some of the necessary training resources could play an important role.

The current trend in Arabic NLP regarding the lack of dialectal resources is to try to tackle this problem piecemeal, where researchers build custom tools and methods for a small subset of similar dialects, mainly with the aid of manually crafted datasets. While there is nothing wrong in following such an approach, repeating it for all Arabic dialects is a laborious task.

We believe that there is an alternative that could ease this problem. The proposed approach is based on the use of social media data and carefully crafted clustering and classification methods in order to identify local dialects and link them to geographic areas. The publicly available Twitter messages (also known as *tweets*) offer an opportunity to tag textual data by their geographic location or an approximation of it.

The current research on Arabic dialect classification (which we view as a special case of the task of language identification) only covers a small subset of broadly defined Arabic dialects: Egyptian, Levantine, Iraqi, Jordanian, Tunisian and the one spoken in the Gulf (Zaidan and Callison-Burch, 2014; Huang, 2015; Malmasi et al., 2015). The map in Figure 1 represents a simplified description of the geographic distribution of Arabic dialects, with the actual number of dialects being closer to 25. These, in turn can be further subdivided into variants of each dialect, thus forming a tree. Approaching the problem of dialect classification on a case-by-case basis is a laborious task, and alternatives are needed, as nowadays, language identification is at the heart of many NLP

The Proceedings of the First Workshop on Human-Informed Translation and Interpreting Technology (HiT-IT), pages 52–57,
Varna, Bulgaria, Sept 7, 2017
https://doi.org/10.26615/978-954-452-042-7_007

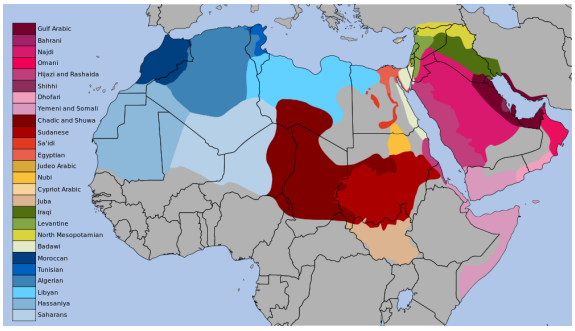

Figure 1: Geographic spread of Arabic dialects (Wikipedia, 2011)

tasks, such as machine translation, question answering and sentiment analysis. The limited capacity for Arabic dialect identification therefore has implications on the ability to carry out NLP for this language, in all its incarnations.

2 Previous Work

Previous work on Arabic dialect identification has mainly used supervised classification methods trained on manually annotated data of n-gram features at both word and character levels (Darwish et al., 2014). Zidan *et al.* (Zaidan and Callison-Burch, 2011, 2014) created a dialect dataset by harvesting reader comments from three local newspapers from three different countries, then used manually annotated data to train a classifier. In the same vein, Bouamor *et al.* (Bouamor et al., 2014) built a multidialectal Arabic parallel corpus of 2,000 sentences cross-translated by native dialect speakers. This dataset includes five dialects in addition to MSA and English.

Diab *et al.* (Diab et al., 2010) started the project COLABA, an effort to create resources and processing tools from dialectal Arabic blogs. COLABA employs human annotators to annotate dialectal words harvested from the blogs in question. The same authors developed DIRA, a term expansion tool for dialectal Arabic. Darwish *et al.* (Darwish et al., 2014) used Morfessor (Creutz and Lagus, 2005), an unsupervised morpheme segmentation tool, to add morphological features to the traditionally used lexical features. Recently, F. Huang from Facebook (Huang, 2015) has adopted a semi-supervised learning approach. Huang trained a classifier on weakly annotated data and another classifier on a small human annotated dataset, then used a combination of both to classify unlabelled data. The reported accuracy gain is 5% compared to previous methods.

Closer to our work is that of Mubarak and Darwish (Mubarak and Darwish, 2014) who show that Twitter can be used to collect dialectal corpora for different Arabic countries using geolocation information associated with Twitter data. They also built a classifier to identify different dialects with accuracy ranging from 95% for the Saudi dialect to 60% for the Algerian. They used a manually extracted list of dialectal n-grams to identify dialectal tweets. Their work is of a special interest for us, as it points out the possible challenges we might face. What differentiates our work is the way in which we collect our data (see below) and our aim to minimise the manual work by using unsupervised learning methods, namely, Expectation Maximisation (Dempster et al., 1977), in addition to supervised learning.

3 Clustering Twitter Data

Twitter data, i.e. tweets and their metadata, present opportunities for various text analysis applications. Tweets are short text messages, with a maximum length of 140 characters, posted by people on the micro-blogging website Twitter. Each tweet comes with its set of metadata fields and values, which contain information such as: the author, creation timestamp, the message, location and more. Table 1 lists some of these fields with their description. Figure 2 presents a detailed view of the data structure of a sample tweet.

Figure 2: Metadata of a sample tweet

[Field]	Description
[id]	The integer representation of the unique identifier for this Tweet.
[text]	The actual UTF-8 text of the Tweet.
[user]	The user who posted this Tweet.
[coordinates]	The geographic location of this Tweet.
[lang]	language identifier corresponding to the machine-detected language of the Tweet text.
[entities]	Entities mentioned in the text, could be other [user]s, hashtags and/or URLs.

Table 1: Metadata fields of tweets

At this stage we have a particular interest in three fields: $[user]$ to identify dialect speakers, $[text]$ to build our corpora and $[coordinates]$ to identify text by location. We also identified other fields potentially useful if we later chose to use information from social networking between the $[user]$s (Wang et al., 2010) to support our methods.

3.1 Data Collection

Our current primary data collection method is based on filtering the Twitter stream by geographic area. Using Twitter Stream API and its geographic bounding box filter, we are able to collect Tweets from a predefined geographic region. At this point, we are collecting Tweets from the geographic area of Libya as defined by a rectangular bounding box. Figure 3 shows a heatmap distribution of our Tweets data (approx. 700,000 Tweets to date), which is in line with the demographics of the country.

The Twitter API restricts free data collection to just 10% of its actual stream of data. Only paying accounts could get a 100% of the stream, in a package called the Firehose. Since we are using a free account, our data collection is limited to roughly 2700 Tweets a day. Although it would be better to have an access to the Firehose, we managed to overcome some of the limitations of the free data API. The Tweets we are currently collecting on a 24/7 basis are a welcome addition to our dataset, yet our primary aim is to collect as many relevant Twitter accounts as possible. Even one Tweet per account is sufficient. It is easy then to use the Twitter Stream API again to get many more Tweets from each account, with the average around 3000 Tweets per account. (In the future, it is also possible to extend our data by using the Twitter Search API to find Tweets containing already known dialect keywords.)

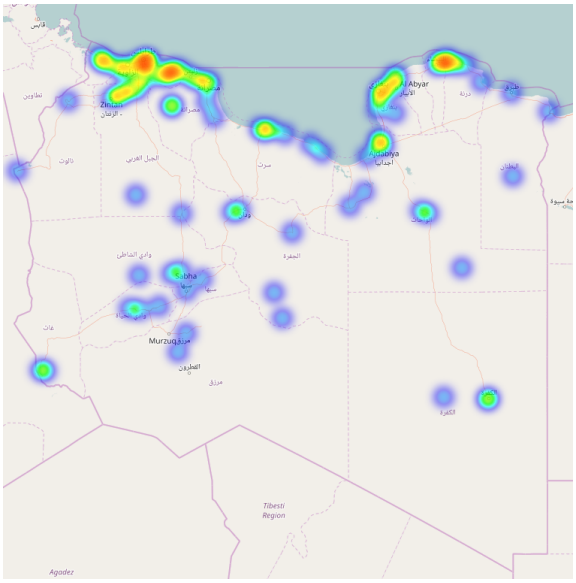

Figure 3: Libyan tweets distribution heatmap

3.2 Data Preprocessing

As with almost all text from social media, processing Tweets comes with its own set of challenges. Unlike some of the other Internet materials, social media including Tweets show a great variance in text quality. Text could come mixed with non-textual content, URLs, and can contain spelling mistakes and acronyms. Also, the restricted length of the Tweets text limits the amount of information available for text similarity measures (Phan et al., 2008), which are an essential part in most methods and applications of language processing (Hu et al., 2008), be it clustering or classification. The small size also makes it inefficient to represent Tweets using the traditional vector space model.

To tackle the problem of text impurity, we have already implemented python scripts that remove all non-Arabic alphabet text from our data. Since

we have no spell checking tools available for the majority of Arabic dialects, we rely on the assumption that the majority of misspelt words would be rare enough to be filtered out by a tf-idf weighting threshold. Since most of our text processing methods are largely dependent on data clustering and classification algorithms, we need an efficient representation of Tweets text that works well with different similarity measures. Although other research has dealt with this problem (Liu et al., 2011; Rosa et al., 2011; Tsur et al., 2013) with different level of efficiency, we decided it will be more natural and convenient for us to cluster the content of entire accounts rather than individual Tweets, as, after all, we are trying to identify different dialect speakers. Therefore, we merge all Tweets from each account into a separate text document and use the results as input to the clustering algorithm. When one user has tweeted from several locations, the most common one is used to provide the geographic coordinates for this account.

3.3 Data Clustering

At this stage we have only run a set of baseline clustering tests to help us understand the problem and the set of challenges we face. To reiterate, we treat the content of each account as a single document, therefore we cluster accounts rather than separate Tweets. This data representation also alows us to overcome the issue of very sparse vector representation of individual Tweets. Each account is represented by an n dimensional vector standing for the words with the n highest scores after the application of tf-idf weighting. Both k-means (MacQueen, 1967) and hierarchical clustering are to be used in order to tune the number of clusters k.

4 Preliminary Results

The results of clustering using geographic distances for $k = 3$ are shown in Figure 5, and appear consistent with the borders of Libyan provinces established since the Antiquity.

Where linguistic distances are concerned, the initial results show some interesting observations. The first observation is that setting the number of clusters k to 3 in k-means gives the most stable clustering results when repeated. The result corresponds well to the above mentioned outcome of spatial clustering, and, in the intuition of the first

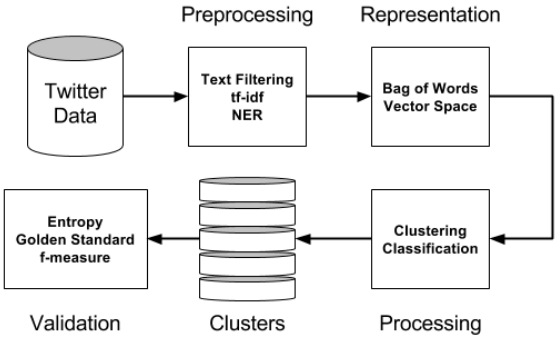

Figure 4: Text processing workflow

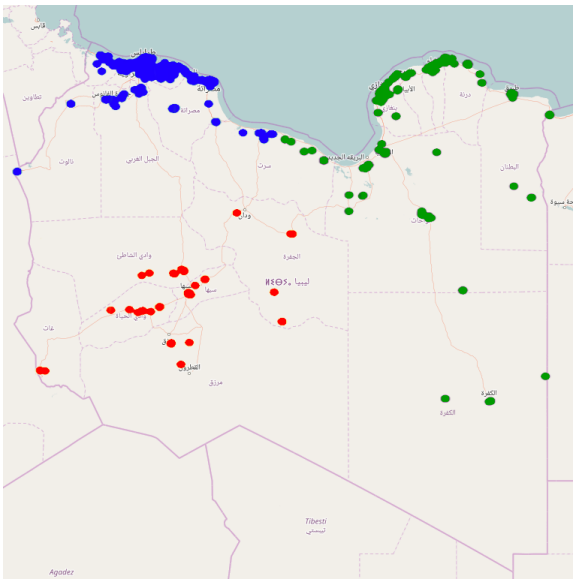

Figure 5: Spatial distance clustering map, $k = 3$

author, this number reflects well the number of main dialects in Libya. The second observation of note is that for $k = 3$, the centroids of the two largest clusters contain toponyms that represent well the geographic location of these clusters' members through words such as city names and places. This requires careful consideration though as a large number of local toponyms in the data could dominate all text features and create clusters that are naturally correlated with the geographic distribution of Tweets. We still need to establish whether removing the toponyms from the data has a significant effect on the composition of clusters and their spatial distribution.

5 Conclusion and Future Work

In our next set of experiments, we are planing to use the Mantel test (Mantel, 1967) in order to mea-

sure the correlation between the geographic distances and the lexical distances between pairs of accounts. Clearly, if this correlation was perfect, either set of distances would produce the same clustering. Using the clusters of one set of distances to generate the prior for the other clustering (e.g. using the cluster centroids of spatial clustering to seed k-means for the linguistic clustering step) and vice versa would produce an iterative algorithm that takes into account both metrics, which we are planning to study, along with the effect of different kernels/text features (e.g. word bigrams and part of speech bigrams) on the result. We plan to make our data and dialectal maps available for translators and researchers in general.

Acknowledgments

The authors want to thank the two anonymous reviewers, as well as the workshop chairs and Samy Hedaya for their helpful comments.

References

Houda Bouamor, Nizar Habash, and Kemal Oflazer. 2014. A Multidialectal Parallel Corpus of Arabic. In *the Proc. of the 9th International Conference on Language Resources and Evaluation*. pages 1240–1245.

Mathias Creutz and Krista Lagus. 2005. Unsupervised morpheme segmentation and morphology induction from text corpora using Morfessor 1.0.

Kareem Darwish, Hassan Sajjad, and Hamdy Mubarak. 2014. Verifiably Effective Arabic Dialect Identification. In *the Proc. of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. pages 1465–1468.

Arthur P Dempster, Nan M Laird, and Donald B Rubin. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)* pages 1–38.

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

Jian Hu, Lujun Fang, Yang Cao, Hua-Jun Zeng, Hua Li, Qiang Yang, and Zheng Chen. 2008. Enhancing text clustering by leveraging Wikipedia semantics. In *the Proc. of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, pages 179–186.

Fei Huang. 2015. Improved Arabic Dialect Classification with Social Media Data. In *the Proc. of the 2015 Conference on Empirical Methods in Natural Language Processing*. pages 2118–2126.

Xiaohua Liu, Kuan Li, Ming Zhou, and Zhongyang Xiong. 2011. Collective semantic role labeling for tweets with clustering. In *the Proc. of the 22nd International Joint Conference on Artificial Intelligence*. volume 3, pages 1832–1837.

James MacQueen. 1967. Some methods for classification and analysis of multivariate observations. In *the Proc. of the 5th Berkeley Symposium on Mathematical Statistics and Probability*. Oakland, CA, USA, pages 281–297.

Shervin Malmasi, Eshrag Refaee, and Mark Dras. 2015. Arabic dialect identification using a parallel multidialectal corpus. In *the Proc. of the International Conference of the Pacific Association for Computational Linguistics*. Springer, pages 35–53.

Nathan Mantel. 1967. The detection of disease clustering and a generalized regression approach. *Cancer Research* 27(2 Part 1):209–220.

Hamdy Mubarak and Kareem Darwish. 2014. Using Twitter to collect a multi-dialectal corpus of Arabic. In *the Proc. of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP)*. pages 1–7.

Xuan-Hieu Phan, Le-Minh Nguyen, and Susumu Horiguchi. 2008. Learning to classify short and sparse text & web with hidden topics from large-scale data collections. In *the Proc. of the 17th International Conference on World Wide Web*. ACM, pages 91–100.

Kevin Dela Rosa, Rushin Shah, Bo Lin, Anatole Gershman, and Robert Frederking. 2011. Topical clustering of tweets. In *Proc. of the ACM SIGIR 3rd Workshop on Social Web Search and Mining (SIGIR–SWSM)*.

SLD Trados. 2017. Terminology Management. [Online; accessed 15-August-2017]. http://www.sdltrados.com/solutions/terminology-management/.

Oren Tsur, Adi Littman, and Ari Rappoport. 2013. Efficient clustering of short messages into general domains. In *the Proc. of the 7th International AAAI Conference on Weblogs and Social Media*.

Xufei Wang, Lei Tang, Huiji Gao, and Huan Liu. 2010. Discovering overlapping groups in social media. In *the Proc. of the 2010 IEEE 10th International Conference - Data Mining(ICDM)*. IEEE, pages 569–578.

Wikipedia. 2011. Varieties of Arabic. [Online; accessed 22-July-2017]. https://en.wikipedia.org/wiki/Varieties_of_Arabic/.

Omar F Zaidan and Chris Callison-Burch. 2011. The Arabic online commentary dataset: An annotated dataset of informal Arabic with high dialectal content. In *the Proc. of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers-Volume 2*. Association for Computational Linguistics, pages 37–41.

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