Computer Stylometric Comparison of Writings by Qassim Amin and Mohammed Abdu on Women's Rights

Ahmed Ibrahim Ahmed Omer and Michael P. Oakes
Research Institute of Language and Information Processing
University of Wolverhampton
Wolverhampton, England
a.omer@wlv.ac.uk

Abstract

Computer stylometry is the computer analysis of writing style. We use the computer stylometric techniques of Hierarchical Cluster Analysis, Principal Component Analysis and Machine Learning to examine the authorship of "The Liberation of Women" which is normally attributed to Qassim Amin. In particular we examine the assertion by Mohamed Emara that certain chapters of this book were written secretly by Mohammad Abdu, who was the Grand Mufti of Egypt. In our experiments, we consistently find that Qassim Amin is the more likely author of the disputed text. The experiments described in this paper were done using the "Stylometry in R" package of Eder et al. (2016).

1 Introduction

Women's liberation was started in Egypt in the nineteenth century when Egypt was ruled by Mohammed Ali (1769-1849), and at that time the first school for training women was established. The school was used to train women to become medical assistants. Later Mohammed Ali opened the first primary school for girls. Ali sent many students to France to study there and to become leaders in different positions in the government. One of the students who was sent to France to do a Ph.D. degree was Qassim Amin. Qassim Amin finished his degree in law and traveled to France to stay there for about four years. Amin's traditional view of society was altered, and he started to see Egyptian women's lives through different eyes. He started to believe that life in the Egyptian community would not be improved unless the status of the women in society could improve. He believed that the main reason for their inferior position in the Egyptian community was ignorance about women and the lack of education. Accordingly, in 1899 he introduced his book "The Liberation of Women" (Amin, 2000) and used both rational Islamic arguments and emotional arguments to put forward his view. In his book, Amin called for women's education, removing the veil, and reformation of marriage and divorce laws. Mohamed Emara who was born in December 1931 is an Islamic scholar, author, investigator, and member of the Islamic Research Academy at Al-Azhar, Cairo. He did his Master's degree in Islamic philosophy at the Faculty of Science at Cairo University. He also got his Ph.D. from the same University in 1975. Mohamed Emara published many books about the life of many Islamic scholars including Jamal al-Deen al-Afghani, Abdul Razzaq Sanhouri Pasha, Sheikh Mohammed al-Ghazali, Rashid Rida and Muhammad Abdu. Emara stated in his book "Islam and Women in Mohammed Abdu's opinion" (Emara, 1997) that many chapters of Qassim Amin's book "Liberation of Women" were written secretly by Mohammed Abdu (1849-1905) who was a teacher of Qassim Amin and a religious scholar, jurist, and liberal reformer. In this paper we used techniques of computer stylometry to examine whether it is more likely that Mohammed Abdu wrote these chapters secretly or they were indeed written by Qassim Amin as generally thought.

2 Related Work

Discriminating between different authors is a challenging task, especially when there is a dispute about a text, and two or more authors have claimed that they have written this disputed text. Many studies in this field have tried to analyse the texts to find proof of authorship, and different approaches are used by researchers in the domain to find this evidence. A typical approach is finding the frequency of specific patterns appearing in the author's texts. These can be, for example, the frequency of words, the word lengths, or the sentence lengths. Shaker and Corne (2010) used a set of 104 function words to analyze texts derived from samples extracted from the website of the Arab Writers' Union (www.amu-dam.net). Their work was mainly inspired by Mosteller and Wallace's (1964) typology of English function words. By shifting their focus to function words, they were able to effectively capture the author's writing style regardless of the text's topic, as the use of these words in a text is normally unrelated to the topic, and yet there appear to be significant differences in the way different authors use them in writing. Ouamour and Sayoud (2012) tested different character and word features using an SMO-SVM classifier on text samples extracted from textbooks. These features included character bigrams, character trigrams, character tetragrams, single words, word bigrams, word trigrams, word tetra grams, and rare words. The texts were collected from ten different authors who wrote their texts in the domain of travel. Kumar and Chaurasia (2012) used character n-grams, especially bigrams and trigrams, to solve the problem of authorship verification. The authors investigated the bigram in different positions in words. They tested bigrams in the initial, the middle position, and at the end of words. The tests were conducted for both English and Arabic texts. The results showed that the initial bigrams and trigrams were the most useful in accomplishing the task. Howedi and Mohd (2014) examined many features to classify authors according to style. The authors used different linguistic features such as character bigrams, character trigrams, single words, word bigrams, and rare words. The dataset used in these experiments was the same data as used by Siham and Sayoud (2012), the AAAT Corpus of ten ancient Arabic travellers. Al-Zubaidi and Ehsan (2017) used the 300 most frequent features to predict the authors in Arabic texts. The dataset consisted of 18 books written by three old Arabic philosophers, Ibnjuzia, Sakhawy, and Tusi. Each one of the authors was represented by six books. Five books were used for training, and the sixth book was used for testing purposes. In this paper we used different linguistic features to discriminate between authors. To predict the authors of the texts, we represented each text sample by a vector. The vectors were numerical representations containing the frequency of the linguistic feature used in each sample. We then created a matrix where the columns corresponded to linguistic features (i.e. word frequencies), and the rows corresponded to individual texts. Finally, the classic Delta distance was used to calculate the distance between vectors. A matrix was produced of the distances between each pair of texts, and these distances were used as the basis of the clustering techniques we used, Hierarchical Agglomerative Cluster Analysis (HACA) and Principal Component Analysis (PCA). The initial vectors were also used as inputs to a set of machine learning techniques.

3 Corpus Description

To check whether the chapters included in the book "Liberation of Women" were written by Qassim Amin or Mohammed Abdu, we built a corpus containing their writings about women. The corpus contained three different sources. Two of these were written by Qassim Amin, and the last one was written by Mohammed Abdu. The first texts used in the experiments were extracted from the first book "The Liberation of Women" which were the chapters about women, and these were the chapters which Mohammed Emara stated were written by Mohmmed Abdu and included in Amin's book without any mention that they were Abdu's contribution to this book. Emara assumed that Amin was not qualified enough to discuss this topic and supported his view by citing new interpretations for verses from the Holy Quran in these chapters. He assumed that only Amin's teacher Mohammed Abdu was able to do this at that time. He also stated that the writing style of these chapters was more similar to Abdu's style than Amin's style. The second set of texts used in the experiments, written by Amin, were extracted from the book “The New Women”. This book, written later by Qassim Amin after the death of Abdu also spoke about women’s rights and asked the community to give women more rights. The last texts contained in the corpus were a collection of Mohammed Abdu’s texts which were written about the rights of women. The texts were extracted from a book written by Mohammed Emara to discuss Abdu’s opinions on women’s rights in Islam. The following table describes the texts used to build the corpus:

Book	Author	The extracted texts
The Liberation of Women	Qassim Amin	G_L1 to GL5
The New Women	Qassim Amin	G_N2 to GN5
Mohammed Abdu's Opinion on Women	Mohammed Emara	M_MAB2_1 to M_MAB2_5

Table 1: Corpus Description.

4 Experiments

In this case study we extracted various samples from each book to compare the style of the writing. Many linguistic features of the texts were then examined including the most frequent words, character 3 grams, and character 6 grams, to discriminate between Abdu and Qassim Amin. Two different methods for analysis were used to investigate the case which were Agglomerative Clustering, and Principal Component Analysis (Eder *et al.,* 2016). Machine learning techniques were also used to label the texts according to author.

4.1 Most Frequent Words Cluster

In this experiment we used the most frequent words as a feature set to discriminate between the two authors. This set of most frequent words is usually a set of the function words. This set can be used by the same author to write on two different topics because these words are topic independent. The most frequent hundred words were used to automatically cluster the texts according to style. The following graph shows that the texts which were extracted from the book “The Liberation of Women” was clustered beside the texts which were extracted from the book “The New Women” which confirm that these texts were written by Qassim Amin, and the texts written by Abdu were clustered in a different branch on the left as shown below:

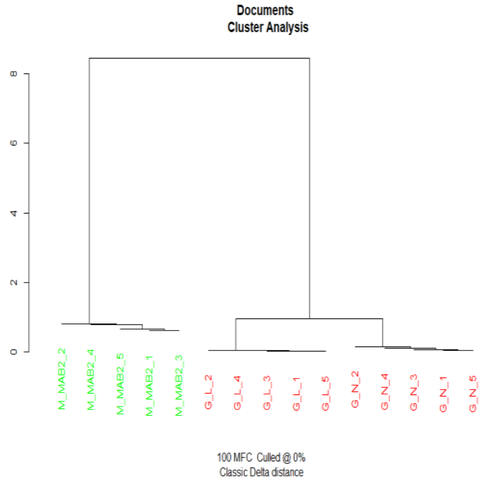


Figure 1: The most frequent words cluster

4.2 Character n-grams Clusters

In this experiment, we used the features character 3-grams and character 6-grams. We uploaded the corpus after doing the tokenization process to produce the corpus in the required format. The tokenizer made a window of a specific length and then cut the words into the required length assuming that the space between words is a character so that we can find a token consisting of one character from a word and another character from the next word, concatenated with a space. For example, the token [y m] can exist in the corpus among the character 3-grams tokens as a result of the two words (happy man). The following graphs show the results of the Agglomerative Clustering method using the features character 3-grams and character 6-grams.

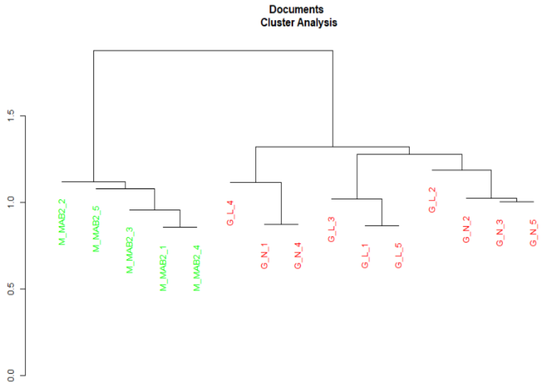


Figure 2: Character 3-grams cluster

From Figure 2 we can see that the texts from the two books which were written by Qassim Amin were clustered together while the texts from Mohammed Abdu were clustered in the leftmost branch of the tree. In addition, the samples from the two books written by Amin were mixed inside the cluster, and this shows that this feature was a very useful feature to capture the fingerprint of Qassim Amin. The same scenario occurred when we used character 6- grams as a feature set to discriminate between the two authors. Figure 3 below was produced by using character 6-grams as the feature set.

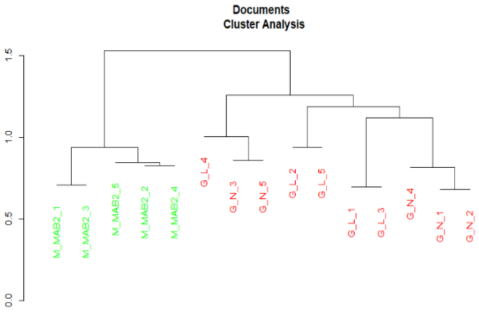


Figure 3: Character 6-grams cluster

4.3 Principal Components Analysis (PCA)

To confirm the results obtained by using the clustering technique using different feature sets, we used PCA analysis to find whether any groups of tokens occurred together in a specific group of documents more than the others so these could be used as a feature set to discriminate between the two authors. We ran the experiment using the 100 most frequent words as the feature set and found out that the first principal component was useful to sperate the two authors in a clear graph showing that the texts which were extracted from the book "The Liberation of Women" were more similar to the texts which were extracted from the book "The New Women" than the text by Muhammad Abdu. Figure 4 shows that the texts which were written by Amin are separated on the left-hand side and the features used to build the model were also shown in the graph. The most frequent words show that Mohammed Abdu uses the term "Man" (الرجل) while Qassim Amin use the term "Men" (الرجال) when they speak about the male gender.

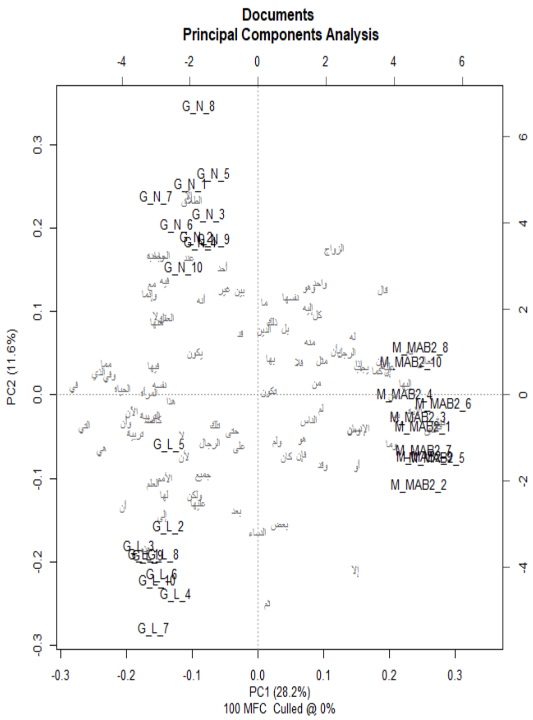


Figure 4: PCA using the most frequent words

In another experiment, we extracted the different morphemes which were available in the corpus by using the Farasa tools (http://alt.qcri.org/farasa/) to do word segmentation and the Stylo package (Eder et al., 2016) to find the most frequent features and used them as a feature set to discriminate between the two authors. The PCA showed that the different morphemes could be used to discriminate between the two authors as the texts of Qassim Amin were separated from Abdu's texts by the first principal component. The following list of morphemes (هن/ ل /وا /ب/ت/ / ف/ ال) were observed on Abdu's side while the following list of morphemes (ات/ نا/ها / ي/ة) were observed on Amin's side on the graph.

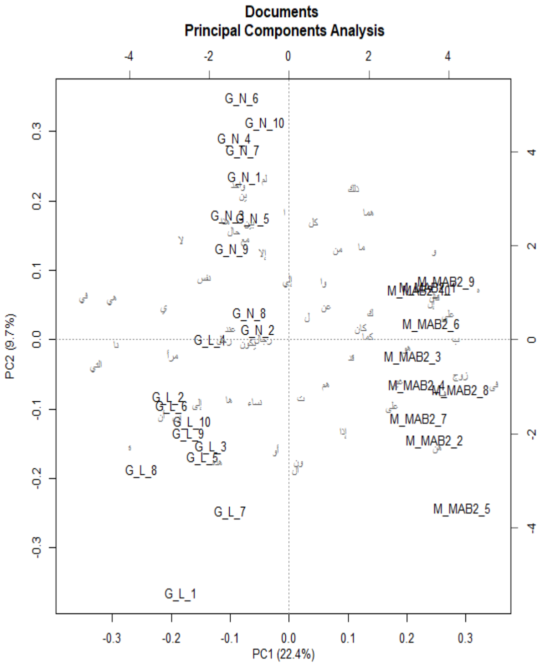


Figure 5: PCA using morphemes and Function words

4.4 Machine Learning Techniques

To check the results obtained by Agglomerative Clustering and PCA, we used machine learning techniques to find whither the chapters which were extracted from the book “Liberation of Women” would be classified or labelled as texts written by Qassim Amin or Mohammed Abdu. Ten samples were extracted from the book “The Liberation of Women” to be labelled by the classifier, and the texts from the book “The New Women”, besides the texts by Mohammed Abdu, were used as a training corpus to extract the useful patterns which could be used to predict the disputed texts. Five different machine learning classifiers were then used to predict the author of the ten samples, and the results confirmed that the sample texts were written by Qassim Amin. The following table shows the classifiers used together with the accuracy achieved using the different classifiers:

Classifier	**Words**	**Character 6-gram**
NSC	90%	90%
SVM	90%	90%
Naïve Bayes	80%	70%
Delta	90%	80%
KNN	80%	70%

Table 2: Classifiers Accuracies

This table shows the different accuracies obtained by using the different classifiers. For example, the accuracy 90% means that 9 samples out of the ten were classified as written by Qassim Amin and one sampled was classified as written by Abdu.

5 Conclusion

In this paper, we introduced a case study about Qassim Amin’s book “The Liberation of Women.” Qassim Amin was a liberal reformer who advocated giving women more rights in the Arab and Muslim communities. He assumed that the lack of education for women could affect not only women but also the whole society. Later some scholars including Mohammed Emara (Emara, 1997) reported that there are some chapters included in Amin’s book “The liberation of Women” which were written secretly by his teacher Mohammed Abdu. We decided to use the Computational Stylometry to investigate this argument and to find whither Abdu wrote these chapters or not. To do the experiment we built a corpus which contained different texts extracted from Amin and Abdu’s books. The disputed texts together with other texts extracted from the book “The New Women” were used to compare Amin’s style against the texts which were extracted from Abdu books on the same topic. Different features were used to investigate the texts’ writing style including the most frequent words, Character 3-grams, and Character 6-grams. The character n-grams were very useful features as they captured the fingerprint of the author. The extracted texts from the two books written by Amin clustered together in one branch in the clustering tree. To confirm the results obtained using the clustering technique we used Principal Component Analysis (PCA) to analyse the texts. The results confirmed that the disputed texts were written by Amin and not by Mohammed Abdu, and the texts from Amin were perfectly separated by the first principal component. We then used machine learning techniques to check whether we can successfully label the texts according to the author’s writing styles. To do that we extracted samples from “Liberation of Women” and used them as a testing corpus to be labeled by the classifiers. We also extracted samples from “New Women” together with texts written by Mohammed Abdu on women’s rights to form a training corpus. Five different classifiers were then used to find the fingerprint of the authors from the training corpus to build the model and to automatically predict the author of the texts available in the testing corpus. The results showed that the disputed texts were more similar to Qassim Amin’s style than Abdu’s style. In the future we would like to extend our experiments by adding more works written by Abdu and Qassim, possibly on different topics, as the most frequent words set of features is topic independent.

References

Abdelali, Ahmed, Kareem Darwish, Nadir Durrani, and Hamdy Mubarak. "Farasa: A fast and furious segmenter for arabic." In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations, pp. 11-16. 2016.

Mustafa, Tareef Kamil, Ammar Adil Abdul Razzaq, and Ehsan Ali Al-Zubaidi. "Authorship Arabic Text Detection According to Style of Writing by using (SABA) Method." Asian Journal of Applied Sciences (ISSN: 2321–0893) 5, no. 02 (2017).

Amin, Qasim. "the Liberation of Women." In Modernist and Fundamentalist Debates in Islam, pp. 163-181. Palgrave Macmillan, New York, 2000.

Amin, Qasim, Qāsim Amīn, and قاسم أمين،. The liberation of women: And, the new woman: Two documents in the history of Egyptian feminism. American Univ in Cairo Press, 2000.

Eder, Maciej, Jan Rybicki, and Mike Kestemont. "Stylometry with R: a package for computational text Analysis." R journal 8, no. 1 (2016).

Emara, Mohammed., 2004. Islam and Women in Mohammed Abdu’s Opinion. Dar al-rashad, Cairo.

Hadjadj, Hassina, and Halim Sayoud. "Towards an authorship analysis of two religious documents." In 2016 8th International Conference on Modelling, Identification and Control (ICMIC), pp. 369-373. IEEE, 2016.

Howedi, Fatma, and Masnizah Mohd. "Text classification for authorship attribution using Naive Bayes classifier with limited training data." Computer Engineering and Intelligent Systems 5, no. 4 (2014): 48-56.

Kumar, Sushil, and Mousmi A. Chaurasia. "Assessment on stylometry for multilingual manuscript." IOSR Journal of Engineering 2, no. 9 (2012): 1-6.

Ouamour, Siham, and Halim Sayoud. "Authorship attribution of ancient texts written by ten arabic travelers using a smo-svm classifier." In 2012 International Conference on Communications and Information Technology (ICCIT), pp. 44-47. IEEE, 2012.

Ouamour, Siham, and Halim Sayoud. "Authorship attribution of short historical arabic texts based on lexical features." In 2013 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery, pp. 144-147. IEEE, 2013.

Shaker, Kareem, and David Corne. "Authorship attribution in arabic using a hybrid of evolutionary search and linear discriminant analysis." In 2010 UK Workshop on Computational Intelligence (UKCI), pp. 1-6. IEEE, 2010.

Compiling and Analysing a Corpus of Transcribed Spoken Gulf Pidgin Arabic Based on Length of Stay in the Gulf

Najah Albaqawi, Michael Oakes
Research Group in Computational Linguistics, University of Wolverhampton

N.Albaqawi@wlv.ac.uk, Michael.Oakes@wlv.ac.uk

Abstract

The focus of this paper is on how to compile and analyse a transcribed spoken Gulf Pidgin Arabic (GPA) corpus with a specific focus on the influence of length of stay in the Gulf on foreign expat female speakers of GPA. GPA is a simplified contact variety of the Arabic language used in the Gulf states for communication between native Arabic speakers and foreign workers and among the workers themselves. This study provides a quantitative analysis of language variation in GPA based on five morpho-syntactic features that are related to the length of stay in the Gulf: definiteness and indefiniteness, coordination, copular verbs, pronouns, and agreement in the verb phrase and in the noun and adjective phrase. Digital recorders and planned interviews were used for collecting accurate naturalistic data. Through a comparative corpus-based analysis of approximately 72,000 words spoken by GPA female participants, evidence from this corpus data indicates that length of stay in the Gulf seems to have a little effect on informants 'choice between GPA linguistic variants. Newcomers and long-term resident GPA female speakers in the Gulf shift towards Gulf Arabic (GA), the lexifier language, in only two features: definiteness and use of conjunction markers.

1 Introduction

The field of corpus linguistics has gained huge popularity in recent years. It has become one of the most wide-spread methods of linguistic investigation not only among the experts, but also many researchers who would not consider themselves to be corpus linguists have begun to apply methods of corpus linguistics to their linguistic statements and assumptions. Joseph (2004:382) states: 'we seem to be witnessing as well a shift in the way some linguists find and utilize data – many papers now use corpora as their primary data, and many use internet data'.

GPA has received relatively little attention in the literature apart from a few descriptive works such as Albakrawi (2013); Albaqawi (2016); Alghamdi (2014); Almoaily (2008, 2012); Alshammari (2010); Al-Azraqi (2010); Al-Zubeiry (2015); Avram (2014, 2015); Gomaa (2007); Hobrom (1996); Næss (2008); Salem (2013); Smart (1990); and Wiswal (2002). In this paper, we are particularly interested in what a corpus of GPA spoken data, ideally in the form of recordings aligned with an orthographic Arabic transcription, might tell us about the use of language. Length of stay in the Gulf and GPA language variation will be examined in this study from a sociolinguistic point of view, since the study of linguistic variation in contact languages can make a valuable contribution to the field of sociolinguistic variation and change. Traditionally, researchers in sociolinguistics were not interested in using corpora in their investigations (Baker, 2010:1) until 1996 when McEnery and Wilson suggested a first possible relation with corpora. They showed the value of supplementing the qualitative analysis of language with quantitative data (McEnery & Wilson, 2003). In 2006, McEnery et al. also indicated that along with the speed of information processing, there are specialised software which can classify and select words to look at their frequencies between major classes, for example, male and female usage.

This paper will start with a brief definition of pidgin and the situation of pidgin Arabic in Saudi Arabia. This will be followed by a discussion of compiling and analysing a spoken variety of Arabic, GPA, and the difficulties associated with that. Then we will analyse the impact of the number of years of residency by Asian female workers located in the Gulf as a potential factor conditioning language variation in GPA. The final section will provide some conclusions and suggestions for future studies on GPA.

2 Definition of Pidgin

In this section we will try to give a simple definition of pidgin and creole, regardless of the diverging views in defining these two contact language types.

Pidgin: Pidgin is defined by Velupillai (2015) as "a language that emerges when groups of people are in close and repeated contact, and need to communicate with each other but have no language in common". McWhorter (2001, 2004) also defined pidgins as the languages that result from maximal contact and adult language learning, and their speakers use them as "transitory tools" for passing exchanges. If people use this simplified version of language, pidgin, as an everyday language, a pidgin can become a real language, a creole.

Usually a pidgin language is a blend of the vocabulary of one major language (i.e. language of the dominant group which is referred to as the 'lexifier' or 'superstrate', in our case GA) with the grammar of one or more other languages (i.e. languages which are spoken by groups with lesser social status to the lexifier speakers which are referred to as the 'substrate languages'. In our case they are from the following six different language groups: Tagalog, Punjabi, Sinhala, Malayalam, Sunda, and Bengali).

3 The Situation of Pidgin Arabic in the Gulf States

The situation in which GPA was developed is a textbook case of the situations that create a pidginised variety. Sakoda and Siegel (2003:1) write:

> Nowadays, the term "pidgin" has a different meaning in the field of linguistics. It refers to a new language that develops in a situation where speakers of different languages need to communicate but don't share a common language.

According to their definition, the situation in the Gulf States is ideal for the birth of a new contact language as the Arab Gulf States are located in the centre of the Old World[1]. Following the October 1973 "oil boom," the Arab Gulf States (GCC)[2] have experienced radical social, political and demographic changes in a very short time. This has led to an extremely rapid increase in the demand for foreign labour. The number of foreign labourers in the Gulf countries, especially the Kingdom of Saudi Arabia, rapidly increased, amounting to almost 4.4 million in 1985, a more than three-fold increase within a single decade. Also, the kingdom is the biggest economy in the Arab world, endowed with the world's second largest proven oil reserves. This makes Saudi Arabia a major hub for population movements (De Bel-Air, 2014). Saudi Arabia, as stated by Avram (2013b), has a multilingual setting as do all Gulf countries; Gulf Arabic (GA) is a form of Colloquial Arabic language spoken by the indigenous people of the Gulf Region. Migrant workers, who come from various linguistic backgrounds and usually do not speak Arabic, come into contact with GA speakers as well as speakers of other Arabic dialects, and there is an urgent need for communication between the two groups, "Arabic-speaking locals and expats on one hand and non-Arabic speaking expats on the other" (Almoaily, 2012, p. 1). Thus, a simplified form of Arabic has developed as a result of this

[1] Some geographers use the term Old World to refer to Asia, Africa, and Europe (see Mundy, Butchart, and Ledger 1992).

[2] Gulf Cooperation Council, which includes: Saudi Arabia, Kuwait, Bahrain, Qatar, United Arab Emirates, and Oman

contact which is known as ‘Gulf Pidgin Arabic’ (henceforth GPA). GPA is a reduced system of language that is used for communication between foreign workers and the native speakers of Arabic. Indeed, GPA and GA are two distinct forms of language, with lexical, phonological, syntactic, and morphological differences. At the level of phonology, Albaqawi (2016) who conducted a study that investigated the phonetic variation within GPA spoken by Asian migrant workers in the Gulf countries concluded that the basic GPA phonetic inventory is either reduced or simplified and differences in phonology are limited in GPA varieties. However, one vital question should be asked: Does a local speaker use GPA when he/she is speaking to a GPA speaker? To answer this question, Almoaily (2008) asked 77 Saudi respondents if they ‘don’t mind using GPA with speakers who are not fluent in GA’. Half of the Saudi respondents agreed to use GPA with non-Arabic speaking foreigners (especially among the younger generation of locals) and the other half either disagreed or strongly disagreed with this statement. He also claimed that locals’ use of GPA when speaking to GPA speakers was higher than 50%.

However, this issue is still a controversial and it depends on the quantity and quality of input which GPA speakers are daily exposed to the superstrate language, GA.

4 Corpus and Methodology

When spoken language is addressed, traditionally, a corpus linguistics work starts with deriving an orthographic transcription from a recording of large stretches of speech. The main aim of building a spoken language corpus is to acquire large amounts of data reflecting the natural use of language. Thus, all the data in this research are collected via interviews with female informants who do not speak Gulf Arabic as their first language. In order to examine such data, a quantitative variationist analysis of GPA variability was used. In this study we attempt to provide a quantitative analysis which aims to discover the potential effect of the number of years spent in the Gulf on variability in GPA morpho-syntax.

4.1 The Corpus

The corpus consists of the speech of informants participating in interviews which were conducted in Saudi Arabia[3]. To test the influence of the length of stay on the GPA female speakers’ language variation, face-to-face recorded interviews were conducted between the subjects and the interviewer (the first author) by using a high-quality digital voice recorder[4] and ranged from 16 to 27 minutes. The data-base consists of interviews with 72 GPA speaking female informants from six linguistic backgrounds (Malayalam, Punjabi, Bengali, Tagalog, Sunda, and Sinhala) as these substrate languages are the largest number of speakers in Saudi Arabia based on the results of the Population Census from De Bel-Air (2018). Half of the data was produced by informants who have spent five years or less in the Gulf while the other half had spent ten years or more in the Gulf at the time the researcher interviewed them. This paper seeks to investigate whether the long-term residents have actually shifted towards GA or not. The structural patterns of GA that were collected from the newcomers of each language group were compared with those of long-term residents (e.g. newly-settled Tagalog speakers vs. Tagalog speakers who spent more than a decade in the Gulf). In other words, we compared their proportional use of GA tokens of the morpho-syntctic phenomena investigated in this study: Arabic definiteness markers (i.e. the prefix *al-*), Arabic conjunction markers (these markers are mostly the free morphemes *wa* ‘and’, *laakin* ‘but’, and *aw* ‘or’), object or possessive pronoun (i.e. subject pronouns in GA are the free morphemes , 1SG *ana* whereas object and possessive pronouns are always bound morphemes, 1SG -i.), copula (i.e. the GA copula *fi* is used overtly only in the past and future whereas it is covert in the present tense), and agreement in the verb phrase and the noun phrase with that produced by their newly settled counterparts. We opt to examine these morpho-syntactic features as we believe that they are adequate to test the proposed typological features (reduced inflection; reduction of agreement markers in verb and noun and adjective agreement, and reduced inventory of function words; copulas, definite and indefinite

[3] All the interviews were conducted in the Saudi Central Province where Najdi Arabic – a sub-dialect of GA – is spoken

[4] Olympus vn-7800pc

articles, and pronouns) that might be found in all pidgin and creole languages worldwide (irrespective of their input languages) see Almoaily (2013); Bakker (1995, 2003); Roberts and Bresnan (2008); Bakker, Daval-Markussen, Muysken, and Parkvall (2011); Sebba (1997); and Siegel (2004).

Counting the lexical features has been excluded for two reasons: First, the purpose of this paper is to examine the structure of GPA rather than its lexicon. Second, vocabulary studies are more related to developed languages. For example, Malmasi, et al. (2016) identified a set of four regional Arabic dialects (Egyptian, Gulf, Levantine, North African) and Modern Standard Arabic (MSA) which are all native languages unlike the GPA which has no natives.

General principles for the quantification of variability above the level of phonology are still a matter of debate (Macaulay, 2002). A number of researchers have come up with several approaches for the quantification of tokens. Some quantify them by the number of words as was done by Precht (2008) and Cheshire, Kerswill and Williams (2005). On the other hand, some researchers prefer to quantify them per minute or hour of speech in a sociolinguistic interview, as was done by Rickford and McNair-Knox (1994). In our case, we preferred to calculate the tokens per number of words as Almoaily (2013) suggested, irrespective of the length of the turn or the number of words produced in a minute of speech. Our reason was that the informants of our study have been exposed to GPA over a period ranging from eight months to twenty-five years, and newly arrived speakers are expected to speak slower than those who have spent more than ten years in the Gulf.

4.2 Transcribing the Interviews

The first author transcribed the interviews herself. It took nearly three hours to transcribe and revise only ten minutes of speech. She used Express Scribe Transcription Playback Software[5]and transcribed that segment of the interview manually[6] (since the transcriptions of the whole interviews are in Standard Arabic script).

4.3 Annotation of Counting the Tokens

In the corpus each variant of a variable is labelled with a unique code[7]. The example below shows a code and its meaning:

Code: (+روابط)/ (- روابط)

Meaning: The conjunction marker is present (CONJ +)/ The conjunction marker is dropped (CONJ -).

In order to count and retrieve the tokens from the transcribed interviews, we used the AntConc software[8]. AntConc is one of the best tools for analysing a corpus. Froehlich (2015) refers to AntConc as a very good toolkit for finding patterns in language which would be difficult to identify just by reading the text. Figure 1 below shows how a transcribed interview appears with the AntConc program:

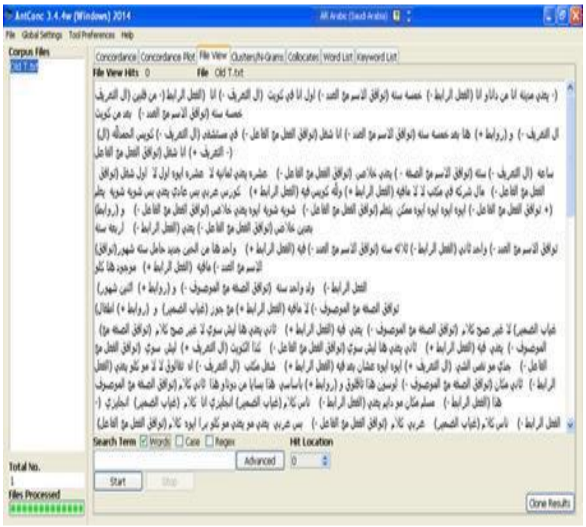

AntConc 3.4.4w (Windows) 2014
File Global Settings Tool Preferences Help
Corpus Files
Old T.txt
Concordance Concordance Plot File View Clusters/N-Grams Collocates Word List Keyword List
File View Hits 0
File Old T.txt
Search Term ☑ Words □ Case □ Regex
Hit Location
Advanced
0
Start
Stop
Total No.
1
Files Processed
Clone Results

Figure 1: Old Tagalog corpus

We tried to find the frequency of occurrence for every linguistic feature chosen in the study (e.g. definiteness). The Concordance view showed whenever the chosen linguistic feature (e.g. definiteness) appeared in our corpus (e.g. Tagalog corpus newcomers) and some context of it (such as a window of x words). We did the same for all the corpus files that we had. We then calculated the percentage of tokens produced in every variant.

[5] Professional audio player software designed to assist the transcription of audio recordings (Free, cross- platform).

[6] Arabic transcription/dictation software tools for non- standard Arabic varieties or Arabic-based contact languages are inaccurate and thus were avoided in transcribing the data for the current study.

[7] We used Arabic characters as a unique code.

[8] A freeware corpus analysis toolkit for concordancing and text analysis.

To compare the use of the given variant by members of a sub- group with that of other sub-groups (e.g. newly-settled female Tagalog speakers vs. long-term female Tagalog residents), the researchers used statistical analysis to look at the differences between two corpora.

This was used to establish the significance of the effect of the years of residency in the Gulf on variation in GPA.

5 Issues in Compiling and Analysing a Spoken Corpus

This study depends on using a suitable corpus and since GPA is only a spoken variety of the Arabic language, there was no such corpus previously available in electronic form. In addition, the GPA corpus is different from Arabic Learner Corpus as in Alfaifi and Atwell (2013). In the Arabic Learner Corpus, the students are all trying to learn Standard Arabic, while in the GPA corpus, the target language, whether GA or GPA, is a matter of debate. Thus, we had to design and build our own corpus. A number of difficulties and challenges were associated with implementing such a corpus. These included, size, balance (choosing informants), representativeness, and annotation. We will discuss the question of annotation here.

Annotation: Annotating a corpus written in Arabic script presents challenges. Many dialects are written in different scripts, have no conventions for spelling and no large body of literature. In our case we have "code-mixed" text, interspersed with other languages (Arabic and English). As a first attempt, we labelled each variant of a variable with a unique Roman code (e.g. CONJ+ if the conjunction is used and CONJ- if the conjunction is dropped). This attempt failed because the AntConc software was not able to detect accurately the linguistic code switching within Arabic script text as Arabic script starts from right-to-left where English Roman script starts from left-to-right. To overcome these systematic changes in writing direction, we decided to retranscribe all our corpus files in a unified spelling system by using Arabic code instead of Roman code for the annotation (e.g. الفعل الرابط- the copula is used and الفعل الرابط+ the copula is dropped). This revised annotation works very well and it has been adopted in the main corpus.

6 Results and Discussion

6.1 New versus Old participants

Each language group was split into two groups based on their length of stay in Saudi Arabia, or any other GA speaking country (5 years or less—referred to as "New" or 10 years or more—referred to as "Old"). Chi-squared tests were run to establish the significance of the effect of years of residency in the Gulf on variation in GPA.

Results of simple concordance comparisons of the new and old participants are presented in Table 1. Comparing the percentages of occurrence of each variable gives us the opportunity to contrast the proportion of use of the GA variants as opposed to the proportion of use of the GPA variants.

These were variants in definiteness, conjunction markers, the copula, object and possessive pronouns and agreement in the VP and in the NP and in the ADJP presented in table 1 and Figure 2:

GA Linguistics Feature	GPA Informants	
	new	old
Definiteness	10.7%	33.7%
Conjunction Markers	12.9%	41.5%
Copula Fi	54.7%	59.2%
Object and Possessive Pronouns	18%	22.9%
Subject-Verb Agreement	5.2%	8.4%
Nominal Agreement	19.3%	26.6.%

Table 1: Concordance and percentage used in the corpus of the new and old participants

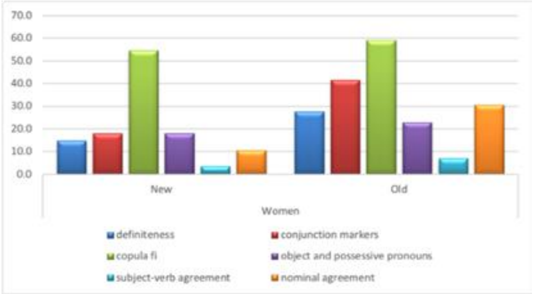


Figure 2: Data Comparison between New and Old GPA speakers

6.2 Variation in Definiteness

We noticed a possible link between the length of stay in GA speaking countries and the use of the definiteness marker *al-*. This shift towards GA was seen in all six language groups. The newly-arrived GPA speakers produced the definiteness marker in 10.7% of the cases, whereas the old members produced them in 33.7%. The chi-squared test revealed that the difference between the new informants and those who stayed longer in the Gulf in producing definiteness markers is significant at a p-value of 0.002.This noticeable shift towards using the GA definiteness marker among the long-term residents could potentially be a result of the fact that definiteness in GA is one of the morpho-syntactic features that are easiest to learn as it only involves adding the prefix *al* – or one of its allophones – to the target noun.

6.3 Variation in the Use of Conjunction Markers

The data reveals a major shift towards GA in the use of conjunction markers. This effect was seen in all six language groups. The newly-arrived GPA speakers produced conjunction markers in 12.9% of the cases, whereas the old informants produced them in 41.5%. The chi-square test reveals that the difference between the new informants and those who stayed longer in the Gulf in producing conjunction markers is significant at a p-value of 0.001. This significant difference could be due to the fact that learning GA conjunction markers is not hard. GA conjunction markers are free morphemes (e.g. *wa* ‘and’, and *aw* ‘or’). This result is in parallel with Almoaily (2013)’s study of male GPA speakers.

6.4 Variation in the Use of the Copula

In GA, there is no copula in the present tense. Thus, the focus here is on the use of the copula *fi* in the present tense in GPA. If long-term residents are found to drop the copula more than the newcomers, this might be an indication of a shift towards GA. The data reveals that the relation between the years of stay and the shift towards GA seems to be slightly negative at a p-value = 0.35. Overall, there is no significant shift towards Gulf Arabic in the data of speakers participating in this study regarding the use of a copula, as new speakers dropped it on average 54.7% of the time and old speakers dropped it in 59.2% of the time.

6.5 Variation in the Use of the Object and Possessive Pronouns

GA personal pronouns inflect for number, person, and gender. In GPA, there are four variants for object and possessive pronouns: GA bound pronoun which agrees with the noun in person, number, and gender, GA bound pronoun which does not agree with the noun, free pronoun, and dropping the object or possessive pronoun. On average, newly-settled informants in all six language groups produced bound object and possessive pronouns in 24.2% of the cases, while the long-term residents produced them in 49.2% of the cases. The difference is significant at a p-value of 0.0001. Note that the newcomers produced tokens of pronouns in free forms 71% of the time and the old group counterparts produced them 75.5% of the time. In fact, this high rate of free object and possessive pronouns indicates that the overall shift is clearly not towards GA (bound pronouns) but GPA (free pronouns). Since this feature (free pronoun) is found in the informants’ L1s, it could probably have some influence on GPA speakers and lead them to learn it at the first stage as reported in Almoaily (2012).

6.6 Variation in Subject-verb Agreement

In Gulf Arabic, the verb inflects for gender, number, tense, and person (Feghali, 2004). The data shows that there is a positive development related to the informant’s length-of-stay in the use of verbs: members of the new group tend to drop verbs more frequently (35.6%) than their old group counterparts (8.4%). The rate of dropping the verb is significantly higher in the data of new informants at a p-value of 0.0002. However, it seems that there is no development in acquiring agreement in the GA verbal system. Overall, the data revealed that all of the informants rarely produced the form of the verb that is used in GA (i.e. fully inflected verb forms that agree with the subject in number, gender, and person). Compare the overall percentage of new-comers who produced a fully inflected GA verb only in 5.2% of the total number of tokens, with that of old informants who produced it in 8.4%. Yet, the difference is not significant (p-value= 0.22). Note, the overall shift is clearly not towards GA, as the use of forms of verb markers which do not agree with the noun in gender, number, and person are predominant in the data of both new and old speakers.

6.7 Agreement in the NP and in the ADJP

In GA, the adjectives agree with the head noun in gender, number, and definiteness (Feghali 2004, Smart 1990, Almoaily 2008). The data show that there is a little positive improvement in the occurrence of nominal agreement by the participants who stay long in the Gulf as compared to their new counterparts. We have noticed that long-term residents produce a few more tokens of noun-adjective agreement in number and gender than their new counterparts. The new informants produced agreement tokens in 19.3% of the total number of cases, while their long-term counterparts produced it in 26.6 % of the total number of cases. Although the difference is not statistically significant (p-value = 0.08), and even though there is obviously a vast amount of variation within the groups, there seems to be a trend towards the acquisition of GA norms.

7 Conclusion

The main aim of this study was to examine how to build and analyse a spoken corpus for a sociolinguistic investigation. Indeed, we expected to face difficulties when deciding on size, balance, representativeness, and annotation of our spoken corpus. Compiling and analysing the corpus for this investigation were the most demanding task and time-consuming task (see section 5). First, choosing GPA speakers who meet certain criteria and convincing them to participate in the interview was not an easy task. Many simply refused to be interviewed and many others were too busy to take part in this study. Also, transcribing the interviews and choosing the appropriate transcription protocol for Arabic script presented greater challenges. The strategy we employed to overcome, or lessen the impact of these problems was by transcribing all our corpus files in a unified spelling system by using Arabic code instead of Roman code. It was very fruitful technique (see section 4).

The study also was aimed to investigate language variation in GPA resulting from the speakers' length of stay in the Gulf. The analysis suggests that this factor seems to have a little effect on informants 'choice between GPA linguistic variants. We expected long-residence speakers to produce more GA tokens than the newly-settled GPA speakers. They have made a significant shift to GA after spending ten years in the Gulf in two linguistic features only: definiteness and conjunction (p-value=0.002, p-value=0.001) respectively. There are some factors which we believe could have had an effect on the informants' choice between the selected features' variants. This could potentially be a result of the fact that most of the informants are female maids living with a local family who mostly use GA when communicating with them, which could play a major role on the process of acquiring a language. This in turn leads them to rapidly learn the language of the host community and effortlessly adopt the system of Gulf Arabic (the target language). Another effect on the informants' choice between the selected feature variants is that it may depend more on the amount of GA input that GPA speakers receive during their stay in the Gulf (rather than the language of origin), different learning abilities to learn a language, and motivation.

We conclude this study with a set of recommendations for future research on this pidgin language. First, we suggest considering the role of input in pidgin formation. Second, we will conduct a comparison study to investigate male and female GPA production and effect of the language of the origin. Finally, it would be fruitful to conduct and computationally analyse more data-based studies of Arabic-based pidgins which are less known in the literature of non-Indo-European pidgin languages.

References

Abdullah Alfaifi and Eric S. Atwell. 2013. Arabic learner corpus v1: A new resource for arabic language research. Leeds.

Abdul-Qadir Wiswall. 2002. Gulf pidgin: An expanded analysis, *unpublished pro-seminar paper,* Ohio State University.

Andrei A. Avram. 2013b. On the periphery of the periphery: Gulf Pidgin Arabic. *Proceedings of 10th Conference of Association* Internationale *de Dialectologie Arabe.* Qatar University, Doha.

Andrei A. Avram. 2015. On the developmental stage of Gulf Pidgin Arabic. *In Arabic varieties: Far and wide. Proceedings of the 11th International Conference of AIDA* (p.87-98). Bucharest, Romania.

Andrei A. Avram. A. 2014. Immigrant Workers and Language Formation: Gulf Pidgin Arabic. *Lengua y Migración,* 6 (2). 7- 40.

Ashraf A. Salem. 2013. Linguistic Features of Pidgin Arabic in Kuwaiti. *English Language Teaching*, 6 (5). 105-110.

Brian Joseph. 2004. On change in Language and change in language. *Language*, 80(3), 381-383.

David. Evans. 2007. Corpus building and investigation for the Humanities. *University of Nottingham* http://www. corpus. bham. ac. uk/corpus-building. shtml.

Emad A. Alghamdi. 2014. Gulf Pidgin Arabic: A Descriptive and Statistical Analysis of Stability. *International Journal of Linguistics* 6, no. 6, p.110.

Françoise De Bel-Air. 2014. Demography, Migration and Labour Market in Saudi Arabia. *Gulf Research Center Knowledge for All.* Retrieved from http://gulfmigration.eu/media/pubs/exno/GLMM_EN_2014_01.pdf.

Françoise De Bel-Air. 2018. Demography, migration and labour market in Saudi Arabia.

Gisle Andersen. 2010. How to use corpus linguistics in sociolinguistics, in O'Keeffe A. & M. McCarthy (ads.), *The* Routledge handbook of corpus linguistics, 547-62. 1" ed. London: Routledge.

Gomaa, Y. 2007. Arabic Pidginization: The Case of Pidgin in Saudi Arabic. *Journal of the Faculty of Arts, 19*, 85-120, Assiut University, Egypt.

Habaka J. Feghali. 2004. Gulf Arabic: the dialects of Riyadh and eastern Saudi Arabia: grammar, dialogues, and lexicon. *Springfield, VA, Dunwoody Press.*

Hameed Y. Al-Zubeiry. 2015. Linguistic Analysis of Saudi Pidginized Arabic as Produced by Asian Foreign Expatriates. *International Journal of Applied Linguistics & English Literature*, 2(4), 47-53.

Heather Froehlich. 2015. Corpus analysis with AntConc. *Programming Historian*.

Hussien Albakrawi. 2012. The linguistic effect of foreign Asian workers on the Arabic Pidgin in Saudi Arabia. *language* 2, no. 9.

Jack R. Smart. 1990. Pidginization in Gulf Arabic: A first report. *Anthropological Linguistics,* 32, 83-118.

Jeff Siegel. 2004. Morphological Simplicity in Pidgins and Creoles, *Journal of Pidgin and Creole Languages,* 19: 139–62.

Jenny Cheshire, Paull Kerswill, and Ann Williams. 2005. Phonology, Grammar and Discourse in Dialect Convergence. In: Peter Auer, France Hinskens, and Paull Kerswill (eds). *Cambridge: Cambridge University Press.*

John H. McWhorter. 2004. The story of human language (Course Guidebook). Chantilly, VA: *The Teaching Company.*

John R. Rickford and Faye McNair-Knox. 1994. Addressee and Topic Influenced Style Shift: a quantitative sociolinguistic study. In: Douglas Biber and Edward Finegan, (eds). Oxford: *Oxford University Press.*

Kent Sakoda and Jeff Siegel. 2003. Pidgin grammar: An introduction to the creole English of Hawai'i. Honolulu, *Hawaii: Bess Press*.

Kristen Precht. 2008. Sex Similarities and Differences in Stance in Informal American Conversation, *Journal of Sociolinguistics*, 12 (1) 89-111.

Mark Sebba. 1997. Contact Languages: pidgins and creoles. *London, Macmillan*.

Mohammad Almoaily. 2008. A data-based description of Urdu Pidgin Arabic. Unpublished MA dissertation, *Newcastle University*.

Mohammad AlMoaily. 2012. Language Variation in Gulf Pidgin Arabic (Doctoral dissertation). *Newcastle University,* the United Kingdom.

Munira Al-Azraqi. 2011. Pidginisation in the eastern region of Saudi Arabia: Media presentation. In *Arabic and the Media*, pp. 159-173. Brill.

Najah S. Albaqawi. 2016. Unity and Diversity within Pidginized Arabic as Produced by Asian Migrant Workers in the Arabian Gulf.

Natalie Schilling-Estes. 2007. Sociolinguistic Fieldwork. In: Robert Bayley and Ceil Lucas, (eds.) Sociolinguistic Variation Theories, Methods, and Applications: *Cambridge: Cambridge University Press.*

Peter Bakker, Aymeric Daval-Markussen, Mikael Parkvall, and Ingo Plag. 2011. Creoles are Typologically Distinct from Noncreoles, *Journal of Pidgin and Creole Languages*, 26 (1) pp. 5-42.

Peter Bakker. 1995. *Pidgins*. In Jacques Arends, Pieter Muysken , and Norval Smith (eds) Pidgins and creoles an introduction.1995. *Amsterdam; Philadelphia: J. Benjamins.*

Peter Bakker. 2003. Pidgin inflectional morphology and its implications for creole morphology, *Yearbook of Morphology*, Part 1, 3-33.

R Core Team. 2013. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL http://www.R-project.org/.

Robert Bayley and Dennis R. Preston. 1996. Second Language Acquisition and Linguistic Variation. *Amsterdam: Benjamins*.

Ronald Macaulay. 2002. Discourse Variation. In: Jack K. Chambers, Peter Trudgill, and Natalie Schilling-Estes (eds.) The handbook of language Variation and change. *New York: Blackwell Publishing Co.*

Sean Wallis. 2014. What might a corpus of parsed spoken data tell us about language. In *Complex Visibles Out There. Proceedings of the Olomouc Linguistics Colloquium* (pp. 641-662).

Sarah J. Roberts and Joan Bresnan. 2008. Retained Inflectional Morphology in Pidgins: A typological study, *Linguistic Typology* 12: 269–302.

Shervin Malmasi, Marcus 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 *Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial3)* (pp. 1-14).

Tony McEnery, Richard Xiao, and Yukio Tono. 2006. Corpus- based language studies: An advanced resource book. Taylor & Francis.

T-Wilson McEnery and A Wilson. 1996. Corpus Linguistics Edinburgh: Edinburgh University Press.

T-Wilson McEnery and A Wilson. 2003. Corpus linguistics. The Oxford handbook of computational linguistics, 448-463.

Viveka Velupillai. 2015. Pidgins, Creoles and Mixed Languages. An Introduction. Amsterdam and Philadelphia: *John Benjamins*.

Wafi Alshammari. 2010. An Investigation into Morpho-syntactic Simplification in the Structure of Arabic Based Pidgin in Saudi Arabia (Master's theses). *Mu'tah University, Jordan*.

William Labov. 1972. Language in the Inner City; studies in the Black English vernacular. Philadelphia: *University of Pennsylvania Press*.

Writing Styles of Salwa and Al-Qarni

Ahmed Ibrahim Ahmed Omer and Michael P. Oakes
Research Institute of Language and Information Processing
University of Wolverhampton
Wolverhampton, England
a.omer@wlv.ac.uk

Abstract

This paper follows a recent court case which judged that “Don't despair” (لا تيأس) by Aaidh ibn Abdullah Al-Qarni was plagiarized from Salwa Aladian’s “That is how they defeat Despair” (هكذا هزموا اليأس). We use techniques of computational stylometry, Hierarchical Agglomerative Clustering Analysis and Principal Components Analysis, to show that the disputed sections not only resemble Salwa’s work in content, but also in writing style.

1 Introduction

Aaidh ibn Abdullah al-Qarni was born in 1960 in Saudi Arabia. He graduated from Imam Muhammad Ibn Saud University and became one of the most respected scholars in his country. Later he published more than 80 books in a very short time and became a very famous writer. In 2003 he introduced his book “Don’t be sad” (لا تحزن) and it was one of the most successful books not only in Saudi Arabia but also in the world. The book addressed both Muslims and non-Muslims and was translated into many different languages. More than 10 million copies of his book were sold, and his followers increased. In 2012 Al-Qarni was found guilty of plagiarism, as Salwa Aladian claimed that he took many “Success stories” from her book “That is how they defeat Despair” (هكذا هزموا اليأس) and used them in his book “Don't despair” (لا تيأس). without putting any referencing to her work. Salwa stated that she collected these stories and used them in her book using her writing style. She said that Al-Qarni took the stories ready from her book and had stolen her book’s introduction as well.

Al-Qarni’s followers started to attack Salwa on social media and a fight of words took place between Salwa’s family and Al-Qarni’s supporters. Abdallah (2019) said that “The cleric had used his religious standing and media exposure to rally a group of dedicated students and followers against Salwa. These followers promptly used online forums and social media websites to attack the young author.” The case took about one year, and Al-Qarni continued to deny the fact that he took the stories from Salwa’s book. Salwa said that she trusted the justice in Saudi Arabia, and she introduced her case to the court. After reviewing the books, the court found that Al-Qarni was guilty and fined him. Al-Qarni paid 300,000 Saudi Riyals to Salwa and apologized to her after he stated that this was not his fault. Al-Qarni said that he had asked one of his students to collect stories about successful people and the student collected all of them from Salwa’s book. In 2018, Al-Qarni was found guilty in another academic dishonesty case. The London-based Arabic language newspaper Arabi21 said that the heirs of the Syrian writer Abdel Rahman Raafat Pasha had won a case against Al-Qarni for allegedly stealing their father's book "Pictures of the Lives of the Companions." (See: https://m.arabi21.com/Story/1072807). Al-Qarni was convicted of reading specific paragraphs from the book on a television program without giving any reference to the original author. Al-Qarni had to pay a fine of 30,000 Saudi Riyals for infringing intellectual property rights and 120,000

Saudi Riyals to Abdel Rahman Raafat's family, in addition to the obligation to stop broadcasting or rebroadcasting the program. "After five years of litigation, we got a verdict against one of the preachers on behalf of a group of heirs. The preacher raided a book by their father, may Allah have mercy on him, and the judge sentenced him to a fine of 30 thousand, and ordered compensation of our client with 120 thousand, and we will ask for more than this" said lawyer Abdul Rahman Al-Lahim, who was appointed by the heirs of Pasha.

2 Related Work

Ouamour and Sayoud (2012) tested different character and word features using an SMO-SVM classifier on text samples extracted from textbooks. These features included character bigram and rare words. The texts were collected from ten different authors who wrote their texts in the domain of travel. Sayoud (2012) also studied the difference in writing style between the Quran (The Muslim holy book) and the Hadith (sayings of the Prophet Muhammad). In his experiments, he extracted four segments from each book and used a hierarchical clustering algorithm to cluster the text according to style. For each book, four segments were extracted. In the first experiment, Sayoud investigated the use of discriminative words such as 'those' (الذين) and the word 'earth' (الأرض). He noticed that these words appeared more often in one of the books than in the other, so he decided to use them as a feature set. In the second experiment, he used word length to discriminate between the two styles. Finally, in the third experiment, he used a new parameter called COST. The COST parameter is a cumulative distance measuring the similarity between the ending of one sentence and the ending of the next. This gives an estimation measure of the poetic form of the text. When Arabic poets write a series of poems, they make a termination similarity between the neighboring sentences of the poem, such as the final syllable or letter. This known in Arabic as rhythm or "Qafeia".

Hadjadj and Sayoud (2016) also investigated the authorship of the Quran and the Hadith, implementing two experiments to explore whether their writing styles are similar of different. The first experiments used Manhattan centroid distance and the SMO-SVM classifier, while the second experiment used hierarchical agglomerative clustering. Three main features were extracted from the dataset: interrogative words, the discriminative words, and COST. The purpose of Sayoud's experiments on the Quran and Hadiths was to challenge the assumption that the Quran was invented by the Prophet Mohammed (and therefore not handed to him by God). He tried to show that the books have two distinctive styles and therefore could not have been written by the same author (Sayoud, 2017).

Alwajeeh, Al-Ayyoub, and Hmeidi (2014) manually collected texts from Arabic news websites. The texts consisted of 500 different articles written by five different authors. They then ran their data through two well-known classifiers, i.e. Naïve Bayes and SVM. In their experiment, they achieved near-perfect accuracy for both classifiers. AbdulRazzaq and Mustafa (2014) claim to be the first to the use classic Delta distance (Burrows, 2002), a measure of difference between two texts, to find the authorship of Arabic texts. In their study, they demonstrated the suitability of this method for Arabic texts by using a database containing 30 books written by five different authors. The results showed that word bigrams and word trigrams were the most suitable features for Arabic authorship studies. Rabab'ah *et al.,* (2016) used two common approaches, i.e. BOW and Stylometry Features (SF), to find authorship in Arabic tweets. The authors collected tweets from twelve famous Arabic Twitter users, professionals working in different fields, e.g. religion, politics, sport, academia, and music, each with many followers. Some of the features were extracted by the morphological analysis tool MADAMIRA (Pasha *et al.*, 2014). This tool returns useful information about the words like aspect, gender, mood, and part of speech. Other features like the unigram and BOW were extracted using the Weka tool (Hall *et al.*, 2009). The following classifiers were tested to find which set of features produced the highest accuracy: Naïve Bayes, Decision trees, and SVM. The results show that combining all the feature sets they computed yields the best result.

Shrestha et al. (2017) used Convolutional Neural Networks (CNNs) to perform authorship attribution task of tweets. They used character n-grams as the feature set and provided a strategy to improve model interpretability by estimating the importance of input text fragments in the predicted classification. The results showed that CNNs outperformed the previous methods.

3 Corpus Description

To build the experimental corpus we used some sample texts from Salwa's book "That is how they defeat the Despair" (هكذا هزموا اليأس) and sample texts from Al-Qarni's books "Thirty reasons for happiness" (ثلاثون سبب للسعادة) and "Characters from the Holy Quran" (شخصيات من القرآن الكريم). In addition, four samples from the disputed text were taken from the document produced by Salwa to compare the plagiarised text with her book. The length of each sample in the corpus was 2000 words. Table 1 shows the texts which were used in the experiments:

Text	Book	Author
X_1_1	The disputed text	?
X_2_1	The disputed text	?
X_3_1	The disputed text	?
X_4_1	The disputed text	?
Q_1_10_15_1[1]	Thirty reasons for happiness	Al-Qarni
Q_1_3_8_1	Thirty reasons for happiness	Al-Qarni
Q_2_30_40_1	Characters from the Holy Quran	Al-Qarni
Q_2_15_25_1	Characters from the Holy Quran	Al-Qarni
Q_2_3_9_1	Characters from the Holy Quran	Al-Qarni
S_30_40_1[2]	That is how they defeat the despair	Salwa
S_45_56_1	That is how they defeat the despair	Salwa
S_15_25_1	That is how they defeat the despair	Salwa
S_60_71_1	That is how they defeat the despair	Salwa
S_90_100_1	That is how they defeat the despair	Salwa

Table 1: Corpus Description.

In these experiments, we used Principal Components Analysis (PCA) to find whether the disputed texts would cluster with Al-Qarni's texts or with Salwa's text. We also used cluster analysis using the hierarchical agglomerative algorithm and the classic Delta intertextual distance measurement to cluster the texts according to textual similarity which is a proxy for writing style.

3.1 PCA Analysis Using the Most Frequent Words

Principal component analysis (Everitt, 2006) is a feature extraction technique. Sets of features (such as most frequent words) tend to co-occur in similar documents, and together they make up a principal component. This technique can be used to reduce the dimensionality of many variables by ranking and sequentially extracting the components according to how much they contribute to the overall variance in the model. The features which show up more in a specific group of texts and show up less in another group are used to discriminate between the texts. PCA is very useful when we have little data, and we had few texts to compare between Salwa's style and Al-Qarni's style. It would have been better if we could have used all the texts from the book La-Tayaas, but the decision of the court made it very difficult to find the whole book. Thus we used the texts which were produced by Salwa to compare her book with Al-Qarni's book and used the PCA technique to extract the most important features from these texts. The initial feature set which we used to discriminate between the two authors was the most frequent words. Figure 1 shows that the disputed texts which were represented by X_1, X2_1, X3_1, and X4_1 were placed on the right-hand side together with Salwa's samples. It is possible to plot the texts and the words which most characterize them on the same graph. Words which occur frequently in the texts are plotted near those texts, and those which

[1] Q_1_10_15_1 means the sample was taken from the first book of Alqarni pages from 10 to 15

[2] S_30_40_1 means the sample was taken from Salwa's book pages from 30 to 40

occur infrequently in those texts are plotted far away. The axes show how correlated the texts and words are with each principal component. In figure 1, the most useful extracted words are shown together with the different samples. Figure 1 shows that the writing styles of Salwa and Al-Qarni can be distinguished by the words of the first principal component, since Al-Qarni's texts appear on the left, and Salwa's texts (including the disputed samples) appear on the right. The following list of words was seen on Salwa's side:

مع: With / قبل: Before /أنه: Which is / لم: Not/ غير: Except /في: In / إليه: To / بين: Between /كل: All /التي: Which was/ كانت: Was / لها: For it

This list of words was seen on Alqarni's side: إن: So / إذا: If / أو: Or / عند: Have / فلا: Not/ لك: For you/ علي: On/ الي: To/ هذا: This/ لا: Not / بها: On that/ كيف: How/ هو: He/ يا: Oh /إلا: Except/ وما: And not/ وقال: Said/ يقول: Is saying/ لما: When.

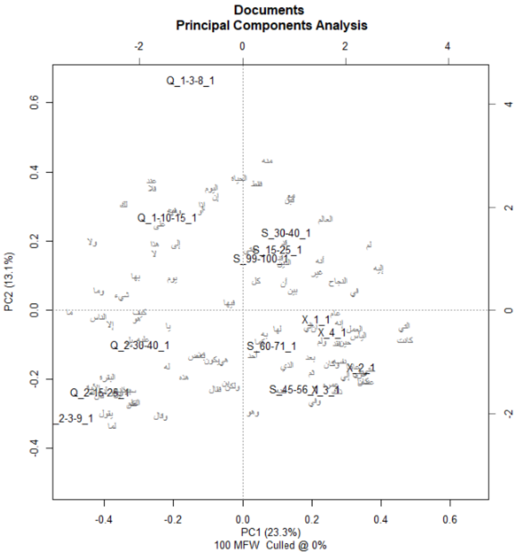


Figure 1: PCA including the MFW

3.2 PCA Using Morphemes

In this experiment we used the Farasa tool (Abdelali *et al.,* 2016) to extract the different morphemes contained in words as the feature set to discriminate between the two authors. For example the following morphemes (ه / ها /ت /ب) were observed on Salwa's side on the graph, while the morphemes (ون / ف / ل /وا) were on Alqarni's side. The following PCA graph (figure 3) shows the results obtained using this feature set. Once again, the disputed texts appeared on Salwa's side of the graph, showing that they were written in her writing style.

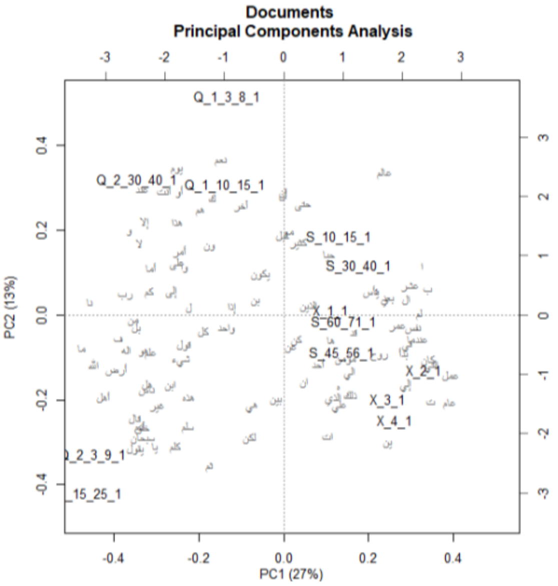


Figure 2 PCA using morphemes

3.3 Cluster Analysis Using the Most Frequent Words

In this experiment we used the Hierarchical Agglomerative Clustering Algorithm (HACA) (Everitt, 2005) to cluster the texts according to style. The most frequent words were used as a feature set and the classic Delta measure was used to measure the distance between the different texts. HACA displays the texts under analysis in a form of upside-down tree called a dendrogram, where the leaves are the texts and the branches show the distances between them. Thus, texts in a similar writing style will be placed close together, and dissimilar texts will be placed far apart. From figure 4, it is clear that the disputed tests X_1, X_2, X3_, and X_4 were clustered under the same branch which contained Salwa's samples. In addition to that, the samples were mixed with Salwa's samples, and they did not form a subset group. The samples which were taken from Al-Qarni's books were clustered together, and as we can see the samples from the first book formed a subset group and the samples from the second book also did so. This indicates that the stories of the "successful people" were taken from the book together with Salwa's style and very little paraphrasing was done for the texts. Salwa stated that she collected these stories and wrote them using her style to motivate the readers, and Al-Qarni took her effort without even putting any reference.

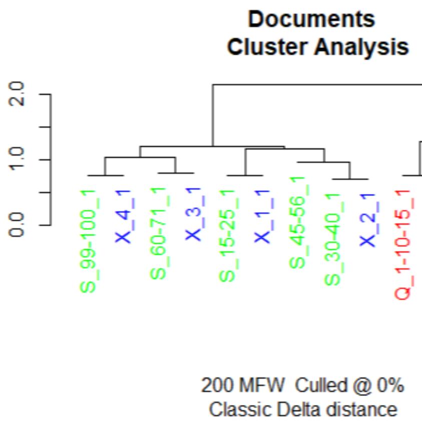


Figure 3: Cluster analysis using MFW

4 Conclusion

In this paper we introduced a recent case which occurred between two Saudi writers. Al-Qarni who is a very famous writer in Saudi Arabia was found guilty after the young female writer Salwa introduced a case against him to the court. To investigate Salwa's claim we collected some texts written by Al-Qarni and others written by Salwa to find which texts the disputed texts were more similar to. We used different features including single words, and morphemes contained in words. In addition, two different methods were used to do the analysis namely Hierarchical Agglomerative Clustering Analysis and Principal Component Analysis.

To sum up, Al-Qarni confirmed that the stories were taken from Salwa's book as the designated student for the task of collecting them took all the stories from one source which was Salwa's book. This was a problem mentioned by Al-Qarni himself, but another problem was that, as we can see from the results above, the collected stories were included in Al-Qarni's book without doing more paraphrasing to reproduce the stories in Al-Qarni's writing style. This made Salwa's fingerprint still visible in the texts, as we saw when the disputed texts clustered together with the texts in Salwa's branch of the HACA dentrogram, and the disputed texts appeared on Salwa's side of the PCA plot.

References

Abbasi, Ahmed, and Hsinchun Chen. "Applying authorship analysis to extremist-group web forum messages." IEEE Intelligent Systems 20, no. 5 (2005): 67-75.

Abdallah, Mariam. 2019. Academic Theft!. [online] Muftisays.com. Available at: https://www.muftisays.com/forums/77-taqleed--the-straight-path/9852-academic-theft.html [Accessed 27 Feb. 2019].

Abdelali, Ahmed, Kareem Darwish, Nadir Durrani, and Hamdy Mubarak. "Farasa: A fast and furious segmenter for arabic." In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations, pp. 11-16. 2016.

AbdulRazzaq, Ammar Adil, and Tareef Kamil Mustafa. "Burrows-Delta Method Fitness for Arabic Text Authorship Stylometric Detection." (2014).

Albadarneh, Jafar, Bashar Talafha, Mahmoud Al-Ayyoub, Belal Zaqaibeh, Mohammad Al-Smadi, Yaser Jararweh, and Elhadj Benkhelifa. "Using big data analytics for authorship authentication of arabic tweets." In Proceedings of the 8th International Conference on Utility and Cloud Computing, pp. 448-452. IEEE Press, 2015.

Al-Qarni, Aaidh. "Don't be sad." Saudi Arabia: International Islamic Publishing House (2003).

Burrows, John. "'Delta': a measure of stylistic difference and a guide to likely authorship." Literary and linguistic computing 17, no. 3 (2002): 267-287.

Eder, Maciej, Jan Rybicki, and Mike Kestemont. "Stylometry with R: a package for computational text Analysis." R journal 8, no. 1 (2016).

El-Fiqi, Heba, Eleni Petraki, and Hussein A. Abbass. "A computational linguistic approach for the identification of translator stylometry using Arabic-English text." In 2011 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2011), pp. 2039-2045. IEEE, 2011.

Everitt, Brian S. An R and S-PLUS® companion to multivariate analysis. Springer Science & Business Media, 2006.

Hadjadj, Hassina, and Halim Sayoud. "Towards an authorship analysis of two religious documents." In 2016 8th International Conference on Modelling, Identification and Control (ICMIC), pp. 369-373. IEEE, 2016.

Ouamour, Siham, and Halim Sayoud. "Authorship attribution of ancient texts written by ten arabic travelers using a smo-svm classifier." In 2012 International Conference on Communications and Information Technology (ICCIT), pp. 44-47. IEEE, 2012.

Pasha, Arfath, Mohamed Al-Badrashiny, Mona T. Diab, Ahmed El Kholy, Ramy Eskander, Nizar Habash, Manoj Pooleery, Owen Rambow, and Ryan

Roth. "Madamira: A fast, comprehensive tool for morphological analysis and disambiguation of arabic." In LREC, vol. 14, pp. 1094-1101. 2014.

Rabab'ah, Abdullateef, Mahmoud Al-Ayyoub, Yaser Jararweh, and Monther Aldwairi. "Authorship attribution of Arabic tweets." In 2016 IEEE/ACS 13th International Conference of Computer Systems and Applications (AICCSA), pp. 1-6. IEEE, 2016.

Sayoud, Halim. "Authorship classification of two old arabic religious books based on a hierarchical clustering." In Workshop Organizers, p. 65. 2012.

Sayoud, Halim. "AUTHORSHIP DISCRIMINATION ON QURAN AND HADITH USING DISCRIMINATIVE LEAVE-ONE-OUT CLASSIFICATION." (2017).

Shrestha, Prasha, Sebastian Sierra, Fabio Gonzalez, Manuel Montes, Paolo Rosso, and Thamar Solorio. "Convolutional neural networks for authorship attribution of short texts." In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers, pp. 669-674. 2017.

Classifying Information Sources in Arabic Twitter to Support Online Monitoring of Infectious Diseases

Lama Alsudias
King Saud University / Saudi Arabia
Lancaster University / UK
lalsudias@ksu.edu.sa

Paul Rayson
Lancaster University / UK
p.rayson@lancaster.ac.uk

Abstract

There is vast untapped potential in relation to the use of social media for monitoring the spread of infectious diseases around the world. Much previous research has focussed on English only, but the Arabic twitter universe has been comparatively much less studied. Motivated by important issues related to levels of trust, quality and reliability of the information online, here we consider the variety of information sources. As a first step, we find that numerous accounts disseminate information via Arabic social media, and we group them into five types of sources: academic, media, government, health professional, and public. We perform two experiments. First, native speakers judge whether they can manually classify tweets into these five groups, and then we repeat the experiment using various Machine Learning (ML) classifiers. We find that inter-annotator agreement is 0.84 for this task, and ML classifiers are able to correctly identify the type of source of a tweet with 77.2% accuracy without knowledge of the user and their bio or profile, but with 99.9% accuracy when provided with this information.

Keywords: Arabic, Infectious diseases, Machine Learning, Natural Language Processing, Twitter.

1 Introduction

People participate in the social web to express their opinions and provide information, which also gives researchers the opportunity to analyse those opinions across larger scale populations than is otherwise possible. One aim of this process is to summarise general opinions regarding international, national, or local events or themes in the huge amount of data available on the Internet. Twitter[1], one of the most popular tools for microblogging in real time, is used by people and organisations alike to share information on different topics, and these can include emergency and/or vital public health information. Due to its popularity as a communication platform, it can be difficult to distinguish reliable information from popular opinions or rumours and this is especially problematic in emergency or health-related scenarios.

In this paper, we consider that it is important to assist reliability and trust judgements by taking into account the source of the information alongside the content of tweets. Hence, via the Twitter API we collected 1,266 tweets containing information about infectious diseases which we categorised into five types of sources: academic, media, government, health professional, and public. First, in order to validate the suitability of the groupings, two Arabic native speakers performed an independent manual labelling of each tweet into one of the types. The resulting inter-coder reliability was 0.84. Second, in order to see whether the grouping can be replicated on a much larger scale, we applied several ML models including Logistic Regression, Random Forest Classifier, Multinomial Naïve Bayes Classifier, and Linear Support Vector classifier. We evaluated the results using 10 fold cross validation for each model. The linguistic features we used to train the systems were selected via the best features based on univariate statistical test. The results show that the ML algorithms correctly classified tweets with up to 77.2% accuracy. We also prove that the bio of the tweet source is an important key that can be used in classification.

[1] http://twitter.com/

The rest of paper is organized as follows. Section 2 covers related background work. Section 3 describes the data collection methods. Section 4 presents the manual coding and Inter-rater reliability. Section 5 focuses on the ML models. Section 6 discusses and visualizes the results. Section 7 contains our conclusions and suggests future work.

2 Related work

There has been a growing interest in Arabic language processing on social media data and Twitter in particular, but only a small part of this research is related to health and medicine topics as we show in this section. In contrast, a growing number of studies have analysed tweets for the public health information they contain in English and other languages such as Chinese and German (Charles-Smith et al., 2015). We summarise this research in what follows.

Arabic related research: Very few papers have described the use of Arabic tweets for studies around health and medicine topics. Khalid and others (2015) evaluated the correctness of health information on twitter based on its medical accuracy. They found that 51.2% of tweets contained false information. The study showed the need for policies on using the social media for health care information. The study of (Alayba et al., 2017) used twitter for sentiment analysis of health services. They collected unbalanced twitter data and annotated it using three annotators by selecting the most frequent tags in each case. They used ML and deep neural networks techniques in their experiment and achieved 91% accuracy using support vector machines. A survey performed by (Alsobayel, 2016) concluded that twitter is the most frequently used by health care professionals in Saudi Arabia for the aim of professional development.

English related research: In contrast to Arabic, there is a much larger body of work utilising twitter in English for research on health and medicine related topics. The previous studies of Paul and Dredze (2011), Aramaki et al. (2011), Breland et al. (2017), and Sinnnenberg (2017) have proved that twitter contains valuable information on public health. These studies show the power of Natural Language Processing (NLP) techniques for learning new information from twitter for public health research and to support health informatics hypotheses. The Epidemic Sentiment Monitoring System (ESMOS) is an example of tools that visualise Twitter users' concerns towards specific health conditions developed by Ji, Chun, and Geller (2013) in order to reduce the time of identifying peaks and ongoing monitoring of diseases required by public health officials. They also displayed a knowledge-based approach that utilises a medical ontology, an open source Disease Ontology developed by (Arze et al., 2011), to identify the occurrence of illnesses and to analyse the etymological expressions that provide subjective expressions and polarity of emotions, sentiments, conclusions, individual states of mind, etc. with an opinion classifier (Ji et al., 2016). In (Yepes et al., 2015), pilot results were demonstrated in relation to future directions to investigate when using Twitter information for public health. Other research (Charles-Smith et al., 2015; Paul et al., 2016) has analysed social media articles in supporting public health in order to show the effectiveness of this strategy. These papers recommend a combination of social media with other techniques to measure disease surveillance and spread. The goal of the study in (Sivasankari et al., 2017) was to produce a real time system for the prediction and detection of the spread of an epidemic by identifying disease tweets by graphical location. Good accuracy and quite diverse expressions were uncovered targetting health-related subjects (Doan et al., 2018). Although the results depend only on four months of Twitter data, the paper develops a useful approach to extracting cause-effect relationships from tweets. Other researchers have combined Twitter data with Google Trends data for tracking the spread of infectious diseases (Hong and Sinnott, 2018). A study by Ahmed and others (2018) analysed the twitter data from infectious disease outbreaks. The study developed new insights into how users respond during infectious disease outbreaks and reflects on users' response in association with the sociological concept of the moral panic. They also suggest to examine the tweets in other languages.

Although the above studies use a variety of different approaches in NLP for showing how twitter data can be used for public health applications including monitoring the spread of some diseases, they only consider the information content and do not attempt to study the trust or reliability of the sources of information. While social media users share information about disease outbreaks, symptoms, drug interactions, diet success, and other health behaviors (Paul et al., 2016; Yepes et al., 2015), more than half of the tweets may contain false information (Alnemer et al., 2015). Hence, there is a clear requirement to filter this information in some way, and hopefully to study the ways in which we can reduce the noise of false information and to find tweets with more reliable information of a higher quality. A key first step to do this is to consider the source of the information since this helps the reader to determine how reliable the information is, for example by comparing a tweet on a specific topic by a health professional versus something similar via a member of the general public. Most previous research has focussed on the content of the tweets and not on the source of the information, hence we take a new approach in this paper to combine the two elements together.

3 Data collection

First, we collected 10,000 Arabic tweets via the Twitter API between December 2018 and February 2019. We defined the keywords related to infectious diseases. These keywords were generated by translating an English Disease Ontology, a medical Ontology, developed by Schirmal et al (2011) since we were unable to locate a suitable Arabic equivalent. Using the Disease Ontology allows us to find all terms related to infectious diseases with a set of synonyms (Ji et al., 2016). The Twitter API does not allow us to retrieve enough historical tweets unless we know the user ID in advance. Therefore, we followed two strategies in collecting tweets:

- collect tweets containing words from the keyword list.
- from a suitable user account (discovered using the first step), collect all historical tweets, and filter them depending on the keyword list.

Next, we filtered the tweets manually by removing advertisements, spam, and retweets. After that, we devised Python scripts to clean the tweets by removing URLs, mentions, hashtags, numbers, emojis, repeating characters and non-Arabic words. We also automatically normalised the Arabic tweets and tokenised them. The first author of the paper first classified each of the tweets manually into one of the five groups described in Section 3.1. Second, we asked another Arabic native speaker to independently classify the tweets into the same five groups in order to calculate inter-coder reliability as described in Section 4.2.

3.1 Tweet categorisation

Based on our initial manual reading of the tweets, we decided on five types of users to classify the tweets into, taking into account the various levels of trust that might be associated with each type. For each category listed below there is a small description with examples illustrated in Table 1.

Academic: the tweet is written by academic researchers in higher education. To illustrate, this could be a researcher who carries out studies about infectious diseases.

Media: the tweet is written for newspapers or magazines whether they are general media or health specific ones. In most cases it contains news about infectious diseases.

Government: the tweet is written by a user account that represents the government in some official capacity such as the ministry of health. It may include news, admonition, warning, or general information related to infectious diseases.

Health professionals: the tweet is written by doctors, nurses, or other health service practitioners. In other words, any person who is employed or trained in the health domain and writes the tweet on any information related to infectious diseases.

Public: the tweet is written by members of the general public. It may include information on infectious diseases, feeling sick, or giving advice to someone. Also, it may be written in many dialects since the people come from many Arab countries.

Table 2 shows the number of tweets in each category after filtering and preprocessing. The total number of tweets is 1,266 with only 56 tweets in the media category and 436 tweets in the public one. The reason for this is that there are few media accounts that tweet about infectious diseases whilst many members of the public tweet on the topic. In the other categories (academic, government, and health professionals), there are a relatively well balanced number of tweets which are 239, 258, and 277, respectively.

Category	Tweet in Arabic	Translated tweet to English
Academic	ورقة علمية حديثة تراجع وتتناول فيروس الهربس وتأثيراته الاكلينيكية خاصه على الأطفال ومضى ضعف المناعة .	A recent scientific paper reviews the Herpes virus and its clinical effects on children and patients with immunosuppression.
Media	حاله وفاه بسبب عدوى الكورونا في خميس مشيط .	A case of death due to infection of the corona in Khamis Mushayt.
Government	يتوفر لقاح الانفلونزا في مراكز الرعايه الصحيه الأوليه .	The flu vaccine is available in primary health care centers.
Health Professional	لا تستعمل قطرات للأذن من توصيه مريض سابق ، فعلاج التهابات الاذن الخارجيه البكتيري يختلف عن علاج الالتهاب الفطري .	Do not use ear drops from a previous patient recommendation. Treatment of bacterial external ear infections is different from treatment for fungal infections.
Public	العشبه العجيبه لعلاج أكثر من مرض فى عشبه واحده وهى الزنجبيل .	The wonderful herb to treat more than one disease in one herb is ginger.

Table 1: Examples of tweets in each category

Category	No. of tweets	No. of words
academic	239	3,696
media	56	601
government	258	3,602
health professional	277	6,123
public	436	4,963
total	1,266	18,985

Table 2: Number of tweets and words in each category

4 Manual Coding and Inter-coder Reliability

4.1 Annotation Process

The process starts with labeling the tweets by two Arabic native speakers, including the first author of the paper, who were provided with the guidelines detailed above. The classification depends first on the text in the tweet itself. If the tweet has an ambiguous classification, the annotator may need to look at the bio, a description written by the twitter user, of the user tweet.

4.2 Inter-coder Reliability

We used the Kappa Statistic to test the robustness of the classification scheme (Artstein and Poesio, 2008). The result shows that Cohen's Kappa score is **0.84** which indicates strong agreement between the two manual coders. Figure 1 shows the confusion matrix between the two annotators. The most disparate results between the two coders is between the academic and health professionals categories and this accounts for 10.9% out of the 16% total. This is caused by the similarity between the language of the two groups and the lack of explicit information in the bio to determine which category the tweet fits into.

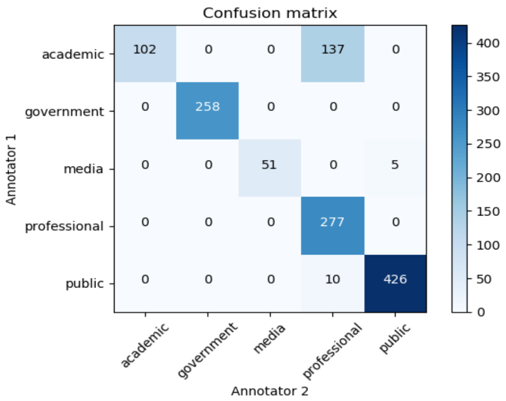


Figure 1: Heat map of confusion matrix between the two annotators.

5 Machine Learning Models

In our study, we used Python Scikit-learn 0.20.2 (Pedregosa et al., 2011) software and applied four ML models: Logistic Regression, Random Forest, Multinomial Naïve Bayes and LSVC: Linear Sup-

Tweet in Arabic	Translated tweet to English	Category
التوصية بأخذ تطعيم فيروس الحصبة للمسافرين .	Recommend vaccination of the measles virus to travelers.	Health professional, government, or academic.
الربيعه : افتتحنا عده مراكز إضافيه لمرضى والاضطرابات وغيرها وتوسعنا في برامج صحه المرأه وخصوصا وأطلقنا مؤشرات لقياس أداء المراكز الصحيه ورضا المراجع .	Alrabiaa: We have opened several additional centers for patients and disorders and others and expanded in women's health programs, especially we have launched indicators to measure the performance of health centers and the satisfaction of references.	Media or government.
بحث لثلاثه أطباء سعوديين يكشف علاقه إنزيم الكبد بالتهاب الزائده الدوديه .	A study by three Saudi doctors reveals the relationship of the liver enzyme to Appendicitis.	Media or academic

Table 3: Examples of tweets with ambiguous categorisation

port Vector Classification. We used 10-fold cross validation to determine accuracy, splitting the entire sample into 90% training and 10% testing for each fold.

5.1 Machine Learning Features

A word frequency approach was used to extract the features from the processed training data after converting them to a matrix of token counts. We designed several features to be used in all four algorithms in order to obtain the best accuracy.

5.1.1 Feature selection

We used various techniques to select the best features automatically (Pedregosa et al., 2011):

- all features: counting unigrams, bigrams and trigrams and ignoring terms that have a document frequency strictly lower than two.
- selecting the best features: Based on a univariate statistical test, keep 60% of the features that have the highest scores.
- selecting from a model: Random Forest Classifier is used to remove unimportant features.
- Using Stanford POST (Part Of Speech Tags) (Manning et al., 2014): Arabic POST is used to annotate the tweet with part-of-speech tags.

5.1.2 Balancing the data

Since the number of tweets is unbalanced across the types, we used two different techniques to re-sample them: under-sampling and over-sampling. We used RUS (Random Undersampling) for under-sampling and ROS (Random Oversampling) for over-sampling. RUS works by removing samples randomly from the majority classes while ROS generates more examples for the minority classes, which has less training data (Burnaev et al., 2015).

5.1.3 Using user bio as a feature

To further resolve the close ambiguity of some types such as academic and health professional, or government and media, we used the bio of the tweet user to provide further features in combination with the tweet text itself. Then we repeated the experiment, including preprocessing, feature extraction and selection, and applied machine learning algorithms, with the new text. Table 3 shows an example of tweets that may be classified into different classes. The first example, which is written by a health professional, can be classified as government because it seems to be providing advice from the government or as an academic as a result from their research.

6 Results and Discussion

Choosing the most frequent class (public) represents 34.4% of the data set, so this represents a simple baseline for our results. Table 4 shows the accuracy, F1-score, Recall, and Precision of the ML models on our training dataset. The Logic Regression classifier and Multinomial Naïve Bayes achieved the highest accuracy (76.0%) with higher recall (0.76) compared to the other algorithms. To evaluate the automatic classification, we compared the algorithms with different sets of features. Figure 2 illustrates the results we achieved running the four ML algorithms with different set of features. The highest accuracy (77.2%) is achieved by the Logic Regression classifier with a selection of the best features based on a univariate statistical test features. It also provided the best score on Random Forest and Linear Support Vector classifications (68.2% and 76.1%, respectively) while selecting all the features reached 76.2% in multinomial NB classifiers. Selecting a model to remove unimportant features did not provide any better results via the four algorithms and POST features had the worst results.

ML	A%	F	R	P
LR	76.0	0.74	0.76	0.74
RF	65.1	0.66	0.68	0.71
MNB	76.0	0.75	0.76	0.75
LSVC	74.1	0.73	0.74	0.75

Table 4: Training results using all features
ML: Machine Learning Model, A%: Accuracy, F: F1-Score, R: Recall, P: Precision
LR: Logistic Regression, RF: Random Forest, MNB: Multinomial Naïve Bayes, LSVC: Linear Support Vector Classification

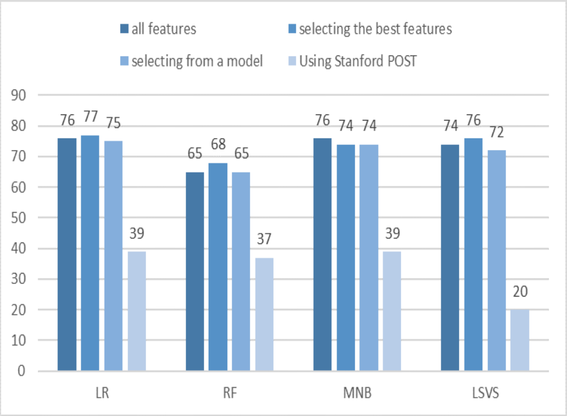


Figure 2: Effect of different features on classification.
LR: Logistic Regression, RF: Random Forest, MNB: Multinomial Naïve Bayes, LSVC: Linear Support Vector Classification

ML	all features	selecting the best features	selecting from a model
LR	76%	79%	77%
RF	62%	72%	72%
MNB	72%	78%	75%
LSVC	73%	78%	76%

Table 5: Effect of over-sampling data on classification.

ML	all features	selecting the best features	selecting from a model
LR	73%	70%	66%
RF	53%	56%	49%
MNB	68%	67%	61%
LSVC	63%	61%	67%

Table 6: Effect of under-sampling data on classification.

We also compared the normal data, which is the original unbalanced data, with the over and under sampled data to show the effect of re-sampling on the classification accuracy, and the results are shown in Table 5 and Table 6. We can see that there is a small enhancement of accuracy using an over-sampling method especially when selecting the best features and selecting features from a model in all four classifiers. On the other hand, under-sampling the data achieves lower accuracy than normal data due to losing some data in the re-sampling process. Figure 3, Figure 4, and Figure 5 show the comparison between normal data, over-sampling, and under-sampling in all features, selecting the best features and selecting features from a model respectively. We can see that when applying all features, the normal data has the best result in all models expect Logic Regression which has the best result when using over-sampling data with accuracy 77.1% (Figure 3). However, over-sampling data with selecting the best features raises the accuracy between 2% to 4% above normal data in all models (Figure 4). It reaches 79.1% in the Logic Regression classifier and 78.2% in the Multinomial Naïve Bayes and Linear Support Vector Classification models. Moreover, the accuracy is highest when over-sampling data with selecting features from a model in all classifiers for instance, Random Forest classifier which increase from 65.1% to 72.2% (Figure 5).

Table 7 represents the accuracy of each model after combining the bio of the tweet user with the tweet text. In many of the results, the accuracy

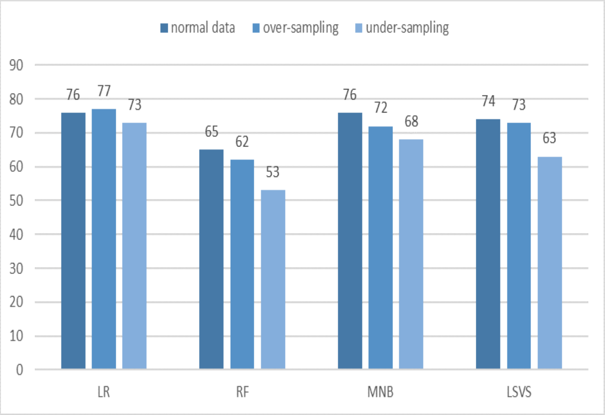


Figure 3: Effect of re-sampling data with all features on classification.

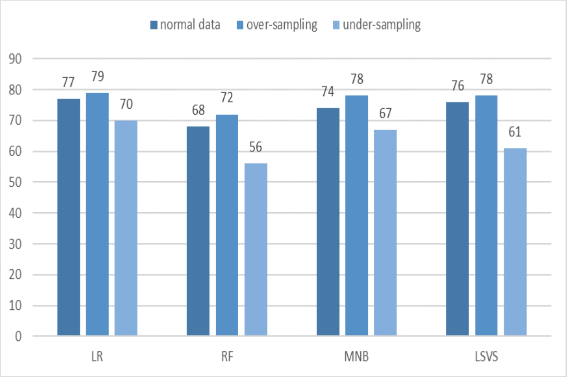


Figure 4: Effect of re-sampling data with selecting the best features on classification.

improves to **99.9%** which shows that the user bio has a very important role to play for classification. For instance, example number 2 in Table 3 can be classified as government after reading the bio of the twitter user which is:

تتمثل رؤية وزارة الصحة في تحقيق (
الصحة بمفهومها الشامل على المستويات للفرد
والأسرة والمجتمع مع العمل على مساعدة
المسنين و ذوي الاحتياجات الخاصة بما يمكنهم.
) which means in English: “The vision of the Ministry of Health is to achieve comprehensive health at the individual, family and community levels while working to help the elderly and those with special needs”. There are some words in the example bio such as ministry which can help in the classification process. The very high accuracy of the classification can be explained as a result of the limited number of twitter accounts in the study.

We performed an analysis using the Logic Regression classifier with a set of the best features based on univariate statistical test features in the heat map in Figure 6. The matrix shows 12 tweets from the academic category are confused with the government category. In Figure 7 representing the worst accuracy (65.1%) resulting from Random Forest with all features, we assess the degree of confusion between categories. The highest confusion across all categories is between the academic and government types.

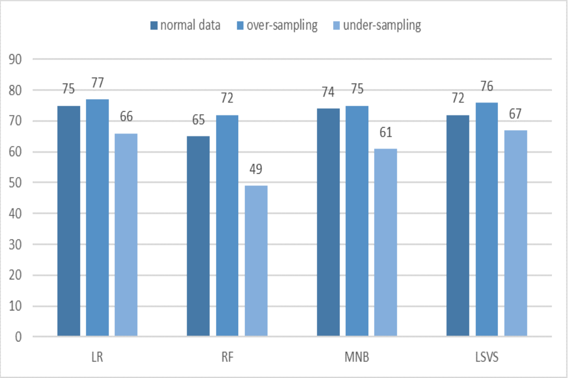


Figure 5: Effect of re-sampling data with selecting from a model on classification.

ML	Tweet text only	Tweet text with bio
LR	77.2%	99.9%
RF	68.2%	99.9%
MNB	76.2%	99.6%
LSVC	76.1%	99.9%

Table 7: Effect of using bio of tweet user as feature on classification.

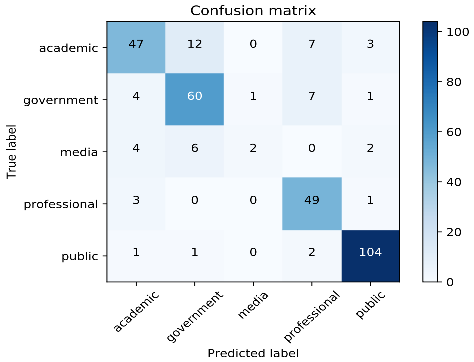


Figure 6: Heat map of confusion matrix of the Logic Regression classifier with selecting the best features based on univariate statistical test features.

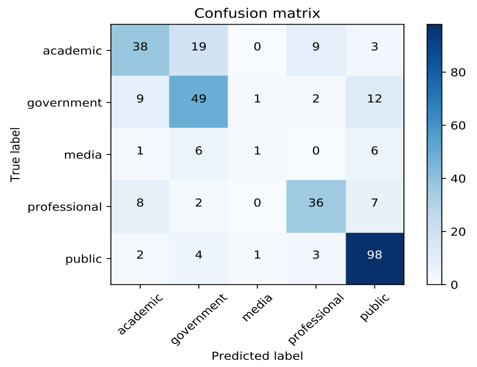


Figure 7: Heat map of confusion matrix of the Random Forest classifier with all features.

7 Conclusion and Future work

In general, social media can be useful as a source of health information. Many government organisations, academic experts, professions, and media outlets in the field of health and medicine release useful health information needed by the public. However, in emergency situations and fast moving scenarios, it is important to understand the veracity of information released in social media, in order to avoid acting on false information. Therefore, we study not only the content of tweets but also the source of the information in order to work towards determining the quality and reliability of public information. We use the bio of the twitter user which contains key information in order to discover reliable accounts that can be trusted.

Here, we introduce a new Arabic social media dataset for analysing tweets related to infectious diseases. The dataset of Arabic tweets has been manually classified into five categories: academic, media, government, health professional, and public, with good inter-rater reliability. We then used ML algorithms to replicate the manual classification. The results showed high accuracy on the classification task, and show that we are able to classify tweets into the five categories with favourable accuracy on the tweet content itself, and highly accurately using the bio information from the user. The dataset, including tweet ID, manually assigned categories, and other resources used in this paper are released freely for academic research purposes[2].

Our future work includes using more NLP techniques and linguistic features such as word embeddings, combining Arabic dialect analysis into the process of classification, and utilising an Arabic medical ontology to be the source of the disease information. Moreover, we will analyse the tweets to support investigation of the spread geographically and over time of infectious diseases in Arab countries.

References

Wasim Ahmed, Peter Bath, Laura Sbaffi, and Gianluca Demartini. 2018. Moral panic through the lens of twitter: An analysis of infectious disease outbreaks. In *Proceedings of the 9th International Conference on Social Media and Society*, pages 217–221. ACM.

Abdulaziz Alayba, Vasile Palade, Matthew England, and Rahat Iqbal. 2017. Arabic language sentiment analysis on health services. In *2017 1st International Workshop on Arabic Script Analysis and Recognition (ASAR)*, pages 114–118.

Khalid Alnemer, Waleed Alhuzaim, Ahmed Alnemer, Bader Alharbi, Abdulrahman Bawazir, Omar Barayyan, and Faisal Balaraj. 2015. Are health-related tweets evidence based? review and analysis of health-related tweets on twitter. *Journal of medical Internet research*, 17(10).

Hana Alsobayel. 2016. Use of social media for professional development by health care professionals: A cross-sectional web-based survey. *JMIR Med Educ*, 2(2):e15.

Eiji Aramaki, Sachiko Maskawa, and Mizuki Morita. 2011. Twitter catches the flu: detecting influenza epidemics using twitter. In *Proceedings of the conference on empirical methods in natural language processing*, pages 1568–1576. Association for Computational Linguistics.

Ron Artstein and Massimo Poesio. 2008. Inter-coder agreement for computational linguistics. *Computational Linguistics*, 34(4):555–596.

Cesar Arze, Gang Feng, Mark Mazaitis, Suvarna Nadendla, Victor Felix, Yu-Wei Wayne Chang, Lynn Marie Schriml, and Warren Alden Kibbe. 2011. Disease Ontology: a backbone for disease semantic integration. *Nucleic Acids Research*, 40(D1):D940–D946.

Jessica Breland, Lisa Quintiliani, Kristin Schneider, Christine May, and Sherry Pagoto. 2017. Social media as a tool to increase the impact of public health research. *American Journal of Public Health*, 107(12):1890–1891. PMID: 29116846.

Evgeny Burnaev, Pavel Erofeev, and Artem Papanov. 2015. Influence of resampling on accuracy of imbalanced classification. In *Eighth International Conference on Machine Vision (ICMV 2015)*, volume 9875, page 987521. International Society for Optics and Photonics.

[2] https://doi.org/10.17635/lancaster/researchdata/303

Lauren Charles-Smith, Tera Reynolds, Mark Cameron, Mike Conway, Eric HY Lau, Jennifer Olsen, Julie Pavlin, Mika Shigematsu, Laura Streichert, Katie Suda, et al. 2015. Using social media for actionable disease surveillance and outbreak management: a systematic literature review. *PloS one*, 10(10):e0139701.

Son Doan, Elly W Yang, Sameer Tilak, and Manabu Torii. 2018. Using natural language processing to extract health-related causality from twitter messages. In *2018 IEEE International Conference on Healthcare Informatics Workshop (ICHI-W)*, pages 84–85. IEEE.

Yang Hong and Richard Sinnott. 2018. A social media platform for infectious disease analytics. In *International Conference on Computational Science and Its Applications*, pages 526–540. Springer.

Xiang Ji, Soon Ae Chun, and James Geller. 2013. Monitoring public health concerns using twitter sentiment classifications. In *Healthcare Informatics (ICHI), 2013 IEEE International Conference on*, pages 335–344. IEEE.

Xiang Ji, Soon Ae Chun, and James Geller. 2016. Knowledge-based tweet classification for disease sentiment monitoring. In *Sentiment Analysis and Ontology Engineering*, pages 425–454. Springer.

Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Michael Paul and Mark Dredze. 2011. You are what you tweet: Analyzing twitter for public health. *Icwsm*, 20:265–272.

Michael Paul, Abeed Sarker, John Brownstein, Azadeh Nikfarjam, Matthew Scotch, Karen Smith, and Graciela Gonzalez. 2016. Social media mining for public health monitoring and surveillance. In *Biocomputing 2016: Proceedings of the Pacific Symposium*, pages 468–479. World Scientific.

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 Édouard Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Lynn Marie Schriml, Cesar Arze, Suvarna Nadendla, Yu-Wei Wayne Chang, Mark Mazaitis, Victor Felix, Gang Feng, and Warren Alden Kibbe. 2011. Disease ontology: a backbone for disease semantic integration. *Nucleic acids research*, 40(D1):D940–D946.

Lauren Sinnenberg, Alison Buttenheim, Kevin Padrez, Christina Mancheno, Lyle Ungar, and Raina Merchant. 2017. Twitter as a tool for health research: A systematic review. *American Journal of Public Health*, 107(1):e1–e8. PMID: 27854532.

Si Sivasankari, Mu Kavitha, and Gi Saranya. 2017. Medical analysis and visualisation of diseases using tweet data. *Research Journal of Pharmacy and Technology*, 10(12):4306–4312.

Antonio Yepes, Andrew MacKinlay, and Bo Han. 2015. Investigating public health surveillance using twitter. *Proceedings of BioNLP 15*, pages 164–170.

Text Segmentation Using N-grams to Annotate Hadith Corpus

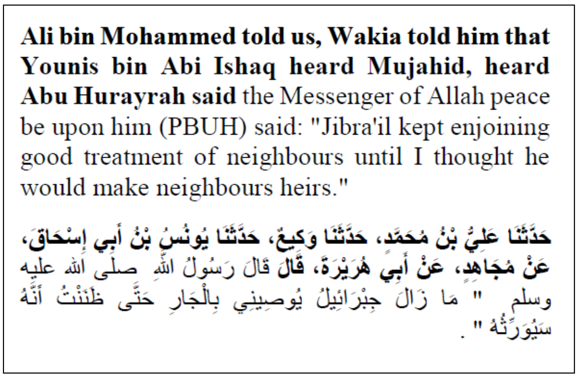
Shatha Altammami
School of Computing
University of Leeds
Leeds,UK
scshal@leeds.ac.uk

Eric Atwell
School of Computing
University of Leeds
Leeds,UK
E.S.Atwell@leeds.ac.uk

Ammar Alsalka
School of Computing
University of Leeds
Leeds,UK
M.A.Alsalka@leeds.ac.uk

Abstract

In this paper, we exploit natural language processing techniques to build a system that automatically segments Hadith into its two main components, Isnad and Matn. We evaluate the previous attempts to segment Hadith and identified the limitations in these studies. Then a Hadith segmentation system is built and tested with Hadith collections extracted from six different Hadith books. The result demonstrates that bi-grams are effective in identifying Hadith segments with 92.5% accuracy.



Ali bin Mohammed told us, Wakia told him that Younis bin Abi Ishaq heard Mujahid, heard Abu Hurayrah said the Messenger of Allah peace be upon him (PBUH) said: "Jibra'il kept enjoining good treatment of neighbours until I thought he would make neighbours heirs."

حَدَّثَنَا عَلِيُّ بْنُ مُحَمَّدٍ، حَدَّثَنَا وَكِيعٌ، حَدَّثَنَا يُونُسُ بْنُ أَبِي إِسْحَاقَ،
عَنْ مُجَاهِدٍ، عَنْ أَبِي هُرَيْرَةَ، قَالَ قال رَسُولُ اللَّهِ صلى الله عليه
وسلم " مَا زَالَ جِبْرَائِيلُ يُوصِينِي بِالْجَارِ حَتَّى ظَنَنْتُ أَنَّهُ
سَيُوَرِّثُهُ " .

Figure 1: Hadith Example, Isnad in bold

1 Introduction

Advancement in Artificial Intelligence (AI), specifically Natural Language Processing (NLP), encouraged researchers to tackle problems associated with textual data. In this work, we exploit NLP methods to build a system that automatically segments Hadith text, which is the collection of narratives reporting different aspects of Prophet Muhammad's life.

Hadith originated in the 7th century and is considered classical Arabic with unique structure, linguistic features, and patterns that make it suitable for applying computational models. Moreover, Hadith possesses historical importance and is still used by Muslims around the world. This is because not all Islamic laws and regulations are mentioned in the Islamic holy book, the Quran. Hence, producing Hadith resources will be useful for a wider community including Islamic scholars, historians, linguists, and computer scientists.

Before building such Hadith resources, it is crucial to explain what constitutes the Hadith literature. It is a huge collection of Hadiths[1] that record every aspect of the prophet's life. In addition to that, there are supporting documents which include commentaries that explain the Hadith text, and biographic material that identify narrators of Hadith, which is central in studying Hadith authenticity.

Hadith types vary, it could be a short sentence or long paragraph describing what the prophet said in a specific incident, a dialogue of the prophet's conversation with someone, or a story told by the prophet's companions that explain the prophet's actions in a specific matter like 'prayer'. Every Hadith consists of two parts, as shown in Figure 1, the **Isnad** which is a chain of narrators shown in bold face text, followed by **Matn** which is the actual teaching. However, these parts exist as one sentence or paragraph where researchers manually segment Isnad from Matn to focus on one part (Luthfi et al., 2018). For example, researchers analyse Isnad to visualizing the chain of narrators (Muazzam Siddiqui, 2014) and identify how Hadith travelled through time. While other researchers focus on Matn with the aim to categorize Hadith into subtopics (Saloot et al., 2016).

Although there are research in the area of Hadith computation, it is still in its infancy with limit contributions (Bounhas, 2019). In fact, there are no common datasets, benchmarks, or evaluation measures as pointed in a recent survey of research

[1] The plural of Hadith is AHadith, but we will use Hadiths

on Arabic NLP (Guellil et al., 2019). It shows that only 11% of the available Arabic resources are classical-Arabic and none dedicated for Hadith. These resources are mostly dedicated for the Quran, such as Dukes and Atwell (2012). Therefore, we aim to build a segmentation model to produce a Hadith corpus where Isnad and Matn are annotated to serve the wider research community and advance the field.
This paper is organized as follows, we survey previous attempts to identify Isnad segments in section 2 and discuss their limitations. Then we introduce the data used to build and test our system in section 3. After that we present our Hadith segmentation model in section 4, and discuss the results in section 5.

2 Literature Review

Surveying the literature, we found a number of attempts to detect Isnad patterns. Table 1 shows summaries of such work and the following paragraphs discuss these papers and their limitations.

Muazzam Siddiqui (2014) attempted to segment Isnad from Matn by using supervised machine learning (ML) algorithms that require an annotated corpus. Thus, a native Arabic speaker annotated Hadith tokens extracted from Sahih Al-Bukhari[2] into five classes (beginning of Person, inside Person, beginning of Narrator, inside Narrator, and Other) where Narrator corresponds to names in Isnad, and Person corresponds to names in Matn.

After annotating the corpus, they studied Hadith contextual patterns to identify features to be used in classification. For example, the word '*told us* -حدثنا' is followed by a narrator's name in Hadith Isnad, and the word '*son of* - بن' is part of a name. Another example is the honorific phrases that always follows a person's name. The classifier takes the training data and features in the form of 'feature, class' where each word is classified as 'beginning/inside Person, beginning/ inside Narrator, or Other'. Then the system classifies new Hadith tokens and segments the Hadith by finding the end of the consecutive list of narrators.

Their system's performance was measured based on two factors. First, its ability to assign each token the correct type irrespective of the boundaries as long as there is an overlap. Second, its ability to correctly find the boundary of each name independent of type assigned (narrator or person). The system produced 90% accuracy in the testing phase. Then for evaluation, they used another manually labeled Hadith book titled 'Musnad Ahmed' which contains 5K tokens that produced 80% accuracy.

Harrag (2014) built a Finite State Transducers (FST) system to extract Hadith segments which include Num-Kitab, Title-Kitab, Num-Bab, Title-Bab, Num-Hadith, Isnad, Matn, Taalik, and Atraf. He used the beginning of words like 'K' for Kitab to identify the book title. Furthermore, he used punctuation to identify other parts of the Hadith assuming that all Matn is surrounded by parenthesis. These features depend on the Hadith book used and how well and correctly it is punctuated. Thus, it cannot be applied to all kinds of Hadith books. His work measures the system's performance to identify several components of Hadith that range from Isnad and going deeper into identifying the narrator's names. However, for the purpose of our study we only report the result of Isnad extraction which was 44% precision.

Azmi and Bin Badia (2010) built a system that aims to draw the tree of narrators, but first Isnad must be extracted. To extract Isnad, they identified pre-processing steps which include removing diacritics and punctuation; apply shallow parsing to handle noise and leave out words which it is not able to parse. Using the shallow parsing output, Noun phrases were the candidate of being a narrator's name. After pre-processing the data, they applied context free grammar to identify each segment in the Hadith by comparing the tokens to the list of Hadith lexicon they compiled earlier. Since the goal of their study is to build the tree, their result reflects the system's success to draw the tree and not the segmentation part.

Maraoui et al. (2018) compiled a list of trigger words that come before, after, and between narrator's names. Furthermore, they identified words that mark the termination of each Hadith which are 'أطرافه' or 'تحفه'. Using these comprehensive lists of words, they were able to segment Isnad from Matn for Sahih Albukhari. However, it is not clear whether it can be used to segment other Hadith books.

Boella (2011) presented HedExtractor system which uses regular expressions (Regex) to extract

[2] Mohammad AlBukhari, Sahih AlBukhari (2002). Damascus: Dar Ibn Kathir.

Paper	Approach	Technique	Pre-processing	Manual annotation	Data	Result
(Muazzam Siddiqui, 2014)	Machine Learning	Nave Base, KNN, Decision tree	Remove diacritics, stemming	Person, Narrator, other	Albukhari Musnad Ahmed	80%
(Harrag, 2014)	Finite State Transducer	-	Tokenize,	None	Albukhari	44%
(Azmi and Bin Badia, 2010)	Rule Based	Context Free Grammar	Shallow parsing Remove diacritics and punctuation	Hadith Lexicons	Albukhari	87%
(Maraoui et al., 2018)	Rule Based	Dictionary Lookup	None	Hadith Lexicons	Albukhari	96%
(Boella, 2011)	Rule Based	Regular Expressions	Transliteration	Hadith Lexicons	Albukhari	97%
(Mahmood et al., 2018)	Rule Based	Regular Expressions	None	None	Muslim, Albukhari, Abu Dawud, Imam Malik	98%

Table 1: Review of Previous Research on Hadith Segmentation

Hadith. First, it extracts each Hadith from the book by finding the number of each Hadith. Then the Arabic text is converted to its transliteration to find the words of transmission based on a list they compiled. It assumes any words between these transmission words are the narrator's names. Once there are no transmission terms detected, the system marks the end of Isnad. However, the exact point of Hadith segmentation is sometimes ambiguous even for humans. To overcome this problem, they have set a threshold of 100 characters which they picked based on trial and error. This threshold tells the system if no other transmission word is detected within the next 100 characters then Matn starts.

Mahmood et al. (2018) extracted Hadiths from different sources, but did not mention any Hadith lexicon list compiled to be used in the Regex. In fact, Hadith lexicons were encoded in the Regex which is not fit for re-usability.

2.1 Limitation of Previous Studies

In the previous research, Hadith segmentation was done using three approaches. First, rule-based that consists of whitelists (or gazetteers) to identify names and Isnad specific words, a filtration mechanism (or Blacklists) to identify Matn words, and grammar rules (as a set of regular expressions) to identify the segmentation point. Second is the ML approach which consists of data annotation, feature and algorithm selection, training and classification. The third is the FST which depends on the degree of consistency in a well-structured text.

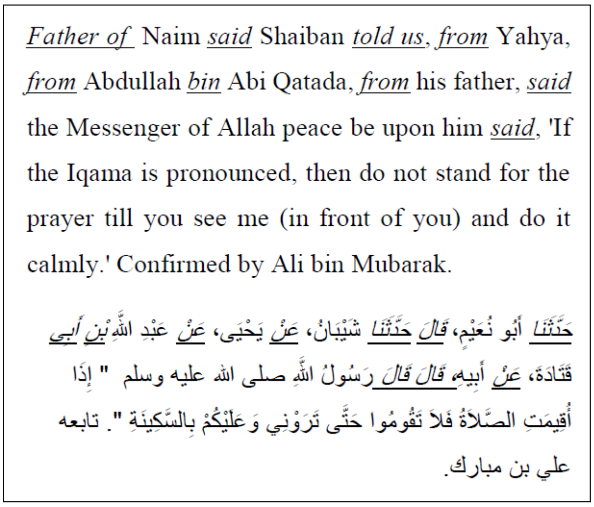
Looking at Table 1 above, it is evident that rule-based produced better results. However, it is not clear if the rule-based approach designed for one book can be applied to all Hadith books. In fact, researchers in Mahmood et al. (2018) created different regular expressions for the different Hadith books which imply that rule-based approaches cannot be universal. In the other hand, the ML approach is no better since it requires training data that represent all kinds of Hadith to make its performance acceptable when applied to the different Hadith books. For example, the study presented in Muazzam Siddiqui (2014) reported a drop in performance by 10 points once the model was tested with a different book. Another problem is associated with FST, segmentation will not work if the Hadith book does not use unique punctuation that surrounds each segment e.g. parenthesis around Matn.

Although we try to compare systems performance in the table above, it is crucial to clarify that the approaches are not comparable for two reasons. First, the data used to test the systems are different in terms of size and type. Second, the way system performance was measured is different in every study. For example. The study in Muazzam Siddiqui (2014) measured the precision of the system's ability to annotate the person's name as Narrator or not. That is whether each name is part of Isnad or Matn. Therefore, their system goal is not to segment but rather to identify narrators. To sum up, the results column in the table above for papers (Harrag, 2014)(Maraoui et al., 2018) (Boella, 2011)(Mahmood et al., 2018) reflect the segmentation performance, while the other studies report the performance of named entity recognition(NER) of narrators in Hadith, which can be used in segmentation.

3 Data Preparation

Before building the segmenter, testing data must be prepared. There is a countless number of Hadith books with a varying degree of authenticity. For the purpose of our project, we include the six famous books, commonly referred to *The Authentic Six* or canonical Hadith books. These books are *Sahih Albukhari*, *Sahih Muslim*, *Sunan Abu Dawood*, *Sunan Altarmithi*, and *Sunan Ibn Maja*. From each book, 40 Hadiths were carefully chosen to form 240 Hadiths that include Hadiths with irregular patterns. This is to ensure we accomplish two goals. First, overcome the limitation of previous studies that relied on one book; second, to produce a realistic performance of a segmenter that can deal with all types of Hadith books.

Then we gathered data required by our system to segment Hadith, which we refer to as training data. It is a list of Isand and Matn segments extracted from a well-structured Hadith book 'Sahih Albukhari'. To automate this task, we scrutinize Hadith parts to find that Isnad consists of a closed set of words that includes narrators' names and transmission words. The example of Isnad in Figure 2 underlines the unique words in Isnad. A common pattern in Isnad is the narrator's name which takes the form of *'first name - son of- father's name'*, so it is two names connected by a 'relation' word. Narrator's names are usually followed by 'transmission' words that reflect how the Hadith was reported e.g. *x 'heard' y* or *x 'said'*. Hence, transmission words will appear four words apart at most. Using this information, we created a list of Isnad lexicons that consists of 'transmission' and 'relation' words. Then we created a python script that tokenizes a Hadith and takes four words at a time to check if an Isnad lexicons is present. Once it detects a group of four words with no Isnad lexicons, it assumes the beginning of Matn text and separates the Hadith at that point. This approach will automatically detect Isnad with regular patterns only, so this step intends to collect the various names of narrators instead of segmenting the Hadith.



Father of Naim said Shaiban told us, from Yahya, from Abdullah bin Abi Qatada, from his father, said the Messenger of Allah peace be upon him said, 'If the Iqama is pronounced, then do not stand for the prayer till you see me (in front of you) and do it calmly.' Confirmed by Ali bin Mubarak.

حَدَّثَنَا أَبُو نُعَيْمٍ، قَالَ حَدَّثَنَا شَيْبَانُ، عَنْ يَحْيَى، عَنْ عَبْدِ اللَّهِ بْنِ أَبِي
قَتَادَةَ، عَنْ أَبِيهِ، قَالَ قَالَ رَسُولُ اللَّهِ صلى الله عليه وسلم " إِذَا
أُقِيمَتِ الصَّلاَةُ فَلاَ تَقُومُوا حَتَّى تَرَوْنِي وَعَلَيْكُمْ بِالسَّكِينَةِ ". تابعه
علي بن مبارك.

Figure 2: Isnad Example, Isnad lexicons underlined

We manually checked the result of this bootstrapping approach that produced a collection of more than four thousand Hadiths to form our gold standard of Isnad and Matn segments.

4 Segmentation Model

In this section, we discuss the techniques and algorithms used to build the Hadith segmenter. Figure 3 shows the structure and components of the Hadith segmenter model which takes in a new Hadith that goes through a preprocessing phase to remove diacritics and punctuations. Then Hadith is tokenized into N-grams depending on the technique applied, next each token is labelled as Isnad, Matn or Neither by comparing it with pre-compiled lists obtained from the gold standard created earlier as explained in section 3. Once every token is labelled, the model finds the best segmentation point of the Hadith. In the following lines, we give details of the techniques used to annotate Hadith tokens.

4.1 Tri-gram Technique

In a previous study, we have shown that considering the meaning of words by using the word em-

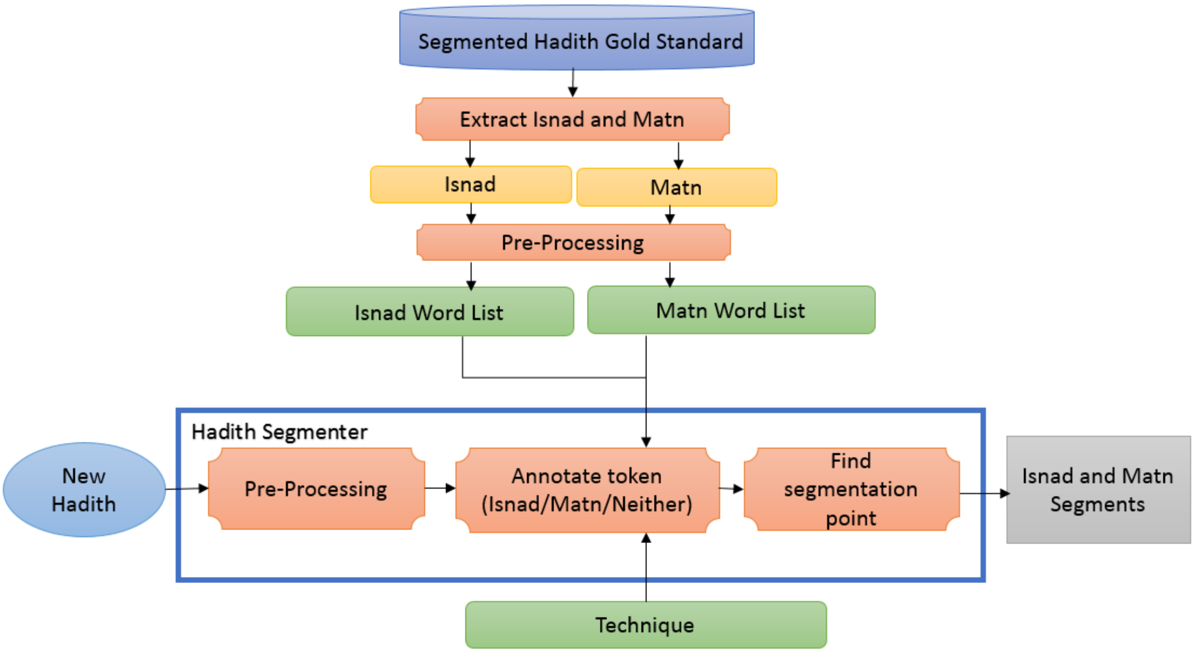


Figure 3: Hadith Segmenter Model

bedding technique does not perform well in Hadith segmentation. This is because such an approach relies on uni-grams that do not capture the unique pattern in Isnad. Furthermore, some words exist in both Isnad and Matn segments, Hence, in this experiment, we aim to capture Isnad patterns by using the N-gram technique. As illustrated in Figure 3, Isnad and Matn segments are extracted from the gold standard of segmented Hadiths. Then they are pre-processed to remove diacritics and punctuations. Finally, tri-gram, bi-gram and uni-gram lists for Isnad and Matn are created to be the evaluation lists in the annotation phase. The reason three lists are created is that a back-off approach can handle irregularity and missing information. For example, if an encountered tri-gram has no match in the tri-gram list of the training data, it can be annotated according to its components. Consider a narrator's full name is not captured in the lists, then Hadith lexicons like 'بن - *son of*' will enable the system to identify this tri-gram as part of the narrator's chain and label it 'Isnad'. This approach is detailed in Algorithm 1. Once every token is labelled, the system finds the segmentation point as detailed in Algorithm 2.

This approach produced 48% accuracy where only 115 out of 240 Hadiths were correctly segmented. To understand this disappointing result, we inspect the incorrectly segmented Hadith and found that the system rarely used the tri-gram fea-

Algorithm 1 Annotate Tri-gram tokens

```
Tokenize Hadith into Tr-igrams $T$
for $t \in T$ do
    if $t \in IsnadTrigramList$ then
        Label $t$ Isnad
    else if $t \in MatnTrigramList$ then
        Label $t$ Matn
    else
        Convert $t$ to Bigrams $b$
        if $b \in IsnadBigramList$ then
            Label $t$ Isnad
        else if $b \in MatnBigramList$ then
            Label $t$ Matn
        else
            Convert $t$ to Unitgram $u$
            if $u \in IsnadUnigramList$ then
                Label $t$ Isnad
            else if $u \in MatnUnigramList$ then
                Label $t$ Matn
            else
                Label $t$ Niether
```

Output:

Token1	Lable1	Token2	Label2	...

Isnad	Matn
حدثنا قتيبة حدثنا مروان بن معاوية الفزاري عن أبي يعفور عن الوليد بن العيزار عن أبي عمرو الشيباني أن رجلا قال لابن مسعود أي العمل أفضل قال سألت عنه رسول الله صلى الله عليه وسلم فقال الصلاة على مواقيتها قلت وماذا Qutaiba told us Marwan bin Muawiya al-Fizari from Abu Yafour from Al-Walid bin Al-Azar from Abu Amr AlShibani that a man said to Ibn Masood, which work is better? He said I asked the Messenger of Allah (PBUH): ”Which action is dearest to Allah?” He (PBUH) replied, ”Performing the prayer at its earliest fixed time.” I asked, ”What is next ?”	يا رسول الله قال وبر الوالدين O prophet, He said, ”Kindness towards parents.”

Table 2: Example of incorrect segmentation when applying tri-gram technique.

Algorithm 2 Find Segmentation point

```
for every token in Output List do
    if $label$ is Matn then
        if followed by Matn or Neither then
            Mark it as Segmentation Point
            Break
    else if $label$ is Neither then
        if followed by two $labels$ Matn
          or Neither then
            Mark it as Segmentation Point
            Break
```

ture, but rather relied on the bi-gram and uni-gram features to annotate tokens. Consider the example in Table 2, feeding this Hadith to the system produces 79 tri-grams, of which only 15 found a match in the tri-gram training set. The remaining 64 tri-grams relied on the bi-gram and uni-gram training set to be annotated. This dependency on bi-gram/uni-gram features to annotate Hadith tri-grams produced unreliable results as illustrated in the example. The phrase 'أن رجلا قال - *that a man said*' should mark the beginning of Matn, instead it was labelled as Isnad. This is because when the system did not find a match in the tri-gram training set, it applied the back-off approach and searched in the bi-gram and uni-gram lists. Since it found a match for the term 'قال *said* ' in the Isnad lists, it labelled the phrase accordingly. Therefore, using tri-grams did not prove useful in our case for two reasons. First, the training data is not large enough to cover all known narrators. Second, it is obtained from only one Hadith book which does not include all Hadith lexicons and patterns.

4.2 Bi-gram Technique

To improve the system performance, we omit tri-gram features and use bi-grams and uni-grams only.

The bi-gram technique produced better results as expected with 222 Hadith out of 240 were correctly segmented, showing 92.5% accuracy. In fact, it is able to segment Hadiths having different structures. For example, the traditional ones where a Matn start with a prophetic saying as shown in Table 3. Other Hadith structures include those containing irregular patterns where Matn starts with an introductory phrase followed by the prophetic saying as shown in Table 4, a dialogue with the prophet as shown in Table 5, or an explanation of a prophetic deed as in Table 6.

Then we analyse the faulty output and found that our system incorrectly segmented some Hadiths with irregular patterns. For example, a Hadith may contain a parallel Isnad, which is a chain of narrators that ends at the prophet followed by another chain of narrators that ends at the prophet again, as shown in Table 7. Another example of an irregular pattern in Isnad is shown in Table 8 which illustrates that Isnad may contain Matn patterns. Finally, Table 9 shows that some Hadith posses a vague segmentation point. Note that for space issues some Hadiths in the examples were truncated as indicated by (...).

Isnad	Matn
حدثنا كثير بن عبيد الحمصي حدثنا محمد بن خالد عن عبيد الله بن الوليد الوصافي عن محارب بن دثار عن عبد الله بن عمر قال Kathir bin Obeid Al-Homsi told us Mohammed bin Khalid from Obidallah bin Walid Al-Wasafi from Moharib bin dathar from Abdullah bin Omar said	قال رسول الله صلى الله عليه وسلم ابغض الحلال الى الله الطلاق The prophet (PBUH) said, Of all the lawful acts the most detestable to Allah is divorce.

Table 3: Correct segmentation, regular pattern.

Isnad	Matn
حدثنا أبو معمر قال حدثنا عبد الوارث عن عبد العزيز قال أنس Abu Muammar told us that Abdul Warith told us from Abdul Aziz said that Anas said	إنه ليمنعني أن أحدثكم حديثا كثيرا أن النبي صلى الله عليه وسلم قال من تعمد علي كذبا فليتبوأ مقعده من النار I refrain from telling you many things about the prophet because I heard the prophet (PBUH) said, ”He who deliberately forges a lie against me let him have his abode in the Hell.”.

Table 4: Correct segmentation, introductory statement.

Isnad	Matn
حدثنا قتيبه قال حدثنا الليث عن يزيد بن ابي حبيب عن ابي الخير عن عبد الله بن عمرو Qaytibah told us Alith from Yazid ibn Abi Habib from Abi Al-Khair from Abdullah bin Amr	ان رجلا سال رسول الله صلى الله عليه وسلم اي الاسلام خير قال تطعم الطعام وتقرا السلام على من عرفت ومن لم تعرف A man asked the Messenger of Allah (PBUH): ”Which act in Islam is the best?” He (PBUH) replied, ”To give food, and to greet everyone, whether you know or you do not.”

Table 5: Correct segmentation, conversation of the Prophet.

Isnad	Matn
حدثنا إسمعيل بن موسى الفزاري حدثنا شريك عن أبي إسحق عن الحارث عن علي بن أبي طالب قال Ismail bin Musa al-Fazari told us Sharik said Abu Ishaq from AlHarith from Ali bin Abi Talib said	من السنة أن تخرج إلى العيد ماشيا وأن تأكل شيئا قبل أن تخرج It is the Sunnah (prophetic tradition) to go out to the Eid prayer walking and eat something before you go out.

Table 6: Correct segmentation, no prophetic words.

Isnad	Matn
حدثنا مسدد قال حدثنا يحى عن شعبة عن قتادة عن أنس رضي الله عنه عن النبي Mosadad said Yahya told us Shoba heard Qatada from Anas may Allah be pleased with him, the Prophet	صلى الله عليه وسلم وعن حسين المعلم قال حدثنا قتادة عن أنس عن النبي صلى إلله عليه وسلم قال لا يؤمن أحدكم حتى يحب لأخيه ما يحب لنفس (PBUH), and from Husayn al-Muallim said Qatada told us from Anas that the Prophet (PBUH) said: ”No one of you becomes a true believer until he likes for his brother what he likes for himself”.

Table 7: Incorrectly segmented, Parallel Isnad.

Isnad	Matn
حدثنا نصر بن علي الجهضمي وأبو عمار والمعنى Nasser bin Ali Juhadhmi and Abu Ammar told us and the meaning	واحد واللفظ لفظ أبي عمار قالا أخبرنا سفيان بن عيينة عن الزهري عن حميد بن عبد الرحمن عن أبي هريرة قال أتاه رجل فقال يا رسول الله هلكت... Is the same but the words are of Ammar they said, Sufian bin Aayneh from Alzahri from Hamid bin Abdul Rahman on the authority of Abu Hurayrah said a man came and said, ”O Allah’s Apostle! I have been ruined.” ...

Table 8: Incorrectly segmented, Isnad contain Matn lexicons.

Isnad	Matn
أخبرنا محمد بن منصور قال حدثنا سفيان قال حدثنا يحيى بن سعيد عن مسلم بن أبي مريم شيخ من أهل المدينة ثم لقيت الشيخ فقال سمعت على بن عبد الرحمن يقول صليت إلى جنب ابن عمر فقلبت الحصى فقال لي ابن عمر Muhammad bin Mansour told us, that Sufian said Yahya bin Said told us about Muslim bin Abi Maryam a Sheikh from Madinah then I met the Sheikh and he said he heard Ali bin Abdul Rahman say I prayed beside Ibn Omar, while I turned the gravel he said	لا تقلب الحصى فإن تقليب الحصى من الشيطان وافعل كما رأيت رسول الله صلى الله عليه وسلم يفعل قلت وكيف رأيت رسول الله صلي الله عليه وسلم يفعل قال هكذ ... Do not fluctuate the gravel, turning the gravel is from the devil and do as I saw the Messenger of Allah peace be upon him do...

Table 9: Incorrectly segmented, names should be part of Matn.

5 Discussion

The findings of this study clearly show that using bi-grams for Hadith segmentation works better than tri-grams specifically because our training data is limited. Although the segmenter result is promising, not all Hadiths with irregular patterns were correctly segmented. In fact, this can be vague even for human annotators who are not experts in Hadith studies. For this reason, we argue that Hadith can be segmented to fine-grained segments that go beyond Isnad and Matn. This is because some Hadith contain information in the Isnad that was identified as Matn segments by our system. For example, a Hadith Isnad may include information about where a specific narrator comes from, then it continues the chain of narrators. Another example is a Hadith that starts a Matn segment with a piece of introductory information containing names of people which was identified as Isnad pattern by our segmenter as in Table 9. Thus, we aim to make an enhancement to Algorithm 2 to enable the segmenter output several segments instead of two, then apply probabilistic measures to identify the exact point of segmentation.

6 Conclusion

In this paper, we demonstrate the need for Hadith common datasets and our initiative to bridge the gap by automatically annotating Hadith corpus using NLP. The main objective of this study is to build a system that segments and annotates Hadith components, Isnad and Matn. Before building our system, we evaluated previous attempts to segment Hadith and found that the successful techniques rely on hand-crafted tools that cannot be generalized to segment all Hadith types. Furthermore, these systems were tested on a limited number of Hadiths from a single book. To address these limitations, our segmenter rely on NLP techniques and tested with Hadiths extracted from six books to ensure coverage of all Hadith types. Although it was successful in segmenting Hadith with 92.5% accuracy, examining the incorrect results points us to ways of improvements discussed in the paper.

References

Aqil Azmi and Nawaf Bin Badia. 2010. itree - automating the construction of the narration tree of hadiths (prophetic traditions). In *Proceedings of the 6th International Conference on Natural Language Processing and Knowledge Engineering(NLPKE-2010)*, pages 1–7.

Marco Boella. 2011. Regular expressions for interpreting and cross-referencing hadith texts. *Langues et Littératures du Monde Arabe (LLMA)*, 9(3):25–39.

Ibrahim Bounhas. 2019. On the usage of a classical arabic corpus as a language resource: related research and key challenges. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, 18(3):23.

Kais Dukes and Eric Atwell. 2012. Lamp: a multimodal web platform for collaborative linguistic analysis. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC12)*, pages 3268–3275. European Language Resources Association (ELRA).

Imane Guellil, Houda Saâdane, Faical Azouaou, Billel Gueni, and Damien Nouvel. 2019. Arabic natural language processing: An overview. *Journal of King Saud University-Computer and Information Sciences*.

Fouzi Harrag. 2014. Text mining approach for knowledge extraction in sahîh al-bukhari. *Computers in Human Behavior*, 30:558–566.

Emha Taufiq Luthfi, Nanna Suryana, and Adbulsamad Hasan Basari. 2018. Digital hadith authentication: A literature review and analysis. *Journal of Theoretical & Applied Information Technology*, 96(15).

Ahsan Mahmood, Hikmat Ullah Khan, Fawaz K Alarfaj, Muhammad Ramzan, and Mahwish Ilyas. 2018. A multilingual datasets repository of the hadith content. *International Journal of Advanced Computer Science and Applications*, 9(2):165–172.

Hajer Maraoui, Kais Haddar, and Laurent Romary. 2018. Segmentation tool for hadith corpus to generate tei encoding. In *International Conference on Advanced Intelligent Systems and Informatics*, pages 252–260. Springer.

Abobakr Bagais Muazzam Siddiqui, Mostafa Saleh. 2014. Extraction and visualization of the chain of narrators from hadiths using named entity recognition and classification. *Int. J. Comput. Linguist. Res*, 5(1):14–25.

Mohammad Arshi Saloot, Norisma Idris, Rohana Mahmud, Salinah Jaafar, Dirk Thorleuchter, and Abdullah Gani. 2016. Hadith data mining and classification: a comparative analysis. *Artificial Intelligence Review*, 46(1):113–128.

Can Modern Standard Arabic Approaches be used for Arabic Dialects? Sentiment Analysis as a Case Study

Kathrein Abu kwaik **Stergios Chatzikyriakidis** **Simon Dobnik**
CLASP and FLOV, University of Gothenburg, Sweden
{kathrein.abu.kwaik,stergios.chatzikyriakidis,simon.dobnik}@gu.se

Abstract

We present the Shami-Senti corpus, the first Levantine corpus for Sentiment Analysis (SA), and investigate the usage of off-the-shelf models that have been built for Modern Standard Arabic (MSA) on this corpus of Dialectal Arabic (DA). We apply the models on DA data, showing that their accuracy does not exceed 60%. We then proceed to build our own models involving different feature combinations and machine learning methods for both MSA and DA and achieve an accuracy of 83% and 75% respectively.

1 Introduction

There is a growing need for text mining and analytical tools for Social Media data, for example Sentiment Analysis (SA) tools which aim to distinguish people's views into positive and negative, objective and subjective responses, or even into neutral opinions. The amount of internet documents in Arabic is increasing rapidly (Ibrahim et al., 2015; Abdul-Mageed et al., 2011; Abdul-Mageed and Diab, 2011; Mourad and Darwish, 2013). However, texts from Social media are typically not written in Modern Standard Arabic (MSA) for which computational resources and corpora exist. These systems achieve reasonable accuracy on the designated tasks. For example, Abdul-Mageed et al. (2011) achieve an accuracy of 95% on the news domain. On the other hand, research on Dialectal Arabic (DA) in terms of SA is an open research question and presents considerable challenges (Badaro et al., 2019; Ibrahim et al., 2015).

The degree to which tools trained on MSA can be used on DA is still also an open research question. This is partly because different dialects differ from MSA to varying degrees(Kwaik et al., 2018). Furthermore, the speakers of Arabic present us with clear cases of Diglossia (Ferguson, 1959), where MSA is the official language used for education, news, politics, religion and, in general, in any type of formal setting, but dialects are used in everyday communication, as well as in informal writing (Versteegh, 2014).

In this paper, we examine whether it is possible to adapt classification models that have been trained and built on MSA data for DA from the Levantine region, or whether we should build and train specific models for the individual dialects, therefore considering them as stand-alone languages. To answer this question we use Sentiment Analysis as a case study. Our contributions are the following:

- We systematically evaluate how well the ML models on MSA for SA perform on DA of Levantine;
- We construct and present a new sentiment corpus of Levantine DA;
- We investigate the issue of domain adaptation of ML models from MSA to DA.

The paper is organised as follows: in Section 2, we briefly discuss the task of SA and present related work on Arabic. In Section 3, we describe an extension of the Shami corpus of Levantine dialects (Qwaider et al., 2018) annotated for Sentiment, Shami-Senti. In Section 4, we present the experimental setting and results of adapting MSA models to the dialectal domain as well as training specific models. We conclude and discuss directions for future research in Section 5.

2 Arabic Sentiment Analysis

Manually gathering information about users' opinions and sentiment data is time-consuming. This is why more and more companies and organisations are interested in automatic SA methods to help them understand it. SA refers to the usage of variety of tools from Natural Language Processing (NLP), Text Mining and Computational Linguistics to examine a given piece of text and identify the dominant sentiment subjectivity in it (Liu, 2012; Ravi and Ravi, 2015). SA is usually categorised into three main sentiment polarities: Positive (POS), Negative (NEG) and Neutral (NUT). SA is frequently used interchangeably with Opinion Mining (Abdullah and Hadzikadic, 2017).

At first glance, Sentiment Analysis is a classification task. It is a complex classification task as if one dives deeper, they are faced with a number of challenges that affect the accuracy of any SA model. Some of these challenges are: (i) Negation terms (Farooq et al., 2017), (ii) Sarcasm (Ghosh and Veale, 2016), (iii) Word ambiguity and (iv) Multi-polarity.

As a result of the rapid development of social media and the use of Arabic dialectal texts, there is an emerging interest in DA. Farra et al. (2010) propose a model of sentence classification (SA) in Arabic documents. They extract sets of features and calculate the total weight for every sentence. A J48 Decision tree algorithm is used to classify the sentences w.r.t. sentiment, achieving an accuracy of 62%.

Gamal et al. (2018) collect tweets from different Arabic regions using different keywords and phrases. The tweets include opinions about a variety of topics. They annotate their polarity by checking if they contain positive or negative terms and without considering the reverse polarity in the presence of negation terms. Then, they apply six machine learning algorithms on the data and achieve an accuracy between 82% and 93%.

Oussous et al. (2018) build an SA model to classify the sentiment of sentences. The authors construct a Moroccan corpus, where the data are collected from Twitter, and annotate it. Multiple algorithms are used, e.g. Support Vector Machines (SVM), Multinomial Nave Bayes (MNB) and Mean Entropy (ME). The SVM model achieves an accuracy of 85%. Ensemble learning by majority voting and stacking is also tried. Using the three aforementioned algorithms in the two models, they attain an accuracy of 83% and 84% respectively. Another work using the same classifiers is described in (El-Halees, 2011). The dataset covers three domains: education, politics and sports. The resulting accuracy is 80%.

A framework for Jordanian SA is proposed in (Duwairi et al., 2014). The authors create a corpus of Jordanian tweets and build a mapping lexicon from Jordanian to MSA that turns any dialectal word into an MSA word, before classifying the tweet. In order for the tweets to be annotated, crowd-sourcing is used. They further use Rapid Miner for pre-processing, filtering, and classification. Three classifiers are used to evaluate the performance of the proposed framework with 1000 tweets: Nave Bayes (NB), (SVM) and k-nearest neighbour (KNN). The NB model gets the highest accuracy with 76.78%.

Binary sentiment classification for Egyptian using a NB classifier is investigated in (Abdul-Mageed et al., 2011). An accuracy of 80% is achieved. Similarly, the Tunisian dialect is addressed in (Medhaffar et al., 2017). Here, the authors create a Tunisian corpus for SA containing 17K comments from social media. Applying Multi-Layer Perceptron (MLP) and SVM on the corpus they get 0.22 and 0.23 error rate respectively. Another line of work addresses the Saudi dialects (Al-Twairesh et al., 2018; Rizkallah, Sandra and Atiya, Amir and ElDin Mahgoub, Hossam and Heragy, Momen", editor="Hassanien, Aboul Ella and Tolba, Mohamed F. and Elhoseny, Mohamed and Mostafa, Mohamed , 2018) and some addresses the United Arab Emirates dialects (Baly et al., 2017a,b).

Several works exploit lexicon-based sentiment classifiers for Arabic. A sentiment lexicon is a lexicon that contains both positive and negative terms along with their polarity weights (Badaro et al., 2014; Abdul-Mageed and Diab, 2012; Badaro et al., 2018). The SAMAR system (Abdul-Mageed et al., 2014) involves two-stage classification based on a sentiment lexicon. The first classifier detects subjectivity and objectivity of documents, which is followed by another classifier to detect the polarity. They employ different datasets and examine various features combinations. Similar work is reported in (Mourad and Darwish, 2013; Al-Rubaiee et al., 2016), where both NB and SVM are explored, achieving an accuracy between 73% and 84% .

Abdulla et al. (2013) compare the performance of corpus-based sentiment classification and lexicon-based classification in Arabic. The accuracy of the lexicon approach does not exceed 60%. They conclude that corpus-based methods perform better using SVM and light stemming.

Overall, there is a considerable amount of work on SA and DA but none of these approaches considered the performance of the classifiers across the domains for which limited data exist.

Lexicon	Negative	Positive	Negation
LABR	348	319	37
Moarlex	13411	4277	▨
SA lexicon	3537	855	▨

Table 1: The number of terms in sentiment lexicons

3 Building Shami-Senti

The question of sentiment analysis has not yet been fully examined for Levantine dialects: Palestinian, Jordanian, Syrian and Lebanese. For this reason, we extend the Shami corpus (Qwaider et al., 2018) by annotating part of it for sentiment. We call the new corpus Shami-Senti.

We build Shami-Senti as follows:

1. Manually extract sentences that contains sentiment words, reviews, opinions or feelings from the Shami corpus;
2. Split the sentences and remove any misleading words or very long phrases (set sentences be no longer than 50 words);
3. Try to avoid ironic and sarcastic text where the intended sentiment is reversed. For example, sentences like the following: تصدقوا احنا ناكرين الجميل الرجال زي الفل "I believe we are ungrateful, this man is perfect" (Karoui et al., 2017), are avoided.

3.1 Sentiment annotation

Two methods have been used to annotate the corpus, a lexicon-based annotation and human annotation. The sentence is marked as positive if it contains positive terms or negated negative terms. It is considered negative if it contains negative terms or negation of positive terms. Any sentence that contains a mixture of positive and negative terms or no sentiment terms is marked as mixed or neutral.

In the lexicon-based annotation, we use three sentiment lexicons: the one provided by LABR (Aly and Atiya, 2013) which contains negative, positive and negated terms; the Moarlex (Youssef and El-Beltagy, 2018) and the SA lexicon (ElSahar and El-Beltagy, 2014) which contain only positive and negative terms. Table 1 illustrates the numbers of terms in each lexicon.

First, for the lexicon-based annotation we extracted 1,000 sentences from the Shami corpus and commissioned a Levantine native speaker to annotate them for sentiment. Then, we implemented Algorithm 1 to automatically annotate the same 1,000 sentences. We computed the inter-annotator agreement but the result was very bad, the disagreement was up to 80%. As a result, we did not consider this method as reliable for annotation, hence we chose to annotate the data set manually.

```
Result: Annotate 1,000 sentences
Build Positive, Negative, Negation lists of
  words extracted from the three lexicons;
Polarity = 0;
for sentence in Shami-Senti do
    count number of positive terms; Then
      Polarity $++$;
    count number of negative terms; Then
      Polarity $--$;
    check if there is a negation,Then Polarity
      $*-1$;
    if Polarity $> 0$ then
        Polarity is Positive;
    else if Polarity $< 0$ then
        Polarity is negative;
    else
        Polarity is mixed;
    end
end
```

Algorithm 1: Lexicon-based annotation of 1,000 Shami sentences

For the human annotation method, we asked two native speakers, one from Palestine and another from Syria, to annotate 533 sentences with 1 if these are positive, 0 if negative and -1 if neutral or mixed sentences. Then we calculated the inter-annotator agreement between them using Kappa statistics (Carletta, 1996) giving us $\kappa = 0.838$ which is a very good agreement. Since the data was split into separate dialects, we asked the annotators to annotate the parts that they were most familiar with, for example, the Palestinian speaker annotated the sentences in Palestinian and Jordanian, while the Syrian speaker annotated the Syrian and Lebanese sentences. We extracted more than 5,000 sentences/tweets for this purpose, and have annotated nearly 2,000 of them so far. Table 2 shows the number of tweets per category.

4 Experiments

In order to estimate the performance of the SA models, which have built on MSA data, on DA evaluation data, we use the following two corpora in our experiments.

- LABR (Aly and Atiya, 2013): this is one of the largest SA datasets to-date for Arabic. It consists of over 63k book reviews written

Corpus	NEG	POS	Mix
Shami-Senti	935	1064	243
LARB 3 Balanced	6580	6578	6580
LABR 2 Balanced	6578	6580	▨
ASTD	1496	665	738

Table 2: The number of instances per category in Shami-Senti and other sentiment corpora used in our experiments

in MSA with some dialectal words. LABR is available with different subsets: the authors split it into 2,3,4 and 5 sentiment polarities with balanced and unbalanced divisions. They depend on the user ratings to classify sentences. Thus, 4 and 5 stars ratings are taken as positive, 1 and 2 star ratings are taken as negative and 3 star ratings are taken as mixed or neutral. The fact that LABR is limited to one domain, book reviews, makes it difficult to use it as a general SA model.

- ASTD (Nabil et al., 2015): it is an Arabic SA corpus collected from Twitter and focuses on the Egyptian dialects. It consists of about 10k tweets, which are classified as objective, subjective positive, subjective negative, and subjective mixed.

Table 2 shows the number of instances of each polarity label in different corpora.

In all experiments, we use the same machine learning algorithms that have been used by the LABR baseline. These are:

1. Logistic Regression (LR)
2. Passive Aggressive (PA)
3. Linear Support Vector classifier (LinearSVC)
4. Bernoulli Naive-Bayes (BNB)
5. Stochastic Gradient Descent (SGD)

The choice is motivated as follows. LR is strong in explaining the relationship between one dependent variable and independent variables (Feng et al., 2014), while PA is suitable for large-scale learning (Crammer et al., 2006). LinearSVC is effective in cases where the number of dimensions is greater than the number of samples (Kumar and Goel, 2015). BNB is suitable for discrete data (Shimodaira, 2014), and SGD is a linear classifier which implements regularised linear models with stochastic gradient descent (SGD) learning. It is a simple baseline classifier related to neural networks (Günther and Furrer, 2013).

In addition, we also use some popular linear and probabilistic classifiers. Hence, we use Multinomial Naive-Bayes (MNB), which is suitable for classification of discrete features. The multinomial distribution normally requires integer feature counts and it works well for fractional counts like tf-idf (Xu et al., 2017). We further use Complement Naive-Bayes (CNB), which is particularly suited for imbalanced data sets. CNB uses statistics taken from the complement of each class to compute the models weights.[1] Generally speaking, a NB classifier converges quicker than discriminative models like logistic regression, so one need less training data. The last one is the Ridge Classifier (RC). Its most important feature is that it does not remove irrelevant features but rather minimise their impact on the trained model (Drucker et al., 1997). All of the algorithms are implemented using the `scikit learn` library in Python (Pedregosa et al., 2011) .

4.1 Three class sentiment classification

We start with the baseline from LABR, and use the 3-class balanced data set. Table 3 states the number instances of each polarity class for both training and testing. The baseline method from LABR uses the language model to predict the polarity class. We conduct two experiments: one with unigrams, and one with both unigrams and bigrams. We build the models by transforming the data into a numerical vectors using the Term Frequency vectorize method. First, a Language Model is built by extracting unigrams and bigrams from the dataset and computing their term-frequencies to create the two models, the unigrams, and the combined unigrams and bigrams. Then, every sentence goes through a classifier which produces a probability of the class the sentence belongs to. Table 4 shows the accuracy of the classifiers on the test set trained on the 3-class balanced LABR. The unigram and bigram TF method is doing marginally better than the unigram language model, particularly with the PA classifier. The four classifiers achieve an accuracy between 58% and 59% to classify MSA sentences. BNB is the worst performing classifier with 35% and 34% accuracy respectively. The reason for this might be that we have a large number of features (i.e. individual words) and since BNB models are counting the words that are not present in the document they do not perform well.

[1] https://scikit-learn.org/dev/modules/naive_bayes.html

	Positive	Negative	Mix
Train	4936	4935	4936
Test	1644	1643	1644

Table 3: The number of instances per category in balanced LABR3

Classifier	Accuracy TF_wg1	Accuracy TF_wg1+2
Logistic Regression	59	59
Passive Aggressive	54	58
Linear SVC	57	58
Bernoulli NB	35	34
SGD Classifier	59	59

Table 4: Accuracy of the baseline on LABR3 (Tf-wg : is the Term Frequency on Word grams)

	Training Dataset	
Classifier	LABR3	Shami-Senti
Logistic Regression	46	62
Passive Aggressive	43	64
Linear SVC	44	64
Bernoulli NB	11	48
SGD Classifier	45	65

Table 5: Accuracy of the baseline TF_wg1+2 trained on LABR3 and Shami-Senti and tested on Shami-Senti

MSA has been researched more from an NLP perspective than DA, and therefore several sentiment analysis approaches have been built for it. The question we want to ask, is whether we can apply these NLP approaches directly on DA or new resources and models are required for DA. We, thus, test the reliability of models that are built on MSA data and adapt them to DA data. Here, we test the baseline bigram TF model on the test part of the Shami-Senti corpus. Table 5 shows the accuracy from this experiment where we trained the baseline by LABR3 and tested it using Shami-Senti. The accuracy is significantly worse, with a drop of more than 10%. The table also shows the accuracy of the baseline when we trained and tested it on Shami-Senti. The highest accuracy was 65% using SGD classifier.

Given the baseline model's poor performance on DA, we build a new SA model. This model also depends on language modelling, where we use a combination of both word-level and character-level n-grams. After several experiments, we observe that a language model that combines features of word-level unigrams and bigrams with character-level n-grams from 2 to 5 gives the best accuracy. We test eight different machine learning algorithms to predict sentiment classification.

Classifier	Model 1	Model 2
Ridge Classifier	57	59
Logistic Regression	59	60
Passive Aggressive	55	58
Linear SVC	57	59
SGD Classifier	59	60
Multinomial NB	57	59
Bernoulli NB	49	49
Complement NB	57	59

Table 6: Accuracy of the proposed model trained and tested on LABR3; Model 1: unigram word level with (2,5) character grams; In Model 2 (unigram,bigrams) word level with (2,5) character grams

Classifier	Accuracy
Ridge Classifier	43
Logistic Regression	46
Passive Aggressive	43
Linear SVC	45
SGD Classifier	50
Multinomial NB	40
Bernoulli NB	44
Complement NB	42

Table 7: Accuracy of the proposed model trained on LABR3 and tested on Shami-Senti

Table 6 shows the accuracy of our model on the LABR 3-class balanced dataset. In Model 1, we test using only unigram words and character grams from 2 to 5, while in Model 2 we add an extra bigram word-level to Model 1. The SGD and LR classifiers give the highest accuracy 60% on Model 2 which is slightly higher than the base line where it was 59%. In all experiments later we will refer to Model 2 as our proposed model. We test this model which was trained on LABR 3 on Shami-Senti. Table 7 shows the results. The model is not performing well on DA achieving an accuracy of 50% using the SGD classifier. This indicates that MSA models are not transferable to DA.

We also train the selected classifier configurations on the Shami-Senti corpus (Table 8). NB algorithms give the highest accuracy with 71%,

Classifier	Accuracy
Ridge Classifier	69
Logistic Regression	67
Passive Aggressive	68
Linear SVC	69
SGD Classifier	68
Multinomial NB	71
Bernoulli NB	71
Complement NB	71

Table 8: Accuracy of the proposed model 3-class classification trained and tested on Shami-Senti

while the differences between the classifiers are marginal. We train the model using 1,000 samples and get an accuracy of 69% by MNB which indicates that increasing the size of the data set has a significant impact on the model accuracy.

4.2 Binary Sentiment classification

The accuracy obtained for the 3-class classification is not very high. This seems to be, at least partly, because the mixed class contains both positive and negative examples which makes the classification task difficult. LABR considers a 3-star rating as a mixed or neutral class. This is not very accurate since, in some cases, users use this rating as negative, while in others as somewhat positive. Table 9 shows three samples from the third neutral class in LABR that we consider should potentially belong to different classes.

We reduce the classification to a binary classification task, by focusing on the positive and negative classes only. Using the LABR, we build a baseline with bigram word counts and another model based on term frequency of unigram and bigram words. After that, we build a unigram and bigram TF words model and a (2-5) TF character model (the proposed model) and apply the LABR 2 classes dataset. The accuracy for the three models, in addition the accuracy of the same models tested on Shami-Senti are shown in Table 10.

We also test the transfer of models between different dialects. We train the classifiers with the proposed configurations to build a model on the ASTD corpus that contains Egyptian dialect data, and test it on both the ASTD and the Shami-Senti corpus. The results are shown in Table 11. The proposed model gives an accuracy up to 83% using linear classifiers like SVC and SGD when it is trained and tested on MSA LABR data set, while it gives an accuracy up to 58% when it is tested on Shami-Senti. We also get an accuracy of 83% when we train and test the model on the ASTD corpus and using an MNB classifier and 57% accuracy when we test it on Shami-Senti.

Models which are trained and built on MSA data can not fit well in dialectal data, even though both of them are considered similar languages. The accuracy for any model tested on Shami-Senti does not exceed 60% (Table 10 and Table 11) in all experiments. Table 12 shows that the model works better for binary sentiment classification with 74% accuracy using MNB, when the model is trained and tested on Shami-Senti. The high accuracy could be due to the quality of the data and human performed annotations. The high accuracy achieved (83%) on both LABR and ASTD indicates that increasing the size of the corpus improves the classification task.

4.3 Feature engineering

In order to improve 3-class sentiment classification, we consider adding more features to the language model. The classifiers with the new features are applied to both the LABR and the Shami-Senti corpus. Based on the three lexicons, (LABR, Moarlex and SA lexicon) we count the number of positive and negative terms in the sentence, and then calculate their probability using Equation 1 and 2. In addition, we use an additional binary feature to indicate if the sentence contains a negation term or not.

$$P(POS) = \frac{\#pos\_terms\_in\_the\_sentence}{total\_length} \quad (1)$$

$$P(NEG) = \frac{\#neg\_terms\_in\_the\_sentence}{total\_length} \quad (2)$$

The three extra features and the word and character n-gram features are combined through the FeatureUnion estimator function in scikit-learn [2] to build and train the models. After many trials we chose to specify the weight of the transformer matrix to 0.4 for the positive feature, 0.2 for the negative feature, 0.4 for the negation feature and 2 for the language model features. The weight for the language module feature is doubled in order to increase their impact. Table 13 shows the result for the SGD and MNB classifiers on both the MSA and Shami corpus. On the MSA data set we get an accuracy of 58.1% and 58.2% using SGD and MNB respectively, which is not a valuable improvement compared to the results in Table 4.

[2] https://scikit-learn.org/0.18/modules/pipeline.html

Sentence		Corrected Polarity
Arabic	بعض الكلمات استوقفتني وجعلتني أفكر.وبعضها الاخر جعلني أبتسم. والبعض جعلني أغرق في الضحك. اشتقت لهذا الأسلوب في الكتابة	Positive
English	Some words stopped me and made me think. Some of them made me smile. And some made me drowned in laughter !!! I missed this method in writing.	
Arabic	الكتاب ليس بسيء ولكنه أثار ضجة اعلانية أكثر من اللازم	Mix
English	The book is not bad but it has too much publicity more than it deserves	
Arabic	بالكاد اكملتها تفاصيلها كثيرة ومبهمة ومملة وبشعه جدا أشبه بالكوابيس	Negative
English	Barely completed, the details are many, opaque, boring and very ugly like nightmares	

Table 9: Examples annotated as neutral in LABR3 and our corrected polarity

	counting 2g		**TF_wg 1+2**		**OUR Model**	
Classifier	**LABR**	**Shami**	**LABR**	**Shami**	**LABR**	**Shami**
Ridge Classifier	78	53	81	54	83	57
Logistic Regression	80	57	80	56	82	58
Passive Aggressive	78	53	81	53	82	56
Linear SVC	78	55	81	55	83	58
SGD Classifier	80	53	82	54	83	56
Multinomial NB	78	52	80	53	82	55
Bernoulli NB	76	48	76	47	74	48
Complement NB	78	51	80	53	82	55

Table 10: Accuracy for binary classifiers with different feature sets trained on the LABR2 dataset and tested on LABR2 and Shami-Senti

	Testing Dataset	
Classifier	**ASTD**	**Shami-Senti**
Ridge Classifier	81	55
Logistic Regression	77	55
Passive Aggressive	82	57
Linear SVC	81	56
SGD Classifier	82	56
Multinomial NB	83	57
Bernoulli NB	82	58
Complement NB	82	58

Table 11: Accuracy of the proposed model on binary classification trained on ASTD and tested on ASTD and Shami-Senti

Classifier	**2 classes**
Ridge Classifier	73
Logistic Regression	74
Passive Aggressive	73
Linear SVC	73
SGD Classifier	73
Multinomial NB	74
Bernoulli NB	72
Complement NB	75

Table 12: Accuracy of the proposed model on binary classification trained and tested on Shami-Senti

On the dialectal data set, the accuracy of the SGD classifier is decreased from 68% in Table 8 to 66%. We hypothesise that this is because of the lexicon which includes primarily MSA terms and Egyptian terms rather than Levantine sentiment terms so the probabilities of features are less accurate. Even though, MNB is still able to improve the classification accuracy from 71% to 75.2%.

The effect of feature engineering has more effect on the dialectal data, as the size of the dataset

	F.Eng	
Classifier	**LABR**	**Shami**
SGD Classifier	58.1	66
Multinomial NB	58.2	75.2

Table 13: Accuracy of two classifiers using feature engineering on 3-class classification task

	Accuracy	
Experiment name	**LABR**	**Shami-Senti**
LSTM(100)	42	64.7
BiLSTM(200)	41.3	61.8

Table 14: Accuracy of deep learning models 3-class LABR and Shami-Senti

plays an important rule. Adding more informative features to a small dataset help the system to learn and predict the correct class.

4.4 Deep learning models

Deep learning has emerged as a powerful machine learning technique and has already produced state-of-the-art prediction results for SA (Zhang et al., 2018; Rojas-Barahona, 2016; Tang et al., 2015). In this section, we conduct a small experiment implemented using the Keras library to test two standard deep learning models to classify sentiment in our datasets.

The first model is a Long Short-Term Memory (LSTM) model. It consists of:

1. an embedding layer with max_features (MF) equal to the maximum number of words (7000), weighted matrix which is a 7000 * 100 matrix extracted from Aravec, a pre-trained Arabic word embedding model (Soliman et al., 2017), and max_lenght = 50 as the maximum number of words in each sentence;
2. an LSTM layer with an output of 100 and 50% of dropout rate;
3. a dense layer with an output of 30 followed by a final sigmoid layer with 3 sentiment classes.

The second model, BiLSTM(200), uses a Bidirectional LSTM layer with an output of 200 rather than an LSTM layer with an output of 100. We train the model using the Adam optimiser and a batch size of 50. We train the two models on the LABR3 balanced corpus. In addition, we do the same experiments on Shami-Senti. Table 14 shows the results for both datasets.

The test accuracy, in general, is not at the desired level. It is clear that feature-based machine learning classifiers outperform deep learning networks.

5 Conclusion and future work

In this paper, we have investigated different ML algorithms and built a model for SA that combines word n-grams with character n-grams, in addition to other supportive features. The model outperforms the baseline on both big and small datasets, and gets an accuracy of 83% for MSA and 75.2% for Shami-Senti. What is more important, we have shown that using a model trained on MSA SA data and then testing it on dialectal SA data, does not produce good results. This suggests that MSA models cannot be easily, if at all, used in dealing with DA. There is, thus, a growing need for the creation of computational resources, not only for MSA, but also for DA. The extent of this need, and whether some resources can be re-used up to some point, is something that needs to be further investigated. In the case we have been looking at in this paper, it seems that the existing MSA approaches will not be very usable when thrown at dialectal data. It goes without saying that the same situation holds when one tries to use computational resources used for a specific dialect of Arabic to another one, modulo the closeness (in some computational measure to be defined) between the two varieties.

In the future, we plan to continue our work on the annotation of the Shami-Senti corpus exploiting more automatic ways and aiming at enhancing it in terms of size, quality and distribution. Once this happens, we plan to investigate the application of the same deep learning models used in this paper, as well as more sophisticated ones. On a similar note, we are currently working on using more sophisticated deep learning models for the same sized dataset we have been using in this paper. This is part of a more general question of using deep learning with small datasets: whether such an endeavour is possible, and if yes, what are the techniques and network tweaks that make this possible.

Acknowledgements

The authors are supported by grant 2014-39 from the Swedish Research Council, which funds the Centre for Linguistic Theory and Studies in Probability (CLASP) in the Department of Philosophy, Linguistics, and Theory of Science at the University of Gothenburg.

References

Muhammad Abdul-Mageed and Mona Diab. 2012. Toward building a large-scale Arabic sentiment lexicon. In *Proceedings of the 6th international global WordNet conference*, pages 18–22.

Muhammad Abdul-Mageed, Mona Diab, and Sandra Kübler. 2014. SAMAR: Subjectivity and sentiment analysis for Arabic social media. *Computer Speech & Language*, 28(1):20–37.

Muhammad Abdul-Mageed and Mona T Diab. 2011. Subjectivity and sentiment annotation of modern standard Arabic newswire. In *Proceedings of the 5th linguistic annotation workshop*, pages 110–118. Association for Computational Linguistics.

Muhammad Abdul-Mageed, Mona T Diab, and Mohammed Korayem. 2011. Subjectivity and sentiment analysis of modern standard Arabic. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pages 587–591. Association for Computational Linguistics.

Nawaf A Abdulla, Nizar A Ahmed, Mohammed A Shehab, and Mahmoud Al-Ayyoub. 2013. Arabic sentiment analysis: Lexicon-based and corpus-based. In *2013 IEEE Jordan conference on applied electrical engineering and computing technologies (AEECT)*, pages 1–6. IEEE.

Malak Abdullah and Mirsad Hadzikadic. 2017. Sentiment analysis on Arabic tweets: Challenges to dissecting the language. In *International Conference on Social Computing and Social Media*, pages 191–202. Springer.

Hamed Al-Rubaiee, Renxi Qiu, and Dayou Li. 2016. Identifying Mubasher software products through sentiment analysis of Arabic tweets. In *2016 International Conference on Industrial Informatics and Computer Systems (CIICS)*, pages 1–6. IEEE.

Nora Al-Twairesh, Hend S. Al-Khalifa, AbdulMalik Al-Salman, and Yousef Al-Ohali. 2018. Sentiment Analysis of Arabic Tweets: Feature Engineering and A Hybrid Approach. *CoRR*, abs/1805.08533.

Mohamed Aly and Amir Atiya. 2013. Labr: A large scale Arabic book reviews dataset. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 494–498.

Gilbert Badaro, Ramy Baly, Hazem Hajj, Wassim El-Hajj, Khaled Bashir Shaban, Nizar Habash, Ahmad Al-Sallab, and Ali Hamdi. 2019. A Survey of Opinion Mining in Arabic: A Comprehensive System Perspective Covering Challenges and Advances in Tools, Resources, Models, Applications, and Visualizations. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, 18(3):27.

Gilbert Badaro, Ramy Baly, Hazem Hajj, Nizar Habash, and Wassim El-Hajj. 2014. A large scale Arabic sentiment lexicon for Arabic opinion mining. In *Proceedings of the EMNLP 2014 workshop on Arabic Natural Language Processing (ANLP)*, pages 165–173.

Gilbert Badaro, Hussein Jundi, Hazem Hajj, Wassim El-Hajj, and Nizar Habash. 2018. ArSEL: A Large Scale Arabic Sentiment and Emotion Lexicon. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Paris, France. European Language Resources Association (ELRA).

Ramy Baly, Gilbert Badaro, Georges El-Khoury, Rawan Moukalled, Rita Aoun, Hazem Hajj, Wassim El-Hajj, Nizar Habash, and Khaled Shaban. 2017a. A characterization study of Arabic twitter data with a benchmarking for state-of-the-art opinion mining models. In *Proceedings of the third Arabic natural language processing workshop*, pages 110–118.

Ramy Baly, Georges El-Khoury, Rawan Moukalled, Rita Aoun, Hazem Hajj, Khaled Bashir Shaban, and Wassim El-Hajj. 2017b. Comparative evaluation of sentiment analysis methods across Arabic dialects. *Procedia Computer Science*, 117:266–273.

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

Koby Crammer, Ofer Dekel, Joseph Keshet, Shai Shalev-Shwartz, and Yoram Singer. 2006. Online passive-aggressive algorithms. *Journal of Machine Learning Research*, 7(Mar):551–585.

Harris Drucker, Christopher JC Burges, Linda Kaufman, Alex J Smola, and Vladimir Vapnik. 1997. Support vector regression machines. In *Advances in neural information processing systems*, pages 155–161.

Rehab M Duwairi, Raed Marji, Narmeen Sha'ban, and Sally Rushaidat. 2014. Sentiment analysis in Arabic tweets. In *2014 5th International Conference on Information and Communication Systems (ICICS)*, pages 1–6. IEEE.

Alaa M El-Halees. 2011. Arabic opinion mining using combined classification approach. *Arabic opinion mining using combined classification approach.*

Hady ElSahar and Samhaa R El-Beltagy. 2014. A fully automated approach for Arabic slang lexicon extraction from microblogs. In *International conference on intelligent text processing and computational linguistics*, pages 79–91. Springer.

Umar Farooq, Hasan Mansoor, Antoine Nongaillard, Yacine Ouzrout, and Muhammad Abdul Qadir. 2017. Negation Handling in Sentiment Analysis at Sentence Level. *JCP*, 12(5):470–478.

Noura Farra, Elie Challita, Rawad Abou Assi, and Hazem Hajj. 2010. Sentence-level and document-level sentiment mining for Arabic texts. In *2010 IEEE international conference on data mining workshops*, pages 1114–1119. IEEE.

Jiashi Feng, Huan Xu, Shie Mannor, and Shuicheng Yan. 2014. Robust logistic regression and classification. In *Advances in neural information processing systems*, pages 253–261.

Charles A Ferguson. 1959. Diglossia. *word*, 15(2):325–340.

Donia Gamal, Marco Alfonse, El-Sayed M. El-Horbaty, and Abdel-Badeeh M.Salem. 2018. Opinion Mining for Arabic Dialects on Twitter. *Egyptian Computer Science Journal*, 42(4):52–61.

Aniruddha Ghosh and Tony Veale. 2016. Fracking sarcasm using neural network. In *Proceedings of the 7th workshop on computational approaches to subjectivity, sentiment and social media analysis*, pages 161–169.

Tobias Günther and Lenz Furrer. 2013. GU-MLT-LT: Sentiment analysis of short messages using linguistic features and stochastic gradient descent. In *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, volume 2, pages 328–332.

Hossam S Ibrahim, Sherif M Abdou, and Mervat Gheith. 2015. Sentiment analysis for Modern Standard Arabic and colloquial. *arXiv preprint arXiv:1505.03105*.

Jihen Karoui, Farah Banamara Zitoune, and Veronique Moriceau. 2017. Soukhria: Towards an irony detection system for Arabic in social media. *Procedia Computer Science*, 117:161–168.

Suresh Kumar and Shivani Goel. 2015. Enhancing Text Classification by Stochastic Optimization method and Support Vector Machine. *International Journal of Computer Science and Information Technologies, 6 (4)*, pages 3742–3745.

Kathrein Abu Kwaik, Motaz Saad, Stergios Chatzikyriakidis, and Simon Dobnika. 2018. A Lexical Distance Study of Arabic Dialects. *Procedia computer science*, 142:2–13.

Bing Liu. 2012. Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1):1–167.

Salima Medhaffar, Fethi Bougares, Yannick Estève, and Lamia Hadrich-Belguith. 2017. Sentiment analysis of Tunisian dialects: Linguistic ressources and experiments. In *Proceedings of the third Arabic natural language processing workshop*, pages 55–61.

Ahmed Mourad and Kareem Darwish. 2013. Subjectivity and sentiment analysis of modern standard Arabic and Arabic microblogs. In *Proceedings of the 4th workshop on computational approaches to subjectivity, sentiment and social media analysis*, pages 55–64.

Mahmoud Nabil, Mohamed Aly, and Amir Atiya. 2015. Astd: Arabic sentiment tweets dataset. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2515–2519.

Ahmed Oussous, Ayoub Ait Lahcen, and Samir Belfkih. 2018. Improving Sentiment Analysis of Moroccan Tweets Using Ensemble Learning. In *International Conference on Big Data, Cloud and Applications*, pages 91–104. Springer.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-learn: Machine learning in Python. *Journal of machine learning research*, 12(Oct):2825–2830.

Chatrine Qwaider, Motaz Saad, Stergios Chatzikyriakidis, and Simon Dobnik. 2018. Shami: A corpus of levantine arabic dialects. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*.

Kumar Ravi and Vadlamani Ravi. 2015. A survey on opinion mining and sentiment analysis: tasks, approaches and applications. *Knowledge-Based Systems*, 89:14–46.

Rizkallah, Sandra and Atiya, Amir and ElDin Mahgoub, Hossam and Heragy, Momen", editor="Hassanien, Aboul Ella and Tolba, Mohamed F. and Elhoseny, Mohamed and Mostafa, Mohamed . 2018. Dialect Versus MSA Sentiment Analysis. In *The International Conference on Advanced Machine Learning Technologies and Applications (AMLTA2018)*, pages 605–613, Cham. Springer International Publishing.

Lina Maria Rojas-Barahona. 2016. Deep learning for sentiment analysis. *Language and Linguistics Compass*, 10(12):701–719.

Hiroshi Shimodaira. 2014. Text classification using naive bayes. *Learning and Data Note*, 7:1–9.

Abu Bakr Soliman, Kareem Eissa, and Samhaa R El-Beltagy. 2017. Aravec: A set of Arabic word embedding models for use in Arabic nlp. *Procedia Computer Science*, 117:256–265.

Duyu Tang, Bing Qin, and Ting Liu. 2015. Deep learning for sentiment analysis: successful approaches and future challenges. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 5(6):292–303.

Kees Versteegh. 2014. *The Arabic language*. Edinburgh University Press.

Shuo Xu, Yan Li, and Zheng Wang. 2017. Bayesian multinomial Naïve Bayes classifier to text classification. In *Advanced multimedia and ubiquitous engineering*, pages 347–352. Springer.

Mohab Youssef and Samhaa R El-Beltagy. 2018. MoArLex: An Arabic Sentiment Lexicon Built Through Automatic Lexicon Expansion. *Procedia computer science*, 142:94–103.

Lei Zhang, Shuai Wang, and Bing Liu. 2018. Deep learning for sentiment analysis: A survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 8(4):e1253.

Classifying Arabic dialect text in the Social Media Arabic Dialect Corpus (SMADC)

Areej Alshutayri
College of Computer Science and Engineering
University of Jeddah
Jeddah, Saudi Arabia
aoalshutayri@uj.edu.sa

Eric Atwell
School of Computing
University of Leeds
Leeds, United Kingdom
e.s.atwell@leeds.ac.uk

Abstract

In recent years, research in Natural Language Processing (NLP) on Arabic has garnered significant attention. This includes research about classification of Arabic dialect texts, but due to the lack of Arabic dialect text corpora this research has not achieved a high accuracy. Arabic dialects text classification is becoming important due to the increasing use of Arabic dialect in social media, so this text is now considered quite appropriate as a medium of communication and as a source of a corpus. We collected tweets, comments from Facebook and online newspapers representing five groups of Arabic dialects: Gulf, Iraqi, Egyptian, Levantine, and North African. This paper investigates how to classify Arabic dialects in text by extracting lexicons for each dialect which show the distinctive vocabulary differences between dialects. We describe the lexicon-based methods used to classify Arabic dialect texts and present the results, in addition to techniques used to improve accuracy.

1 Introduction

Textual Language Identification or Dialect Identification is the task of identifying the language or dialect of a written text. The Arabic language is one of the world's major languages, and it is considered the fifth most-spoken language and one of the oldest languages in the world. Additionally, the Arabic language consists of multiple variants, both formal and informal (Habash, 2010). Modern Standard Arabic (MSA) is a common standard written form used worldwide. MSA is derived from Classical Arabic which is based on the text of the Quran, the holy book of Islam; MSA is the primary form of the Arabic language that is spoken and studied today. MSA is taught in Arab schools, and promoted by Arab civil as well as religious authorities and governments. There are many dialects spoken around the Arab World; Arabic dialectologists have studied hundreds of local variations, but generally agree these cluster into five main regional dialects: Iraqi Dialect (IRQ), Levantine Dialect (LEV), Egyptian Dialect (EGY), North African Dialect (NOR), and Gulf Dialect (GLF). Arabic dialectologists have traditionally focused mainly on variation in phonetics or pronunciation of spoken Arabic; but Arabic dialect text classification is becoming important due to the increasing use of Arabic dialect in social media text. As a result, there is a need to know the dialect used by Arabic writers to communicate with each other; and to identify the dialect before machine translation takes place, in order to ensure spell checkers work, or to accurately search and retrieve data. Furthermore, identifying the dialect may improve the Part-Of-Speech tagging: for example, the MADAMIRA toolkit identifies the dialect (MSA or EGY) prior to the POS tagging (Pasha et al., 2014). The task of Sentiment Analysis of texts, classifying the text as positive or negative sentiment, is also dialect-specific, as some diagnostic words (especially negation) differ from one dialect to another. Text classification is identifying a predefined class or category for a written document by exploring its characteristics or features (Ikonomakis et al., 2005; Sababa and Stassopoulou, 2018). However, Arabic dialect text classification still needs a lot of research to increase the accuracy of classification due to the same characters being used to write MSA text and dialects, and also because there is no standard written format for Arabic dialects. This paper sought to find appropriate lexical features to classify Arabic dialects and build a more sophisticated filter to extract features from Arabic-character written dialect text files. In this paper, the corpus was annotated with dialect labels and used in automatic dialect lexicon-extraction and text-classification experiments.

2 Related Work

There are many studies that aim to classify Arabic dialects in both text and speech; most spoken Arabic dialect research focuses on phonological variation and acoustic features, based on audio recordings and listening to dialect speakers. In this research, the classification of Arabic dialects will focus on text, One example project focused on Algerian dialect identification using unsupervised learning based on a lexicon (Guellil and Azouaou, 2016). To classify Algerian dialect the authors used three types of identification: total, partial and improved Levenshtein distance. The total identification meant the term was present in the lexicon. The partial identification meant the term was partially present in the lexicon. The improved Levenshtein applied when the term was present in the lexicon but with different written form. They applied their method on 100 comments collected from the Facebook page of Djezzy and achieved an accuracy of 60%. A lexicon-based method was used in (Adouane and Dobnik, 2017) to identify the language of each word in Algerian Arabic text written in social media. The research classified words into six languages: Algerian Arabic (ALG), Modern Standard Arabic (MSA), French (FRC), Berber (BER), English (ENG) and Borrowings (BOR). The lexicon list contains only one occurrence for each word and all ambiguous words which can appear in more than one language are deleted from the list. The model was evaluated using 578 documents and the overall accuracy achieved using the lexicon method is 82%. Another approach to classify Arabic dialect is using text mining techniques (Al-Walaie and Khan, 2017). The text used in the classification was collected from Twitter. The authors used 2000 tweets and the classification was done on six Arabic dialects: Egyptian, Gulf, Shami, Iraqi, Moroccan and Sudanese. To classify text, decision tree, Naïve Bayes, and rule-based Ripper classification algorithms were used to train the model with keywords as features for distinguishing one dialect from another, and to test the model the used 10-fold cross-validation. The best accuracy scored 71.18% using rule-based (Ripper) classifier, 71.09% using Naïve Bayes, and 57.43% using decision tree. Other researchers on Arabic dialect classification have used corpora limited to a subset of dialects; our SMADC corpus is an International corpus of Arabic with a balanced coverage of all five major Arabic dialect classes.

3 Data

The dataset used in this paper is the Social Media Arabic Dialect Corpus (SMADC) which was collected using Twitter, Facebook and comments from online newspapers described in (Alshutayri and Atwell, 2017, 2018b,c). We plan to make the Social Media Arabic Dialect Corpus (SMADC) available to other researchers for non commercial uses, in two formats (raw and cleaned) and with a range of metadata. This corpus covers all five major Arabic dialects recognised in the Arabic dialectology literature: EGY, GLF, LEV, IRQ, and NOR. Therefore, five dictionaries were created to cover EGY dialect, GLF dialect, LEV dialect, IRQ dialect, and NOR dialect. (Alshutayri and Atwell, 2018a) presented the annotation system or tool which was used to label every document with the correct dialect tag. The data used in the lexicon based method was the result of the annotation, and each comment/tweet is labelled either dialectal document or MSA document.

The MSA documents in our labelled corpus were used to create an MSA word list, then we added to this list MSA stop words collected from Arabic web pages by Zerrouki and Amara (2009), and the MSA word list collected from Sketch Engine (Kilgarriff et al., 2014), in addition to the list of MSA seed words for MSA web-as-corpus harvesting, produced by translating an English list of seed words (Sharoff, 2006). The final MSA word list contains 29674 words. This word list is called "StopWords1" and was used in deleting all MSA words from dialect documents, as these may contain some MSA words, for example due to code switching between MSA and dialect.

The dialectal documents consist of documents and dialectal terms, where the annotators (players) were asked to write the dialectal terms in each document which help them to identify dialect as described in (Alshutayri and Atwell, 2018a). The dialectal documents were divided into two sets: 80% of the documents were used to create dialectal dictionaries for each dialect, and 20%, the rest of the documents, were used to test the system. To evaluate the performance of the lexicon based models, a subset of 1633 documents was randomly selected from the annotated dataset and divided into two sets; the training dataset which contains 1383 documents (18,697 tokens) are used to create the dictionaries, and the evaluation dataset which contains 250 documents (7,341 tokens). The evaluation dataset did not include any document used to create the lexicons as described previously.

4 Lexicon Based Methods

To classify the Arabic dialect text using the Lexicons, we used a range of different classification metrics and conducted five experiments, all of which used a dictionary for each dialect. The following sections show the different methods used and describe the difference between the conducted experiments, and the result of each experiment.

4.1 Dialectal Terms Method

In this method, the classification process starts at the word level to identify and label the dialect of each word, then the word-labels are combined to identify the dialect of the document. The dialectal terms produced from the annotation tool were used as a dictionary for each dialect. The proposed system consists of five dictionaries, one for each dialect: EGY dictionary contains 451 words, GLF dictionary contains 392 words, IRQ dictionary contains 370 words, LEV dictionary contains 312 words from LEV, and NOR dictionary contains 352 words.

According to the architecture in Figure 1, to classify each document as being a specific dialect, the system follows four steps:

- Detect the MSA words in the document by comparing each word with the MSA words list, then delete all MSA words found in the document.
- The result from the first step is a document containing only dialectal words.
- Detect the dialect for each word in the document by comparing each word with the words in the dictionaries created for each dialect.
- Identify dialect.

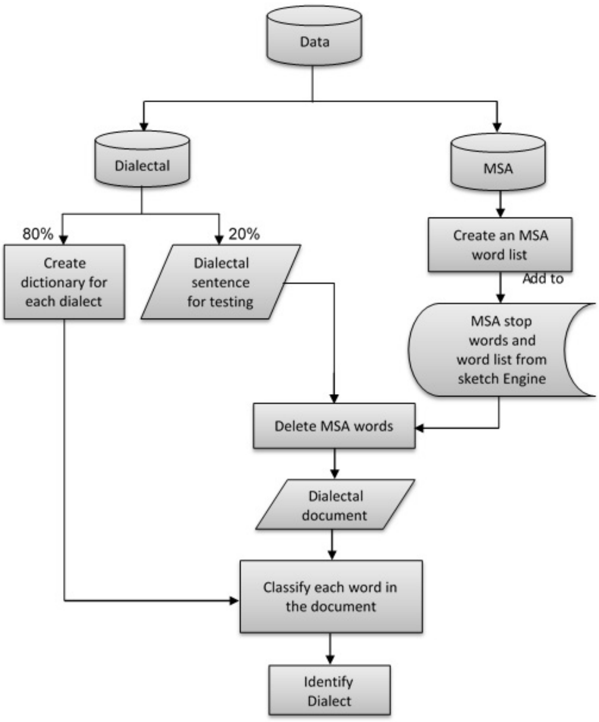
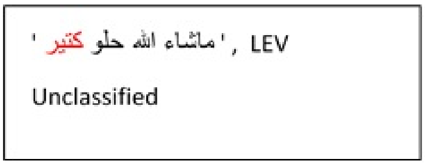

Data
Dialectal
MSA
80%
20%
Create dictionary for each dialect
Dialectal sentence for testing
Create an MSA word list
Add to
MSA stop words and word list from sketch Engine
Delete MSA words
Dialectal document
Classify each word in the document
Identify Dialect

Figure 1: The architecture of classification process using lexicon based.

Using this method based on the dialectal terms written by the annotators produces some unclassified documents due to words that occur in more than one dialect. For example, the document in Figure 2 was labelled as LEV and the structure of the document is also LEV dialect, but the word (كتير) (kti:r) which appears in the text is also used in EGY. Therefore, when classifying each word in the document the model found the word (كتير) (kti:r) in EGY dictionary and also in LEV dictionary, so the model was not able to classify this document as the other words are MSA words or shared dialectal words. Unclassified documents indicate that using this dialectal terms method is not effective in dealing with ambiguous words.


' ماشاء الله حلو كتير ', LEV

Unclassified

Figure 2: Example of unclassified document.

Table 1 shows the accuracies achieved by applying the dialectal terms method on the testing set. The first column represents using MSA words list, and the second column represents the achieved accuracies based on using SMADC to create dictionaries. The best accuracy is 56.91 with 140 documents correctly classified using StopWords1. Based on this method, 110 documents were unclassified to a specific dialect because they contain some ambiguous terms which are used in more than one dialect, as in the example of Figure 2. As a solution to this problem, a voting method is used and another way is using a frequent term method.

MSA	**SMADC**
StopWords1	**56.91**%
Without delete MSA Words	55.60%

Table 1: Results of dialectal terms method using the dictionaries created from SMADC.

4.2 Voting Methods

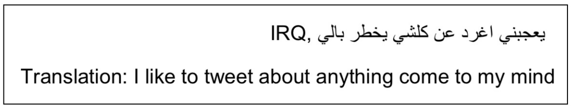
Another method to classify Arabic dialect text is to treat the text classification of Arabic dialects as a logical constraint satisfaction problem. The voting method is an extension of the dialectal term method presented previously. The classification starts at the word level based on the dictionaries created from the 80% training set of documents described in Section 3. So, the annotated training set of documents was used instead of the dialectal terms list. In this method, we looked to the whole document and count how many words belong to each dialect. Each document in the voting method was represented by a matrix C. The size of the matrix is $C_{|n| \times |5|}$, where n is the number of words in each document, and 5 is the number of dialects (EGY, NOR, GLF, LEV, and IRQ).

4.2.1 Simple Voting Method

In this method, the document is split into words and the existence of each word in the dictionary is represented by 1 as in Equation 1.

$$c_{ij} = \begin{cases} 1, & \text{if} \quad word \quad \epsilon \; dialect \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

The following illustrates the method. We apply Equation 1 on the following document **A** labelled as IRQ dialect as in Figure 3:



يعجبني اغرد عن كلشي يخطر بالي ,IRQ

Translation: I like to tweet about anything come to my mind

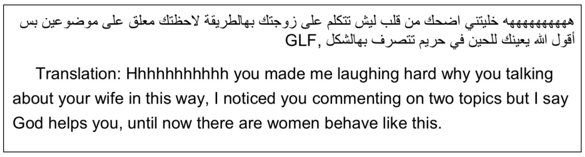
Figure 3: The text in document **A**.

Words	**NOR**	**EGY**	**IRQ**	**LEV**	**GLF**
يعجبني	0	0	1	0	0
اغرد	0	0	1	0	0
عن	1	1	1	1	1
كلشي	1	0	1	0	0
يخطر	0	0	0	0	0
بَالي	0	1	0	0	0
Total	2	2	**4**	1	1

Table 2: The matrix representation of document **A** with simple voting.

The result of classification is IRQ according to Table 2; the total shows that four words in this document belong to IRQ dialect in comparison with two words belong to NOR and EGY, and one word belong to LEV and GLF.

The proposed model identifies the document correctly but sometimes this model cannot classify a document and the result is unclassified when more than one dialect gets the same count of words (total), like document **B** labelled as GLF dialect as shown in Figure 4:



هههههههههه خليتني اضحك من قلب ليش تتكلم على زوجتك بهالطريقة لاحظتك معلق على موضوعين بس
أقول الله يعينك للحين في حريم تتصرف بهالشكل ,GLF

Translation: Hhhhhhhhhh you made me laughing hard why you talking about your wife in this way, I noticed you commenting on two topics but I say God helps you, until now there are women behave like this.

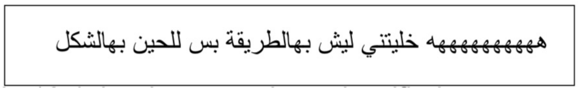
Figure 4: The text in document **B**

Using the StopWords1 to delete MSA words from the document, the result is the following dialectal document containing only dialectal words as in Figure 5.

According to the result in Table 3 the document is unclassified because more than one dialect has the same total number of words.

4.2.2 Weighted Voting Method

This method is used to solve the problem of unclassified documents in simple voting method. To solve this problem, we proposed to change the value of the word from 1 to the probability of the word to belong to this dialect as a fraction of one divided by the number of dialects the word is found in their dictionaries as in Equation 2. If a word can belong to more than one dialect, its vote



هههههههههه خليتني ليش بهالطريقة بس للحين بهالشكل

Figure 5: Example of unclassified document.

Words	NOR	EGY	IRQ	LEV	GLF
هّهّهّهّه	0	1	1	1	0
خليتني	0	0	0	0	0
ليش	1	0	1	1	1
بهَالطريقة	0	0	0	0	0
بس	1	1	1	1	1
للّحين	0	0	0	0	1
بهَالشكل	0	0	0	0	0
Total	2	2	3	3	3

Table 3: The matrix representation of document **B** with simple voting.

is shared between the dialects.

$$c_{ij} = \begin{cases} \frac{1}{m}, & \text{if} \quad word \quad \epsilon \, dialect \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

$\frac{1}{m}$ is the probability of the word belonging to the specific dialect, where m the number of dialects which the word belongs to. By applying the new method on the unclassified document, the document is classified correctly as GLF dialect, according to Table 4.

Words	NOR	EGY	IRQ	LEV	GLF
هّهّهّهّه	0	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{3}$	0
خليتني	0	0	0	0	0
ليش	$\frac{1}{4}$	0	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$
بهَالطريقة	0	0	0	0	0
بس	$\frac{1}{5}$	$\frac{1}{5}$	$\frac{1}{5}$	$\frac{1}{5}$	$\frac{1}{5}$
للّحين	0	0	0	0	1
بهَالشكل	0	0	0	0	0
Total	0.45	0.5333	0.7833	0.7833	**1.45**

Table 4: The matrix representation of document **B** with Weighted voting.

4.2.3 Results of Voting Method

Voting method is focused on the existence of the word in the dictionary, so, the frequency of the word is ignored, unlike the frequent term method which described in Section 4.3. The highest accuracy achieved is 74.0% without deleting MSA words from the classified document. Moreover, using the value of one to express the existence of the word in the dictionary showed low accuracy due to the similarity between the sum of ones for each dialect, as described in Section 4.2.1. Table 5 shows the different accuracies achieved using SMADC. The first column in Table 5 shows using of MSA stop words. The second and the third columns represent the methods used to classify documents. The cells inside the second and third columns present the achieved accuracies using these methods. The voting method scored 74% using the weighted voting method and SMADC to create dictionaries. After cleaning the MSA word list, the accuracy increased to 77.60%.

MSA	Simple Vote	Weighted Vote
StopWords1	69.19%	**72.0%**
Without delete MSA Words	65.60%	**74.0%**

Table 5: Results of Voting methods using the dictionaries created from SMADC.

4.3 Frequent Terms Methods

Another method is presented in this section to solve the problem shown in the dialectal terms method described in Section 4.1 and to improve the accuracy of classification achieved using the voting method. In frequent terms method, new dictionaries with word frequencies were created from the 80% training set of documents. The documents were classified into the five dialects. Then, for each dialect a .txt file was created to contain one word per line with the word's frequency based on the number of times the word appeared in the documents. The frequency for each word showed if the word is frequent in this dialect or not, which helps to improve the accuracy of the classification process. In comparison to the first method, the third step in Figure 1 was used to detect the dialect for each word in the document by comparing each word with the words in the dictionaries created for each dialect. If the word is in the dictionary, then calculate the weight (W) for each word by dividing the word's frequency (F) value by the Length of the dictionary (L) which equals the total number of words in the word's dialect dictionary, using the following equation:

$$W(word, dict) = \frac{F(word)}{L(dict)} \tag{3}$$

For each document, five vectors were created, one per dialect, to store the weight for each word in the document; so the length of each vector is equal to the length of the document. By applying the Equation 3 on ”كتير”, we found the weight of the word ”كتير” in LEV dialect is bigger than the weight of it in EGY dialect, as shown in the following equations.

$$W(\text{”كتير”}, EGY) = \frac{F(\text{”كتير”})}{L(EGY)} = \frac{3}{2032} = 0.00147$$

$$W(\text{”كتير”}, LEV) = \frac{F(\text{”كتير”})}{L(LEV)} = \frac{8}{2028} = 0.00394$$

Two experiments were done after calculating the weight for each word. The first experiment was based on summing the weights and calculating the average. The second experiment was based on multiplying the weights together.

4.3.1 Weight Average Method (WAM)

This method based on calculating the average of the word weights for each document. Table 6 shows the values of the weight for each word in the document after deleting MSA words. Five vectors were created to represent five dialects and each cell contains the weight for each word in the document. The model calculated the average for each dialect by taking the summation of the weight (W) values for each vector then dividing the summation of weights by the length (L) of the document after deleting the MSA words, as in the following equation:

$$Avg_{dialect} = \frac{\sum W_{dialect}}{L(document)} \tag{4}$$

Words	NOR	LEV	IRQ	GLF	EGY
مَاشَاء	0	0.00049309	0	0.00026143	0
حلو	0	0.00295857	0.00053304	0.00026143	0.00049212
كتير	0	0.00394477	0	0	0.00147637

Table 6: Results of WAM using the dictionaries created from SMADC.

By calculating the average for the dialect vectors using the Equation 4, the model classified the document as LEV dialect, after comparing the results of the average obtained from the following equations.

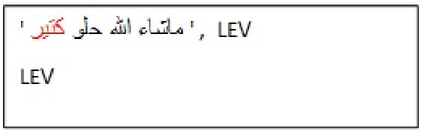
$$Avg_{EGY} = \frac{\sum W_{EGY}}{L(document)} = \frac{0.00196849}{3} = 0.00065616$$

$$Avg_{LEV} = \frac{\sum W_{LEV}}{L(document)} = \frac{0.00739643}{3} = 0.00246547$$

$$Avg_{GLF} = \frac{\sum W_{GLF}}{L(document)} = \frac{0.00052286}{3} = 0.00017428$$

$$Avg_{IRQ} = \frac{\sum W_{IRQ}}{L(document)} = \frac{0.00053304}{3} = 0.00017768$$

By applying the proposed model on the same unclassified example in Figure 2, we found that the model classified the document correctly as in Figure 6.



'ماشاء الله حلو كتير', LEV

LEV

Figure 6: Example of correctly classified document.

4.3.2 Weight Multiplied Method (WMM)

The WAM model is based on summing the word weights and calculating the average. According to probability theory, probabilities are generally combined by multiplication. So, for an alternative model, the Weight Multiplied Method (WMM), we multiplied the word weights for each document to compute the accuracy of classification in comparison to the average method used in the previous section.

$$P(doc|c) = \prod W(word, dict) \tag{5}$$

We applied Equation 5 on the weights in Table 6. There is a problem with combining weights by multiplication: if any of the weights to be combined is zero, the combined weight will be zero. So, we change the value of not found words in the dialect dictionary from zero to one. However, in the Table 6 if the values in NOR vector changed to one this will affect the result of multiplication. For that reason the result of multiplication was checked as to whether or not it equal one then we changed the result to zero.

According to Equation 5 the document is classified as IRQ dialect, which is a wrong prediction.

$$P_{EGY} = \prod W(word|EGY) = 1 \times 0.00049212 \times 0.00147637 = 0.00000072$$

$$P_{LEV} = \prod W(word|LEV) = 0.00049309 \times 0.00295857 \times 0.00394477 = 0.0000000057$$

$$P_{GLF} = \prod W(word|GLF) = 0.00026143 \times 0.00026143 \times 1 = 0.000000068$$

$$P_{IRQ} = \prod W(word|IRQ) = 1 \times 0.00053304 \times 1 = 0.00053304$$

To solve wrong predictions which result from using WMM and to improve the classification accuracy, we replace one when the word is not in the dictionary with one divided by the number of words in each dictionary to not affect the result of multiplication. By applying the new value to Equation 5 the document is correctly classified as LEV dialect. Table 7 shows the improved accuracy resulted using WMM when using one divided by the number of words in each dictionary to represent the absence of a word in the dictionary.

$$P_{EGY} = \prod W(word|EGY) = \frac{1}{L(dic_{EGY})} \times 0.00049212 \times 0.00147637 = \frac{1}{2032} \times 0.00049212 \times 0.00147637 = 0.0000000003575$$

$$P_{LEV} = \prod W(word|LEV) = 0.00049309 \times 0.00295857 \times 0.00394477 = 0.0000000057$$

$$P_{GLF} = \prod W(word|GLF) = 0.00026143 \times 0.00026143 \times \frac{1}{L(dic_{GLF})} = 0.00026143 \times 0.00026143 \times \frac{1}{3472} = 0.000000068$$

$$P_{IRQ} = \prod W(word|IRQ) = \frac{1}{L(dic_{IRQ})} \times 0.00053304 \times \frac{1}{L(dic_{IRQ})} = \frac{1}{1889} \times 0.00053304 \times \frac{1}{1889} = 0.0000000000149$$

$$P_{NOR} = \prod W(word|NOR) = \frac{1}{L(dic_{NOR})} \times \frac{1}{L(dic_{NOR})} \times \frac{1}{L(dic_{NOR})} = \frac{1}{1436} \times \frac{1}{1436} \times \frac{1}{1436} = 0.0000000003376$$

The following sections will compare the first model based on summation and calculate average with the multiplication method, and show the achieved results using average method and the multiplication method.

4.3.3 Result of Frequent Terms Method

The frequent term method which is based on using word frequencies gave good results in showing whether the words in the tested document is used in the specific dialect. The model was tested using the test dataset described in Section 3. Based on the average method, the model achieved 88% accuracy using the MSA StopWords1 list. However, using the multiply method achieves low accuracy due to replacing zero with one when the word does not exist in the dictionary; so instead we divided by the number of words in the dictionary..

Table 7 reports the different accuracies achieved when using SMADC based on using one divided by the number of words in the dictionary to represent words which are not found in the dictionary. The frequent terms method scored 88% using the weight average metric when dictionaries were created using SMADC. The accuracy improved to 90% after cleaning the MSA word list from some dialectal words as a result of mislabelling process.

MSA	WMM	WAM
StopWords1	55.60%	**88.0%**
Without delete MSA Words	43.0%	**64.0%**

Table 7: Results of frequent term methods using the dictionaries created from SMADC.

By comparing the Weight Average Method (WAM) model based on summation and calculating average with the Weight Multiplied Method (WMM), we found that the WAM achieved a higher accuracy than the WMM multiplication method.

5 Conclusion

The classification of Arabic dialect text is a hot topic attracting a number of studies over the last ten years (Sadat et al., 2014; Zaidan and Callison-Burch, 2014; Elfardy and Diab, 2013; Mubarak and Darwish, 2014; Harrat et al., 2014; Shoufan and Alameri, 2015). In this paper, we classified Arabic dialects text using a lexicon based method, and explored different metrics for scoring dialect words from lexicons: weight average method, weight multiplied method, simple voting method and weighted voting method. The lexicons were dictionaries created for each dialect from our Arabic dialect corpora. The classification process was based on deleting all MSA words from the document then checking each word in the document by searching the dialect dictionaries. The voting method scored 74% using the weighted voting method and SMADC to create dictionaries. After cleaning the MSA word list, the accuracy increased to 77.60%. The frequent terms method scored 88% using the weight average metric when dictionaries were created using SMADC. The accuracy improved to 90% after cleaning the MSA word list from some dialectal words as a result of mislabelling process. These scores compare favourably against other Arabic dialect classification research on subsets of Arabic dialects.

References

Wafia Adouane and Simon Dobnik. 2017. Identification of languages in Algerian Arabic multilingual documents. In *Proceedings of the Third Arabic Natural Language Processing Workshop*, pages 1–8, Valencia, Spain. Association for Computational Linguistics.

Mona Al-Walaie and Muhammad Khan. 2017. Arabic dialects classification using text mining techniques. In *2017 International Conference on Computer and Applications (ICCA)*, pages 325–329.

Areej Alshutayri and Eric Atwell. 2017. Exploring twitter as a source of an arabic dialect corpus. *International Journal of Computational Linguistics (IJCL)*, 8:37–44.

Areej Alshutayri and Eric Atwell. 2018a. Arabic dialects annotation using an online game. In *2nd International Conference on Natural Language and Speech Processing (ICNLSP)*, pages 1–5.

Areej Alshutayri and Eric Atwell. 2018b. Creating an arabic dialect text corpus by exploring twitter, facebook, and online newspapers. In *Proceedings of Open-Source Arabic Corpora and Processing Tools. OSACT'2018 Open-Source*, Miyazaki, Japan.

Areej Alshutayri and Eric Atwell. 2018c. *A Social Media Corpus of Arabic Dialect Text*. Computer-Mediated Communication and Social Media Corpora, Clermont-Ferrand: Presses universitaires Blaise Pascal.

Heba Elfardy and Mona Diab. 2013. Sentence level dialect identification in Arabic. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, volume 2, pages 456–461, Sofia, Bulgaria. Association for Computational Linguistics.

Imène Guellil and Faiçal Azouaou. 2016. Arabic dialect identification with an unsupervised learning (based on a lexicon). application case: Algerian dialect. In *IEEE Intl Conference on Computational Science and Engineering (CSE) and IEEE Intl Conference on Embedded and Ubiquitous Computing (EUC) and 15th Intl Symposium on Distributed Computing and Applications for Business Engineering (DCABES)*, pages 724–731.

Nizar Y. Habash. 2010. *Introduction to Arabic Natural Language Processing*. Morgan and Claypool.

Salima Harrat, Karima Meftouh, Mourad Abbas, and Kamel Smaïli. 2014. Building resources for algerian arabic dialects. In *15th Annual Conference of the International Communication Association (Interspeech)*, pages 2123–2127.

Emmanouil Ikonomakis, Sotiris Kotsiantis, and V Tampakas. 2005. Text classification using machine learning techniques. *WSEAS transactions on computers*, 4:966–974.

Adam Kilgarriff, Vít Baisa, Jan Bušta, Miloš Jakubíček, Vojtěch Kovář, Jan Michelfeit, Pavel Rychlý, and Vít Suchomel. 2014. The sketch engine: ten years on. *Lexicography*, 1(1):7–36.

Hamdy Mubarak and Kareem Darwish. 2014. Using twitter to collect a multi-dialectal corpus of Arabic. In *Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP)*, pages 1–7, Doha, Qatar. Association for Computational Linguistics.

Arfath Pasha, Mohamed Al-Badrashiny, Mona Diab, Ahmed El Kholy, Ramy Eskander, Nizar Habash, Manoj Pooleery, Owen Rambow, and Ryan Roth. 2014. MADAMIRA: A fast, comprehensive tool for morphological analysis and disambiguation of Arabic. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 1094–1101, Reykjavik, Iceland. European Language Resources Association (ELRA).

Hanna Sababa and Athena Stassopoulou. 2018. A classifier to distinguish between cypriot greek and standard modern greek. pages 251–255.

Fatiha Sadat, Farnzeh Kazemi, and Atefeh Farzindar. 2014. Automatic identification of Arabic language varieties and dialects in social media. In *Proceedings of the Second Workshop on Natural Language Processing for Social Media (SocialNLP)*, pages 22–27, Dublin, Ireland. Association for Computational Linguistics and Dublin City University.

Serge Sharoff. 2006. Open-source corpora: Using the net to fish for linguistic data. *International Journal of Corpus Linguistics*, 11:435–462.

Abdulhadi Shoufan and Sumaya Alameri. 2015. Natural language processing for dialectical arabic: A survey. In *Proceedings of the Second Workshop on*

Arabic Natural Language Processing, pages 36–48, Beijing, China. Association for Computational Linguistics.

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

Verbs in Egyptian Arabic: a case for register variation

Michael Grant White
Department of Linguistics, Brigham Young University
Provo, Utah, USA 84602
mgrantwhite@gmail.com

Deryle W. Lonsdale
Department of Linguistics, Brigham Young University
Provo, Utah, USA 84602
lonz@byu.edu

Abstract

The limited availability of Egyptian Arabic (EA) corpus resources, especially speech corpora, has left open opportunity for research into such dialect phenomena as register. In this paper we introduce a new two-million-word EA corpus, CALM. We perform a register analysis on EA between two subcorpora of CALM (i.e. Movies and Blogs), showing several features that vary between the two. A discussion follows about how annotation was carried out automatically, how it was hand-corrected, and what the prospects are for carrying out similar studies using CALM.

1 Introduction

The advent of the internet has made written Egyptian Arabic much more accessible than in the past. Traditional sources of written language like books, newspapers, academic journals, and government documents are composed in Modern Standard Arabic which differs morphologically, lexically, and syntactically from Egyptian Arabic. However, as access to the internet spreads, so does the appearance of written Egyptian Arabic on blogs and social media sites. Collections of digital texts provide linguists with a new opportunity to collect large samples of the written dialect from a variety of sources on numerous topics.

One active area of corpus linguists involves the identification and characterization of registers (Gries, 2006), a type of language use that is determined by the situation or circumstance in which the speech act occurs (Johnstone, 2008). Situations that cause speakers to change their lexical and grammatical choices are said to belong to different registers. Some overlap exists between the notions of register and genre (Biber and Conrad, 2001), but a fine-grained distinction between the two is not necessary for this discussion.

Modern studies of register variation rely on annotated corpora. Unfortunately, an annotated corpus containing both written and spoken Egyptian Arabic is not available preventing studies of Egyptian Arabic register variation.

To help explore this problem, this paper introduces a new two-million word corpus of Egyptian Arabic. We perform a preliminary analysis of variation between two registers found in the corpus, on the basis of partial annotation, namely that of verbs. We compare verb frequency, lexical diversity, and other phenomena between two subcorpora and confirm a quantitative difference between two putative registers: Movies and Blogs. We show that, although the examination of a single part of speech cannot capture the true extent of the variance of two registers, it provides a platform from which to launch an in-depth analysis by answering the preliminary questions concerning register analysis for Egyptian Arabic.

We also offer comments on the use of automatic tools for annotation, and mention various types of post-processing that help improve the annotations for corpus analyses like the one discussed here.

2 Background

A corpus is a collection of texts or transcriptions gathered for the purpose of conducting an empirical study of language (Hunston, 2002; Kübler and Zinsmeister, 2015), and they have become a valuable resource in nearly every field of linguistics (Teubert, 2005). Corpora assume many different forms and sizes in accordance with their intended purpose. Varying amounts of corpus content are available across different languages, dialects, and text types.

In this paper we focus on Arabic corpus linguistics and associated annotation and analysis. Several types of corpora are available for Arabic: examples include ones for general language (ArabiCorpus[1]), transcribed speech (CALLHOME Egyptian Arabic and MGB-3) (Canavan et al., 1997; Ali et al., 2017), specialized (The Quranic Arabic Corpus[2]), parallel (OPUS[3]), and learner corpora (Arabic Learner Corpus) (Alfaifi and Atwell, 2015). With these corpora and others, our understanding of Arabic and how it is used has increased (Buckwalter and Parkinson, 2011; Bentley, 2015; Ismail, 2015; Alasmari et al., 2017; Dickins, 2017; Henen, 2018). One area that has been largely overlooked in Arabic corpus studies is discourse analysis, especially for learner Arabic (Ryding, 2006) and for different dialects.

2.1 Register

Research on registers in a corpus targets the identification of features that distinguish one register from another. Biber (1993) gives eight parameters to use in classifying registers: the primary channel of delivery, format, setting, addressee, addressor, factuality, purpose, and topic. Separating texts according to these parameters helps establish appropriate registers.

Once texts are collected in a principled way with the aim of representativeness, feature-based register analysis can begin. Each register has characteristics or dimensions of language associated with it (Biber, 1993) based on pertinent lexical, grammatical, and syntactic features. For example, the narrative dimension in English is characterized by past tense verbs, third person pronouns, public verbs, synthetic negation, and present participle clauses. By comparing the frequency of the features in different types of texts, the dimensions in which they fall can be determined. The dimensions for each text type are then taken as characteristic of the register to which the texts belong.

In this paper we deal with two primary registers: the oral and the literate. In daily life, the most common registers in the oral dimension come from spontaneous speech. However, this type of data is currently costly to collect and transcribe. Often corpora contain scripted speech from television and movies to represent oral language. Spontaneity suffers somewhat: since scripted speech is first written, it affords the author time to craft each utterance and edit it until it achieves the desired effect. Utterances made spontaneously do not often reflect this luxury.

This has led to debate within the corpus community, with some asserting that scripted language reflects artificial settings created by the same author, and hence may not be completely realistic (Sinclair, 2004). On the other hand, others have found only minimal differences between movie language and spontaneous speech (Taylor, 2004; Brysbaert and New, 2009; Forchini, 2012). This issue becomes particularly interesting in Egyptian Arabic.

Written registers are very frequently used in corpus studies. Sample text types that fall within the written registers include literature, newspaper articles, academic articles, encyclopedia entries, personal correspondence, and official documents (Biber and Conrad, 2001; Biber et al., 2006). With its widespread availability of texts, the internet is also a source for written register data. Biber et al. (2015) found that English texts from the internet can be categorized into several registers: narrative, information description/explanation, opinion, interactive discussion, how-to/instructional, informational persuasion, lyrical, spoken, and hybrid.

2.2 Arabic corpus analysis

The oral/written distinction and corpus registers in general is more complex and nuanced for Arabic's diglossic situation: the standarized version of the language, MSA, is used in many written situations, whereas a wide array of dialects is used for everyday spoken language. Collecting, annotating and analyzing corpus data is hence more complicated and much remains to be done in examining the variation that exists between spoken and written Arabic. Examples of such work include Fakhri's (2009) investigation of the variation between academic Arabic in the disciplines of the humanities and the law. Johnstone (2008) examined Arabic expository prose and identified the use of three features—repetition, parataxis, and formulaicity—which are typically associated with spoken language.

Several Egyptian Arabic film and television transcript corpora have been used in recent stud-

[1] See `http://arabiCorpus.byu.edu`.

[2] See `http://corpus.quran.com`.

[3] See `http://opus.nlpl.eu`.

ies. Hussein (2016) used a corpus of Egyptian movie transcripts to study the pragmatic and syntactic functions of the Egyptian word[4] كده kɪdʌ. This corpus contains 231,542 words from seventeen different films. Production dates for these movies range from 1958-2008 with the majority of words in the corpus coming from movies made pre-1990, which makes the content somewhat dated for contrasting it with recent content such as internet texts.

Such a corpus was used by Sayed (2018) to study the use of the discourse marker معلش maʕleʃ. This corpus contains transcripts of 76 episodes from the 2017-2018 Egyptian television serial سابع جار sæ:biʕ ga:r. One potential weakness to using this corpus in a register study is that all of the transcripts come from one television show. Most of the content is produced by only a handful of speakers/characters, calling into question representativeness.

The issue of representativeness is also important in choosing a suitable blog corpus. Two general Egyptian Arabic blog corpora are the Arabic Multi-Dialect Text Corpus (Almeman and Lee, 2013), which contains thirteen million words and Yet Another Dialectal Corpus (YADC) (Al-Sabbagh and Girju, 2012b) which contains six million. Both were created by performing web searches using dialect-specific words and then scraping the text from the webpages returned by the search engine. The Arabic Multi-Dialect Text Corpus used 139 different words determined to be unique to Egyptian Arabic as the search terms or seeds. The frequency of these words does not seem to have played a role in their choice.

In building a corpus with web searches, the frequency of the seeds is important for representativeness (Sharoff, 2006; Biber et al., 2015). No such frequency lists exist for Egyptian Arabic, and no documented effort was made by the creators of the Arabic Multi-Dialect Text Corpus to choose frequent words or phrases. The creator of YADC, on the other hand, took measures to create a more representative corpus of the texts available online. The queries contained multiple Egyptian exclusive function words. One downside to using function words is that many of them are found in several dialects (Qafisheh, 1992; Tamis and Persson, 2013). For our purposes, a corpus that contains only Egyptian Arabic is preferable.

3 Introducing CALM

Because of the need for a sizable corpus of Egyptian Arabic language, we collected and annotated a new corpus designed in an attempt to more accurately represent both oral and written language. This paper introduces CALM (**C**orpus **al**-**L**ogha al-**M**usriya, Corpus of Egyptian) a two-million-word corpus of Egyptian Arabic. CALM contains transcripts from 65 movies (comprising 655,858 word tokens), 88 scripted television programs (396,734 word tokens), and internet texts (1,092,442 word tokens). Some of the content has been annotated, as described in this paper, and annotation is ongoing. The corpus is available via download[5].

For the purposes of this paper, two subcorpora were extracted from CALM: a subcorpus of movie/television transcripts, and a subcorpus of internet texts.

3.1 The transcript subcorpus

The transcripts of CALM make up the largest known collection of transcribed Egyptian Arabic movies and TV programs produced in Egypt and written for Egyptian audiences. In other languages a quicker and cheaper method to build a comparable corpus would be from subtitles, but in Arabic foreign movies are subtitled using MSA. Only movies and programs popular to Egyptian audiences were selected for transcription based on the belief that they contain more mainstream language and are written by those who are able to skillfully mimic everyday speech.

Note that, as mentioned earlier, some debate exists about whether movie transcripts truly represent spontaneous speech (vs. the author's creative voice). A comparison of a script in CALM from the Egyptian film حسن ومرقص ɦʌsʌn wi murʔosˁ (Maati, 2008) with the movie transcripts reveals several instances where actors stray from the script, both omitting words from the script and adding their own content spontaneously.

Most movies and TV programs are from the year 2000 and later. No conscious effort was made to choose movies and TV based upon genre, or to balance the content across genres. However, care was

[4] When necessary in this paper, Arabic text is followed by an IPA transcription or by an English gloss.

[5] See http://linguistics.byu.edu/thesisdata/CALMcorpusDownload.html.

taken to make sure that one genre does not dominate the subcorpus created for movies and TV.

Once a movie was selected for inclusion in the corpus, it was transcribed and then reviewed for accuracy by a native Egyptian speaker. A second reviewer was used to determine the ability of the reviewers to catch all of the mistakes in the transcription. This process was necessary as some reviewers were not able to successfully read a transcript while listening to a movie.

3.2 The blog subcorpus

The other content in CALM was created from internet texts and will be called the blog subcorpus. Although internet texts can be classified into many different genres (Biber et al., 2015), in this paper they will be treated as a single register. We exclude internet texts that contain transcriptions of speeches, movies, television programs, and songs. Some of the blog texts were collected from the internet based upon seeded n-gram searches via Bing and Google, as discussed in the previous section, though this time relying on frequent dialect-specific words to decrease the chances of dialect mixing. We also used BootCat, a do-it-yourself web-to-corpus text conversion pipeline (Baroni and Bernardini, 2004), to find, scrape, and convert other webpages written in Egyptian Arabic into text files. A cursory review of the files was completed to remove non-EA texts that were returned by the process. However, some MSA is contained in CALM because it is interwoven throughout posts written in Egyptian; posts completely written in MSA, though, were removed.

EA exhibits numerous orthographic, lexical, morphological, and syntactic differences from MSA that will be familiar to many Arabists (El-Tonsi, 1982; Hassan, 2000; Ryding, 2005; Abdel-Massih et al., 2009). Even the representation of lemmas (base forms, or dictionary citation forms) and their orthography varies across EA dictionaries, necessitating a custom representation for CALM annotation. A discussion of these is beyond the scope of this paper, but an extensive list of the ones relevant to CALM corpus creation and annotation is available (White, 2019, forthcoming).

One area lacking in research is that of register variation within Egyptian Arabic, especially of the features that distinguish the spoken form from the written. Such a study could be undertaken with the use of two corpora said to represent different registers within the oral and literate dimensions of the language. To represent the oral dimension, film and television transcripts could be used because of the features that they share with spontaneous speech. The literate dimension could be represented by the language contained in blogs, since registers traditionally used to represent this dimension are written using MSA. Since it differs from Egyptian Arabic lexically, syntactically, and morphologically, comparing an MSA corpus and an Egyptian corpus would not increase our understanding of how Egyptians write the dialect. Therefore, the two corpora must be of EA.

Once these corpora have been decided upon, the next step is to determine the features that should be counted and compared. These features can be large or relatively few in number. In the next section, we perform a feature-based examination of two subcorpora from CALM which, we will show, represent two different registers of EA.

4 Case study: verb register variation

To assure tractability for this study, two subcorpora were created from CALM: (1) the Movies subcorpus, consisting of transcriptions from movies (113,163 word tokens) and television shows (115,236 word tokens), for a total of 228,399 word tokens including 38,768 verbs; and (2) the Blogs subcorpus, containing 141,318 word tokens and 27,616 verbs. While not exactly balanced, they are of reasonably comparable size.

For this study only the verbs were annotated, in part because they are slightly easier to identify and annotate than nouns and adjectives (Al-Sabbagh and Girju, 2012a), and because of their widespread use in determining register (Ferguson, 1983; Friginal, 2009; Staples, 2016). Table 1 gives sample annotations for several verbal features.

In this section, then, we perform a register variation analysis on verb features to characterize the two dimensions of content in CALM: oral versus literate (or spoken versus written), as represented by the Movies and Blogs subcorpora, respectively.

The first step was to annotate each verb in the two subcorpora. Once each verb was assigned a part-of-speech tag, a verbal category, and a lemma, each of these features were counted from each subcorpus and compared in order to determine whether verbs are used differently. Since the size of the two subcorpora was not exactly the same, counts from each were normalized. We used two

IMPERFECT	ممكن تدلني على حد يديني عنوانه
PERFECT	مراد لو كان عايز يتجوزني فعلا مكنش ادى ودنه لسارة
IMPERATIVE Positive	طب وانا ذنبي ايه يا شوقي بيه اديني التمثالين بتوعي وحاجتك جاية في الطريق
IMPERATIVE Negative	متدنيش الوش الكئيب ده. كفاية البومة اللي عندي في البيت
HABITUAL	فولتارين! من امتى بندي حقن فولتارين احنا
FUTURE	على كل حال أنا مش هديهم اي كلمة إلا لما أسمع رأيك بقى

Table 1: Sample feature-based verb annotations

statistical tests to compute significance: (1) log-likelihood because of its frequent appearance in corpus linguistics studies (Wilson, 2013); and (2) the Bayesian Information Criterion (BIC) for its reliability over chi square when dealing with word counts that fall on either end of the frequency spectrum (Dunning, 1993; Rayson and Garside, 2000).

Verbs are more common in the Blogs subcorpus than in Movies (see Table 2). However, not all verbal categories (IMPERFECT, PERFECT, IMPERATIVE, HABITUAL, and FUTURE) in Blogs occur more frequently than in Movies. Table 3 shows the category differences, all of which are statistically significant except for FUTURE. However, some of these differences change when we compare frequency to the total amount of verbs in each corpus rather than the total number of words. This is because of the higher concentration of verbs in Blogs, causing counts of the verbal categories taken out of the total number of words to be misleading (Gries, 2006).

Total	Movies	Blogs
# of words	228,399	141,318
# of verbs	38,768	27,083
% of verbs	16.97	19.54

Table 2: Totals and percentages of verbs

Category	% of Words		% of Verbs	
	Movies	Blogs	Movies	Blogs
Imperfect	6.56	7.90	38.67	40.45
Perfect	5	6.4	26.86	32.54
Habitual	1.83	2.46	10.8	12.58
Imperat.	2.57	1.52	15.15	7.8
Future	1.44	1.3	8.51	6.67

Table 3: Frequency of verbal categories

When factored by the total amount of words in each subcorpus, IMPERFECT is significantly more frequent in Blogs; however, this significance disappears when the frequency is compared with the total number of verbs. The opposite is true of the verbs hosting the FUTURE morpheme. Although the comparative frequency of this verb in Movies was insignificant when compared to the total words, its frequency becomes significant when compared only to verbs. PERFECT and HABITUAL are significantly higher in Blogs by both comparisons.

The IMPERATIVE represents the largest difference in usage between the two subcorpora. We investigate further by separating the imperatives into four categories: negative IMPERATIVE, positive 2SG.MASC.IMPERATIVE, positive 2SG.FEM.IMPERATIVE, and positive 2PL.IMPERATIVE.

Instead of comparing the frequencies of the imperatives against the total number of words in the corpus, we compared them to the total number of verbs. The numbers for each category of imperative are given in Table 4.

Type	Per 100 . . .	Movies	Blogs
Negat.	verbs	1.1	0.7
	imperatives	7.29	9.01
Male Posit.	verbs	10.14	5.88
	imperatives	66.89	75.83
Female Posit.	verbs	3.31	0.69
	imperatives	21.87	8.87
Plural Posit.	verbs	0.6	0.49
	imperatives	3.95	6.3

Table 4: Frequency of imperatives across subcorpora

The frequencies of IMPERATIVEs in Movies are all significantly higher than in Blogs except for the positive 2PL.IMPERATIVE. However, the table reveals that as a percentage of the total imperatives used, the Blogs uses the positive 2SG.MASC.IMPERATIVE and positive 2PL.IMPERATIVE significantly more than Movies. Negative imperative use is significantly greater in Movies compared to all verbs, but not significantly greater when compared to all IMPERATIVE verbs. Only the positive 2SG.FEM.IMPERATIVE remains more frequent in Movies regardless of its comparison set.

Another feature used to prove register variation is lemma frequency across verbs from each register. Among the verbs that occur in either subcorpus with a frequency of over 100, seventeen verbs show a frequency significantly higher in one register over the other (see Table 5). Table 6 illustrates percentage of IMPERATIVE verb forms in each subcorpus for these seventeen verbs.

More common in...			
Movies		Blogs	
اتفضل	please, come in	بدأ	to begin
خشّ	leave	دخل	to enter
هدي	to calm oneself	كتب	to write
أكل	to eat	حاول	to try
ساب	to leave	لقى	to find
استنى	to wait	حسّ	to feel
شرب	to drink	فتح	to open
مشي	to walk	رد	to respond
		قرأ	to read

Table 5: Contrastive distribution of high-frequency verbs across registers

Word		% Imperat.	Meaning
اتفضل	M	93.6	please, come in
هدي	M	83.2	to calm oneself
استنى	M	56.8	to wait
خشّ	M	43.4	to enter
فتح	B	32.3	to open
ساب	M	31.7	to leave
مشي	M	29.9	to walk
رد	B	26.7	to respond
دخل	B	20.5	to enter
قرأ	B	15.9	to read
أكل	M	11.3	to eat
كتب	B	11.1	to write
حاول	B	10.2	to try
شرب	M	5.6	to drink
بدأ	B	2.6	to begin
حسّ	B	0.09	to feel
لقى	B	0	to find

Table 6: IMPERATIVE usage of verbs most common to each subcorpus (M=Movies, B=Blogs)

Having a lemmatized corpus also permits comparison of lexical diversity between the two registers. Using the Biber (2006) formula for normalizing lexical diversity counts, statistics for each register are given in Table 7. All differences displayed are statistically significant, suggesting that Blogs is richer in verb types as well as in verb diversity.

	Types	Diversity	Types/1M Verbs	Types/1M Words
Movies	2,279	5.88	11,574	4,768
Blogs	2,079	7.53	12,510	5,530

Table 7: Diversity of verbs

4.1 Reflections

The data reported above provide enough evidence to warrant a wider investigation into the variations that exist between these potential registers. In English, use of the PERFECT has been identified as a feature of narration (Biber and Conrad, 2001; Staples, 2016). If Egyptian Arabic behaves like English, then the higher frequency of the PERFECT in Blogs signals a greater reliance on narration than in Movies, hence forming different registers. The possibility of the Egyptian Arabic PERFECT being a feature of narration is further supported by Biber et al. (2006), who found that most English internet texts could be classified as narrative.

Similarly, the frequency of the IMPERATIVE in Movies could easily be a feature of involved and non-narrative speech as found in Somali (Biber and Conrad, 2001). Therefore, the frequency of the PERFECT and IMPERATIVE in both subcorpora suggests a difference in narrative-based register separation. Distinct differences in HABITUAL versus FUTURE could also be linked to narration, but possibly some other feature. As little is known about the features of each dimension of Egyptian Arabic, a deeper investigation is needed so that the frequency of these verbal tenses and aspects can be put into context.

The variance of the frequencies of the verbal aspects and moods further supports register variation as one corpus alone cannot be used to generate a description of how verbs are used. In both subcorpora IMPERFECT is used more than any other verb aspect or mood followed by PERFECT. If this description were based solely on Movies, IMPERATIVE would be the third most common verb form. However, this is not true of Blogs. Therefore, the omission of one subcorpus from analysis would skew the description of the dialect: both need to be taken into account when producing a description of the language.

The subjects of IMPERATIVE verbs also seem to differ across register. The number of female positive imperatives in Blogs is trivial when compared to Movies. However, many factors independent of register could affect this result. Further investigation is needed to determine whether this difference can be used to indicate register variation.

Greater frequency of verb tokens and types in Blogs in EA is also interesting. Biber (2006) found, for academic English, that verbs were much more common in spoken academic registers (e.g. lectures vs. journal articles); features associated with verbs were also found to be characteristic of the oral dimension of Spanish and English more generally (Biber, 1999; Biber et al., 2006). However, the opposite appears to be true for Egyptian Arabic: Blogs contains a statistically higher number of verbs than Movies does.

Blogs also contains a greater diversity of verbs, which is consistent with English and Spanish; this may be expected as authors have time to think about the words they will use and revise their choices (Biber, 2006; Biber et al., 2006). In this study our differences in EA verb usage (i.e. number of verbs and their variety) suggest that the language contained in Blogs and Movies is different. In theory, both are written and revised; therefore, the difference in the diversity of verbs cannot be due to the fact that one of the registers is written. One factor that could have contributed to this is the size of the annotated corpus, but it could also be true that a feature of spoken Egyptian Arabic—like Spanish and English—is a lack of verbal diversity. Therefore, if this pattern holds as more of the corpus becomes annotated, it would constitute further evidence of register variation.

5 Notes on annotation

Linguistic annotation is the process by which additional linguistic information is added to a corpus in order to facilitate quantitative analyses of corpus content and user queries (Kübler and Zinsmeister, 2015). Manual annotations are performed by humans, automatic annotations are done by a computer program, and automatic annotations that are checked by a human for accuracy are called semi-automatic annotations.

Automatic annotators available for EA are somewhat limited, and although more resources exist for MSA, the morphological and lexical differences cause MSA annotators a challenge in annotating EA texts (Maamouri et al., 2014). In 2004, a part-of-speech annotator for MSA was achieving an accuracy 95.49% (Diab et al., 2004) versus a contemporary analyzer for EA with an accuracy at 62.76% (Duh and Kirchhoff, 2005). One reason for the disparity was the lack of large corpora or a complete lexicon of EA for annotator training (Habash and Rambow, 2006).

Abo Bakr et al.'s (2008) annotator translated Egyptian Arabic sentences into MSA and then tagged the MSA for part of speech, which would then be applied back to the Egyptian words. Conversion of the Egyptian Arabic to MSA was successful 88% of the time, and overall accuracy ratings for tokenization and part-of-speech tagging for EA were 90% and 85% respectively.

Al-Sabbagh and Girju (2012a) created an Egyptian Arabic tagger that did not depend upon MSA. Originally trained on three language types (Twitter, QA Pairs, and blogs), its highest reported F-measure among them for POS tagging is 0.907 (QA Pairs), though it had less success on blogs (with an F-measure of 0.888).

MADAMIRA (Arfath Pasha et al., 2014) analyzes each word according to the possible morphemes attached to it. It then uses language models to provide a morphological analysis, part-of-speech, lemma, and diacritics for each word in a text. Its accuracy score for part-of-speech tagging is 0.923. MADAMIRA's ability to provide lemmatization makes it valuable tool for register variation studies.

One issue regarding annotators involves whether they are accurate enough to be used without the need for a manual review of the results. Another annotator issue is how well accuracy persists when annotating texts in a different domain from the training set. There is an apparent lack of published research on using MADAMIRA in this cross-domain fashion. MADAMIRA was trained on transcripts of speech (Habash et al., 2012), but the literature is less clear about the register of Egyptian Arabic on which it was evaluated (Arfath Pasha et al., 2014). We would expect the average accuracy of MADAMIRA to shift either up or down when applied to other registers of the dialect as this phenomenon has been found in other languages (Tseng et al., 2005; Derczynski et al., 2013).

Numerous tagsets are available to use for Arabic part-of-speech tagging (Arfath Pasha et al., 2014;

Alian and Awajan, 2018). A modified version of the tagest employed by MADAMIRA was used for CALM annotation. Verbs were annotated as such, even in the presence of pronominal object suffixes and prepositional proclitics. Additionally, a second layer of annotation was applied to all verbs to indicate certain verbal categories. MADAMIRA divides verbs into three groups: imperfect (i), perfect (p), and command (c). In CALM, two more categories were created from the imperfect category. Although verbs in the HABITUAL (h) and FUTURE (f) are IMPERFECT, these were promoted as separate categories for ease of searching. Another change to MADAMIRA's annotations in CALM is the identification of negative imperatives and their inclusion into the "command" category. (The default tagset collapses imperfect verbs and negative imperatives into one class.)

Annonation of CALM also includes a few other adjustments to MADAMIRA's output: (1) MADAMIRA does not view the passive verbs as a verb form but adds an extra layer of annotation; these were folded into the basic verb paradigm in CALM. (2) Slight differences in lemmatization involved clarification by adding short vowels where necessary.

Overall MADAMIRA performed relatively well in annotating Movies and Blogs from CALM, and a combination of post-processing, both manual and automatic, made corrections when necessary. Hereafter we refer to raw annotations as "non-gold", and corrected annotations as "gold". Table 8 shows both the non-gold and gold statistics for the content shown earlier in Table 7.

	Types	Divers.	Types/1M Verbs	Types/1M Words
Non-gold				
Movies	2,867	7.44	14,608	5,999
Blogs	2,751	10.08	16,651	7,318
Gold				
Movies	2,279	5.88	11,574	4,768
Blogs	2,079	7.53	12,510	5,530

Table 8: Diversity of verbs (non-gold and gold)

Table 9 gives the counts for each of the verb types as annotated by the automatic tagger (the "non-gold" annotations) and after human correction (the "gold" annotations), and the percent change between the two annotation types. In all cases, the cross-register differences in verb usage that were significant in the gold subcorpora also held in the non-gold subcorpora. This nearly holds for the imperatives as well, except that the non-gold corpora do not report a significant difference in the use of the 2PL.IMPERATIVE in Blogs. As explained earlier, the automatic tagger does not attempt to categorize negative imperatives. For that reason, each cell in its row contains 'NA'.

For verb diversity measures, MADAMIRA data are nearly identical to the lists generated by hand (i.e. those in Table 5) except for six verbs whose counts were not accurate enough to reveal the statistically significant register differences. Regarding comparative verbal diversity, MADAMIRA scores diversity in Movies at 7.44% and in Blogs at 10.08% (a difference of 2.64%) whereas manual correction yields a difference in diversity of only 1.65%.

In conclusion, the annotations produced solely by MADAMIRA would have led researchers to nearly the same conclusions as those reached above with hand-corrected annotated data. The counts for overall verbs and verbal categories varied in every case from the numbers provided by the corrected annotations; however, the variations were not enough to change the results. Except for the IMPERATIVE category, MADAMIRA's total number of verbs in each category in Blogs changed by less than 5% after hand-correction. In both subcorpora, MADAMIRA was consistent with the categories that it over- and under-represented: IMPERFECT and PERFECT were both overrepresented, and IMPERATIVE and HABITUAL were both underrepresented. The only exception was the FUTURE category, which showed an underrepresentation in Movies and the opposite in Blogs.

One difficulty MADAMIRA had was in differentiating proper nouns from verbs, a challenge since Arabic has no capital letters. IMPERFECT and PERFECT were overrepresented due to misclassification, precision was lower on Movies, and recall on proper nouns suffered. In Movies, 7 of the top 10 words incorrectly tagged as verbs were actually names and titles given to people. Seventeen word forms represent 1,239 of the 3,290 recall errors of this type, comprising 37.6% of all the false positives. Names in the Blogs subcorpus were also problematic; in the top 30 false positives there, 8 were names (totaling 223 occurrences). This type of ambiguity only accounts for 10.5% of the total number of false positives, though, likely due to lower use of personal names in the Blogs.

	Movies		Blogs	
	Non-Gold	% Change	Non-Gold	% Change
All verbs	38,518	-0.65	27,296	-1.16
Imperfect	15,807	+5.44	11,448	+3.96
Perfect	11,714	+12.47	9,343	+3.96
Command	3,762	-35.97	1,324	-38.22
Habitual	3,962	-5.35	3,301	-4.95
Future	3,273	-0.82	1,880	+2.01
Imperatives				
Neg.	NA	NA	NA	NA
Pos. 2SG.MASC	1,455	-40.95	692	-35.33
Pos. 2SG.FEM	877	-31.75	179	-9.6
Pos. 2PL	165	-28.88	89	-34.07

Table 9: Effect of hand correction for frequency counts

Overall MADAMIRA performed relatively well in annotating the verbs in Movies and Blogs from CALM. However, in order to achieve higher accuracy for this paper, the annotations were manually reviewed and corrected. Throughout the process of manual correction, high-frequency errors made by MADAMIRA became apparent and a supplemental Python post-processor was developed to target these mistakes. This program was able to boost MADAMIRA's precision score from 0.922 to 0.944. Although the post-processor was able to reduce the number of corrections needed, every automatically assigned annotation was manually reviewed. Details are discussed elsewhere (White, 2019, forthcoming).

6 Conclusions and future work

This paper discussed the need for an Egyptian Arabic corpus of spoken language transcripts and introduced CALM, a new two-million word corpus of spoken EA. It also conducted an analysis into the use of verbs in two potential registers of EA.

The results show significant variance in the usage of verbs in Movies versus Blogs. These differences are consistent with variations found between other registers in previous multidimensional analyses. These results also lay the groundwork for future studies by providing a description of some of the dimensions of EA based upon empirical data.

We also showed that in spite of the challenges in annotating Egyptian Arabic, an automatic tagger was able to produce results that were not appreciably different from those produced through a process of manual correction.

The scope of this work was to show how a non-trivial subset of CALM could serve as data for a register analysis. It was limited in several ways, all of which can be extended via further research. First, a finer distinction into register types (especially blog subtypes) could be enacted, as has been done for other languages. In addition, this work involved annotations based on only one part of speech (i.e. verbs), whereas other categories could serve for similar analyses once annotations are available. Third, given the ongoing debate about whether transcripts of scripted speech can be used to represent speech, more study should ascertain how exactly dialogue and narration are characterized for register in EA. Finally, insight could be sought concerning the frequent use of the HABITUAL in Blogs. Is this due to the narrative dimension, or some other one represented in Blogs? Answers to this question can inform curricula for Egyptian Arabic learners, who often find this feature difficult.

References

Ernest T. Abdel-Massih, Zaki N. Abdel-Malek, and El-Said Badawi. 2009. A Reference Grammar of Egyptian Arabic. Georgetown University Press, Washington D.C.

Hitham Abo Bakr, Khaled Shaalan, and Ibrahim Ziedan. 2008. A Hybrid Approach For Converting Written Egyptian Colloquial Dialect Into Diacritized Arabic. In The 6th International Conference on Informatics and Systems (INFOSYS 2008).

Rania Al-Sabbagh and Roxana Girju. 2012a. A Supervised POS Tagger for Written Arabic Social Networking Corpora. In Proceedings of the Conference on Natural Language Processing (KONVENS), pages 39–52.

Rania Al-Sabbagh and Roxana Girju. 2012b. YADC: Yet another Dialectal Arabic Corpus. In Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC), pages 2882–2889.

Jawharah Alasmari, E. Atwell, and J. Watson. 2017. Using the Quranic Arabic Corpus for Comparative Analysis of the Arabic and English Verb Systems. International Journal on Islamic Applications in Computer Science and Technology, 5.

Abdullah Alfaifi and Eric Atwell. 2015. Arabic learner corpus.

Ahmed Ali, Stephan Vogel, and Steve Renals. 2017. Speech recognition challenge in the wild: Arabic MGB-3. In 2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), pages 316–322.

Marwah Alian and Arafat Awajan. 2018. Arabic Tag Sets. In Proceedings of SAI Intelligent Systems Conference, pages 592–606. Springer.

Khalid Almeman and Mark Lee. 2013. Automatic Building of Arabic Multi Dialect Text Corpora by Bootstrapping Dialect Words. In Proceedings of 1st International Conference on Communications, Signal Processing, and their Applications (ICCSPA), pages 1–6, Sharjah, UAE. Institute of Electrical and Electronics Engineers (IEEE).

Mohamed Al-Badrashiny Arfath Pasha, Mona T. Diab, and Ahmed El Kholy. 2014. MADAMIRA: A Fast, Comprehensive Tool for Morphological Analysis and Disambiguation of Arabic. In Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC).

Marco Baroni and Silvia Bernardini. 2004. BootCaT: Bootstrapping Corpora and Terms from the Web. In Proceedings of the 4th Langauge Resources Evaluations Conference (LREC), page 1313.

Randell Bentley. 2015. Conditional Senteces in Egyptian Colloquial Arabic and Modern Standard Arabic: A Corpus Study. Master's thesis, Brigham Young University.

Douglas Biber. 1993. Representativeness in Corpus Design. Literary and Linguistic Computing, 8(4):2430–257.

Douglas Biber. 1999. Longman Grammar of Spoken and Written English. Harlow, England.

Douglas Biber. 2006. University Language: A Corpus-based Study of Spoken and Written Registers. John Benjamins, Philadelphia.

Douglas Biber and Susan Conrad. 2001. Register Variation: A Corpus Approach. In The Handbook of Discourse Analysis. Blackwell Publishers, Massachusetts.

Douglas Biber, Mark Davies, James K. Jones, and Nicole Tracy-Ventura. 2006. Spoken and Written Register Variation in Spanish: A Multidimensional Analysis. Corpora, 1(1):1–37.

Douglas Biber, Jesse Egbert, and Mark Davies. 2015. Exploring the Composition of the Searchable Web: A Corpus-Based Taxonomy of Web Registers. Corpora, 10(1):11–45.

Marc Brysbaert and Boris New. 2009. Moving beyond Kučera and Francis: A Critical Evaluation of Current Word Frequency Norms and The Introduction of a New And Improved Word Frequency Measure For American English. Behavior Research Methods, 41(4):977–990.

Tim Buckwalter and Dilworth Parkinson. 2011. A Frequency Dictionary of Arabic: Core Vocabulary for Learners. Routledge, New York.

Alexandra Canavan, George Zipperlen, and David Graff. 1997. CALLHOME Egyptian Arabic Speech. Linguistic Data Consortium Web Download, LDC97S45. Philadelphia, PA.

Leon Derczynski, Alan Ritter, Sam Clark, and Kalina Bontcheva. 2013. Twitter Part-of-speech Tagging for All: Overcoming Sparse and Noisy Data. In Proceedings of the International Conference Recent Advances in Natural Language Processing (RANLP), pages 198–206.

Mona Diab, Kadri Hacioğlu, and Daniel Jurafsky. 2004. Automatic Tagging Of Arabic Text: From Raw Text To Base Phrase Chunks. In Proceedings of HLT-NAACL 2004: Short papers, pages 149–152. Association for Computational Linguistics.

James Dickins. 2017. The Pervasiveness of Coordination in Arabic, with Reference to ArabicgtEnglish Translation. Languages in Contrast, 17(2):229–254.

Kevin Duh and Katrin Kirchhoff. 2005. POS Tagging Of Dialectal Arabic: A Minimally Supervised Approach. In Proceedings of the acl Workshop on Computational Approaches to Semitic Languages, pages 55–62. Association for Computational Linguistics.

Ted Dunning. 1993. Accurate Methods for the Statistics of Surprise and Coincidence. Computational Linguistics, 19(1):61–74.

Abbas El-Tonsi. 1982. Egyptian Colloquial Arabic: A Structure Review, volume 1. American University In Cairo, Cairo, Egypt.

Ahmed Fakhri. 2009. Rhetorical variation in Arabic academic discourse: Humanities versus law. Journal of Pragmatics, 41(2):306–324.

Charles A. Ferguson. 1983. Sports announcer talk: Syntactic aspects of register variation. Language in Society, 12(2):153–172.

Pierfranca Forchini. 2012. Movie Language Revisited: Evidence from Multi-Dimensional Analysis and Corpora. Peter Lang, Bern.

Eric Friginal. 2009. Language of Outsourced Call Centers: A Corpus-based Study of Cross-cultural Interaction. John Benjamins, Philadelphia.

Stefan Th. Gries. 2006. Exploring variability within and between corpora: some methodological considerations. Corpora, 1(2):109–151.

Nizar Habash, Ramy Eskander, and Abdelati Hawwari. 2012. A morphological analyzer for Egyptian Arabic. In Proceedings of the Twelfth Meeting of the Special Interest Group on Computational Morphology and Phonology (SIGPHON), pages 1–9. Association for Computational Linguistics.

Nizar Y. Habash and Owen C. Rambow. 2006. MAGEAD: A Morphological Analyzer And Generator For The Arabic Dialects.

Gadalla Hassan. 2000. Comparative Morphology of Standard and Egyptian Arabic. LINCOM EUROPA.

David Henen. 2018. “ya” between Vocative and Non-Vocative Use in Egyptian Film Language A Corpus Analysis: Pragmatic Functions and Formal Features. American University in Cairo, Egypt.

Susan Hunston. 2002. Corpora in Applied Linguistics. Cambridge University Press, Cambridge.

Mona Hussein. 2016. Propositional and Non-Propositional Functions of /Keda/ in the Language of Egyptian Film. American University in Cairo, Egypt.

Ahmad Ismail. 2015. ṭab asta’zen ana ba’a: A corpus-based Study of Three Discourse Markers in Egyptian Film Language. American University in Cairo, Egypt.

Barbra Johnstone. 2008. Discourse Analysis. Blackwell, Malden, MA.

Sandra Kübler and Heike Zinsmeister. 2015. Corpus Linguistics and Linguistically Annotated Corpora. Bloomsbury, New York.

Mohamed Maamouri, Ann Bies, Seth Kulick, Michael Ciul, Nizar Habash, and Ramy Eskander. 2014. Developing an Egyptian Arabic Treebank: Impact of Dialectal Morphology on Annotation and Tool Development. In Proceedings of LREC, pages 2348–2354.

Yousef Maati. 2008. ɦʌsʌn wi murʔosʕ. Al-daar Al-Masriya Al-lubnaniya, Cairo, Egypt.

Hamdi A. Qafisheh. 1992. Yemeni Arabic Reference Grammar. Dunwoody Press, Kensington, MD.

Paul Rayson and Roger Garside. 2000. Comparing corpora using frequency profiling. In Proceedings of the workshop on Comparing corpora, volume 9, pages 1–6. Association for Computational Linguistics.

Karin C. Ryding. 2005. A Reference Grammar of Modern Standard Arabic. Cambridge University Press, Cambridge, England.

Karin C. Ryding. 2006. Teaching Arabic in the United States. In Handbook for Teaching Arabic Language Professionals in the 21st Century, pages 13–20. Routledge, New York.

Mukhtar Sayed. 2018. maʕleš maʕleš: A CORPUS-BASED STUDY ON THE DISCOURSE MARKER maʕleš. American University in Cairo, Egypt.

Serge Sharoff. 2006. Creating general-purpose corpora using automated search engine queries. In Baroni and Bernardini, editors, WaCky! Working papers on the Web as Corpus, pages 63–98. Gedit.

John Sinclair. 2004. Corpus Creation. In Geoffrey Sampson and Diana McCarthy, editors, Corpus Linguistics: Readings in a Widening Discipline, pages 78–84. Continuum, New York.

Shelly Staples. 2016. Identifying Linguistic Features of Medical Interactions: A Register Analysis. Talking at Work. Palgrave Macmillan, London.

Rianne Tamis and Janet Persson, editors. 2013. Sudanese Arabic-English; English-Sudanese Arabic: A Concise Dictionary. SIL International.

Christopher John Taylor. 2004. The Language of Film: Corpora and Statistics in the Search for Authenticity. Notting Hill (1998)-A Case Study. Miscelánea, pages 71–86.

Wolfgang Teubert. 2005. My Version of Corpus Linguistics. International Journal of Corpus Linguistics, 10(1):1–13.

Huihsin Tseng, Daniel Jurafsky, and Christopher Manning. 2005. Morphological features help POS tagging of unknown words across language varieties. In Proceedings of the Fourth SIGHAN Workshop on Chinese Language Processing.

Michael Grant White. 2019, forthcoming. Verb usage in Egyptian Arabic: a case for register variation. Master's thesis, Brigham Young University.

Andrew Wilson. 2013. Embracing Bayes Factors for key item analysis in corpus linguistics. In M. Bieswanger and A. Koll-Stobbe, editors, New Approaches To The Study Of Linguistic Variability, pages 3–9. Peter Lang, Frankfurt.

Crisis Detection from Arabic Tweets

Alaa Alharbi[1,2] and Mark Lee[1]

[1]School of Computer Science, University of Birmingham, Birmingham, UK
[2]College of Computer Science and Engineering, Taibah University, Medina, KSA
alaharbi@taibahu.edu.sa & m.g.lee@cs.bham.ac.uk

Abstract

Social media (SM) platforms such as Twitter offer a rich source of real-time information about crises from which useful information can be extracted to support situational awareness. The task of automatically identifying SM messages related to a specific event poses many challenges, including processing large volumes of short, noisy data in real time. This paper explored the problem of extracting crisis-related messages from Arabic Twitter data. We focused on high-risk floods as they are one of the main hazards in the Middle East. In this work, we presented a gold-standard Arabic Twitter corpus for four high-risk floods that occurred in 2018. Using the annotated dataset, we investigated the performance of different classical machine learning (ML) and deep neural network (DNN) classifiers. The results showed that deep learning is promising in identifying flood-related posts.

1 Introduction

Social media (SM) platforms provide a valuable source of real-time information about emergency events. During mass emergencies, microblogging sites such as Twitter are used as communication channels by people and organisations to post situational updates, provide aid, request help and search for actionable information. Examples of Twitter's effectiveness during crises include the Manila floods in 2013 (Olteanu et al., 2015), the Louisiana flood in 2016 (Kim and Hastak, 2018) and tropical storm Cindy in 2017 (Kim et al., 2018). Twitter was used to report the protests that followed the Iranian presidential elections of 2009 (Khondker, 2011). It also played an important role in the Arab Spring (Arafa and Armstrong, 2016). For instance, Twitter was used as a means of communication by protesters during the Egyptian revolution in February 2011 (Tufekci and Wilson, 2012). Petrovic et al. (2013) found that Twitter often breaks incoming news about disaster-related events faster than traditional news channels. The early identification of disaster-related messages enables decision-makers to respond quickly and effectively during emergencies.

The huge volume of user-generated Twitter data related to numerous daily events has given rise to the need for automatic event extraction and summarising tools. Event extraction from Twitter streams poses challenges that differ from traditional media. In particular, traditional text extraction techniques are challenged by the noisy language used in social media, including colloquialisms, misspelled words and non-standard acronyms. Because of the imposed character limit (280 characters), Twitter users tend to use more abbreviations and may also post non-informative messages that require some knowledge of the situational context for interpretation. In addition, Twitter's popularity makes it appealing for spammers who spread propaganda, pornography and viruses (Benevenuto et al., 2010; Kabakus and Kara, 2017). Another challenge posed by Twitter is that the increasing volume and high-rate data stream of user-generated messages create significant computational demand.

Previous studies that have explored the problem of extracting crisis-related messages from SM have proposed various matching-based and learning-based approaches. Supervised machine learning (ML) and deep learning models have been used to identify event-relevant messages and classify them into several categories. A significant percentage of such studies have been conducted on English SM text. Very little work has focused on Arabic text. The Arabic language has its own peculiarities that make classifying Arabic SM text more challenging. For example, SM users sometimes write in their own dialects. There exist many spoken Arabic dialects that differ in their phonology, morphology and syntax (Chiang et al., 2006). People tend to write the dialectical words according to their own pronunciations. There is no spelling standard for written dialectical words. Dialects are region-based. Hence, a classifier trained on data collected from one region may not perform well when tested on data collected from another region.

Unlike English, Arabic has poor available resources. To the best of our knowledge, there is no publicly available Arabic crisis-related dataset. We therefore built our own. In this work, we focused on flooding crises as they are a major hazard in the Middle East. A crisis usually occurs after heavy rain and subsequent flash flooding. In October and November 2018, heavy rainfall caused severe flooding in various Middle Eastern countries including Saudi Arabia, Kuwait, Jordan, Qatar, Iraq and Iran. According to civil defence authorities in Saudi Arabia, 1,480 individuals were rescued, 30 died and 3,865 were evacuated during floods that occurred in the period between 19 October and 14 November.[1] In Jordan, the flash flood on 9 November left at least 12 people dead and 29 injured.[2] On the same day, Kuwait had heavy rain that resulted in infrastructure and property damage and left at least one person dead.[2]

This research used different supervised learning approaches to extract flood-related tweets for the purpose of enhancing crisis management systems. We investigated the ways in which deep neural networks (DNN) compare to ML models in identifying crisis-related SM messages. Inspired by Nguyen et al. (2017), we also explored how different models perform when they are trained on historical event data, as labelling data from current events is expensive. Furthermore, continually retraining a model from scratch using data from current events is undesirable as it delays the timely processing of messages. The contributions of this paper as follows:

- We provide an annotated Arabic Twitter dataset of flood events.
- We benchmark the dataset using different supervised learning approaches.
- We evaluate the performance of two classical ML models and four DNNs on extracting flood-related messages, under two training settings: (1) train and test on the same event data; and (2) train on previous in-domain events and test on the current event.

The rest of this paper is organised as follows: section 2 surveys related work. Section 3 describes the process of building the Arabic flood Twitter dataset. Section 4 presents the used ML and DNNs models. The experimental settings and the results are detailed in sections 5 and 6, respectively. Finally, section 7 concludes the paper and discusses future work.

2 Related Work

A review of the recent literature confirms widespread interest in detecting and extracting information from Twitter posts that describe current events. Recently, extracting crisis-related events from social media has received considerable attention.

Kireyev et al. (2009) experimented with latent Dirichlet allocation (LDA) topic models to detect disasters from Twitter posts. Sakaki et al. (2010) developed an earthquake reporting system by processing Twitter data. They used a support vector machine (SVM) to classify Twitter messages into two groups: event and non-event. They also proposed temporal and spatial models to estimate the earthquake's location. Cameron et al. (2012) presented a model to detect crises from Twitter using burst detection and incremental clustering. Abel et al. (2012) described a framework called Twitcident for searching, filtering and analysing Twitter streams during incidents. Twitcident monitors broadcasting services and translates incident-related messages into profiles for use as Twitter search queries to extract relevant tweets.

Using a supervised ML approach, Imran et al. (2013a) classified Twitter posts into fine-grained classes and extracted the relevant information from the messages. In a subsequent work, they described a method for extracting disaster information using conditional random fields (CRF) (Imran et al., 2013b). Ashktorab et al. (2014) described a supervised learning-based approach to identifying disaster-related tweets and extracting actionable information.

Singh et al. (2017) developed a classification-based system to extract flood-related posts and classify them as high or low priority to identify victims who need urgent assistance. Nguyen et al. (2017) and Caragea et al. (2016) used convolutional neural network (CNN) to identify in-

[1] https://sabq.org/jGVvgZ

[2] http://floodlist.com/asia/jordan-flash-floods-november-2018

formative (useful) messages from crisis events. Nguyen et al. (2017) highlighted that CNN performed better than many classical ML approaches. Going further, Neppalli et al. (2018) compared the performance of a naïve Bayes (NB) classifier to two deep neural models in identifying informative crisis-related posts. Their results demonstrated that CNN outperformed both the recurrent neural network (RNN) with gated recurrent unit (GRU) model and the NB with handcrafted features. Unlike the described work, which focused on the classification of English tweets to extract the relevant event messages, Alabbas et al. (2017) used supervised ML classifiers to identify high-risk flood-related tweets that were written in Arabic.

Using different classical ML and deep learning approaches, we also classified the Arabic tweets as flood-related or irrelevant. Our work differs from that of Alabbas et al. (2017) in the classification techniques and data collection. Instead of tracking the Arabic words سيول، فيضانات (floods), we based our collection on event-related keywords as described in the following section.

3 Corpus Collection and Annotation

Using the Twitter API,[3] Arabic tweets were collected by tracking certain keywords and hashtags related to 10 flood events. The tracked floods occurred in the Middle East in October and November, 2018. The initial set of tweets for each event were crawled based on the event-related trendy hashtags or by searching for tweets containing the terms سيول (floods) and the flood location name. Then, the dataset were expanded by tracking all the relevant hashtags found in the collected set. This step was repeated until no new event-related hashtag could be found. Different numbers of messages were obtained per event. While we managed to crawl thousands of tweets for some events, we ended up with just a few hundred for others. The size of candidate flood-related data might depend on the popularity and severity of the event.

In this research, only four events were considered for annotation. The events were: Jordan floods, Kuwait floods, Qurayyat floods and Al-Lith floods. The selected events took place in different areas of the Arab world: Jordan, Kuwait, northern Saudi Arabia and western Saudi Arabia, respectively. Hence, we believed that the dataset should include tweets written in different Arabic dialects. In addition, each of these events trended on Twitter. We collected plenty of candidate flood-related tweets, at least 5,000 for each of the four crisis under consideration. The four floods led to property and infrastructure damage. Three of them left several people displaced or dead. Therefore, we assumed that the collected messages would convey different types of disaster-related actionable information.

To construct the dataset, we first extracted the tweet IDs and texts from the event-related JSON files obtained by the Twitter streaming API. Each retweet was replaced with the original text of the retweeted message. We removed duplicates (i.e., tweets that had exactly the same text). After that, a random sample of around 1,050 distinct tweets was selected from each event to be annotated by a human. As the Qurayyat flood had only 954 distinct messages, we labelled them all. The corpus was annotated by four native Arabic speakers. They were provided with the annotation instructions, examples of ten labelled tweets and a brief description of each event. Annotators were asked to provide the appropriate label based on the tweet's text; they were not required to open any included hyperlinks. Each tweet was judged by two annotators who selected the most suitable label for the two tasks described below.

1. **Relevance:** The first task was to decide whether a message was on-topic/event-related or off-topic/not related. Very short and understandable messages that did not convey any meaning, such as those that only included hashtags, were ignored.

2. **Information type:** In order to build classifiers that could identify informative crisis-related messages, tweets that communicated useful information were labelled based on the information category they provided. This task followed the annotation scheme described by Olteanu et al. (2015), which labelled each message as one of the following broad categories:

    - Affected individuals: included reports on affected, dead, missing, trapped, found or displaced people
    - Infrastructure and utilities damage

[3] https://help.twitter.com/en/rules-and-policies/twitter-api

- Donations, assistance and volunteering services
- Caution and advice
- Sympathy, prayers and emotional support
- Other useful information: messages that did not belong to the previous categories but helped in understanding the emergency situation
- Not applicable: the message was either irrelevant or did not communicate any useful information, e.g., personal opinions.

We measure inter-rater agreement with Cohen's Kappa, resulting in $k \approx 0.82$ for relevance and $k \approx 0.9$ for information type. In cases when the two annotators disagreed, the tweet was judged by a third person. The final dataset[4] included 4,037 labelled Twitter messages for four flood events. Table 1 presents a general description of the dataset along with the number of relevant and irrelevant messages per event. In our corpus, 24.69% of tweets were irrelevant. Table 2 shows the distribution of information categories per event.

4 Models

4.1 Classical ML Models

The performance of classic ML models depends mainly on how the features are extracted and selected. To benchmark the dataset, we experimented with SVM and NB for flood-related message identification.

4.2 Deep Learning Models

Deep learning has profound generalisation ability and has proven to perform well in text classification, achieving state-of-the-art results on standard natural language processing (NLP) benchmark problems. In this research, we experiment with the following deep learning models:

- Convolutional Neural Network (CNN): The network architecture was similar to that proposed by Kim (2014). We used two 1D convolutions that were applied in parallel to the input layer vectors, extracting local patches from sequences using convolution windows of sizes 3 and 5 with 100 feature maps each. A sliding max-pooling operation of size 2 was applied over each feature map to obtain the maximum value, representing the most important feature. The output vectors of the two convolutions were concatenated and a 0.5 dropout rate was applied for regularisation. The output was fed into a 100-dimension fully connected layer with rectified linear unit (ReLU) activation.

- Long Short-Term Memory (LSTM): LSTM (Hochreiter and Schmidhuber, 1997) is a type of recurrent neural network (RNN) that can learn over long input sequences. In our experiments, this model involved one LSTM layer with 196 hidden output dimensionalities. As proposed by Gal (2016), we applied a dropout rate for input units of the LSTM layer and a dropout rate of the recurrent units for regularisation. In the experiments, both were set to 0.2.

- Convolution LSTM (CLSTM): This model is similar to the CNN described above except that the fully connected dense layer is replaced by an LSTM layer similar to the one presented above. In this architecture, the CNN was used to extract features that were fed into an LSTM layer, which processed down-sampled high-level input sequences.

- Bidirectional LSTM (BiLSTM): BiLSTM is a another type of RNN. In processing input sequences in both forward and backward directions, BiLSTM merges their representations to capture patterns that might be missed by order-dependent RNNs such as LSTM. The bidirectional model in our experiments had 196 hidden dimensions and dropout rates equal to the ones used in the LSTM model.

The input sequences, the embedding and output layers were similar for all previously described DNN models. The embedding layer was used as the first hidden layer to map words (input sequences) to dense vectors. In our experiments, vectors were initialised from an external embedding model and fine-tuned during training. The output layer mapped its input vectors – which were obtained from the last hidden layer in each model – to a probability between 0 and 1 using the sigmoid activation function.

[4] It is available for research purposes at https://www.cs.bham.ac.uk/~axa1314/

Crisis	Country	# of Labelled Posts	# of On-topic Posts	# of Off-topic Posts
Jordan floods	Jordan	1009	761	248
Kuwait floods	Kuwait	1056	822	234
Qurayyat floods	Saudi	954	705	249
Al-Lith floods	Saudi	1018	752	266

Table 1: Dataset Description.

Flood Name	Affected individuals	Infrastructure & utilities	Donations & volunteering	Caution & advice	Sympathy & emotional support	Other useful information	Not applicable
Jordan	200	60	15	79	269	76	310
Kuwait	96	106	30	62	191	53	518
Qurayyat	184	58	8	244	131	37	292
Al-Lith	70	196	81	212	72	58	329

Table 2: Distribution of tweets by information types.

5 Experiments

In this study, we performed a binary classification task to identify flood-related messages. Identifying the information category of the relevant tweets is left for future work. As the dataset had imbalanced classes, we first up-sampled the minority class to have a relatively equal class distribution. Then we preprocessed the tweets as described below.

5.1 Text Preprocessing:

To improve model generalisation, we replaced each URL with the Arabic word رابط (hyperlink). In the same way, each user handle was substituted with the word مستخدم (username), while numbers were replaced with the word رقم (number). We also normalised repeated letters and elongation (Tatweel). Diacritics or short vowels, non-Arabic characters, punctuation and special characters were removed. We performed three types of letter normalisations: the variant forms of *alef* (أ، إ، آ) were normalised to (ا) , *alef maqsora* (ى) to *ya* (ي) and *ta marbouta* (ة) to *ha* (ه) . This was done because people often misspell various forms of *alef* and do not distinguish between *ta marbouta* and *ha* when these letters occur at the ends of words. In addition, stop words were removed. While stop word removal is not useful for some NLP tasks such as sentiment analysis, it can enhance the performance of some classification tasks as they do not affect the overall topic/meaning of a document. Finally, tweets were tokenised using the CMU Tweet NLP tool (Gimpel et al., 2010). We did not apply stemming to the tokens, as the previous work confirmed that stemming does not improve classification accuracy (Alabbas et al., 2017).

With respect to classic ML models, unigrams, bigrams and trigrams of words were extracted. In case of NB, text was represented as bags of words as we experimented with multinomial NB classifier which is suitable for classification with discrete features. For SVM model, the features were transformed into term-frequency inverse-document-frequency (TF-IDF) vectors, in which each tweet represented a document. For DNN models, texts were segmented into words. The maximum length of input sequences per tweet was set to 60 words. Messages comprising fewer than 60 words were zero-padded. Each word was transformed into a vector. Word vectors were initialised from Ara Vec (Soliman et al., 2017). Ara Vec was trained on Arabic Twitter text of 1,090 million tokens using a continuous bag of words (CBOW) technique with a window size of three words. In both types of models, we limited the vocabulary to the most common 5,000 words in the training corpus.

5.2 Training Settings:

We first examined the performance of the learning models in identifying the relevant messages when they were trained using data from the same event. In this case, the data was split into subsets of 80% for training and 20% for testing using 5-fold cross validation. Assuming that labelled data were not available for the current event, the second experiment evaluated the models' performance when they were trained using the historical events. Here, the entirety of the data pertaining to the event under consideration was used for training and testing.

5.3 Models' Settings:

Classic ML classifiers were implemented using the scikit-learn library (Pedregosa et al., 2011). We experimented with linear kernel SVM and multinomial NB classifiers. Deep learning models were built using the Keras library.[5] The DNN models were trained for 10 epochs in mini-batches of 10 samples. The optimiser and loss function arguments were set to adam and binary crossentropy, respectively.

6 Results

Table 3 shows the average accuracy scores for the first experiment, in which classifiers were trained on the event data using 5-fold cross-validation. The table indicates that DNNs performed very well despite the relatively small training dataset. The deep learning models yielded comparable performance. RNNs outperformed the ML models in all cases. LSTM and BiLSTM achieved the best accuracy scores. SVM returned results that were competitive with DNN models. Looking at the classification errors of the LSTM and BiLSTM models, we found that the most common error is the incorrect classification of minority class (off-topic tweets). This is due to the imbalanced dataset. The random over-sampling can increase the likelihood of overfitting the data as it creates exact copies of existing instances. We also found that some of the uninformative flood-related messages were mistakenly classified as off-topic. For instance, 14% of such messages in Kuwait data were incorrectly classified by the LSTM model.

As the identification of crisis-related messages is a time-critical task, it is unlikely to obtain sufficient labelled data from the current event. Hence, we explored how the classifiers perform in detecting relevant posts from different events within the same domain. The accuracy scores are displayed in Table 4. BiLSTM achieved the best accuracy in most cases. CLSTM and LSTM showed competitive results in certain instances. LSTM outperformed the CNN in 9 out of 11 experiments. This showed that RNNs could be more suitable to address such problems as they represent the whole input sequence instead of relying on some key local features. Feeding the extracted CNN features into an LSTM layer instead of a fully connected dense layer resulted in improved accuracy when training on one event. The structure of RNNs allows such models to learn problem-specific information about the mapping they approximate, which could reduce the training data requirement. As the number of training examples increased, CNN achieved performance comparable with CLSTM. Table 4 shows that SVM generalised better than NB model. Generally, it can be seen that DNNs outperformed the traditional ML models. DNNs use distributed representation of words and learn high-level abstract features (Imran et al., 2018). On the other hand, ML models' performance depends on the training data and manually engineered features and therefore perform poorly when tested in different crises due to the great variation of data.

In the first six cases, we trained the models using data from a single event. Taking chronological order into account, we then increased the number of events in the training set to see whether this could enhance performance. It could be noticed that all models showed the best accuracy in classifying Al-Lith messages when three events were used for training. However, increasing the number of training events did not always result in improved accuracy. For example, training DNNs using data from Kuwait and Qurayyat resulted in lower performance compared to the case when only Qurayyat data was used to classify Al-Lith messages. Similarly, the results acheived by using Jordanian data to train ML models were higher than those obtained by using the joint dataset of Jordan and Kuwait.

7 Conclusion

This paper investigated the problem of extracting flood-related data from Arabic tweets using a supervised learning approach. To the best of our knowledge, it is the first work that uses deep learning to identify crisis-related data from Arabic tweets. Our results show that RNNs are promising in identifying crisis messages using training data from the event or from other in-domain events. We also provided a gold-standard Arabic Twitter dataset for high-risk floods. For future work, we aim to evaluate the same models in multiclass identification to extract information types from flood-related messages. We also plan to utilise domain adaptation approaches to enhance the results of classifiers trained using data within the crisis domain.

[5] https://keras.io/

Event	SVM	NB	CNN	LSTM	CLSTM	BiLSTM
Jordan floods	91.03	79.72	91.77	91.26	91.69	**92.14**
Kuwait floods	89.45	83.76	90.58	**91.91**	89.87	91.21
Qurayyat floods	94.18	90.19	92.87	95.17	94.64	**95.48**
Al-Lith floods	90.83	81.59	93.86	**94.08**	91.64	93.56

Table 3: The accuracy scores of classical ML and DNN models when they are trained on event data.

Train Set	Test Set	SVM	NB	CNN	LSTM	CLSTM	BiLSTM
Jordan floods	Kuwait floods	61.60	63.15	67.51	70.32	67.01	**70.46**
Jordan floods	Qurayyat floods	68.42	56.47	70.95	72.10	71.03	**72.33**
Jordan floods	Al-Lith floods	**71.22**	69.23	64.49	65.82	67.75	71.07
Kuwait floods	Qurayyat floods	**69.73**	59.15	62.22	64.13	67.66	**69.73**
Kuwait floods	Al-Lith floods	63.90	61.98	67.15	67.30	**71.81**	71.59
Qurayyat floods	Al-Lith floods	68.19	68.04	75.22	**76.40**	75.66	76.03
Jordan + Kuwait floods	Qurayyat floods	69.80	60.30	73.10	75.24	73.02	**76.55**
Jordan + Kuwait floods	Al-Lith floods	69.89	68.04	71.30	68.93	71.59	**74.40**
Jordan + Qurayyat floods	Al-Lith floods	73.89	72.04	75.88	75.36	76.62	**77.73**
Kuwait + Qurayyat floods	Al-Lith floods	70.85	70.71	72.63	**75.88**	74.92	74.48
Jordan + Kuwait + Qurayyat floods	Al-Lith floods	75.51	72.11	76.84	**77.95**	77.81	77.66

Table 4: The accuracy scores of classical ML and DNN models when they are trained on out-of-event data.

References

Fabian Abel, Claudia Hauff, Geert-Jan Houben, Richard Stronkman, and Ke Tao. 2012. Twitcident: fighting fire with information from social web streams. In *Proceedings of the 21st International Conference on World Wide Web*, pages 305–308. ACM.

Waleed Alabbas, Haider M al Khateeb, Ali Mansour, Gregory Epiphaniou, and Ingo Frommholz. 2017. Classification of colloquial arabic tweets in real-time to detect high-risk floods. In *2017 International Conference On Social Media, Wearable And Web Analytics (Social Media)*, pages 1–8. IEEE.

Mohamed Arafa and Crystal Armstrong. 2016. " facebook to mobilize, twitter to coordinate protests, and youtube to tell the world": New media, cyberactivism, and the arab spring. *Journal of Global Initiatives: Policy, Pedagogy, Perspective*, 10(1):6.

Zahra Ashktorab, Christopher Brown, Manojit Nandi, and Aron Culotta. 2014. Tweedr: Mining twitter to inform disaster response. In *ISCRAM*.

Fabricio Benevenuto, Gabriel Magno, Tiago Rodrigues, and Virgilio Almeida. 2010. Detecting spammers on twitter. In *Collaboration, electronic messaging, anti-abuse and spam conference (CEAS)*, volume 6, page 12.

Mark A Cameron, Robert Power, Bella Robinson, and Jie Yin. 2012. Emergency situation awareness from twitter for crisis management. In *Proceedings of the 21st International Conference on World Wide Web*, pages 695–698. ACM.

Cornelia Caragea, Adrian Silvescu, and Andrea H Tapia. 2016. Identifying informative messages in disaster events using convolutional neural networks. In *International Conference on Information Systems for Crisis Response and Management*, pages 137–147.

David Chiang, Mona Diab, Nizar Habash, Owen Rambow, and Safiullah Shareef. 2006. Parsing arabic dialects. In *11th Conference of the European Chapter of the Association for Computational Linguistics*.

Yarin Gal. 2016. *Uncertainty in deep learning*. Ph.D. thesis, PhD thesis, University of Cambridge.

Kevin Gimpel, Nathan Schneider, Brendan O'Connor, Dipanjan Das, Daniel Mills, Jacob Eisenstein, Michael Heilman, Dani Yogatama, Jeffrey Flanigan, and Noah A Smith. 2010. Part-of-speech tagging for twitter: Annotation, features, and experiments. Technical report, Carnegie-Mellon Univ Pittsburgh Pa School of Computer Science.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Muhammad Imran, Carlos Castillo, Fernando Diaz, and Sarah Vieweg. 2018. Processing social media messages in mass emergency: Survey summary. In *Companion of the The Web Conference 2018 on The Web Conference 2018*, pages 507–511. International World Wide Web Conferences Steering Committee.

Muhammad Imran, Shady Elbassuoni, Carlos Castillo, Fernando Diaz, and Patrick Meier. 2013a. Extracting information nuggets from disaster-related messages in social media. In *Iscram*.

Muhammad Imran, Shady Elbassuoni, Carlos Castillo, Fernando Diaz, and Patrick Meier. 2013b. Practical extraction of disaster-relevant information from social media. In *Proceedings of the 22nd International Conference on World Wide Web*, pages 1021–1024. ACM.

Abdullah Talha Kabakus and Resul Kara. 2017. A survey of spam detection methods on twitter. *International Journal of Advanced Computer Science and Applications*, 8(3).

Habibul Haque Khondker. 2011. Role of the new media in the arab spring. *Globalizations*, 8(5):675–679.

Jooho Kim, Juhee Bae, and Makarand Hastak. 2018. Emergency information diffusion on online social media during storm cindy in us. *International Journal of Information Management*, 40:153–165.

Jooho Kim and Makarand Hastak. 2018. Social network analysis: Characteristics of online social networks after a disaster. *International Journal of Information Management*, 38(1):86–96.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*.

Kirill Kireyev, Leysia Palen, and Kenneth Anderson. 2009. Applications of topics models to analysis of disaster-related twitter data. In *NIPS Workshop on applications for topic models: text and beyond*, volume 1. Canada: Whistler.

Venkata Kishore Neppalli, Cornelia Caragea, and Doina Caragea. 2018. Deep neural networks versus naïve bayes classifiers for identifying informative tweets during disasters.

Dat Tien Nguyen, Kamela Ali Al Mannai, Shafiq Joty, Hassan Sajjad, Muhammad Imran, and Prasenjit Mitra. 2017. Robust classification of crisis-related data on social networks using convolutional neural networks. In *Eleventh International AAAI Conference on Web and Social Media*.

Alexandra Olteanu, Sarah Vieweg, and Carlos Castillo. 2015. What to expect when the unexpected happens: Social media communications across crises. In *Proceedings of the 18th ACM conference on computer supported cooperative work & social computing*, pages 994–1009. ACM.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-learn: Machine learning in python. *Journal of machine learning research*, 12(Oct):2825–2830.

Sasa Petrovic, Miles Osborne, Richard McCreadie, Craig Macdonald, Iadh Ounis, and Luke Shrimpton. 2013. Can twitter replace newswire for breaking news? In *Seventh international AAAI conference on weblogs and social media*.

Takeshi Sakaki, Makoto Okazaki, and Yutaka Matsuo. 2010. Earthquake shakes twitter users: real-time event detection by social sensors. In *Proceedings of the 19th international conference on World wide web*, pages 851–860. ACM.

Jyoti Prakash Singh, Yogesh K Dwivedi, Nripendra P Rana, Abhinav Kumar, and Kawaljeet Kaur Kapoor. 2017. Event classification and location prediction from tweets during disasters. *Annals of Operations Research*, pages 1–21.

Abu Bakr Soliman, Kareem Eissa, and Samhaa R El-Beltagy. 2017. Aravec: A set of arabic word embedding models for use in arabic nlp. *Procedia Computer Science*, 117:256–265.

Zeynep Tufekci and Christopher Wilson. 2012. Social media and the decision to participate in political protest: Observations from tahrir square. *Journal of communication*, 62(2):363–379.

The Design of the SauLTC application for the English-Arabic Learner Translation Corpus

Maha Al-Harthi
Princess Nourah University, Riyadh, Saudi Arabia
Amal Alsaif
Al-Imam Mohammad Ibn Saud Islamic University, Riyadh, Saudi Arabia

Abstract

This paper reports on the development of two important tools designed specifically for the project entitled “the Saudi Learner Translation Corpus” (SauLTC): the conversion tool and the SauLTC application. The challenges encountered during the different stages of the project, especially the stage of the corpus alignment, were highlighted. SauLTC is a POS-tagged and error annotated parallel corpus proposed to be 3 million tokens, including translation projects required for graduation at the College of Languages at the Princess Nourah bint Abdulrahman University in Riyadh. It comprises a multi-version corpus featuring linguistic annotation, complemented with an interface for monolingual or bilingual querying of the data. The corpus can be used to identify the students’ strategies in translation and analyze their patterns of language use. The paper describes the corpus parameters and compilation process, followed by an explanation of how the textual processing and sentence alignment is being conducted. A detailed description of the SauLTC conversion tool and the application will be provided. Potential uses of the corpus will be suggested for research, training and pedagogical purposes.

1 Introduction

The merge of Learner Corpus Research (LCR) and Corpus-Based Translation Studies (CBTS) was inevitable due to their shared characteristics of interlingual mediation (Granger and Lefer, 2018). Both LCR and CBTS involve assessing the impact of transfer; a kind of transfer from L1 for LCR and from the ST for CBTS (Gilquin et al., 2008). Parallel corpora are also used in a variety of NLP and IE systems, dictionary construction and automatic alignment systems.

In line with these developments, the present paper introduces SauLTC, the first unidirectional, multiversion parallel learner corpus in the Arab world. It consists of final year students’ translation projects required for graduation. The structure of the paper is as follows: in the following section, we reviewed the relevant studies in the area of learner translation corpora. Section 3 is devoted to the description of the design and the development of the SauLTC, its structure, participants, corpus compilation and data normalization. The conversion tool developed for the project is described, together with the SauLTC application. Finally, the corpus statistics and general remarks will be provided. The tool and the application could be used for other languages with slight modification of the used POS tagger. Both can be accessed by contacting the authors.

2 Related Work

Corpora that are specifically designed for use in the translation pedagogy have been a significant development. One of the first endeavors is the use of syllabus driven stratified parallel corpora to address specific teaching and learning tasks and train for specialized areas in translation (Tiayon, 2004). Nevertheless, these corpora remained reference corpora that illustrated best case practice. The primary purpose of corpora of learner translations is to provide new possibilities and insights into the translation training process. Before the availability of translation learner corpora, accessing the process of translation and translation training was mainly conducted through think aloud recordings, questionnaires, key-logging and eye-tracking (Göpferich and Jääskeläinen, 2009). While the results of these methods are informative, their collection tends to be limited due to cost and time constraints. Corpora, on the other hand, is relatively easier to compile and access, which encouraged the development of an increasing number of bi- and multilingual parallel learner corpora around the world. To the best of our knowledge, no English- Arabic leaner translator corpus has been developed so far. However, there are other learner translator language pairs. Historically, we can distinguish two main stages in the development of learner translator corpus research. The first stage comprises early endeavors, which were characterized by being smaller in size and publicly unavailable, while the second stage witnessed more recent projects that are greater in size with their own online interfaces available to the research community at large.

One of the earliest learner translation corpora was compiled in Germany by Robert Spence (1998). Another Learner Translator Corpora are PELCRA (Uzar, 2002) and the student Translation Archive (STA) (Bowker and Bennison, 2003). These early attempts to compile collections of electronically stored learner translations primarily aimed at identifying common problems in learner translations. The Russian Translation Learner Corpus or RuTLC (Sosnina, 2006) is another type that consists of English STs and their translations into Russian as the native language. Like the *PELCRA* corpus, it is also error-tagged, allowing automatic analysis of learner errors. It is used to identify the frequency and distribution of error types in order to detect the most frequent lexical, stylistic and grammatical errors in student translations in order to modify and improve teaching strategies and materials. Finally, Multiple Italian Student Translation Corpus (MISTiC) was developed by Castagnoli (2009) for a corpus-based study on explicitation. Multi-parallel and longitudinal analyses are possible, as there are several translations for each ST and each student contributed more than one translation. Although collecting and analyzing the output of trainee translators can be useful for translation teaching, and research started in the above pioneering projects at the end of the 1990s, these corpora remained exclusive and inaccessible to the wider community of researchers. The corpora also varied in the number of languages they include, the directionality of the translation, and the technologies used.

The second stage was marked by the availability of online learner translator corpora projects such as the ENTRAD project in Spain, MeLLANGE LTC in the UK, RusLTC in Russia, and CorTrad in Brazil. ENTRAD (see Flore´n Serrano and Lore´s Sanz, 2008) is a text-level aligned corpus that can be queried by the metadata such as the translator’s age, gender, and mother tongue. Perhaps the most remarkable achievement in translation learner corpus research was the compilation of the MeLLANGE Learner Translator Corpus (LTC), which was completed in 2007 and provides advanced searchable and user-friendly query interfaces that allow the user to perform an extensive search through rich metadata. The query tool also includes error-tagging system based on prior linguistic annotation.

The one-to-many concordances are used for comparative observations (Castagnoli et al., 2011). The corpus is compiled of originals of four different text types (journalistic, administrative, legal and technical) and their translations produced by learners as well as professional translators (to provide reference translations for comparison with student versions: the trainees). MeLLANGE represents a valuable source of data for universal analyses about translation trainees’ performance, due to its availability online and its relatively big size. Like the MeLLANGE LTC, RusLTC (Kunilovskaya and Kutuzov, 2015) is error-tagged and annotated with various metadata about translators and translation situations. It is a multiple learner translation corpus containing English and Russian source texts together with their translations produced by Russian translation trainees. Similarly, CorTrad (Tagnin et al., 2009) is a multiversion English-Brazilian Portuguese corpus that allows a comparison not only between source texts and translations, but also between the different translations of the same source text.

Recent developments in the field of translation learner corpus research include KOPTE (Corpus Project of Translation Evaluation) (Wurm, 2016), UPF LTC (University of Pompeu Fabra learner translator corpus) (Espunya, 2013), and CELTraC (Czech-English Learner Translation Corpus) (Štěpánková, 2014). However, these corpora are not yet made available online.

Sylviane Granger and her team from the Centre for English Corpus Linguistics of Université Catholique de Louvain proposed a new corpus project initiative entitled Multilingual Student

Translation (MUST) in 2016. This corpus can be searchable on a web-based interface, Hypal4MUST, an adapted version of the Hypal tool designed by Obrusnik (2014) for the processing of parallel texts.

The above-mentioned corpora differ mainly in terms of the languages involved, the translation direction, and the techniques/technology employed for corpus creation and querying. Most of these corpora, similar to the present corpus, focus on translations into the students' native language in order to investigate translation related phenomena, whereas the corpora that include student output in the foreign language were considered as a tool for foreign language teaching (Uzar and Waliński, 2001). The principal objective of these projects is to identify common problems and errors in student translations in order to improve teaching materials.

3 SauLTC

SauLTC is a promising project that was initiated to keep up with the latest developments in linguistic research and to make use of the piling archive of the PNU students' translation projects. SauLTC is a multi-version corpus organized in three parallel sub-corpora. The first corpus comprises the English source texts. The second and third corpora include two versions of the translations of one source text: First, the draft translation (Version 1 of TT) is the first attempt of the student to translate the source text on her own. Second, the final translation (Version 2 of TT) is the student's same translation after making the necessary changes based on her instructor's feedback.

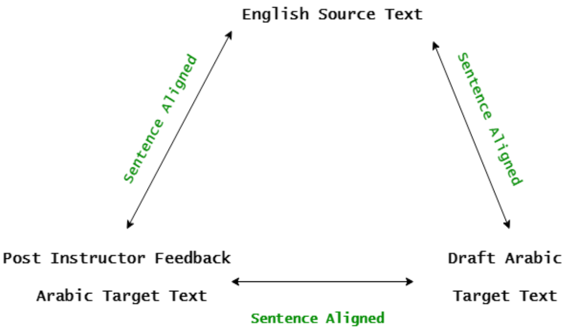


Figure 1: The alignment directions of the SauLTC files

Each student's contribution includes a learner profile, the source text, the draft translation and the post-instructor feedback final submission. All this information is described in the searchable corpus metadata, with all translations and metadata being anonymized. The corpus is sentence aligned across all three versions (see Fig.1). Thus, one of the functionalities available in our corpus allows end users to examine what a trainee translator produces on her own (draft translation) and the effect of an expert translator's feedback (final translation submission).

3.1 Participants

The creation and compilation of the SauLTC involved three types of participants. The first and main type of participant are the students who are Arabic native language speakers. Their explicit participation consent is documented on the profile forms in addition to other background information they provided. The second type of participants are the instructors who provided feedback on the students' drafts and later assessed the students' final submission. The third type of participant is the alignment verifiers, qualified translators, who were later enlisted to double-check the automatic sentence alignment. Each verifier aligned at least three students' projects which entails the double-checking of nine parallelization.

3.2 Corpus Compilation

Currently, there are 186 student participants, 47 instructor participants, and 17 alignment assistants (see section 3.8). As mentioned above, each student's contribution consists of three Microsoft Word files and a learner profile in a Word template. The leaner profile includes detailed information about English language exposure, together with the student's consent form. The source texts are chapters or booklet extractions from extracted fiction, self-help, biography, history, health, psychology, religion, culture, management or science. The source texts are 6000 words on average. All three documents are collected in one folder under the student's name in addition to the student's profile information. We designed naming convention for this first version of the corpus as following: each folder is named 'SauLTC_V1_Seq-No4digits_Year_SsmesterNo', for example, SauLTC_V1_0008_2016_S2; the four Word files in the folder have the same naming that end with one of the following depending on its type (_source, _draft, _final _metadata). We also separately collected metadata information of supervisors and alignment verifiers that were recruited to manually double-check the automatic sentence alignment in online forms. This information include the educational background, professional experience, the consent and work commitment.

3.3 Data Normalization

Due to the large number of illustrations, tables, diagrams and figures that the source and final translated texts included and the various ways that students used to deal with, all texts require a prior stage of normalization to minimize the challenges that the alignment process may face. Some students excluded these illustrations from their draft target texts and subsequently their final target texts. Others translated and recreated them in the target texts. Since these illustrations and tables as well as the strategies followed by students in dealing with them are an integral part of the translation process, we decided to include them in the searchable database. However, the automatic sentence aligner can only handle running text. tables had to be removed and saved separately in order to be manually aligned.

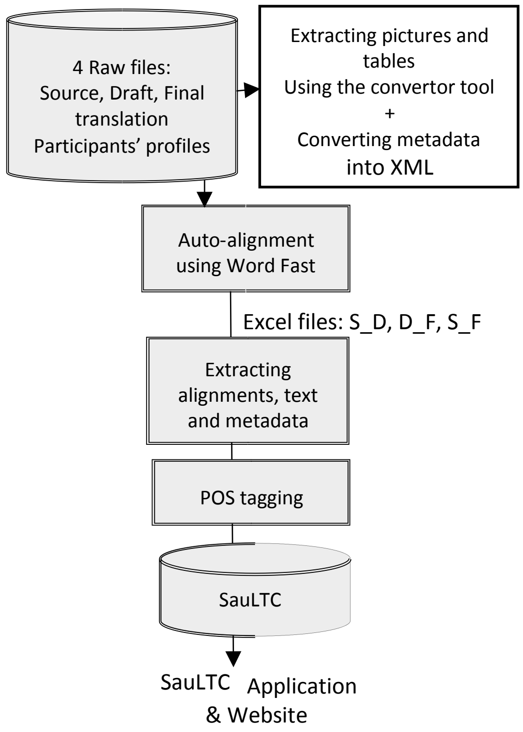

4 Raw files:
Source, Draft, Final translation
Participants' profiles
Extracting pictures and tables
Using the convertor tool
+
Converting metadata into XML
Auto-alignment using Word Fast
Excel files: S_D, D_F, S_F
Extracting alignments, text and metadata
POS tagging
SauLTC
SauLTC Application & Website

Figure 2: The pipeline process of constructing the SauLTC

Due to the large number of student folders, we developed our own tool, SauLTC XML Conversion tool, to extract all these illustrations (See section 3.4). The tool is effective, but it is not able to distinguish automatically in-text illustrations and diagrams from irrelevant paragraph lines, borders and other decorative embellishments. These superfluous additions have to be deleted manually from each student's folder post-extraction. The relevant diagrams and tables are then added to the database automatically to be accessed by the researcher when needed.

3.4 SauLTC XML Conversion tool

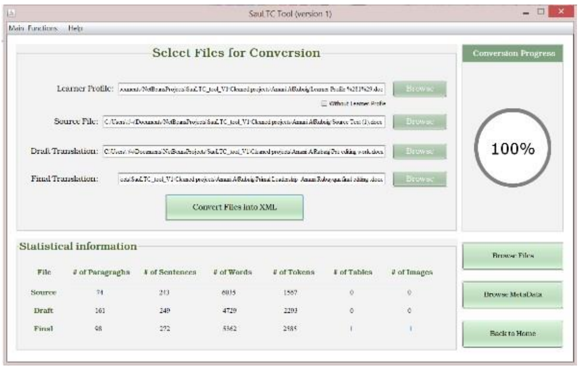
One of the main obstacles of the initial automatic processing of the student folders is the diversity of the translation genres and the formatting. This lack of uniformity led us to develop a converter, SauLTC XML Conversion tool. The tool is able to convert any word text file (English or Arabic) into XML standard format with ability of extracting all figures, tables and formatting shapes separately. Figure 3 shows the main screen of the tool where the user should upload the three Word files: source, draft translation, and final translation with the translation learner metadata. Before the XML conversion, the tool recognizes and removes headers, footers and decorating shapes that student may include in their submission. The tool also clean up the text from any extra spaces. Once the conversion process is complete, the statistical information of the number of paragraphs, sentences, words, unique tokens, tables and images will be shown for all the three files. These statistics could be used to check the quality of the translation and how the student modifies the final version compared to the draft version.

The tool also offers browsing and editing facilities on the extracted text and save the new editing into the XML format. The user also can browse the extraction of metadata and modify any field before converting it into XML format. Due to the inconsistency on filling the earlier metadata form manually by the student, for example, the student may remove some fields or add unwanted information, which make the automatic extraction difficult and need a kind or normalization to ensure that the selected fields are correctly filled. The resulted XML and JPEJ files will be transferred into the next process of the corpus manipulation as in Figure 2.

3.5 Corpus Parallelization

The SauLTC is a sentence aligned bilingual corpus. For more efficiency, our alignment process runs in two stages: automatic alignment (English-Arabic and Arabic-Arabic pairs) and manual verification, as in Figure 2. The Auto-aligner at WordFast Anywhere was utilized for the automatic parallelization of the source text, draft text, and final submission text. WordFast Anywhere is a free web-based set of translation memory products.

The manual verification of the automated aligned files is all handled by the verifiers who should receive at least three student folders; each has the four Word files (source, draft, final translations and the metadata) and follow the instructions in the SauLTC alignment guidelines. We offer a tutorial video to ensure that each verifier had everything explained in a step-by-step format with online assistance by the corpus team. After they complete their double-checking and report comments for any unusual dealings, they fill in an online short form indicating an approximation of the number of hours it took and the number of mis-alignments they found. The parallelized three excel sheets (source_draft, source_final, draft_final) are uploaded to the SauLTC team. These excel sheets are then converted into XML files and used to create the online parallel searchable database automatically (See section 3.7 for more details about the tool).



Learner Profile: ...uments/NetBeansProjects/SauLTC_tool_V1/Cleaned projects/Amani AlRubaiq/Learner Profile %281%29.docx

□ Without Learner Profile

Source File: C:/Users/.../Documents/NetBeansProjects/SauLTC_tool_V1/Cleaned projects/Amani AlRubaiq/Source Text (1).docx

Draft Translation: C:/Users/.../Documents/NetBeansProjects/SauLTC_tool_V1/Cleaned projects/Amani AlRubaiq/Pre editing work.docx

Final Translation: ...cts/SauLTC_tool_V1/Cleaned projects/Amani AlRubaiq/Primal Leadership Amani Rubayqan final editing.docx

File	# of Paragraghs	# of Sentences	# of Words	# of Tokens	# of Tables	# of Images
Source	74	213	6035	1567	0	0
Draft	161	249	4729	2293	0	0
Final	98	272	5362	2585	1	1

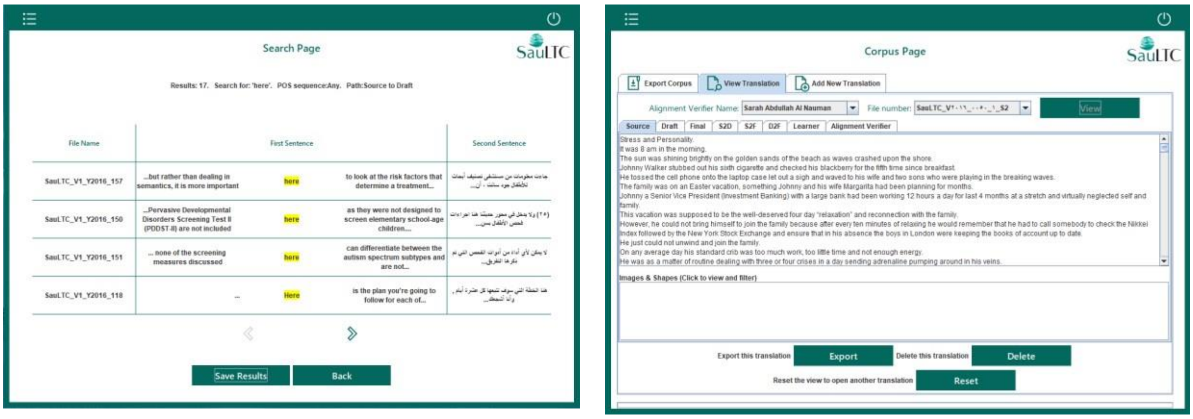
Figure 3: The SauLTC XML Conversion tool main interface

3.6 Part of Speech (PoS) Tagging

To maximize the benefit of using the SauLTC corpus in research, all sentences in the three versions are morphologically tagged using powerful POS automatic taggers for both languages. For the English source texts, the Stanford Automatic Tagger (Toutanova et al., 2003) was used, due to its availability as a powerful open source English tagger. For Arabic, MADAMIRA (Pasha et al., 2014) was used to tag the Arabic texts.

In fact, the two tag sets are not identical which led to a mismatching problem while comparing the word classes in source and target texts in any parallel grammatical investigations. For instance, the user should use the actual POS tags when exploring the corpus in our engine. To overcome this issue, we propose a general tag set to map the two different tag sets: SauLTC General tags. Then, the end user is able to use a specific tag within either the English source files or the Arabic target files using unique tags, while he is able to use the more specific POS tags as well by specifying the language in corresponding files.

The POS tagging is run on sentence level for more accurate tags and this process requires more text processing including tokenization and combining words with corresponding tags. The tags are saved in our database and can be extracted in the XML files using our application and the website.



Results: 17. Search for: 'here'. POS sequence:Any. Path:Source to Draft

File Name	First Sentence			Second Sentence
SauLTC_V1_Y2016_157	...but rather than dealing in semantics, it is more important	here	to look at the risk factors that determine a treatment...	جاءت معلومات من مستشفى تصنيف أبحاث للأطفال جوه ستانث ، أن...
SauLTC_V1_Y2016_150	...Pervasive Developmental Disorders Screening Test II (PDDST-II) are not included	here	as they were not designed to screen elementary school-age children....	(٢٥) ولا يدخل في محور حديثنا هنا اجراءات فحص الأطفال بسن...
SauLTC_V1_Y2016_151	... none of the screening measures discussed	here	can differentiate between the autism spectrum subtypes and are not...	لا يمكن لأي أداة من أدوات الفحص التي تم ذكرها التفريق...
SauLTC_V1_Y2016_118	...	Here	is the plan you're going to follow for each of...	هنا الخطة التي سوف تتبعها كل عشرة أيام ، وأنا أشجعك...

Alignment Verifier Name: Sarah Abdullah Al Nauman

File number: SauLTC_V٢٠١١_٠٠٤٠_١_S2

Stress and Personality.
It was 8 am in the morning.
The sun was shining brightly on the golden sands of the beach as waves crashed upon the shore.
Johnny Walker stubbed out his sixth cigarette and checked his blackberry for the fifth time since breakfast.
He tossed the cell phone onto the laptop case let out a sigh and waved to his wife and two sons who were playing in the breaking waves.
The family was on an Easter vacation, something Johnny and his wife Margarita had been planning for months.
Johnny a Senior Vice President (Investment Banking) with a large bank had been working 12 hours a day for last 4 months at a stretch and virtually neglected self and family.
This vacation was supposed to be the well-deserved four day "relaxation" and reconnection with the family.
However, he could not bring himself to join the family because after every ten minutes of relaxing he would remember that he had to call somebody to check the Nikkei Index followed by the New York Stock Exchange and ensure that in his absence the boys in London were keeping the books of account up to date.
He just could not unwind and join the family.
On any average day his standard crib was too much work, too little time and not enough energy.
He was as a matter of routine dealing with three or four crises in a day sending adrenaline pumping around in his veins.

Images & Shapes (Click to view and filter)
Export this translation
Export
Delete this translation
Delete
Reset the view to open another translation
Reset

Figure 4: Two samples of the SauLTC desktop application for exploring the corpus and extracting words or phrases.

3.7 SauLTC Application

After completing the parallelization process with the manual verification of each excel alignment file, an automatic extracting desktop Java application has been developed to deal with these files (source_draft, source_final, and draft_final). For each alignment file, the application extracts the text of sentences, tokenizes the words, and extracts POS tags (Arabic and English) along with the general POS tag. We designed a comprehensive database to store all information required and to simplify and fasten the searching and corpus extraction processes. The user may only upload the excel files, the application will continue the remaining process automatically.

In the following section, we list the main features of the SauLTC application:

Importing additional translated files: This feature is to add more translations into the corpus. The application requests all aligned excel files (source-draft, source-final, draft-final) that are produced by WordFast and verified by the verifiers and the folder that has the images and table files. The user assigns a translation learner name and the verifier name from a predefined list. If the names are not listed, the user should add the new names in the editing participants tab.

The text in all files are extracted into the database and segmented further into words and saved with the automatic POS tag using Stanford for English and MADAMIERA for Arabic automatically. Any other figures and illustrations are also saved in the database.

Learner interface: This feature is to add, delete, and edit the learner metadata. Only the user who should be the administrator of the corpus can edit any information of the learner: name, age, demographic information, educational background, translation experience, the use of reference material and so on.

Supervisor interface: This feature is to add, delete, and edit supervisor metadata. The user who should be the administrator of the corpus can edit any information of the supervisor: name, educational background, years of experience in teaching translation, and years of experience in supervising translation projects.

Verifier interface: This feature is to add, delete, and edit alignment verifier metadata. The user who should be the administrator of the corpus can edit any information of the verifier: name, age, translation experience and educational level.

Exporting the corpus: This feature is used to extract the whole corpus or a couple of files that belongs to a specific verifier or learner. The extraction will be in seven XML files: the text tokenized with POS tags and the SauLTC general tags of all the source, draft, and final translation, the metadata of the participants (learner, supervisor, and verifier), source-draft alignment, source-final alignment, draft-final alignment files. All these files share the same naming convention as well as the folder name (i.e. SauLTC_V1_0008_2016_S2).

Exploring the corpus: This is the most powerful feature of our application. The user is able to search according to single or multiple criteria at the same time. First, the user has to select the alignment path (source-draft, draft-final, source-final), and enters a word or a phrase s/he wants to search for. He could specify the POS tags either language specific or the SauLTC tags, in addition to any information from the metadata of the learner, the supervisor and the alignment verifier. Figure 4 presents screenshots of exploring the corpus with all metadata, and how the application process enquires of words or phrases. The user may use regular expressions in the search box along with POS tags. The resulting table could be exported in CVS format for more portability.

The Basic SauLTC Statistics	**Total**	
Number of Alignment Verifiers	17	
Number of Learners	209	
Number of Supervisor/Teachers	47	
The SauLTC corpus - Version 1		Avg per translation
Distinct Translation Instances	115	-
Total Files (source, draft, final)	345	
English Sentences in source	36,518	318
Arabic Sentences in draft	32,196	280
Arabic Sentences in final	32,468	282
Total Sentences	101,182	880
Number of English Words	610,370	5,308
Number of Arabic Words	1081,746	9,406
Words in draft	536,177	4,662
Words in final	545,569	4,744
Total Words	1,692,116	14,714
Total Images	1,014	9
The sentence-paralizations		
Source to Draft Translations	30,421	265
Source to Final Translations	30,575	266
Draft to Final Translations	29,628	258

Table 1: The SauLTC Statistics of the first version

3.8 Corpus Statistics

The first version of the SauLTC corpus has 30,421 sentence-parallelization in source_draft, 30,575 sentence-parallelization in source_final, and 29,628 sentence-parallelization in draft_final alignment of only 115 translations in this version. The total number of tokens is 1,692,116, with an average of 14,714 tokens per file, with all corresponding tags: Stanford tags for English words and MADAMIRA tags for Arabic words, in addition to the general tags for both. The total number of translation students in the whole project is 186 who were under the supervision of 47 instructors. The alignment verification is carried out by at least 17 verifiers, participating in the project.

While we have 36,518 English sentences in source files, the aligned sentences are only 30,421 in source_draft alignment, which indicates that there is no one-to-one sentence-parallelization when the students translate the text.

For instance, there are 53 sentences on average in the source file that were merged or deleted when translated into Arabic draft and 52 sentences on average for source_final alignment.

In terms of words, similarly there is around 645 words in English were omitted when translated into Arabic draft, and around omitted 563 words when verified by the supervisors in the final version. In fact, these findings support the claim that Arabic has a richer semantic lexical system than English does, where one Arabic word may be translated into a phrase or multiple words in English to express the same meaning. In addition, the morphological structure in Arabic allows constructing a complete meaningful sentence in one token such as (سنكتبها/we will write it down). There is no significant difference between the number of sentences and words in draft and final versions, both are Arabic. The learner tended to make fewer changes in the final version, following the supervisor’s comments; the sentence average in the final translation was decreased by only 22 sentences compared to the draft version. The verifiers provide any significant remarks and comments during the alignment process to assist the researchers to track the changes in the translations.

4 Conclusion

This paper introduces the first version of the SauLTC, together with some of the tools developed specifically for this corpus: SauLTC XML Conversion tool designed to convert word text files into XML standard format, and SauLTC desktop application which is an automatic extraction tool developed to deal with the alignment excel sheets with automatic POS tagging. SauLTC represents the first learner translator parallel corpus for an English Arabic language pair. It is also one of the first corpora to provide parallelization of pre-edited and post-edited versions of trainee translations. This paper describes the challenges encountered at some of the compilation stages. The most prominent challenge was in the process of text alignment, due to the huge differences in the punctuation mark systems between the English source texts and their Arabic translations in terms of their segmentations, which in turn made automatic alignment imprecise. The practical solution was to hire assistants to manually verify and double check the alignment of sentences between the three documents of the same text. The launching of a website for the corpus and making it available online will be the following stage in the project. This will provide researchers the opportunity of exploring SauLTC with multiple selections of criteria such as extracting specific words or phrases with optional morphological features in translated texts or parallel texts, tracking the errors, investigating strategies followed by translation learners while translating multi-word units and obtaining some statistics of any searchable component. All the features included in the SauLTC application, in addition to some other features will be available in the website.

The corpus was designed to enable researchers to examine the translation process both quantitatively and qualitatively. It is a valuable resource for automatic processing of bilingual text. Translation instructors and translation students and trainees can utilize the corpus for a more data-driven approach to learning and training. Overall, the potential applications of an English-Arabic learner translation corpus are numerous and valuable for research, training purposes of automatic NLP systems such as machine translation, word alignment systems and dictionary construction.

Acknowledgments

This paper is part of a project that was funded by the Deanship of Scientific Research at Princess Nourah bint Abdulrahman University. We are grateful to all the participants who helped throughout the different stages of the project: the students, the instructors and the alignment verifiers. Special thanks goes also to all the research assistants and the team of the project programmers. We are very grateful to the anonymous reviewers for their very insightful comments and suggestions that have greatly improved this paper.

References

Adam Obrusník. 2014. Hypal: A User-Friendly Tool for Automatic Parallel Text Alignment and Error Tagging. Eleventh International Conference Teaching and Language Corpora, Lancaster, pp. 67-69.

Andrea Wurm. 2016. Presentation of the KOPTE Corpus and Research Project. https://www.academia.edu/24012369/Presentation_of_the_KOPTE_Corpus_and_Research_Project.

Anna Espunya. 2014. The UPF learner translation corpus as a resource for translator training. Language Resources and Evaluation 48(1): 33-43.

Arfath Pasha, Mohamed Al-Badrashiny, Mona Diab, Ahmed El Kholy, Ramy Eskander, Nizar Habash, Manoj Pooleery, Owen Rambow, and Ryan M. Roth. 2014. MADAMIRA: A Fast, Comprehensive Tool for Morphological Analysis and Disambiguation of Arabic. Proceedings of the 9th Language Resources and Evaluation Conference (LREC'14): 1094-1101

Charles Tiayon. 2004. “Corpora in translation teaching and learning”. Language Matters, 35 (1): 119-132.

Ekaterina Sosnina. 2006. Development and application of Russian translation learner corpus. In *Proceedings of corpus linguistics—2006, St. Petersburg, Russia:* 365–373.

Gaëtanelle Gilquin, Szilvia Papp, and María B. Díez-Bedmar (eds.). 2008. Linking up Contrastive and Learner Corpus Research. Amsterdam/Atlanta: Rodopi.

Kelly Washbourne. 2015. Learning to Fail: Unsuccessful Translations as Pedagogical Resource. Current Trends in Translation Teaching and Learning E, 2: 285–320.

Kirsten Malmkjær. 2017. The Routledge Handbook of Translation Studies and Linguistics. Abingdon & New York: Routledge.

Kristina Toutanova, Dan Klein, Christopher D. Manning, and Yoram Singer. 2003. Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency

Network. In Proceedings of HLT-NAACL, 252-259.

Kristýna Štěpánková. 2014. Learner Translation Corpus: CELTraC (Czech-English Learner Translation Corpus). Bachelor's Diploma Thesis.

Lynne Bowker and Peter Bennison. 2003. Student Translation Archive: Design, development and application. In F. Zanettin, S. Bernardini & D. Stewart (eds.) Corpora in Translator Education. London & New York: Routledge, 103-117.

Malcolm Williams. 1989. The Assessment of Professional Translation Quality: Creating Credibility out of Chaos. In TTR (Traduction, terminologie, Rédaction) 2 (2): 13-33.

María C. F. Serrano. 2006. ENTRAD, an English Spanish parallel corpus created for the teaching of translation. Paper presented at the 7th Teaching and Language Corpora Conference (TALC 2006).

Maria Kunilovskaya and Andrey Kutuzov. 2015. A quantitative study of translational Russian (based on a translational learner corpus). In Corpus Linguistics 2015. Proceedings of the 7th International Conference: 33-40.

Natalie Kübler. 2008. A comparable Learner Translator Corpus: Creation and use. LREC 2008 Workshop on Comparable Corpora: 73-78.

Rafal Uzar and Jacek T. Waliński. 2001. Analysing the fluency of translators. International journal of corpus linguistics, 6: 155-166.

Rafal Uzar. 2002. A corpus methodology for analysing translation. In Tagnin, S.E.O. (Ed.), Cadernos de Tradução: Corpora e Tradução, 1(9): 235–263.

Robert Spence. 1998. "A corpus of student L1-L2 translation". In Granger S. and Hung J. (eds.). Proceedings of the First International Symposium on Computer Learner Corpora, Second Language Acquisition and Foreign Language Teaching, 110-112.

Sara Castagnoli, Dragos Ciobanu, Kerstin Kunz, Natalie Kübler, and Alexandra Volanschi. 2011. "Designing a learner translator corpus for training purposes." Corpora, Language, Teaching, and Resources: From Theory to Practice 12: 221-248.

Sara Castagnoli. 2009. Regularities and variations in learner translations: A corpus-based study of conjunctive explicitation. PhD Dissertation, University of Pisa.

Stella E. O. Tagnin. 2014. The CoMET Project: Corpora for Teaching and Translation. In T. Sardinha & T. Ferreira (Eds.), Working with Portuguese Corpora, 201-214. Bloomsbury.

Stella E. O. Tagnin, Elisa Duarte Teixeira, Diana Santos. 2009. CorTrad: a multiversion translation corpus for the Portuguese-English pair., Teixeira, E., and Santos, D. (2009). CorTrad: a multiversion translation corpus for the Portuguese-English pair., Teixeira, E., and Santos, D. (2009). CorTrad: a multiversion translation corpus for the Portuguese-English pair. 28th. conference on Lexis and Grammar, Bergen.

Susanne Göpferich and Riitta Jääskeläinen. 2009. Process research into the development of translation competence: Where are we, where do we need to go? Across Languages and Cultures, 10 (2): 169-191.

Sylviane Granger and Marie-Aude Lefer. 2018. MUST: A collaborative corpus collection initiative for translation teaching and research. CECL Papers: 72-73.

Distance-Based Authorship Verification Across Modern Standard Arabic Genres

Hossam Ahmed
Leiden University Institute for Area Studies
Witte Singel 25, 2311 BZ, Leiden, The Netherlands
h.i.a.a.ahmed@hum.leidenuniv.nl

Abstract

Authorship Verification (AV) is a type of stylometric analysis that addresses an authorship problem where, given a document of unknown origin and a set of documents written by a known author, the task is to identify whether the document is indeed written by that author. Previous research uses a number of techniques to address this problem. Most successful techniques in Classical Arabic as well as other languages use an SVM method supported by a distance measure in vector space and a distance/similarity threshold for accepting the document as authentic. While Arabic Authorship Attribution (where the task is to attribute the question document to one of several candidates) surveys and evaluates the usability of different distance measures, this paper is the first to provide such overview for Modern Arabic AV. Using a corpus of short texts from five common Modern Standard Arabic genres, this paper evaluates four common distance measures (Canberra, Manhattan, Cosine, and Jaccard) with a number of lexical, syntactic, and morphological features. The results show that Canberra Distance is a best performing distance measure in most genres, with an accuracy rate of up to 97.8%, well over highest known baseline.

1 Introduction

This paper compares the accuracy of Authorship Verification (AV) in five Modern Standard Arabic (MSA) genres using four popular distance measures: Manhattan Distance, Canberra Distance, Cosine Distance, and Jaccard distance. The genres in question are fiction and non-fiction books, and articles on economics, politics, and newspaper columns.

Authorship Verification (AV) is a type of authorship analysis problem that addresses the question of whether a question document is written by a known author, given a corpus of authentic documents known to be written by that author. AV is often compared to Authorship Attribution (AA), where there is a set of known candidate authors, and the task is to determine which one of them is the author of the question document. Both AV and AA are relevant in the areas of corpus linguistics, stylistic and literary analysis, Digital Humanities, and forensic linguistics.

This paper is organized as follows: section 2 gives an overview of relevant literature and outlines the research question. Section 3 describes the corpus used and feature extraction. Section 4 outlines the Authorship Verification method and distance measured used in the experiments. The results are described and discussed in sections 5 and 6.

2 Related Work

When approaching AA and AV as Machine Learning (ML) tasks, AV differs essentially from AA in that the former involves only positive training data (a corpus known to be written by just one author). AA, on the other hand, involves a set of documents for each of the candidate authors. It can be argued that an AA task is easier, in the sense that all is needed is to determine which corpus is most similar to the question document. In AV, the alternative corpus is virtually that of any other author.

2.1 Arabic Authorship Attribution

AA is often approached as a classification problem. Literature on AA is extensive. For Arabic ML-based research, there has been much progress. Abbasi & Chen (2005a, 2005b) use an elaborate combination of C4.5 and SVM classifiers, combined with an ensemble of linguistic and non-linguistic features to analyze web and forum authorship. They find that SVM outperforms decision trees in AA, reaching accuracy of 94% for Arabic. SVM has also been used with a number of features in other AA contexts with success. Ouamour & Sayoud (2013) achieve 80% accuracy using Rare Words as the SVM feature of choice. Howedi & Mohd (2014) show that a small training set can render high AA accuracy using Naïve Bayes Bag of Words (96.67%) and SVM using character or word tetragram (93.33%). For modern Arabic, Altakrori et al. (2018) investigate AA in Twitter posts using an array of n-gram character, word, and sentence features, to be used with a number of ML algorithms (Naïve Bayes, SVM, Decision Trees, and Random Forests). For tweets, Random Forests seems to outperform other approaches.

2.2 Arabic Authorship Verification

AV tasks are often seen as AA tasks with the added complication that there is only one author to consider. A reasonable and successful approach in AA is to build a profile of certain features for each of the given authors and the question document, compare the profile of the question document to each of the author profiles, and make a decision using any of the approaches outlined in the previous section. This approach is not immediately accessible to AV because there is only one available dataset to create a profile; that of a single author. Two main approaches emerged to overcome this obstacle. The Imposters Method supplements the training data with a corpus of distractors, text known to be written by other authors, converting the task to an AA problem, and using familiar AA techniques. An example of such approach in Arabic can be seen in Arabic Twitter posts (Altakrori et al., 2018) where the stated context of the approach is law enforcement, where the authenticity of a tweet is needed as evidence. The authors frame the problem as attributing a question tweet to one of a number of suspects. This method suffers from two main drawbacks. First, using the Imposters method incurs additional computational cost as multiple profiles will be created. Second, the quality of AV prediction relies, at least partially, on the selection of the supplementary corpus. The perpetrator in the tweets example (ibid) may not be one of the usual suspects – the attribution problem will then return the suspect with the closes style out of the group provided. Similar issues arise in literary analysis contexts.

The Author Profiling method aims at avoiding the problems that arise with the Imposters method. Within this method, features are extracted from the known corpus and suspect document to create author profiles. If the two profiles match or are sufficiently similar, the document is deemed authentic, otherwise it is judged to be written by another author. Determining similarity and deciding on the threshold for acceptance are key questions in this approach. In languages other than Arabic, Halvani, Winter, & Pflug (2016) use an ensemble of n-gram features over 5 European languages using SVM-calculated distance metric based on Manhattan Distance (Burrows, 2002). They determine the similarity threshold of acceptance (θ) through Equal Error Rate (EER), a point where false negatives and false positives in the training data are equal. They achieve accuracy rates in the mid-70% range, depending on the language tested. It can be seen in this example that negative training data is still needed. EER is also used with a distance based metric based on compression models rather than linguistic features (Halvani et al., 2017), also relying on negative training data to determine θ, with remarkable improvement in processing time, yet slightly lower accuracy than best-performing approaches. Jankowska, Milios, & Kešelj (2014) define θ in terms of the maximum dissimilarity within the training set, completely dispensing with negative data in training. Using common character n-grams and Nearest Neighbor technique, this technique achieves accuracies in the high 80% when applied to the English, Spanish, and Greek datasets from PAN-2013 (Stamatatos et al., 2014). Benzebouchi et. al (2018) use word embeddings and a voting system between SVM and NN techniques to produce high-accuracy AV.

There is little research on Arabic AV. In Classical Arabic, Ahmed (2018) uses an author profiling technique, a similarity metric based on Burrows (2002), and defines θ in terms of simple Gaussian technique to show that stem bigrams offer best accuracy performance (87%) for Classical Arabic. There is no research that deals with MSA. Furthermore, it is not immediately clear if the similarity metric (based on Manhattan Distance) is also optimal in non-literary genres. A comparison of the effectiveness of different distance measures is not available for Arabic AV.

2.3 Research Question

The research outlined above indicate that the careful choice of classifier and relevant feature sets contributes to better AA and AV accuracy. Genre distance metric also seem to play a role in AA. García-Barrero, Feria, & Turell (2013) show that AA accuracy is sensitive to genre, even in closely related genres (literary criticism and short stories). Ouamour & Sayoud (2018) conduct a broad survey of distances and feature sets used in Arabic AA, showing that Manhattan centroid gives highest average accuracy in Arabic AA. The effect of distance or genre has not been studied in Arabic AV.

This is the first study to look at the effect of distance measures and feature selection in modern Arabic Authorship Verification. This paper addresses the following questions:

1. Does feature selection affect the accuracy of AV across MSA genres?
2. Does distance measure selection affect the accuracy of AV across MSA genres?

Depending on the feature set under investigation, the first question addresses lexical, grammatical, and stylistic characteristics of an individual writer, but also of the genre under discussion. The second question addresses the role of feature frequency in the success of AV in Arabic.

To answer these questions, this paper reports the results of a number of experiments examining the accuracy of distance-based AV in modern Arabic in five genres: opinion columns, economics, politics, fiction and nonfiction. The paper compares the accuracy of best performing features in the survey conducted for AA by Ouamour & Sayoud (2018): Manhattan Distance, Canberra Distance, Jaccard Distance, and cosine similarity. The feature set and similarity threshold θ used in this paper are similar to those used by Ahmed (2017, 2018), as they report highest accuracies for Classical Arabic AV. Specifically, this paper will use n-grams of tokens, stems, trilateral roots, and part-of-speech tags.

3 Corpus used

A total of 125 documents from five common genres in Modern Standard Arabic are selected as follows. Five authors are selected from each genre. For each author, five documents are collected.

Author	**Source**
Fiction	Hindawi Foundation book repository www.hindawi.org
Ali Al-Jaarim	
Abdul Aziz Baraka Sakin	
Nicola Haddaad	
Nawaal Al-Saadaawi	
Georgi Zidaan	
Non-fiction	
Abbas Al-Aqqaad	
Ismail Mazhar	
Salama Moussa	
Fouad Zakareyya	
Zaki Naguib Mahmoud	
Economics	www.almasryalyoum.com
Musbah Qutb	
Mohammed Abd Elaal	
Bissan Kassab	www.madamasr.com
Waad Ahmed	
Yumn Hamaqi	www.ik.ahram.org.eg
Politics	www.dw.com
Alaa Al-Aswani	
Wael Al-Semari	www.youm7.com
Danadarawy Al-Hawari	
Belal Fadl	www.alaraby.co.uk
Salma Hussein	www.shorouknews.com
Opinion columns	www.shorouknews.com
Ashraf Al-Barbari	
Emad Eldin Hussein	
Fatima Ramadan	
Mostafa Kamel El Sayyed	
Sara Khorshid	

Table 1: Corpus used.

Table 1 lists the authors and source of the documents used for the corpus. Whenever possible, authors and texts are selected from similar backgrounds e.g. Egyptian writers or Egyptian web sites, to minimize the effect of language variation across dialects. The corpus is collected from same source for each genre whenever possible to minimize any potential editorial effect.

Domain	Avg. size
Opinion columns	746
Economics	765
Fiction	1010
Nonfiction	1001
Politics	760

Table 2: Average document length (tokens).

3.1 Preprocessing and feature extraction

For Economics, politics, and opinion columns, documents are downloaded as text-only (UTF-8) documents. Titles, by-lines, and other front matter are removed. For fiction and non-fiction, documents collected are entire books in e-book (epub) format. They are converted to plain text (UTF-8), then sampled by using about 1100 words from the middle of the book using regular expressions delimited by space, to avoid material that may be repeated verbatim for a given author (front matter, acknowledgement, repeated preface, dedication, etc.). Punctuation and non-Arabic characters are removed. Table 2 shows average document length per genre after pre-processing.

The feature token is taken to represent Arabic words, and is defined as a sequence of Arabic characters separated by white space (note that non-Arabic characters, digits, and punctuation marks have been removed in preprocessing). The pre-processed text is passed through MADAMIRA version 2.1 (Pasha et al., 2014) with standard settings. Part-of-speech (POS) tags and word stems are then extracted from the analysis produced by MADAMIRA. Roots are extracted from the plain-text corpus using ISRI Stemmer in NLTK (Bird et al., 2009). Table 3 shows an example of features extracted from the pre-processed word 'المؤلفين'.

4 Verification method

This section outlines the verification method of the experiment.

AV problem: the authorship problem is defined as p(D$_u$, D$_A$)→{1, 0} where D$_u$ is a document of questionable attribution to an author *A*, and D$_A$ = {D$_{A,1}$, D$_{A,2}$,…} is the set of documents of known attribution to *A*. As this is an AV, rather than AA, problem, D$_A$ is of a single author, and there is only one set per problem. The AV procedure should return 1 if D$_u$ is written by *A* and 0 if not. No 'unknown' response is allowed.

Preprocessed word	المؤلفين
Token	المؤلفين
Root	ألف
Stem	مؤلف
POS tag	noun

Table 3: Example of features extracted from an input word.

Data representation: simplifying the problem, all the known documents in D$_A$ are concatenated to create a single document.

Feature engineering: D$_A$ is a document with sequence of tokens, roots, POS tags, or stems produced by preprocessing. N-grams of relevant features are created, where n ∈ {1, 2, 3, 4}. The known and question documents are vectorized over term frequencies of the relevant feature n-grams using Scikit-Learn (Pedregosa et al., 2011).

Computing distance metrics: Four distance metrics are calculated between D$_u$ and D$_A$. based on Ouamour & Sayoud (2018) the four distance measures are Manhattan Distance, Canberra Distance, Cosine Distance, and Jaccard distance. Stamatatos Distance is not implemented, as it performs consistently poorly in their survey.

Manhattan Distance: for unknown document D_u, known corpus D_A, and normalized frequency of feature f n-gram, Manhattan Distance is calculated as:

$$Man(D_u, D_A) = \sqrt{\sum_{f=1}^{n} |D_{u,f} - D_{A,f}|}$$

Canberra Distance is calculated as

$$Can(D_u, D_A) = \sum_{f=1}^{n} \frac{|D_{u,f} - D_{A,f}|}{|D_{u,f}| - |D_{A,f}|}$$

Cosine Distance is defined as

$$CosDist(D_u, D_A) = \frac{D_{u,f} . D_{A,f}}{||D_{u,f}||_2 - ||D_{A,f}||_2}$$

Jaccard Distance is defined as

$$Jacc(D_u, D_A) = \frac{|D_{u,f} \cap D_{A,f}|}{|D_{u,f} \cup D_{A,f}|}$$

Threshold determination: The training phase of this method is comprised of calculating a similarity threshold θ above which D_u is considered authentic. following Ahmed (, 2017), the acceptance threshold θ is dynamically calculated for each D$_A$ = {D$_{A,1}$, D$_{A,2}$,…} by calculating the distance for each known document *k* and the rest of the known documents:

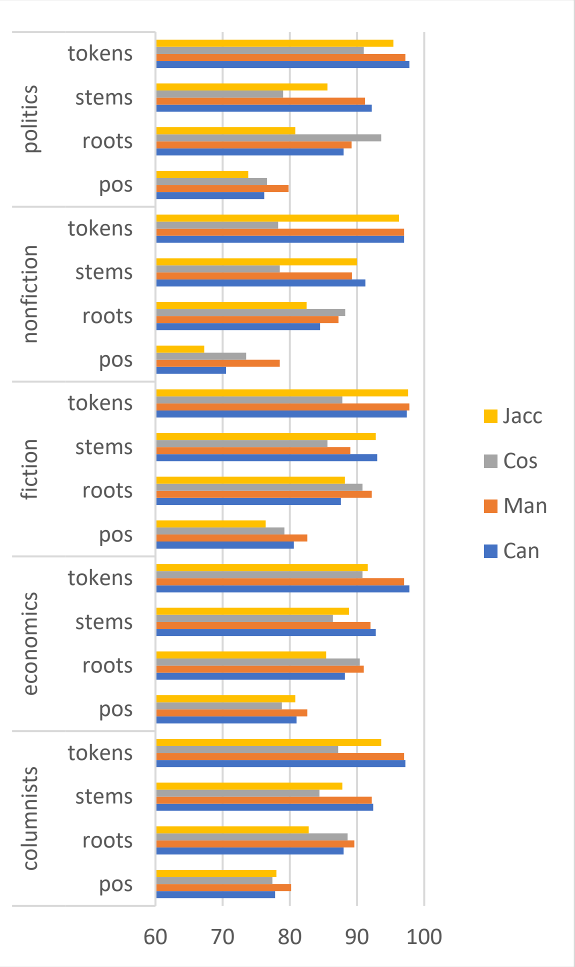


Figure 1: Distance accuracies per genre (unigrams).

$$Dist_k = Dist(D_k, D_{A-k})$$

θ is then defined as the lower bound of the confidence interval of the values of all members of D_A at p = 0.005.

Verification: The testing phase consists of calculating the distance for each document in a given genre against the known corpus for each author. Training and testing data come from the same genre. The document is considered unauthentic if distance $Dist(D_u, D_A) > \theta$ and authentic otherwise.

Evaluation and Baseline: Evaluation of the results is done through the leave-one-out method. Accuracy is defined as follows:

$$accuracy = \frac{Correct\ predictions}{Total\ predictions}$$

The baseline accuracy for this experiment is that used by Ahmed (, 2018) using Manhattan Distance in Classical Arabic and the same θ used in this paper. The best performing feature ensemble for the baseline is stem bigrams.

Domain	**Distance measure**	**Accuracy**
Opinion columns	Canberra	97.2%
Economics	Canberra	97.8%
Fiction	Manhattan	97.8%
Nonfiction	Manhattan, Canberra	97%
Politics	Canberra	97.8%
Baseline		**87.1%**

Table 4: Best performing feature/distance measure per domain.

5 Results

The testing method returned results for all genres that are consistently and considerably above the baseline reported for Classical Arabic. For all genres, the best performing feature is token unigrams, with accuracy ≥ 97%, albeit with some variation in the winning distance measure. Table 4 shows the best performing distance measure per genre.

Figure 1 shows distance accuracies per feature unigram over genres. The figure shows that in four out of the five genres, Canberra Distance is the best performing distance measure to be used with the tested method, with Manhattan Distance coming at a close second. Cosine distance and Jaccard distance perform considerably less accurately, although their best performance is still consistently higher than the baseline.

Another finding of the experiments is that higher n-gram feature assemblies perform worse than their unigram counterparts to varying degrees. Figure 2 compares distance measure accuracies across various n-grams. It shows that for unigrams, the distance measures perform at higher 90% accuracies, while for n = 2 – 4, accuracies drop to mid- and low-80%.

6 Discussion

The overall trend of the results – as far as the research question of this paper is concerned – is expected. AV accuracy is sensitive to frequencies across genres. Overall, distance measures that are least sensitive to frequency (Jaccard distance and cosine distance) underperform compared to those which incorporate frequency (Canberra,

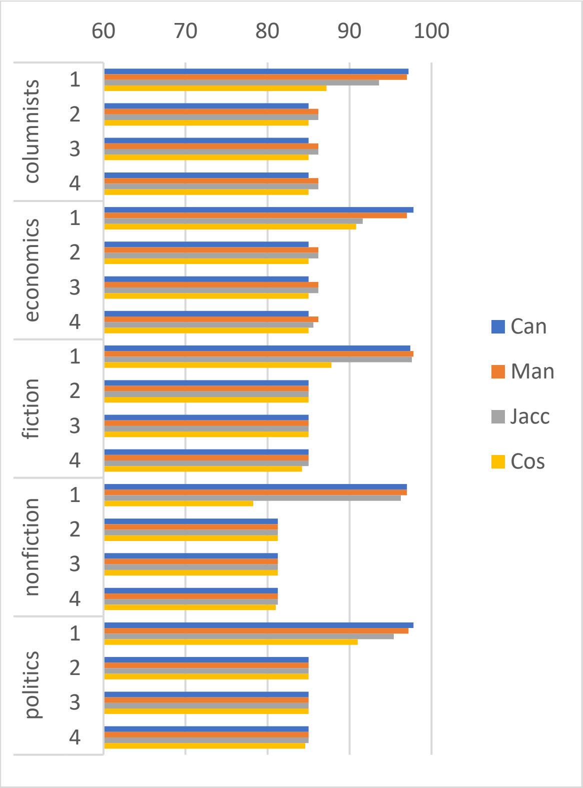


Figure 2: Distance n-gram accuracies per genre.

Manhattan). The slightly improved accuracies delivered by Canberra Distance over Manhattan Distance across all five genres reflects the value of weighing the less common terms (in this case tokens), as Canberra Distance is more sensitive to vectorized values of smaller values than Manhattan Distance.

Some of the unexpected results pertain to the best performing feature, and the improvement of accuracies in this experiment over best known baseline in Classical Arabic. Best performance in this experiment is at least 10% higher accuracy than reported by Ahmed (, 2018). Using the same feature ensemble and distance measures reported for best results in that reference (stem bigrams and Manhattan-distance based similarity) renders accuracies slightly lower than the Classical Arabic data (80% - 85% MSA, depending on genre, compared to 87.1%, and token unigrams are 20% less accurate than results reported here). This difference might be attributed to stylistic variation, change in language convention (higher reliance on loan-words or some similar lexical factor that MSA uses to allow writers to distinguish themselves, while in CA innovation might be said to be at a deeper lexical level). Still, the difference in best-performance is very high. It can be explained in terms of size effects. The Author (Ahmed, 2018) notes that the size of the documents used is very large, and that there is no gain in performance after using more than 1% of the corpus used, and alludes that using even smaller corpora might help improve predictions. While CA texts are volumes in size, the texts used in this experiment are less than 1100 tokens long. Another possibility is the difference in calculating the distance. Using Manhattan Distance and Canberra Distance in their raw form in this experiment causes a tighter cluster, smaller distances, than used to generate the baseline (through the square root or division over separate frequencies). The baseline uses 'delta;' a distance measure based on Manhattan Distance, but does not take the square root (Ahmed, 2017; Burrows, 2002). This means that during the training phase, known documents will generate similarity values that are more spread over the vector space, and a less tight confidence interval for calculating θ.

A related point of difference to existing literature is that the best performing feature in this experiment is Canberra Distance, which ranked low in Classical Arabic AA survey (Ouamour & Sayoud, 2018). This difference can be an additional indicator that MSA differs stylistically from Classical Arabic, note that the discussion above for Arabic AV also compares this work to Classical Arabic. It could also be related to the different nature of the task (AA vs. AV).

Another unexpected finding is consistency across genres. One would expect that authors in different genres would differentiate themselves differently. For example, a genre like novels (fiction) or opinion columns would be expected to give authors more latitude to differentiate themselves by using more varied phrase structures than, say, economics. This in theory would reflect in better differentiation through features such as POS n-grams. However, this does not seem to be the case, and lexical selection is consistently the differentiating factor across the five genres under discussion. On the other hand, this is good news on the computational side; a simple Bag-of-Words, minimal preprocessing, and a simple similarity metric will yield excellent results in efficient computation time.

The superior performance of token unigrams raises a number of questions. The first issue is related to genre characteristics. In genres such as economics, politics, and opinion columns, it is likely that texts go to post-editing prior to publication, ant this could affect certain features more than others. If an editor is more likely to change sentence phrasing and grammatical ‘errors’ than alter word choice, purely lexical features (tokens) would be a better reflection of the author’s style than the stylesheet of the publisher. This, however, does not seem to be the case in the current experiment. Token unigrams are also the most effective feature ensemble in fiction and non-fiction, where post-editing is not expected. In opinion columns, the whole corpus is extracted from a single source, potentially reducing or neutralizing any possible effects of post-editing. Token unigrams are still the most effective feature ensemble.

The second question related to the higher performance of token unigrams comes from the nature of feature extraction. POS tags and stem features are extracted using MADAMIRA with standard settings and roots are generated using ISRI. MADAMIRA is reported to have 95.9% accuracy in POS tagging and 96.0% for stemming (Pasha et al., 2014) while ISRI reports recall and precision values of less than 48% (Taghva et al., 2005). Whether the development of better morphological analyzers could indeed reveal that the value of token unigrams in AV is overstated is an empirical question that I leave for future research.

7 Conclusion

In this paper I have shown that distance measures that are sensitive to term frequency deliver higher accuracies in AV tasks in MSA across five common genres. I have also shown that a simple BoW technique together with a simple non-negative-evidence algorithm that uses Canberra Distance to determine AV can deliver very high accuracies with minimum pre-processing.

Future research should focus on cross-domain AV. Would the same method and distance measures perform with the same behavior if the training set comes from a domain and the test document from another? The fact that tokens are the key features might affect that outcome. On the other hand, as Canberra Distance is weighted to be more sensitive to less common vectors, it may be likely that domain-specific tokens be not so influential as to affect the AV task. I leave this question to future research.

References

Ahmed Abbasi and Hsinchun Chen. 2005a. Applying Authorship Analysis to Arabic Web Content. In Paul Kantor, Gheorghe Muresan, Fred Roberts, DanielD. Zeng, Fei-Yue Wang, Hsinchun Chen, and RalphC. Merkle, editors, *Intelligence and Security Informatics SE - 15*, volume 3495 of *Lecture Notes in Computer Science*, pages 183–197. Springer Berlin Heidelberg.

Ahmed Abbasi and Hsinchun Chen. 2005b. Applying authorship analysis to extremist-group Web forum messages. *IEEE Intelligent Systems*, 20(5):67–75.

Hossam Ahmed. 2017. Dynamic Similarity Threshold in Authorship Verification: Evidence from Classical Arabic. *Procedia Computer Science*, 117:145–152.

Hossam Ahmed. 2018. The Role of Linguistic Feature Categories in Authorship Verification. *Procedia Computer Science*, 142:214–221.

Malik H. Altakrori, Benjamin C. M. Fung, Steven H. H. Ding, Abdallah Tubaishat, and Farkhund Iqbal. 2018. Arabic Authorship Attribution: An Extensive Study on Twitter Posts. *ACM Trans. Asian Low-Resour. Lang. Inf. Process*, 18(1):51.

Nacer Eddine Benzebouchi, Nabiha Azizi, Monther Aldwairi, and Nadir Farah. 2018. Multi-classifier system for authorship verification task using word embeddings. *2nd International Conference on Natural Language and Speech Processing, ICNLSP 2018*:1–6.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit.* O’Reilly Media, Inc.

John Burrows. 2002. “Delta”: a measure of stylistic difference and a guide to likely authorship. *Literary and Linguistic Computing*, 17(3):267–287.

David García-Barrero, Manuel Feria, and Maria Teresa Turell. 2013. Using function words and punctuation marks in Arabic forensic authorship attribution. In Rui Sousa-Silva, Rita Faria, Núria Gavaldà, and Belinda Maia, editors, *Proceedings of the 3rd European Conference of the International Association of Forensic Linguists*, pages 42–56, Porto, Portugal. Faculdade de Letras da Universidade do Porto.

Oren Halvani, Christian Winter, and Lukas Graner. 2017. Authorship Verification based on Compression-Models.

Oren Halvani, Christian Winter, and Anika Pflug. 2016. Authorship verification for different languages, genres and topics. *Digital Investigation*, 16:S33–S43.

Fatma Howedi and Masnizah Mohd. 2014. Text Classification for Authorship Attribution Using Naive Bayes Classifier with Limited Training Data. *Computer Engineering and Intelligent Systems*, 5(4):48–56.

Magdalena Jankowska, Evangelos Milios, and Vlado Kešelj. 2014. Author Verification Using Common N-Gram Profiles of Text Documents. *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics*:387–397.

Siham Ouamour and Halim Sayoud. 2013. Authorship Attribution of Short Historical Arabic Texts Based on Lexical Features. In *2013 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery*, pages 144–147.

Siham Ouamour and Halim Sayoud. 2018. A Comparative Survey of Authorship Attribution on Short Arabic Texts.

Arfath Pasha, Mohamed Al-badrashiny, Mona Diab, Ahmed El Kholy, Ramy Eskander, Nizar Habash, Manoj Pooleery, Owen Rambow, and Ryan M Roth. 2014. MADAMIRA : A Fast , Comprehensive Tool for Morphological Analysis and Disambiguation of Arabic. *Proceedings of the 9th Language Resources and Evaluation Conference (LREC'14)*:1094–1101.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, and Vincent Dubourg. 2011. Scikit-learn: Machine learning in Python. *Journal of machine learning research*, 12(Oct):2825–2830.

Efstathios Stamatatos, Walter Daelemans, Ben Verhoeven, Martin Potthast, Benno Stein, Patrick Juola, Miguel A. Sanchez-Perez, and Alberto Barrón-Cedeño. 2014. Overview of the author identification task at PAN 2014. In *CEUR Workshop Proceedings*, volume 1180, pages 877–897.

Kazem Taghva, Rania Elkhoury, and Jeffrey Coombs. 2005. Arabic stemming without a root dictionary. In *Information Technology: Coding and Computing, 2005. ITCC 2005. International Conference on*, volume 1, pages 152–157. IEEE.