Arabic Dialect Texts Classification

by

Areej Odah O. Alshutayri

Submitted in accordance with the requirements for the degree of
Doctorate of Philosophy

UNIVERSITY OF LEEDS

The University of Leeds
Faculty of Engineering

School of Computing

October, 2018

The candidate confirms that the work submitted is her own, except where work which has formed part of jointly-authored publications has been included. The contribution of the candidate and the other authors to this work has been explicitly indicated overleaf. The candidate confirms that appropriate credit has been given within the thesis where reference has been made to the work of others.

Publications

Chapter 3-9 in parts II, III and V of this thesis are based on jointly-authored publications. I was the lead author and the co-authors acted in an advisory capacity, providing supervision and review. All original contributions presented here are my own.

Part II-

The method which was used to collect tweets from Twitter in Chapter 4 is based on the following paper:

Alshutayri A. and Atwell E. (2017). Exploring Twitter as a Source of an Arabic Dialect Corpus. *International Journal of Computational Linguistics (IJCL)*, **8** (2), pp. 37-44.

The methods which were used to collect comments from online newspaper, Facebook and Twitter in Chapter 5 and 7 are based on the following papers:

Alshutayri A. and Atwell E (2018). *Creating an Arabic Dialect Text Corpus by Exploring Twitter, Facebook, and Online Newspapers.* Proceedings of the LREC 2018 Workshop, The 3rd Workshop on Open-Source Arabic Corpora and Processing Tools (OSACT3), pp.54-61.

Alshutayri A. and Atwell E. (2018). A Social Media Corpus of Arabic Dialect Text. Wigham, C.R. & Stemle, E. (2018). TITLE. Cahiers du Laboratoire de Recherche sur le Langage. No. 7. Clermont-Ferrand: Presses universitaires Blaise Pascal.

The method was used to verify the annotated the corpus in Chapter 8 is based on the following paper:

Alshutayri A. and Atwell E. (2018). Arabic Dialects Annotation using an Online Game. 2nd International Conference on Natural Language and Speech Processing (ICNLSP), IEEE, Algiers, Algeria.

Part III

The initial experiment of classifying Arabic dialects in Chapter 3 are based on the following paper:

Alshutayri A., Atwell E., Alosaimy A., Dickins J., Ingleby M. and Watson J. (2016) *Arabic Language WEKA-Based Dialect Classifier for Arabic Automatic Speech Recognition Transcripts* Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects (VarDial 2016), pp. 204-211.

I carried out the implementation and evaluation of the model. I wrote the majority of the content of the paper. Janet Watson and James Dickins examined some of the texts to extract some features. AbdulRahman Alosaimy translated texts from Buckwalter to Arabic and computed the utterances distribution. Eric Atwell provided supervisory advice. Michael Ingleby and Eric Atwell made suggestions to help clarify the content for submission.

The second experiment of classifying Arabic dialects using three different sources of Arabic dialects in Chapter 6 are based on the following paper:

Areej Alshutayri and Eric Atwell. Classifying Arabic Dialects in Three Different Corpora Using Ensemble Classifier*. (in preparation).*

Acknowledgements

First and above all, I need to thank Allah (Almighty God) who gave me strength, knowledge and patience to bear all the difficulties that I faced during my study, who blessed me with success to do my work.

I dedicate this thesis and all success in my life to my parents Mr. Odah and Mrs. Fatimah whose support me spiritually and physically in my entire life; without their encouragement and support, this work and all of my successes would not have been done. I would like to thank my Mom and Dad for their love, prayers, support, and the confidence they gave me to complete my study abroad.

I am heartily thankful to my supervisor, Professor Eric Atwell for helping me with ideas and for his constant encouragement and appreciation for every work I gave him. Thanks to him for pushing me to publish so I see some parts of my research cited in other research. Thank to him because he respected all my circumstances, he is an example of a pure good man.

I would also like to thank my brothers and sisters for their trust in me and for their support and encouragement. They being such an inspiration to me and I would like to say “I love you” to all of them.

Last but not the least, I would like to thank my lovely sons, Mohammed and AbdulRahman, for their smiles or even their cries which supported me to finish this work, thank for their sharing with me the ups and downs in the alienation; they sacrificed even they are young to understand that but I know, without their presence in my life and without their sacrifices I am sure that this work would not exist.

Thank go to King Abdulaziz University, Jeddah, Saudi Arabia for giving me a scholarship to study in the UK.

Finally, thanks go to my friends, neighbours and all people in the school of Computing for their cooperation and support.

Abstract

This study investigates how to classify Arabic dialects in text by extracting features which show the differences between dialects. There has been a lack of research about classification of Arabic dialect texts, in comparison to English and some other languages, due to the lack of Arabic dialect text corpora in comparison with what is available for dialects of English and some other languages. What is more, there is an increasing use of Arabic dialects 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 from Twitter, comments from Facebook and online newspapers from five groups of Arabic dialects: Gulf, Iraqi, Egyptian, Levantine, and North African. The research sought to: 1) create a dataset of Arabic dialect texts to use in training and testing the system of classification, 2) find appropriate features to classify Arabic dialects: lexical (word and multi-word-unit) and grammatical variation across dialects, 3) build a more sophisticated filter to extract features from Arabic-character written dialect text files.

In this thesis, the first part describes the research motivation to show the reason for choosing the Arabic dialects as a research topic. The second part presents some background information about the Arabic language and its dialects, and the literature review shows previous research about this subject. The research methodology part shows the initial experiment to classify Arabic dialects. The results of this experiment showed the need to create an Arabic dialect text corpus, by exploring Twitter and online newspaper. The corpus used to train the ensemble classifier and to improve the accuracy of classification the corpus was extended by collecting tweets from Twitter based on the spatial coordinate points and comments from Facebook posts. The corpus was annotated with dialect labels and used in automatic dialect classification experiments. The last part of this thesis presents the results of classification, conclusions and future work.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

Abstract Classifier (Abs-Cl)

Application Programming Interface (API)

Arabic Dialect Dataset (ADD)

Arabic Dialects Identification (ADI)

Arabic Multi Dialect Written Corpora (AMDWC)

Arabic Online Commentary Dataset (AOCD)

Automatic Identification of Dialectal Arabic (AIDA)

Automatic Speech Recognition (ASR)

Comma Separated Values (CSV)

Comprehensive Classifier (Comp-Cl)

Computer Mediated Communication (CMC)

Conditional Random Field (CRF)

Dialectal Arabic (DA)

Discriminating Similar Languages (DSL)

Egypt (EG)

Egyptian Dialect (EGY)

Extensible Markup Language (XML)

Gulf Dialect (GLF)

Hypertext Markup Language (HTML)

International Phonetic Alphabet (IPA)

Inverse Document Frequency (IDF)

Iraqi Dialect (IRQ)

Java Server Page (JSP)

K-nearest neighbours (IBK)

Kuwait (KW)

Language Identification (LID)

Language Models (LM)

Levantine Dialect (LEV)

Linguistic code switching (LCS)

Machine Readable Dictionaries (MRDs)

Malay Chat-style-text Corpus (MCC)

Modern Standard Arabic (MSA)

Multinomial Naïve Bayes (MNB)

Natural Language Processing (NLP)

North African Dialect (NOR)

Qatar (QA)

Saudi Arabia (SA)

Sequential Minimal Optimization (SMO)

Social Media Arabic Dialect Corpus (SMADC)

Sprachatlas der Deutschen Schweiz (SDS)

Support vector machine (SVM)

Term Frequency (TF)

Tunisian Sentiment Analysis Corpus (TASC)

United Arab Emirates (UAE)

Weight Average Method (WAM)

Waikato Environment for Knowledge Analysis (WEKA)

Weight Multiplied Method (WMM)

Word Count (WC)

Part I

Introduction, Background and Literature Review

Chapter 1
Introduction

Language Identification or Dialect Identification is the task of identifying the language or dialect of a written text. The task of Arabic dialect identification may require both computer scientists and Arabic linguistics experts.

There are many languages spoken and written by the world's population, and each language has different dialects, which are divided depending on the geographical locations. 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 based on the text of the Quran, the holy book of Islam; and 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) which is a subclass of Peninsular Arabic. Studies in Arabic dialectology focus on phonetic variation (Ali *et al.* 2016; Alorifi 2008; Biadsy *et al.* 2009; Horesh and Cotter 2016).

Arabic dialects classification is becoming important due to the increasing use of Arabic dialect in social media. As a result, there is a need to know the dialect used by speakers or 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 (Lu and Mohamed 2011). 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.

1.1 Background

In recent years, research in Natural Language Processing (NLP) on Arabic Language has garnered significant attention (Shoufan and Al-Ameri 2015). Social Media is a particularly good resource to collect Arabic dialect text for NLP research. Almost all Arabic text is in Modern Standard Arabic (MSA) because most Arab people are taught in school to always write in MSA in all formal situations; however, some Arabs, especially young people, have started to write using their dialect in informal uses such as Computer-Mediated Communication (CMC) and social media. There are some Arabic dialect text corpus data-sets, but many of these corpora are not available, or not covering most of Arabic dialects, or not balanced, or insufficiently labelled. There is a lack of Arabic dialect text corpora in comparison with what is available for dialects of English and other international languages, and this showed the need to create dialect text corpora for use in Arabic dialect text processing.

There are many studies that aim to classify Arabic dialects in both text and speech. In this research, the classification of Arabic dialects will focus on text, because most of Arabic dialect research focuses on phonological variation, based on audio recordings and listening to dialect speakers; this is sufficient to notice and capture phonetic and phonological features in a dialect. There are many studies focusing on speech such as in (Ali *et al.* 2015; Alorifi 2008; Belgacem *et al.* 2010; Biadsy *et al.* 2009) due to the explicit phonological variations between Arabic dialects. However, text classification is a new topic and still needs a lot of research to increase the accuracy of classification due to the same characters being used to write MSA text and many dialects, and also because there is no standard written format for Arabic dialects. In addition, lexical and grammatical differences are also worth studying, and the study of these requires larger text data-sets of transcribed dialect data. The transcription need not, and should not, be phonetic transcription in International Phonetic Alphabet (IPA), since this is too time-consuming and unnecessary to capture dialect-specific words and phrases.

1.2 Purpose and Objective

In general, natural language processing for spoken and written English and other languages has been the subject of many studies in the last fifty years (Biadsy *et al.* 2009). However, Arabic language research has been growing very slowly in comparison to English language research (Alorifi 2008). This slow growth is due to the lack of recent studies on the nature of the variation of the Arabic language resulting from a lack of database of Arabic dialects. Moreover, assessing the similarities and differences between dialects of a language is a challenge in natural language processing.

Almost all available datasets for Arabic computational linguistic research are in MSA, especially those in textual form. Recently, researchers are starting to work with Arabic dialect text (Almeman and Lee 2013; Zaidan and Callison-Burch 2014). Given the increasing use of Arabic dialect in informal settings such as Computer-Mediated Communication (CMC) and social media, these type of texts are now considered for corpus creation. There is a lack of Arabic dialect text corpora which are balanced and cover several Arabic dialects, so we decided to use Twitter and Facebook, because they attract a lot of people who freely write in their dialects. In addition, to cover long dialect texts we used online comments text from Arabic newspapers.

According to Malmasi *et al.* (2015), if we classify Arabic dialects according to countries, we will notice a high degree of confusion and overlap between dialects. Since there are no clear geographical borders between Arabic dialects (Lu and Mohamed 2011). So, grouping overlapping dialects under broad classes is the best method to improve the accuracy of classification.

In this thesis we classified dialects into five classes: Gulf, Iraqi, Egyptian, North African, and Levantine: GLF, IRQ, EGY, NOR, and LEV. These classes cover the major Arabic dialects in the Arab world.

The objective of this work is to build a balanced Arabic dialect text corpus using CMC and social media sources: Twitter, comments from online newspapers, and Facebook. The research aim is contributing to and enhancing the accuracy of classification for Arabic dialectical texts by exploring a new method of classification and extracting Arabic linguistic features.

The research objectives are outlined as follows to guide the research and achieve the aim:

- Collect a dataset of Arabic dialect texts which is a novel data source: dialect data written in Arabic characters by dialect speakers to use it in training and testing processes.
- Focus on lexical (word and multi-word-unit) and grammatical variation across dialects.
- Define the differences between Arabic dialects to decide how to classify them in text.

- Select good features that distinguish accurately between Arabic dialects, which we can test in different classifiers.
- Develop a new filter to extract Arabic dialect features from the dataset.
- Create dictionaries for each dialect.
- Choose a suitable machine-learning algorithm (classifier) to classify dialects texts.
- Check the efficiency of the extracted features by testing them in different classifiers.
- Conduct classification experiments to derive results and make conclusions.

1.3 Research Questions and Contributions

The research addresses questions including the following:

- Which source of dataset provide the best results?
- What are appropriate features?
- Do the selected features improve the classification accuracy?

In this research the contributions are:

- The construction of a large multi-dialect corpus of Arabic.
- An exploration of how to extract geolocation sensitive text from various social and internet media.
- The use of gamification for corpus annotation.
- Identification and extraction of new linguistic features to classify Arabic dialect text which can be tested in different classifiers.
- Creation of dictionaries for each dialect.
- The use of ML and dictionary based approaches to automatically classify dialects.

1.4 Outline of the Thesis

This thesis is split into seven parts with 11 chapters as shown in the following:

- **Part I**
  Introduction, and Literature Review
  - Chapter 1: Introduction and Background

    Chapter 1 provides background information about Arabic language and its dialects, the objectives of this research and the contributions.
  - Chapter 2: Literature Review

    Chapter 2 covers the current and past work within the area of Arabic dialect corpora, classification of Arabic dialect and information about machine learning.
- **Part II**
  Creating the Arabic Dialect Corpus
  - Chapter 3: Exploring Twitter as a Source of Arabic Dialect Texts Corpus

    Chapter 3 explores Twitter as a source of Arabic Dialect Texts and describes the methods that we used to extract tweets and classify them according to the geographic location of the sender.
  - Chapter 4: Creating an Arabic Dialect Text Corpus by Exploring online Newspaper

    Chapter 4 presents our methods to create a corpus of dialectal Arabic by extracting the online comments from electronic Arabic newspapers as another source of a dialectal Arabic text.
  - Chapter 5: Extending an Arabic Dialect Texts Corpus

    Chapter 5 presents how we extended the Social Media Arabic Dialect Corpus (SMADC) by collecting more tweets from Twitter based on spatial coordinate points, and scrape Facebook posts to collect users' comments.
  - Chapter 6: Annotating Arabic Dialect Texts Corpus

    Chapter 6 introduces a new approach to annotate the dataset were collected from Twitter, Facebook, and online newspaper for the five main Arabic dialects: Gulf, Iraqi, Egyptian, Levantine and North African.

  - Chapter 7: Final Version of Arabic Dialect Texts Corpus

    Chapter 7 presents a description of the final version of the corpus that were collected from Twitter, Facebook, and online newspaper.

- **Part III**
  Arabic Dialect Texts Classification
  - Chapter 8: Initial Experiment in Classification

    Chapter 8 describes an Arabic dialect identification system which we developed for the Discriminating Similar Languages (DSL) 2016 shared task.

  - Chapter 9: Classifying Arabic Dialects in Three Different Corpora using Ensemble Classifier

    Chapter 9 describes the method was used to classify a text as belonging to a certain Arabic dialect and presents the comparison between three different data sets to explore which is the best source of written Arabic dialects.

  - Chapter 10: Automatic Dialect Texts Classification

    Chapter 10 introduces the methods were used to classify Arabic dialect texts and the achieved results of these methods.

- **Part IV**
  Conclusions and Future Work
  - Chapter 11: Conclusion and Future Work

    Chapter 11 summarizes the thesis achievements, conclusion and future work.

Chapter 2
Literature Review

This chapter presents a review of Arabic language and its dialects, the phonological and lexical variations between dialect, machine learning algorithms, and some previous works related to this thesis in parts of creating an Arabic dialect corpus and automatic classification of Arabic dialect text. Some parts of this chapter is derived from (Alshutayri and Atwell 2018b; Alshutayri and Atwell 2018c).

2.1 Arabic Language

The Arabic language is a Semitic language originating on the Arabian Peninsula, and it is considered one of the major languages in the world. As a result of the expansion of Islam from Spain to Persia, the Arabic language is spread across many countries.

2.1.1 The Language Situation on the Eve of Spread of Islam

- Levantine Dialect

  The Levantine covered the area occupied by modern Syria, Lebanon, Palestine, and Jordan. The whole of this area had been under the Byzantine control before the Arab conquests. At that time, the majority of the population spoke different dialects of Aramaic. While in the cities, people spoke Greek especially the government officials, merchants, and landowners. However, the Arabic language was spoken in some areas where the nomadic Arab tribes summered in the towns and settlements such as the Bekaa Valley, Zabad, and Aleppo (Holes 2004).

  There are three factors have helped the spread of the Arabic language as a spoken language on the eve of the spread of Islam: the trade-engendered contact between speakers of Arabic and Aramaic, the permanent settlement by Christian Arabs, and the failure of Greek culture to affect outside the cities and coastal ports. As a result of these factors the Arabic language became the first language in this area and the Aramaic speakers started to accept Arabic as a language for communication (Holes 2004).

- Iraqi Dialect

The linguistic situation in Iraq had some similarities to the situation in Syria. The majority of the population were rural and sedentary, Christian or Jewish, and they spoke Aramaic dialects, although the Persian language was spoken in the cities. By the mid-seventh century, the Arabic speaking tribesmen who settled in Mesopotamia mixed with the local Aramaic-speaking people. Regular contact between the Aramaic and Arabic-speaking local people and the Arab tribes of inner Arabia helped Arabic language to spread across different areas in Iraq (Holes 2004).

- Egyptian Dialect

At this time, Egypt was multilingual and the majority of the population was made up of rural people in the Nile Valley and Delta, in addition to the inhabitants of the towns and cities of the Delta and Nile including Alexandria. The rest of the population was the people who lived in cultivable areas in to the east of the River Nile and Delta, and people in the desert to the west of the Red Sea, and people in Sinai (Holes 2004).

The people in the Nile Valley and Delta spoke Coptic because they lived alongside Greek traders and urban Copts. While on the eastern side of the valley and into the deserts, there had been a process of Arabization due to the migration of tribal from the peninsula (Holes 2004).

- North Africa

At the time of the Islamic conquest, the Berber tribes lived on the North Africa coast which was controlled by the Byzantine empire. The Greeks had no authority over, or contact with the Berber which allowed the Berber language to have remain a spoken language up to the present (Holes 2004).

Figure 2.1 shows the language situation on the eve of Islam on the Arab world.

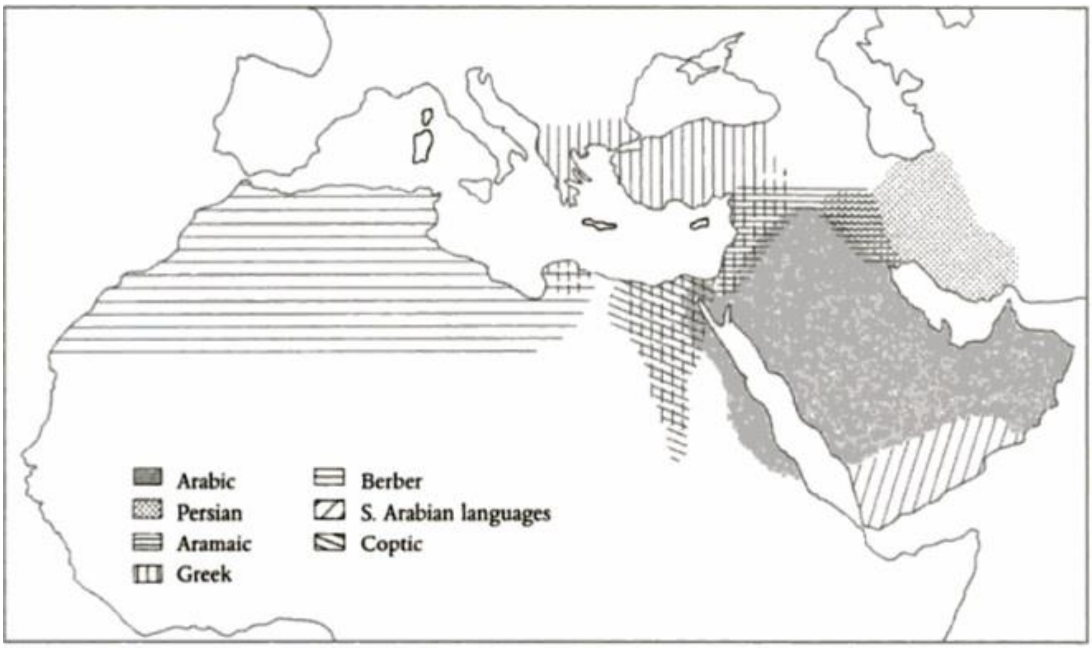


Figure 2.1 The language situation on the eve of the Islamic conquests. Adapted from (Holes 2004).

2.1.2 The Reform of the Arabic Lexicon

By the eighth century, the Arab empire stretching from Spain to Persia helped to spread Classical Arabic in this area. After that, in the nineteenth century, when the French conquered Egypt and North Africa, loan-words were introduced by writers as a result of the influence of the French language and Ottoman Turkish in the second half of the nineteenth century (Versteegh 2014).

In this period, the Arabic lexicon expanded as a result of translating of Greek logical, medical and philosophical writings, but the process of translation did not stop at technical and scientific terminology. Some examples of the effect of the translation process are: the verb talfaza derived from tilifizyun, and the broken plurals bunuk from the noun Bank. The regional variation and the new vocabularies that were borrowed from other languages, both are factors contributing to modify Classical Arabic and create Modern Standard Arabic (Versteegh 2014).

2.2 Arabic Dialect

Each language has different dialects, differentiated mainly by the geographical locations of speakers, as shown in Figure 2.2. Moreover, there other important factors affected on variation between Arabic dialect such as, sociological and communal. The Bedouin societies speak different dialects from the local sedentary societies, and people of different religious have different dialects (e.g. Muslim/Christian/Jewish dialects).

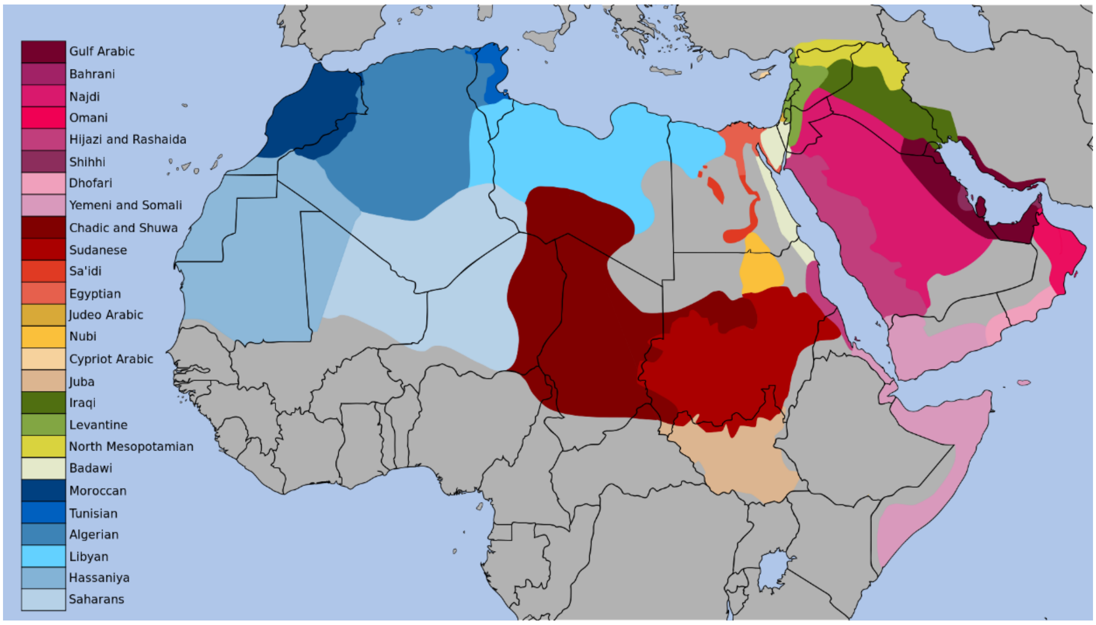


Figure 2.2 Different Arabic varieties in the Arab world. Adapted from Wikipedia.

Arabic language has multiple variants, some formal and some informal (Habash 2010). Modern Standard Arabic (MSA) is a standard formal variant in the Arab world, and it is used and understood by almost all people in the Arab world. MSA is based on Classical Arabic, which is the language of the Qur’an, the Holy Book of Islam. MSA is mostly written, not spoken in daily life (Biadsy *et al.* 2009). MSA is used in media, newspaper, culture and education; additionally, most Natural Language Processing (NLP) research and tools are based on MSA, such as Automatic Speech Recognition (ASR) and Language Identification (LID), Figure 2.3 shows the usage of MSA. Dialectal Arabic (DA) is an informal variant used in daily life communication, TV shows, songs and movies. These dialects are mostly spoken, not written. In contrast to MSA, Arabic dialects are less closely related to Classical Arabic. Arabic dialects vary from each other and from Modern Standard Arabic, Section 2.5 describe the variation between Arabic dialects.

DA is a mix of Classical Arabic and other ancient forms from different neighbouring countries that developed as a result of social interaction between people in Arab countries and people in the neighbouring countries (Biadsy *et al.* 2009).

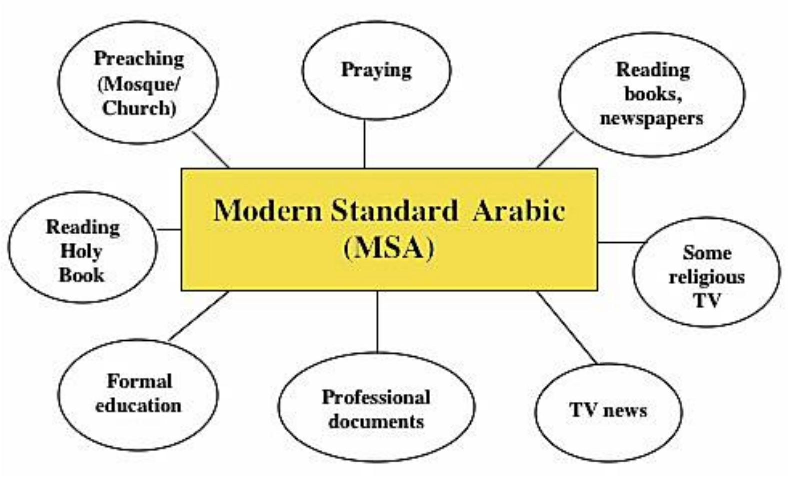


Figure 2.3 The usage of MSA.

The main groupings of Arabic dialects are: GLF, IRQ, LEV, EGY and NOR as shown in Figure 2.4 (Habash 2010).

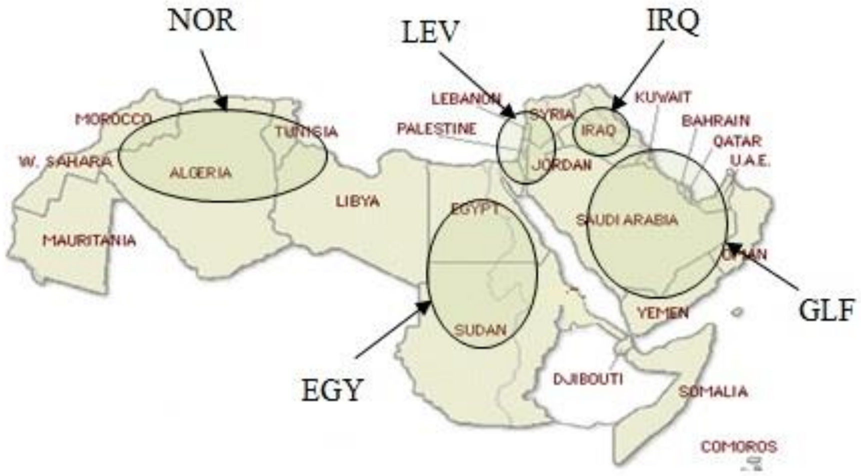


Figure 2.4 Arab World Map. Adapted from ArabBay.com.

GLF is used in countries around the Arabian Gulf, and includes dialects of Saudi Arabia, Kuwait, Qatar, United Arab Emirates, Bahrain, Oman and Yemen. IRQ is used in Iraq, and it is a sub-dialect of GLF (Alorifi 2008; Biadsy *et al.* 2009; Habash 2010). LEV is used in countries around the Mediterranean east coast, and covers the dialects of Lebanon, Syria, Jordan and Palestine. EGY includes the dialects of Egypt and Sudan. Finally, NOR includes the dialects of Morocco, Algeria, Tunisia and Libya (Alorifi 2008; Biadsy *et al.* 2009; Habash 2010).

2.3 Arabic Dialect Text Corpora

In recent years, social media has spread between people because of the growth of wireless Internet networks and several social applications of Smartphones. These media sources of texts contain people's opinions written in their dialects which make it the most viable resource for dialect Arabic. The sources are Twitter, forums, Facebook, blogs, and online commentary.

Arabic dialect studies have developed rapidly in recent years. However, any classification of dialects depends on a corpus to use in training and testing processes. There are several studies that have tried to create Arabic dialect corpora; however, many of these corpora do not cover all the geographical variations in dialects. In addition, several of them are not accessible to the public. The following section describes text corpora that were built by previous studies using Twitter, Facebook, and online newspaper comments.

2.3.1 Twitter Corpus Creation

Twitter is a social medium, which enables users to write texts consisting of 140 characters[1] (Meder *et al.* 2016), increased now to 280 characters. Twitter is a more accessible resource from which to collect data compared to other social media, because the data in Twitter is public. Twitter offer an Application Programming Interface (API) that helps researchers to access the available data on the server, and to extract other metadata, such as location. However, there is a lack of readily available Twitter corpora for specific research

[1] at the time of collecting the tweets

purposes such as balanced training data for Machine Learning of automatic dialect classification, which makes it necessary for researchers to create their own corpora (Saloot *et al.* 2016).

Mubarak and Darwish (2014) used Twitter to collect an Arabic multi-dialect corpus. The researchers classified dialects as Saudi Arabia, Egyptian, Algerian, Iraqi, Lebanese and Syrian.

The Twitter's API[2] allows users to specify a search query or pattern and then extract all tweets which match this query; the query can include words, and/or general patterns such as "lang:X" which matches all tweets written in a specific language X. Mubarak and Darwish (2014) used the general query lang:ar on the Twitter's API to get the tweets which were written in Arabic language. They collected 175M Arabic tweets, then, extracted the user location from each tweet to classify it as a specific dialect according to the location.

Then, Mubarak and Darwish (2014) classified these tweets as dialectal or not dialectal (MSA) using the dialectal words from the Arabic Online Commentary Dataset (AOCD) described in (Zaidan and Callison-Burch 2014). Each dialectal tweet was mapped to a country according to the user location mentioned in the user's profile, with the help of the GeoNames geographical database (Mubarak and Darwish 2014).

The next step was normalization to delete any non-Arabic characters and to delete the repetition of characters. Finally, they asked native speakers from the countries identified as tweet locations to confirm whether each tweet used their dialects or not. At the end of this classification, the total tweets number about 6.5M in the following distribution: 3.99M from Saudi Arabia (SA), 880K from Egypt (EG), 707K from Kuwait (KW), 302K from United Arab Emirates (UAE), 65k from Qatar (QA), and the remaining (8%) from other countries such as Morocco and Sudan. Figure 2.5 shows the distribution of tweets per country.

[2] http://apps.twitter.com

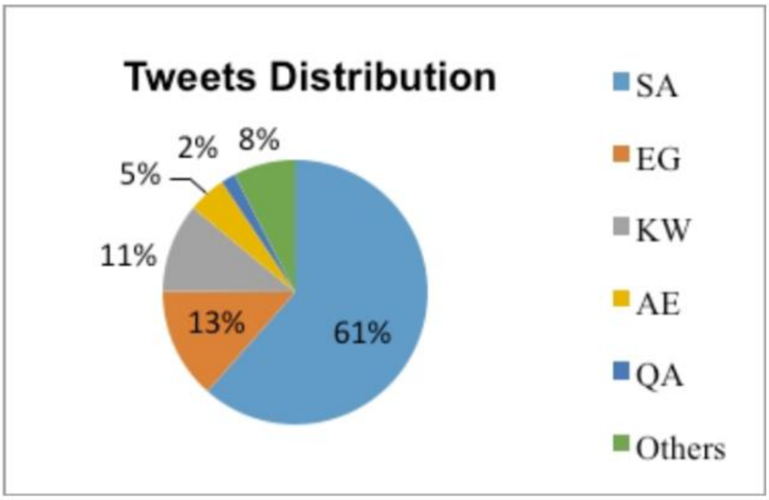


Figure 2.5 Dialectal Tweets Distribution. Adapted from (Mubarak and Darrwish, 2014, p.5, fig. 2).

In the sentiment analysis field, Xiang *et al.* (2012) created an English twitter corpus contained 680 million tweets for training, and 16 million tweets for testing, to detect offensive content in Twitter. Additionally, Pak and Paroubek (2010) collected corpus of an English language. Researchers used popular Twitter accounts of newspapers and magazines to create this corpus for sentiment analysis and opinion mining purposes, to decide if the sentiments for a document were positive, negative or neutral. There are much research studied of sentimental analysis in Arabic, and all these researchers created their dataset from Twitter or other sources because of the lack of a corpus of Arabic dialects (Duwairi 2015; Ibrahim *et al.* 2015; Al-Harbi and Emam 2015).

In the case of Malay Chat-style-text Corpus (MCC), researchers followed ten criteria to create a MCC corpus; Population boundary, Representativeness, Sampling technique, Production and reception text, Variety, Chronology, Anonymization, Share ability, Fragmentation, and Chunking (Saloot *et al.* 2016). In the first criterion, researchers define the boundary of the desired population. In the second criterion, the sampling frame used Twitter user IDs for the users who set their location to Malaysia. In the third criterion, even if the location was set to Malaysia, they checked the language, and if they wrote using a non-Malay language then those user IDs were considered as out-of-coverage. In the fourth criterion, the tweets had to be in chat-style, non- formal Malay language; therefore, any commercial and political tweets are ignored. In the fifth criterion, they tried to cover different writing style considered the differences in using grammar, lexis, and discourse features. In the sixth criterion, the corpus could be built in a synchronic or diachronic way, according to the potential users. In the seventh criterion, user IDs had be hidden to make all tweets anonymized. In the eighth criterion, the corpus should be made available for another research purpose. In the ninth criterion, the corpus must have different version such as text and Extensible Markup Language (XML). In the tenth criterion, the corpus is suitable for extracting sub-corpora. After applying these criteria, researchers found that the sample frame was equal to 321 users who posted their tweets in chat-style Malay language, out of 4,500 users. Then, they used a computer application to extract 3,200 tweets from each user to create a corpus containing one million tweets. In all, MCC consists of 14,484,384 words and 646,807 terms.

2.3.2 Facebook Corpus Creation

Facebook was used to create two corpora for sentiment analysis (Itani *et al.* 2017). The authors manually copied post texts which were written in Arabic dialect to create a news corpus collected from the “Al Arabiya” Facebook page and an arts corpus collected from the Facebook page “The Voice”. Each corpus contained 1000 posts. They found that 5% of the posts were associated with a specific dialect while 95% were common to all dialects. After collecting Facebook posts and comments they processed the texts by removing time stamps and other redundant text. In the last step, the texts were manually annotated by four native Arabic speakers, who were experts in MSA and Arabic dialects. The labels were: negative, positive, dual, spam, and neutral. To validate the result of the annotation step, the authors had to agree the same label. The total number of posts were 2000 divided into 454 negative posts, 469 positive posts, 312 dual posts, 390 spam posts, and 375 neutral posts.

Another piece of research used the text in Facebook to create corpora for improved Arabic dialect classification with social media data (Huang 2015). The authors randomly selected 2700 documents from Facebook public posts. Then labelled each document manually by human annotators. The results showed that 58% of the collected documents was Modern Standard Arabic (MSA), Egyptian dialect in the second place with 34% of the documents followed by Levantine and Gulf. Maghrebi in the last place. In addition to some documents not labelled as Arabic dialect such as verses from the Quran, classical Arabic, foreign words and their transliterations, etc.

Tunisian Sentiment Analysis Corpus (TASC) was created using Facebook users comments for sentiment analysis (Mdhaffar *et al.* 2017). The authors collected comments written on official pages of Tunisian radios and TV channels called Mosaique FM, JawhraFM, Shemes FM, HiwarElttounsi TV and Nessma TV for seventeen months period from January 2015 to June 2016. The corpus consists of 17K comments manually annotated to 8215 comments are positive and 8845 comments are negative.

2.3.3 Web and Online Newspaper Corpus Creation

A multi-dialect Arabic text corpus was built by Almeman and Lee (2013) using a web corpus as a resource. In this research, they focused only on distinct words and phrases which are common and specific to each dialect. They covered four main Arabic dialects: Gulf, Egyptian, North African and Levantine.

They collected 1,500 words and phrases by exploring the web and extracting each dialect’s words and phrases, which must have been found in one dialect of the four main dialects. In the next step, they consulted a native speaker for each dialect to distinguish between the words and confirm that words were used in that dialect only. After the survey, they created a corpus containing 1,000 words and phrases in the four dialects, including 430 words for Gulf, 200 words for North Africa, 274 words for Levantine and 139 words for Egyptian.

Zaidan and Callison-Burch (2014) worked on Arabic Dialects Identification and focused on three Arabic dialects: Levantine, Gulf, and Egyptian. They created a large data set called the Arabic Online Commentary Dataset (AOCD) which contained dialectal Arabic content. Zaidan and Callison-Burch collected words in all dialects from readers' comments on the three online Arabic newspapers which are Al-Ghad from Jordan (to cover Levantine dialect), Al-Riyadh from Saudi Arabia (to cover Gulf dialect), and Al-Youm Al-

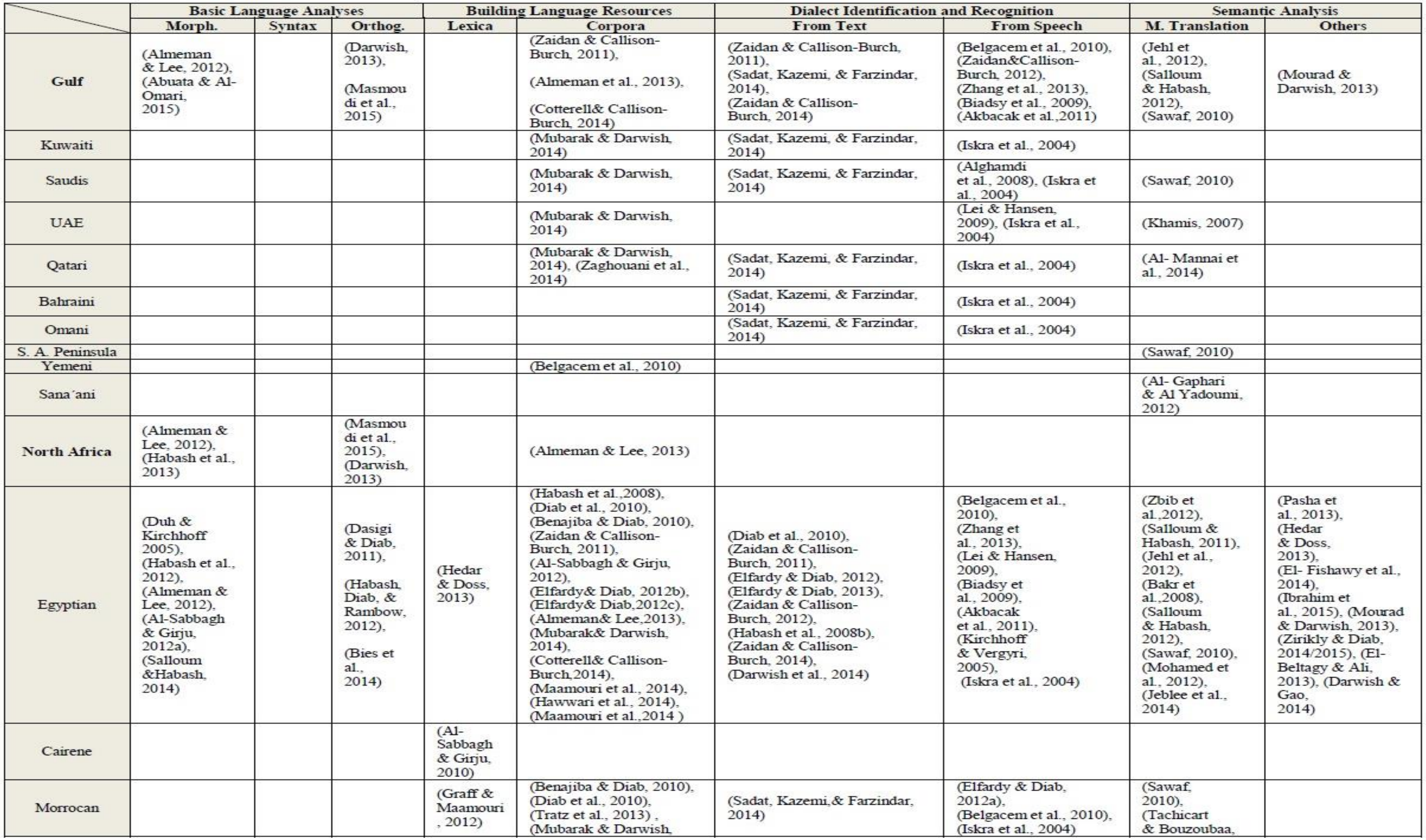
Sabe from Egypt (to cover Egyptian dialect). They used the newspapers to collect 1.4M comments from 86.1K articles. Finally, they extracted 52.1M words for all dialects. They obtained 1.24M words from Al-Ghad newspaper, 18.8M form Al-Riyadh newspaper, and 32.1M form Al-Youm Al-Sabe newspaper.

El-Haj *et al.* (2018) created an Arabic dialect corpus covers four Arabic dialects: Egyptian (EGY), Levant (LAV), Gulf (GLF), and North African (NOR), in addition to Modern Standard Arabic (MSA). The authors collected the corpus by randomly selected comments from the Arabic Online Commentary Dataset (AOCD) (Zaidan and Callison-Burch, 2014) which covers MSA, EGY, GLF and LAV. North African (NOR) were not covered in AOCD so for NOR dialect the authors randomly selected texts from Tunisian Arabic which is a free online corpus of Tunisian dialect (Karen and Faiza 2010) beside randomly selected sentences from the Internet forums. They collected 23,567 documents divided as 5802 for EGY, 3638 for GLF, 3519 for LAV, 5277 for NOR, and 5331 for MSA.

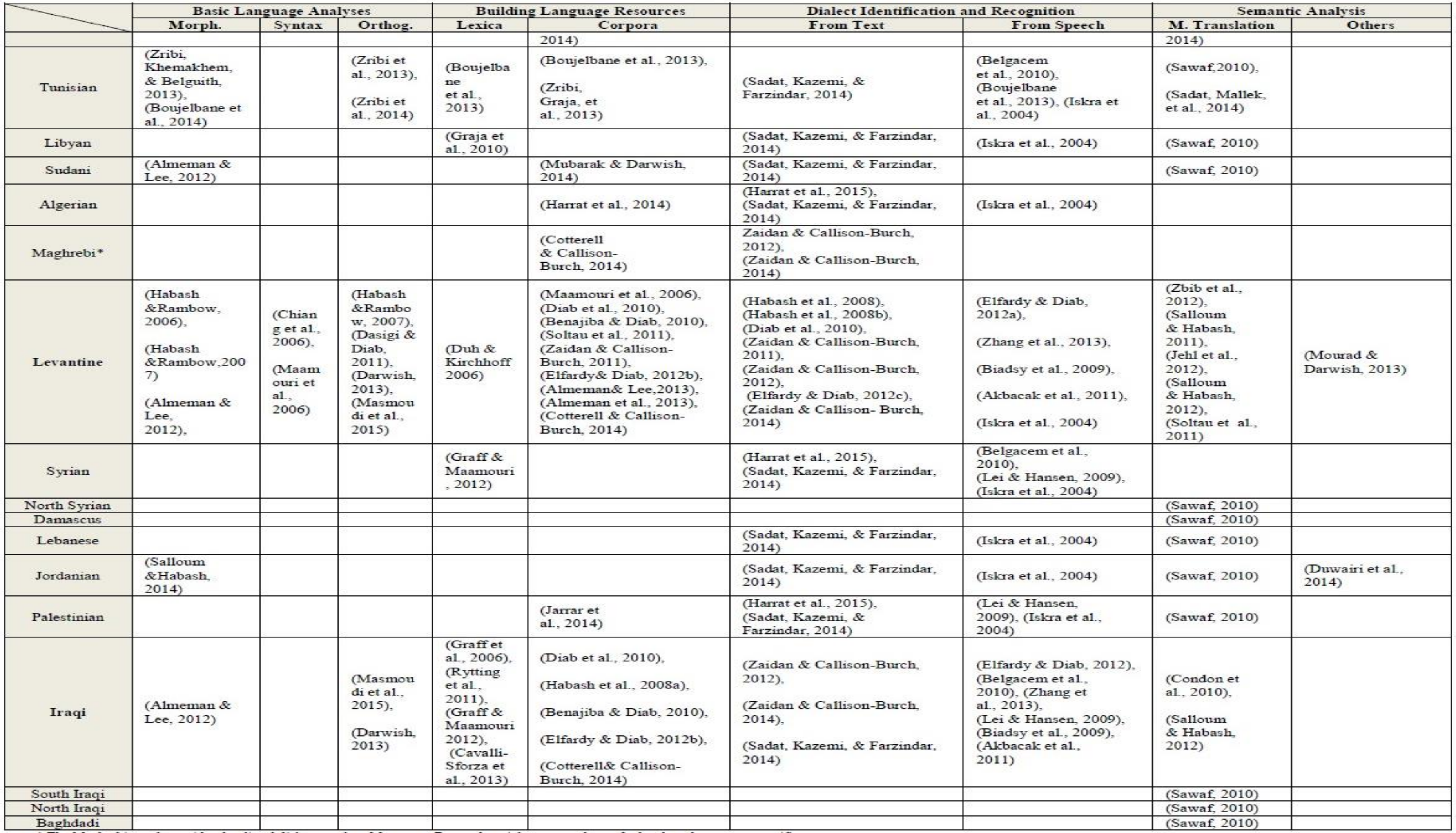
The last research by Bouamor *et al.* (2014) presented a multi-dialectal Arabic parallel corpus. This corpus contains 2,000 sentences in five dialects: Egyptian, Tunisian, Jordanian, Palestinian, and Syrian, in addition to MSA and English. Researchers selected 2,000 sentences from the Egyptian-English corpus, which was built by (Zbib et al., 2012, cited in Bouamor et al., 2014, p.1242) because the Egyptian dialect is the most understood dialect in the Arab world as a result of the Egyptian media industry. After that, they asked four native speakers of Tunisian, Jordanian, Palestinian, and Syrian dialects to translate 2,000 sentences which were written in Egyptian to their own dialects. The fifth translator from Egypt was asked to translate the 2,000 sentences to MSA.

The following is table from a survey of all research on natural language processing on Arabic dialects and created corpora for Arabic dialect. The table shows that there is a lot of research on speech corpora because most of dialect research focuses on speech but working with Arabic dialect text is a more recent development (Shoufan and Al-Ameri 2015).

Table 2.1 Dialectal Arabic NLP- Literature overview (Shoufan and Al-Ameri 2015).



	Basic Language Analyses			Building Language Resources		Dialect Identification and Recognition		Semantic Analysis	
	Morph.	Syntax	Orthog.	Lexica	Corpora	From Text	From Speech	M. Translation	Others
Gulf	(Almeman & Lee, 2012), (Abuata & Al-Omari, 2015)		(Darwish, 2013), (Masmoudi et al., 2015)		(Zaidan & Callison-Burch, 2011), (Almeman et al., 2013), (Cotterell& Callison-Burch, 2014)	(Zaidan & Callison-Burch, 2011), (Sadat, Kazemi, & Farzindar, 2014), (Zaidan & Callison-Burch, 2014)	(Belgacem et al., 2010), (Zaidan&Callison-Burch, 2012), (Zhang et al., 2013), (Biadsy et al., 2009), (Akbacak et al.,2011)	(Jehl et al., 2012), (Salloum & Habash, 2012), (Sawaf, 2010)	(Mourad & Darwish, 2013)
Kuwaiti					(Mubarak & Darwish, 2014)	(Sadat, Kazemi, & Farzindar, 2014)	(Iskra et al., 2004)		
Saudis					(Mubarak & Darwish, 2014)	(Sadat, Kazemi, & Farzindar, 2014)	(Alghamdi et al., 2008), (Iskra et al., 2004)	(Sawaf, 2010)	
UAE					(Mubarak & Darwish, 2014)		(Lei & Hansen, 2009), (Iskra et al., 2004)	(Khamis, 2007)	
Qatari					(Mubarak & Darwish, 2014), (Zaghouani et al., 2014)	(Sadat, Kazemi, & Farzindar, 2014)	(Iskra et al., 2004)	(Al- Mannai et al., 2014)	
Bahraini						(Sadat, Kazemi, & Farzindar, 2014)	(Iskra et al., 2004)		
Omani						(Sadat, Kazemi, & Farzindar, 2014)	(Iskra et al., 2004)		
S. A. Peninsula								(Sawaf, 2010)	
Yemeni					(Belgacem et al., 2010)				
Sana´ani								(Al- Gaphari & Al Yadoumi, 2012)	
North Africa	(Almeman & Lee, 2012), (Habash et al., 2013)		(Masmoudi et al., 2015), (Darwish, 2013)		(Almeman & Lee, 2013)				
Egyptian	(Duh & Kirchhoff 2005), (Habash et al., 2012), (Almeman & Lee, 2012), (Al-Sabbagh & Girju, 2012a), (Salloum &Habash, 2014)		(Dasigi & Diab, 2011), (Habash, Diab, & Rambow, 2012), (Bies et al., 2014)	(Hedar & Doss, 2013)	(Habash et al.,2008), (Diab et al., 2010), (Benajiba & Diab, 2010), (Zaidan & Callison-Burch, 2011), (Al-Sabbagh & Girju, 2012), (Elfardy& Diab, 2012b), (Elfardy& Diab,2012c), (Almeman& Lee,2013), (Mubarak& Darwish, 2014), (Cotterell& Callison-Burch,2014), (Maamouri et al., 2014), (Hawwari et al., 2014), (Maamouri et al.,2014)	(Diab et al., 2010), (Zaidan & Callison-Burch, 2011), (Elfardy & Diab, 2012), (Elfardy & Diab, 2013), (Zaidan & Callison-Burch, 2012), (Habash et al., 2008b), (Zaidan & Callison-Burch, 2014), (Darwish et al., 2014)	(Belgacem et al., 2010), (Zhang et al., 2013), (Lei & Hansen, 2009), (Biadsy et al., 2009), (Akbacak et al., 2011), (Kirchhoff & Vergyri, 2005), (Iskra et al., 2004)	(Zbib et al.,2012), (Salloum & Habash, 2011), (Jehl et al., 2012), (Bakr et al.,2008), (Salloum & Habash, 2012), (Sawaf, 2010), (Mohamed et al., 2012), (Jeblee et al., 2014)	(Pasha et al., 2013), (Hedar & Doss, 2013), (El- Fishawy et al., 2014), (Ibrahim et al., 2015), (Mourad & Darwish, 2013), (Zirikly & Diab, 2014/2015), (El-Beltagy & Ali, 2013), (Darwish & Gao, 2014)
Cairene				(Al-Sabbagh & Girju, 2010)					
Morrocan				(Graff & Maamouri , 2012)	(Benajiba & Diab, 2010), (Diab et al., 2010), (Tratz et al., 2013) , (Mubarak & Darwish,	(Sadat, Kazemi,& Farzindar, 2014)	(Elfardy & Diab, 2012a), (Belgacem et al., 2010), (Iskra et al., 2004)	(Sawaf, 2010), (Tachicart & Bouzoubaa,	



	Basic Language Analyses			Building Language Resources		Dialect Identification and Recognition		Semantic Analysis	
	Morph.	Syntax	Orthog.	Lexica	Corpora	From Text	From Speech	M. Translation	Others
					2014)			2014)	
Tunisian	(Zribi, Khemakhem, & Belguith, 2013), (Boujelbane et al., 2014)		(Zribi et al., 2013), (Zribi et al., 2014)	(Boujelbane et al., 2013)	(Boujelbane et al., 2013), (Zribi, Graja, et al., 2013)	(Sadat, Kazemi, & Farzindar, 2014)	(Belgacem et al., 2010), (Boujelbane et al., 2013), (Iskra et al., 2004)	(Sawaf,2010), (Sadat, Mallek, et al., 2014)	
Libyan				(Graja et al., 2010)		(Sadat, Kazemi, & Farzindar, 2014)	(Iskra et al., 2004)	(Sawaf, 2010)	
Sudani	(Almeman & Lee, 2012)				(Mubarak & Darwish, 2014)	(Sadat, Kazemi, & Farzindar, 2014)		(Sawaf, 2010)	
Algerian					(Harrat et al., 2014)	(Harrat et al., 2015), (Sadat, Kazemi, & Farzindar, 2014)	(Iskra et al., 2004)		
Maghrebi*					(Cotterell & Callison-Burch, 2014)	Zaidan & Callison-Burch, 2012), (Zaidan & Callison-Burch, 2014)			
Levantine	(Habash &Rambow, 2006), (Habash &Rambow,2007) (Almeman & Lee, 2012),	(Chiang et al., 2006), (Maamouri et al., 2006)	(Habash &Rambow, 2007), (Dasigi & Diab, 2011), (Darwish, 2013), (Masmoudi et al., 2015)	(Duh & Kirchhoff 2006)	(Maamouri et al., 2006), (Diab et al., 2010), (Benajiba & Diab, 2010), (Soltau et al., 2011), (Zaidan & Callison-Burch, 2011), (Elfardy& Diab, 2012b), (Almeman& Lee,2013), (Almeman et al., 2013), (Cotterell & Callison-Burch, 2014)	(Habash et al., 2008), (Habash et al., 2008b), (Diab et al., 2010), (Zaidan & Callison-Burch, 2011), (Zaidan & Callison-Burch, 2012), (Elfardy & Diab, 2012c), (Zaidan & Callison- Burch, 2014)	(Elfardy & Diab, 2012a), (Zhang et al., 2013), (Biadsy et al., 2009), (Akbacak et al., 2011), (Iskra et al., 2004)	(Zbib et al., 2012), (Salloum & Habash, 2011), (Jehl et al., 2012), (Salloum & Habash, 2012), (Soltau et al., 2011)	(Mourad & Darwish, 2013)
Syrian				(Graff & Maamouri , 2012)		(Harrat et al., 2015), (Sadat, Kazemi, & Farzindar, 2014)	(Belgacem et al., 2010), (Lei & Hansen, 2009), (Iskra et al., 2004)		
North Syrian								(Sawaf, 2010)	
Damascus								(Sawaf, 2010)	
Lebanese						(Sadat, Kazemi, & Farzindar, 2014)	(Iskra et al., 2004)	(Sawaf, 2010)	
Jordanian	(Salloum &Habash, 2014)					(Sadat, Kazemi, & Farzindar, 2014)	(Iskra et al., 2004)	(Sawaf, 2010)	(Duwairi et al., 2014)
Palestinian					(Jarrar et al., 2014)	(Harrat et al., 2015), (Sadat, Kazemi, & Farzindar, 2014)	(Lei & Hansen, 2009), (Iskra et al., 2004)	(Sawaf, 2010)	
Iraqi	(Almeman & Lee, 2012)		(Masmoudi et al., 2015), (Darwish, 2013)	(Graff et al., 2006), (Rytting et al., 2011), (Graff & Maamouri 2012), (Cavalli-Sforza et al., 2013)	(Diab et al., 2010), (Habash et al., 2008a), (Benajiba & Diab, 2010), (Elfardy & Diab, 2012b), (Cotterell& Callison-Burch, 2014)	(Zaidan & Callison-Burch, 2012), (Zaidan & Callison-Burch, 2014), (Sadat, Kazemi, & Farzindar, 2014)	(Elfardy & Diab, 2012), (Belgacem et al., 2010), (Zhang et al., 2013), (Lei & Hansen, 2009), (Biadsy et al., 2009), (Akbacak et al., 2011)	(Condon et al., 2010), (Salloum & Habash, 2012)	
South Iraqi								(Sawaf, 2010)	
North Iraqi								(Sawaf, 2010)	
Baghdadi								(Sawaf, 2010)	

During my research, I found a number of papers which were not related directly to my research, but some that could help me to choose more seed words, such as the Arabic dialect of Tangier, which belongs to NOR dialect (Aguade 2015), Home Arabic (Kalach 2015), which talks about Hims dialect (LEV dialect). These papers were presented in Association International Dialectologie Arabic (AIDA) (Grigore and Bițună 2015).

According to the previous research that worked to create an Arabic corpus, to build any corpus we need first to decide on the size or length of the corpus (Alsulaiti and Atwell 2005; Mansour 2013). The length of the corpus can be decided depending on the purpose for which it will be used and also the available resources such as funding (Mansour 2013). In addition, the corpus must correspond to the need of the users (Alsulaiti and Atwell 2005). The last consideration in planning a corpus is the type of genres to be included (Mansour 2013).

In my research, I created a dataset by collecting tweets and comments to use it in classification process for training and testing the system. I plan to make the corpus available for other studies after I finish my PhD. I will focus only on what the classifier needs to classify the dialects.

There is a lack of an Arabic dialects corpus, and at the beginning of my research I tried to contact all authors for all papers which I found, in order to create an Arabic dialects corpus. Unfortunately, I did not get an answer except from Almeman and Lee (2013) who sent me their corpus. Moreover, according to what I read, there is no standardization in creating an Arabic dialects corpus, so I used Twitter and Facebook as a social applications that represents a dialectal text and attract a lot of people who freely write in their dialects. Additionally, I used the readers’ comments from online newspaper as a source for long written text.

After I created a new corpus to use it in my research I got access to AOCD (Zaidan and Callison-Burch 2011) and Arabic dialect dataset from (El-Haj *et al.* 2018). I tried also to extract Arabic dialect text from Sketch engine but I found that they label text based on the domain of the website, which sometimes give an incorrect label.

2.4 Dialect Classification

The classification of dialect becomes an important process for other tasks, such as machine translation, dialect-to-dialect lexicons, and information retrieval according to the dialect (Malmasi *et al.* 2015). In fact, there is no standard for writing Arabic dialects because MSA is the formal standardised form of written Arabic (Elfardy and Diab 2012). The following section shows some text classification research that classifies Arabic dialects.

2.4.1 Token and Sentence Level Dialects Identification in Arabic

There are several approaches to classifying Arabic dialects. Some research uses token level to check all tokens one-by-one, and decide if a certain token belongs to this dialect or not; another research study used a sentence-level approach to evaluate a whole sentence and decide whether it belonged to a certain dialect.

A lexicon-based method 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 evaluated using 578 documents and the overall accuracy achieved using the lexicon method is 81.98%.

One paper presents an Automatic Identification of Dialectal Arabic (AIDA). AIDA is a system uses the token level approach to identify a Linguistic Code Switching (LCS) in MSA and Arabic dialects (Egyptian and Levantine). AIDA contains dictionaries, MSA morphological analyser, language models, and sound change rules (Elfardy and Diab 2012). There are two outputs produced for each word; one is a context-insensitive, which means the focus is on the token, not on the context of the word in that sentence, while the second is context-sensitive, which means the focus is on the context of the word in that sentence.

The approach contains four steps:

1- Pre-processing: This is a cleaning step to separate punctuation and numbers and delete any repetition of some characters as a speech effects (Elfardy and Diab 2012). Also, this step includes labelling Latin words, URLs, digits, and punctuation using LAT, URL, NUM, and PUNC class labels.

2- Dialectal Dictionaries: In this step, researchers used the Machine Readable Dictionaries (MRDs) which were developed for the system Tharwa. Tharwa is a three-way dictionary in DA-MSA-English. It consists of 33,955 unique DA lemmas and their equivalents in MSA and English.

3- ALMOR: This step checks if a token is MSA or not, using a system of MSA morphological analysis ALMORGEANA (ALMOR). They assume if the token has an analysis according to ALMOR, and the token is not belong to a pre-defined list of DA, then the token is MSA. Otherwise, the token is DA.

4- Language Models (LM): In this step, to create a language model for MSA they used broadcast news, broadcast conversations, and web-logs; meanwhile, to create a language model for DA, they used dialectal news articles, user commentaries, speech transcription, poems and web-logs (Elfardy and Diab 2012). They collected 13M tokens for each. They then created three lists of n-gram: the first list is Shared-MSA-DA, which contains the shared tokens between MSA and DA; the second list is MSA-Unique, which contains tokens that exist only in MSA; the last list is DA-Unique, which contains tokens that exist only in DA.

The system achieves an accuracy of 74% on words that are context-sensitive, and 84.4% on those that are context-insensitive.

Another research study to classify Arabic dialects used a sentence-level approach to classify whether the sentence was MSA or Egyptian dialect (Elfardy and Diab 2013). They based the study on a supervised approach and used a token level labels approach described in (Elfardy and Diab 2012) to extract sentence-level features. They also used a Naïve Bayes classifier which was trained on labelled sentences. The system used two types of features:

1- Core Features: to indicate if the given sentence is dialectal or non-dialectal (Elfardy and Diab 2013). It was divided into:
   - Token-based Features: used the approach that described in (Elfardy and Diab 2012) to classify each token in the given sentence. In addition, they calculated the percentage of tokens which were analysable by the MSA morphological analyser, and

the percentage of tokens which were analysable by the EDA morphological analyser.

- Perplexity-based Features: calculated the perplexity for MSA and EDA by running each sentence through each of the MSA and EDA LMs. The perplexity indicates the confusion about the sentence, so if the perplexity value is high then this means the given sentence has low priority to match the LM.

2- Meta Features: These are the features that do not directly relate to the dialectal words, but help to estimate whether the sentence is informal or not. It includes, the percentage of punctuation, numbers, and words having word-speech effects. Furthermore, it check to see if the sentence has repeated punctuation, an exclamation mark, or emoticons.

Researchers used WEKA (Hall *et al.* 2009) to train the system by using Naïve-Bayes classifier. The training process consisted of two sets: In the first set, they split the data into training set and held-out test set, while in the second set they used all datasets in the training process (Elfardy and Diab 2013). In the two sets of experiments they applied a 10-fold cross-validation and used an AOCD dataset (Zaidan and Callison-Burch 2011). Table 2.2 shows the number of sentences and tokens used in the datasets. The system accuracy was about 85.5%.

Table 2.2 Number of EDA and MSA sentences and tokens in the training and test datasets. Adopted from (Elfardy and Diab 2013)

	MSA Sent.	**EDA Sent.**	**MSA TOK.**	**EDA TOK.**
Train	12,160	11,274	300,181	292,109
Test	1,352	1,253	32,048	32,648

Another research study introduced AIDA2, which is an improved version of AIDA. They used the same experiments as in the previous studies. They presented a hybrid approach to classify MSA and EDA by using token and sentence-levels classification (Al-Badrashiny *et al.* 2015). The system tried to identify if each token belongs to which dialect and finally decides if the whole sentence belongs to which dialect. In token level classification, they used a

Conditional Random Field (CRF) classifier, which made a decision to label each word in the sentence based on language model and morphological analyser.

In sentence level classification, they used two independent underlying classifiers. After that, they trained another classifier that uses the class labels and the confidence scores generated by each of the two underlying classifiers to decide upon the final class for each sentence.

They first classify each token to one of six tags as defined in (Solorio *et al.* 2014). The tags are:

- lang1: for MSA tokens.
- lang2: for EDA tokens.
- ne: for named tokens.
- ambig: if there is an ambiguity to decide if the token is MSA or EDA.
- mixed: for mixed morphology in the token.
- other: if the token is non Arabic.

To identify the class of a token they used a CRF classifier which is trained using decisions from the following underlying components as shown in Figure 2.6.

- MADAMIRA: is a public morphological tool to analysis and disambiguation of EDA and MSA text (Pasha *et al.* 2014). MADAMIRA uses SAMA (Maamouri *et al.* 2009) to analyse the MSA words and CALIMA (Habash *et al.* 2012) to analyse the EDA words. MADAMIRA uses D3 tokenization method (ex. bAlfryq, “By the team” tokenised as “b+Al+fryq”) (Al-Badrashiny *et al.* 2015).
- Language Model: is built using 119K manually annotated words of the training data from shared task in addition to 8M words from weblogs data, 4M from MSA, and 4M from EDA (Al-Badrashiny *et al.* 2015). The weblogs are automatically annotated based on the word source.
- Modality List: in this step they used ModLex (Al-Sabbagh *et al.* 2013) which is a tool of Arabic modality triggers used to decide the class of lemma; whether it is MSA, EDA, or both depend on context (Al-Badrashiny *et al.* 2015).
- NER: this step works to assign a flag called “isNE” to true for all input entities tagged as ne.

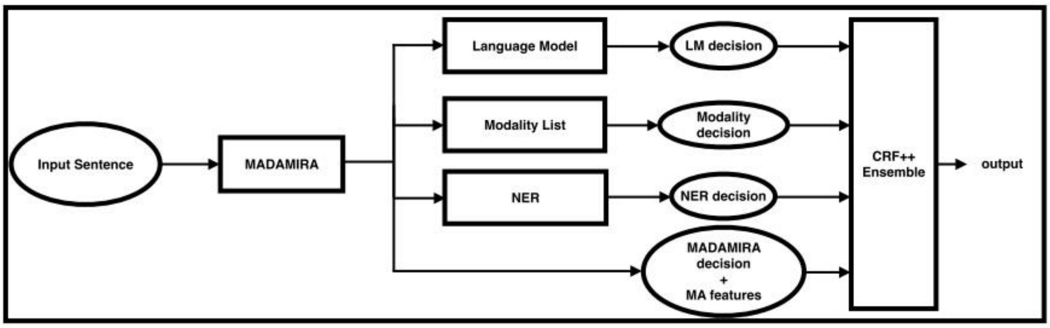


Figure 2.6 Token-level identification pipeline. Adopted from (Al-Badrashiny *et al.* 2015).

By using these components, they generated MADAMIRA-features, LM-features, Modality-features, NER-features, and Meta-features for each word, then they used these features to train the CRF classifier (Al-Badrashiny *et al.* 2015).

The next level is a sentence-level identification, using an ensemble classifier to classify each sentence by generating the class label for each sentence. Figure 2.7 shows the components of sentence-level identification. The process consists of three main components: Comprehensive Classifier (Comp-Cl), Abstract Classifier (Abs-Cl), and DT Ensemble.

- Comp-Cl: This classifier uses the input data as D3 tokenized in addition to the classes for each word generated from Token-Level Identification to cover dialectal statistics, token statistics, and writing style.
- Abs-Cl: This classifier uses the input as surface-level without any tokenisation to covers semantic and syntactic relations between words.
- DT Ensemble: This step takes the results, which are the sentence label and a score for this label from the classifiers to train a decision-tree classifier who decides the class of the input sentence.

The token level achieves an accuracy of 90.6%, and the sentence-level achieves an accuracy of 90.8%.

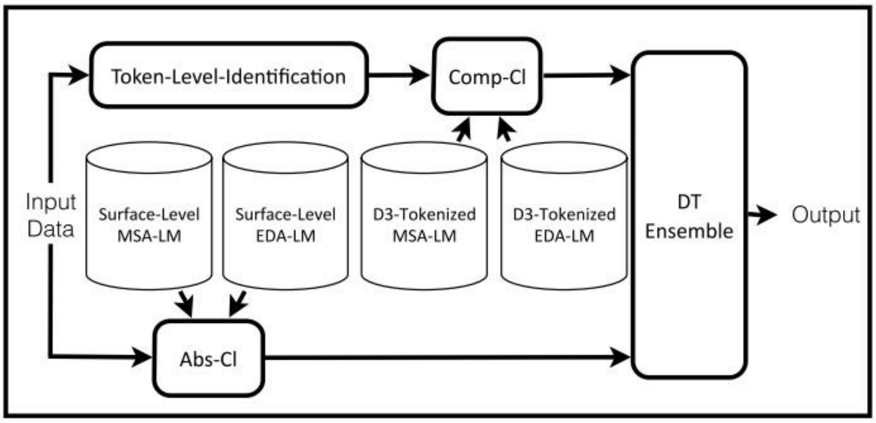


Figure 2.7 Sentence-level identification pipeline. Adopted from (Al-Badrashiny *et al.* 2015).

Algerian dialect identification using an 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 when the term present in the lexicon. The partial identification when the term partially present in the lexicon. The improved Levenshtein when the term present in the lexicon but with different writing. They applied their method on 100 comments collected from Facebook page of Djezzyand the accuracy scored 60%.

We end with a research to classify Arabic dialect 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 word features as a keywords are 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.

2.4.2 Deep Learning for Arabic dialect Identification

Deep learning in classification of Arabic dialect texts is a new topic and recently there is some new research on this topic.

One research applied different deep learning models for classification of Arabic dialectal text (Lulu and Elnagar 2018). The data set used in this paper was Arabic Online Commentary (AOC) (Zaidan and Callison-Burch 2011), which consists of Gulf dialect, Egyptian dialect, and Levantine dialect along with the MSA. The authors used four different deep neural network models to classify Arabic dialect which are Long-Short Term Memory (LSTM), Convolutional Neural Networks (CNN), Bidirectional LSTM (BLSTM), and Convolutional LSTM (CLSTM). The models achieved different accuracies, the highest accuracy scored 71.4% using LSTM, followed by CLSTM with a score of 71.1%, then BLSTM with a score of 70.9%, and the lowest accuracy scored 68.0% using CNN (Lulu and Elnagar 2018).

Another piece of research also used the Arabic Online Commentary (AOC) (Zaidan and Callison-Burch 2011) as a dataset of Arabic dialectal text. The authors used six different deep learning models on the task of classification (Elaraby and Abdul-Mageed 2018). The models were used are: Convolutional Neural Networks (CNN), Long-Short Term Memory (LSTM), Convolutional LSTM (CLSTM), Bidirectional LSTM (BiLSTM), Bidirectional Gated Recurrent Units (BiGRU), and Bidirectional Long-Short Term Memory (BiLSTM). The experiment has been done in three different ways: first way is binary to classify text to dialect or MSA, the second way is 3-way to classify text into one of the three dialects (Egyptian vs. Gulf vs. Levantine), the third way is 4-way to classify text to one of three dialect in addition to MSA. The best accuracy achieved using BiGRU model scored 87.65% on the binary classification, and 87.81% on the 3-way classification, for 4-way classification the accuracy was 83.49% (Elaraby and Abdul-Mageed 2018).

2.5 Arabic Dialect Variations

2.5.1 Phonological Variation

The spoken languages in the Arab world countries before Islam were described in Section 2.1.1 had some effect on the phonology of the Arabic dialect which made Arabic dialects differ phonologically from MSA and each other. Elmahdy *et al.* (2010) and Habash (2010) suggested that these variations between Arabic dialects help users distinguish and recognize one dialect from another. There is no standard orthography or agreed spelling system for Arabic dialect text, and dialect text is often written phonetically, based on the dialect pronunciation of words. The following summary presents some common variations in the pronunciation of some Arabic consonants.

The MSA consonant Qaaf (ق) (q) is pronounced as a glottal stop /ʔ/ in EGY and LEV, as /g/ in GLF, and IRQ. For instance, the word "road" in MSA is pronounced as (طريق) (tˤrjq), in EGY and LEV is pronounced as (طريء) (tˤrjʔ) and in GLF and IRQ is pronounced as (طريج) (tˤrjdʒ). Also, we noticed Qaaf (ق) (q) is pronounced as Kaaf (ك) (k) in IRQ; for instance, the word "time" in MSA is pronounced as (وقت) (wqt) while in IRQ it is pronounced as (وكت) (wkt).

Another variation is in consonant Jiim (ج) (dʒ) which pronounced as (/g/) in EGY and LEV and /j/ in GLF such as the word "chicken" is pronounced as (دجاجه) (ddʒaːdʒh) in MSA, and NOR, while in EGY it is pronounced as (دكاكه) (dgaːgh) , and in GLF and IRQ as (ديايه) (djaːjh), another example, the word “beautiful” is pronounced as dʒamjl in MSA, IRQ and NOR, while in EGY it is pronounced as gamjl and in GLF as jamjl, which means tend to.
Moreover, the consonant Thaa (ث) (θ) in MSA is pronounced as (ت) (t) or (س) (s) in EGY and LEV. For example, the word "three" is pronounced (ثلاثه) (θlaːθh) in NOR, GLF, and IRQ, whereas in EGY and LEV it is pronounced as (تلاته) (tlaːth).

Another example, the word "then" is pronounced as (ثم) (θm) in MSA and GLF; however, in EGY and LEV, it is pronounced as (سم) (sm) which means poison in MSA.

Another difference is in consonant Dhaa (ظ) (ðˤ) , which is pronounced as (ز) (z) in EGY and LEV. The word “appeared” is pronounced as (ظهر) (ðˤhr) in MSA, GLF, and IRQ, while in EGY and LEV it is pronounced as (زهر) (zhr) which means flower in MSA. Table 2.3 summarises the major regional variations in the pronunciation of alphabetic characters in Arabic.

Table 2.3 Regional Variations in Arabic Phonetics

MSA		GLF	EGY	NOR	LEV	IRQ
ق	q	g	ʔ	g	ʔ	k
ج	dʒ	ʤ (or) j	g	ʤ	ʤ	ʤ
ث	θ	θ	s (or) t	t	s (or) t	θ
ذ	ð	ð (or) d	z (or) d	ð	z	ð
ظ	ðˤ	ðˤ	z	ðˤ	z	ðˤ

2.5.2 Phonological and Orthographical Variations

In general, Arabic dialects do not have a standard orthography leading to many spelling variations (Elfardy and Diab 2013).

As mentioned in Section 2.5.1, there are some phonological variations between dialects, and collecting data from Twitter help us to notice some orthographical variations depending on morphological variations.

- To express present verb:
  - NOR dialect: use /k/ and /n/ as a prefix (e.g. كنقولك knqu:lk)
  - IRQ dialect: use /d/ as a prefix (e.g. ديقول djqu:l)
- To express future verb:
  - EGY dialect: use /h/ as a prefix (e.g. هتستخدم htstxdm)
  - LEV dialect: use /t/ as a prefix (e.g. تيكتب tjktb)
- To express question:
  - IRQ dialect: use /ʃ/ as a prefix (e.g. شتريد ʃtri:d)
- To express a pronoun “you”:
  - GLF dialect: /ʤ/ as a suffix (e.g. حقج hgʤ)
- To express a demonstrative pronouns “this”:
  - GLF dialect: /h/ as a prefix (e.g. هالسنين halsni:n)
- To express definite articles:
  - NOR use /l/ in nouns start with moon letters (e.g. لمدرسه lmdrsh)

2.5.3 Lexical Variations

English dialect research has also focussed on phonetic and phonological variation; but lexical variation is also worth study, and can make use of text data written by dialect speakers using standard character sets to try to capture dialect, rather than IPA transcription. For example, "Cheryl Kerl, Woath it? Coase ah am, pet" (Kerl 2010) is a dialect spelling and lexical variant of standard British English "Cheryl Cole, Worth it? Of course I am, dear".

Arabic dialects differ from each other in terms of lexical variation. For instance, the MSA word “tˤa:wlh”, which means “table”, is pronounced as “mi:dh” in NOR, “trbjzh” in EGY, and “mjz” in IRQ. To extract tweets belonging to each dialect, 35 words are used to collect tweets from Twitter. Appendix A contains tables to show the lexical variations between Arabic dialects. Some of these words are used to collect data while the rest of them will be used as features to classify Arabic dialects.

2.6 Machine Learning

Automated learning or Machine Learning (ML) is the process to program computers (machine) to learn from input (training data) and show the output (Shalev-Shwartz and Ben-David 2014).

2.6.1 Types of Machine Learning

Machine Learning has been divided into subfields according to the types of learning tasks and the outcomes (Ayodele 2010; Shalev-Shwartz and Ben-David 2014). The common algorithm types are:

- Supervised learning: This algorithm uses a dependant variables (labels) which is used to predict the outcome by generating a function used to map inputs to desired outputs (Ayodele 2010). Examples of Supervised Learning: Regression, Decision Tree, Random Forest.
- Unsupervised learning: This algorithm does not use a dependant variables (labels), so the model is a set of inputs used for clustering. Examples of Unsupervised Learning: A priori algorithm, K-means (Ayodele 2010).
- Semi-supervised learning: This algorithm uses both labelled and unlabelled inputs to generate a classifier (Ayodele 2010).
- Reinforcement Learning: In this algorithm, the machine is trained to make a decision by observation of the world to learn from past

experience. Example of Reinforcement Learning: Markov Decision Process (Ayodele 2010).

2.6.2 List of Machine Learning Algorithms

The goal of the classification process is to classify items that have similar feature into groups or classes by using supervised learning (Ayodele 2010). The following are points and some descriptions of algorithms based on supervised learning:

- Linear Classifiers
  - Logistic Regression
  - Naïve Bayes Classifier
  - Support Vector Machine
  - Sequential Minimal Optimization
  - Multinomial Naïve Bayes (MNB)
- Quadratic Classifiers
- Boosting
- Decision Tree
  - Random Forest
- Neural Networks
- Bayesian Networks

Linear Classifiers: According to (Ayodele 2010) a linear classifier groups items that have same features "by making a classification decision based on the value of the linear combination of the features" (Timothy Jason Shepard, 1998, cited in Ayodele, 2010, p.24).

- **Naïve Bayes Classifier:** It is used for a very large data set and to solve text classification problems. It calculates a probabilities by counting the frequency of values in the data set (Patil and Sherekar 2013). The algorithm uses Bayes' theorem and works with an assumption of no dependence between attributes, which means any feature in a class is unrelated to any other feature in the class.

- **Support Vector Machine:** Support vector machine (SVM) was developed for numeric prediction classifying data by constructing N-dimensional hyper plane to separate data optimally into two categories

(Ayodele 2010; Witten and Frank 2005). SVM works to find a hypothesis h that reduces the limit between the true error on unseen test data and the error on the training data (Joachims 1998). SVM achieved best performance in text classification task due to the ability of SVM to remove the need for feature selection which means SVM eliminate a high-dimensional feature spaces resulting from the frequent of occurrence of word w_i in text. In addition, SVM automatically find good parameter settings. Figure 2.8 shows an example of the SVM.

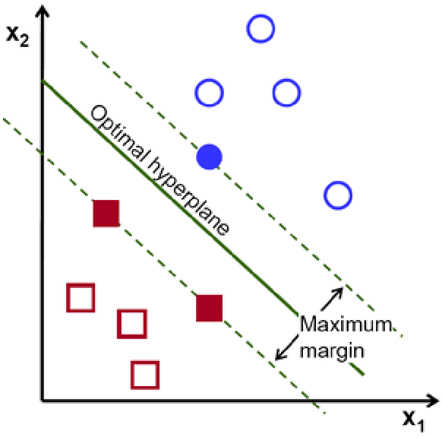


Figure 2.8 The SVM Algorithm. Adopted from OpenCV.com

As in Figure 2.8, the SVM constructs a hyperplane that separates between different set of points based on a vector of features. To predict more accurate classification, the SVM should correctly separate between the different labelled points with a bigger “gap” by normalizing the distances on both sides of the hyperplane from the nearest points which cause the optimization problem (Ma and Saunders 2018).

- **Sequential Minimal Optimization (SMO):** SVM showed a good performance on text categorization, but SVMs training algorithms are slow and complex. For that, Sequential Minimal Optimization (SMO) was developed to solve SVM dual optimization problem (Platt 1998). SMO is an iterative algorithm which works to solve and optimize the quadratic programming problem that appears during the training of SVM by finding the convergence (Ma and Saunders 2018).

- **Multinomial Naïve Bayes (MNB):** The multinomial Naïve Bayes (MNB) used to estimate the conditional probability of a specific word (attribute) according to the frequency of that word in a class (dialect) taking into account the number of appearances of the word in more than one class (Manning *et al.* 2008).

The SMO and MNB were used in the experiments in Chapter 10.

2.7 Feature Selection Methods

Feature selection is one of the important steps in the classification process. It is used to select a subset of tokens or terms that differentiate between classes and exist in the training set to use it as features in text classification (Manning *et al.* 2008; Korde and Mahender 2012). Actually, selecting a good feature will help to decrease the size of the effective vocabulary, will make training more efficient, and will improve the classification accuracy. According to Manning et al. (2008) there are three features selection methods: Mutual, χ2 Feature selection, and Frequency-based feature selection. In order to classify Arabic dialects, the frequency-based feature selection method will be used. This method is based on selecting the most frequent token or term in a class. I used this method to choose some features by using a Sketch Engine (Kilgarriff *et al.* 2014) to create a corpus from the Twitter data, and notice the frequency of words in each dialect. In addition to frequency-based feature selection, this research based on lexical variations to classify dialects.

2.8 Summary

In this chapter Arabic language and its dialects are briefly discussed. The literature review is focused on the previous research on creating Arabic dialect text corpus from Twitter, Facebook, online newspaper, and Web. Moreover, the classification methods used to classify Arabic dialect: token and sentence level. In addition to the phonological and lexical variation between Arabic dialects.

The following chapter presents an initial experiment to classify Arabic dialect text.

Part II

Creating the Arabic Dialect Corpus

Chapter 3
Exploring Twitter as an Arabic Dialect Corpus Source

3.1 Introduction

This chapter explores Twitter as a Source of an Arabic Dialect Corpus source and describes the methods that we used to extract tweets and classify them according to the geographic location of the sender. We classified Arabic dialects by using Waikato Environment for Knowledge Analysis (WEKA) data analytic tool which contains many alternative filters and classifiers for machine learning. Our approach in classifying tweets achieved an accuracy of 79%. This chapter is derived from the published paper under the title Exploring Twitter as a source of an Arabic dialect corpus (Alshutayri and Atwell 2017).

Most research in Arabic dialectology focus on phonetic variation based on audio recordings and listening to dialect speakers (Alorifi 2008; Biadsy *et al.* 2009; Horesh and Cotter 2016; Sadat *et al.* 2014). Horesh and Cotter (2016) confirmed that past and current research is focussed on phonetic and phonological variation between Arabic dialects; all of the examples that they presented are of phoneme variation, and they did not mention any work on text, corpus-based research, lexical, or morpho-syntactic, or grammar variation. Therefore, most Arabic dialectology research collected audio recordings (Horesh and Cotter 2016). In this chapter, we use Twitter to create a dialectal Arabic text corpus by tracking some seed words. Seed words are distinguishing words that are commonly and frequently used in one dialect and not used in any other dialects. In addition, we collect user geographical location information to help verify the results. The chapter is organized as follows: in Section 3.2 we review related work on using Twitter as a source of Arabic Dialects. In Section 3.3 we present our method on how to extract tweets and dialectal words. In Section 3.4 we show the results of the classification process. Finally, Section 3.5 draws conclusion from the data.

3.2 Related Work

Arabic dialect studies have developed rapidly in recent years and most of the previous work has focused on a spoken dialect. Recently people have started using dialect in social media, which makes Twitter a source of written Arabic dialect. A related research project created a Malay text corpus using Twitter (Saloot *et al.* 2016), described in detail in Chapter 2. A multi dialect

Arabic speech parallel corpus was built by an Arabic dialects study (Almeman *et al.* 2013), which created a speech corpus which focused on four main Arabic dialects: MSA, GLF, EGY and LEV; in a domain of travel and tourism. They obtained 67,132 speech files, 15,492 for MSA, 15,492 for GLF, 25,820 for EGY and 10,328 for LEV by recording the dialectal prompts from 52 speakers with an age range of between 16 and 60 years, 49 of which were males and 3 were females. They obtained 32 hours of speech with the average length of prompt being 37 minutes. After recording, they began to segment prompts into audio files in which each file contained one sentence. Mubarak and Darwish (2014) used Twitter to collect an Arabic multi-dialect corpus using the dialectal words from the Arabic Online Commentary Dataset (AOCD), described in (Zaidan and Callison-Burch 2014), both studies are described in Chapter 2.

Another research team, Ali, Mubarak, and Vogel (2014) used the same corpus that was described in (Mubarak and Darwish 2014) to build a language model for the Egyptian dialect as a basis for a speech recognition system which is able to distinguish whether the dialect spoken is Egyptian or not and to recognise the speech accurately (Ali *et al.* 2014). They used 880K tweets written in Egyptian dialect and for speech data they recorded 12.5 hours from Aljazeera Arabic channels. In this thesis, instead of extracting all Arabic tweets like the previous work we tried to extract dialectal tweets by using a filter based on the seed words belonging to each dialect in the Twitter extractor program which connects with Twitter and extracts the dialectal tweets according to the filter conditions. The filter uses a list of seed words for each dialect to decide which tweets to extract for that dialect. In addition, we tried to create a balanced corpus by running the Twitter extractor program for a specific time for each dialect to collect the same number of tweets for all dialects.

3.3 Collecting Tweets

This section is about how we collected tweets and labelled them by the name of the dialect that they represent. In our experiment, we tried to collect dialectal tweets for country groups (5 groups) which are Iraqi Dialect (IRQ), Levantine Dialect (LEV), Egyptian Dialect (EGY), North Africa Dialect (NOR), and Gulf Dialect (GLF). We created an app which connects with the Twitter API[1] to access the Twitter data programmatically.

Our plan for collecting tweets depends on identifying seed words for every dialect. Seed words are distinguishing words that are very common and used very frequently in one dialect and not used in any other dialects. One source for a dialectal word is an Arabic Online Commentary Dataset (AOCD) (Zaidan and Callison-Burch 2011), but we do not have access to this dataset; instead, we have chosen some seed words from Zaidan and Callison- Burch's (2011) paper that described this dataset. The authors collected words for all dialects from readers' comments on the online websites of three Arabic newspapers: Al-Ghad from Jordan to cover the Levantine dialect, Al-Riyadh from Saudi Arabia to cover the Gulf dialect, and Al-Youm Al-Sabe from Egypt to cover the Egyptian dialect. In addition, we used some seed words from (Almeman and Lee 2013). The researchers collected 1,500 words and phrases by exploring the web and extracting the dialects' words and phrases. We did not find a corpus for the Iraqi dialect, but we extracted some IRQ seed words from (Khoshaba 2006). All of the dialect seed words we have chosen seem to be popular and frequently used in its dialect and can usually be heard from native speakers of each dialect, or on TV programs or movies. We tried to use words that could be found in only one dialect and not in other dialects, such as the word مصاري (msˤa:rj), which means "Money" and is used only in LEV dialect; we also used the word دلوقتي (dlwʔti:), which means "now" and is used only in EGY dialect, while in GLF speakers used the word الحين (alħi:n) when they mean "now". In IRQ, speakers change Qaaf (/q/) to (/k/) so they say وكت (wkt), which means "time". Finally, for NOR, which is the dialect most affected by French colonialism and neighbouring countries, speakers used the words بزاف (bza:f) and برشا (brʃa:), which mean "much". Table 3.1 shows examples of the seed words that we used in our experiment.

[1] http://apps.twitter.com

Table 3.1 Example of some seed words for each dialect

GLF	IRQ	LEV	EGY	NOR
lbjh لبيه	ʃtri:d شتريد	mni:ħ منيح	ʕajz عايز	dja:lk ديالك
ʃlwn شلون	ba:wʕ باوع	xtja:r ختيار	bsˤ بص	ʕla:ʃ علاش
amħq أمحق	ʕlmu:d علمود	zlmh زلمه	mfi:ʃ مفيش	gʕmz قعمز

We collected Arabic dialect tweets by using the query lang:ar which extracts all tweets written in the Arabic language, and we tracked 35 seed words all unigram in each dialect, (see Appendix A). Each tweet has a user name and user location. In addition to the tracking of seed words, we used the user location to show the geographical location of the tweets, to be sure that tweets belong to this dialect. The user location was not always available, and sometimes could be a sport club name, street name or landmark name. However, in general, it is usually a country or the name of a city. By running the Twitter extractor for 144 hours, we collected 210,915K tweets with the total number of words equal to 3,627,733 words; these included 44,894K tweets from GLF during 9 hours, 39,582K from EGY during 10 hours, 45,149K from IRQ during 29 hours, 40,248K from LEV during 52 hours, and 41,042K from NOR during 44 hours. Figure 3.1 shows the distribution of tweets per dialect and Table 3.2 shows the number of words that were extracted for each dialect.

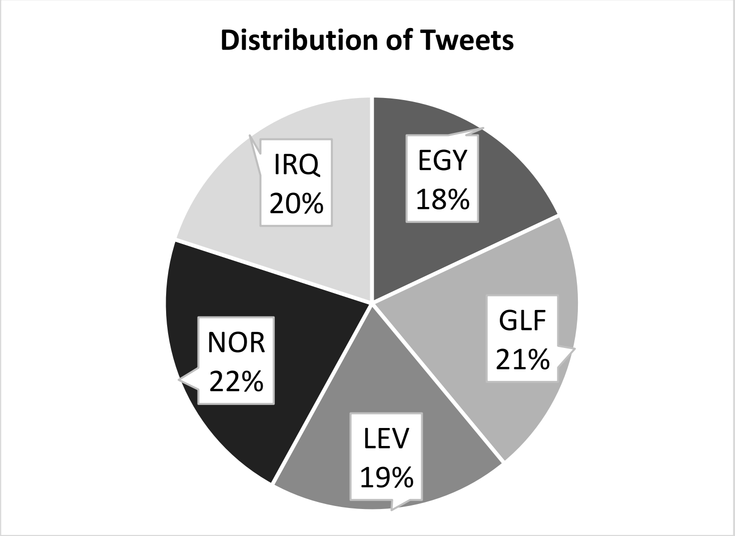


Figure 3.1 The number of tweets collected for each dialect.

Table 3.2 Number of words extracted for each dialect

Dialect	Number of Tokens
GLF	658,893
EGY	558,236
IRQ	905,072
LEV	628,184
NOR	877,348

After collecting the tweets we started to remove noise by using Python to perform a pre-process of the extracted tweets because a lot of tweets contained noise data such as hashtags, emojis, redundant characters, non-Arabic characters, and some bad language.

3.4 Research Experiments and Results

In this section, we describe how we classified the samples of our five major Arabic dialects collected from Twitter using the WEKA toolkit (Hall *et al.* 2009), a widely used tool for data mining that provides a great deal of machine learning algorithms. To classify dialects, the data set is divided into two sets: the first set contains 8,090 labelled tweets used for training and divided unequally between the Arabic dialects: 2,152K from GLF, 1,541K from EGY, 1,585K from NOR, 1,533K from LEV, and 1,279K from IRQ. The second set is for testing and contains 1,764 labelled tweets: 450 from GLF, 326 from EGY, 377 from NOR, 286 from LEV, and 223 from IRQ. For the testing set, we collected new tweets depending only on location, without using any seeds words, then we have manually classified these tweets into the appropriate dialect. We achieved 79% accuracy by using Multinomial Naive Bayes (MNB) algorithm with the WordTokenizer feature to extract words between spaces or any other delimiters such as full-stop, comma, semi colon, colon, parenthesis, question, quotation and exclamation mark.

3.5 Conclusion

Most of the Arabic dialectology corpora are audio recordings, so in this chapter we explored Twitter as a source of Arabic dialect texts to create written corpus of Arabic dialects which is more directly useful for natural language processing research. Our dialect text corpus is more useful for building a classifier to classify dialects than the corpus produced from (Mubarak and Darwish 2014) because we collected a balanced corpus. We have achieved a large corpus of written Arabic dialects texts by dividing the Arab countries into five groups, one for each of the five main dialects: Gulf, Iraqi, Egyptian, Levantine and North African. To distinguish between one dialect and another, we used seed words that are spoken in one dialect and not in the other dialects. In addition, we extracted the user’s location to help us to enhance dialect classification and specify the country and dialect to which each tweet belongs. In general, Twitter can be used as a reference to collect an Arabic dialect text corpus but to make our corpus balanced we had to run the tweet extractor in one dialect longer than another as we noticed that a lot of tweets come from Saudi Arabia, whereas we had fewer tweets from North African countries and Iraq. To classify Arabic dialects we used WEKA and created two sets of data: one as a training set and another as a testing set. We achieved an accuracy of up to 79%.

Chapter 4
Creating an Arabic Dialect Text Corpus by Exploring Online Newspapers

4.1 Introduction

This chapter is about creating an Arabic dialect text corpus by exploring online newspapers. The objective of this chapter is to build an Arabic dialect text corpus using an online commentary from a newspaper. We collected 10,096K comments with a total number of words equal to 309,994K from five groups of Arabic dialects; Gulf, Iraqi, Egyptian, Levantine, and North African. This chapter is derived from the published papers that explored social media as a source of an Arabic dialect corpus (Alshutayri and Atwell 2018b; Alshutayri and Atwell 2018c). It explores an online newspaper as a source and describes the methods that we used to extract comments and then classify them according to the country of the newspaper.

In this chapter, we present our methods to create a corpus of dialectal Arabic by extracting the online commentary from electronic Arabic newspapers as another dialectal Arabic text source.

The chapter is organized as follows: in Section 4.2 we review related works on an Arabic dialects corpus and online newspaper corpus creation. Section 4.3 describes the major variations between Arabic dialects. In Section 4.4 we present the methodology used to collect online newspapers' comments. Finally, Section 4.5 contains the conclusion.

4.2 Related Work

There is a lack of an Arabic dialects corpus, and no standardization in creating an Arabic dialects corpus, so we used Twitter, a social application that represents a dialectal text, because it attracts a lot of people who freely write in their dialects. In addition, in order to incorporate longer dialectal texts, we used online comments texts from Arabic newspapers because Twitter limits the text to140 characters only (at the time of the data collected).

Arabic dialect studies has developed rapidly in recent months. However, any classification of dialects depends on a corpus to use in training and testing processes. There are many studies that have tried to create Arabic dialects corpora; however, many of these corpora do not cover the geographical variations in dialects. In addition, a lot of them are not accessible to the public. This section describes the corpora that were built by previous studies.

A multi dialect Arabic text corpora was built by (Almeman and Lee 2013) using a web corpus as a resource, and has been described in detail in Chapter 2.

Mubarak and Darwish (2014) used Twitter to collect an Arabic multi-dialect corpus, also described in detail in Chapter 2.

Zaidan and Callison-Burch (2014) worked on Arabic Dialects Identification and focused on three Arabic dialects: Levantine, Gulf, and Egyptian. They created a large data set called the Arabic Online Commentary Dataset (AOCD) which contained dialectal Arabic content, described in detail in Chapter 2.

4.3 Arabic Dialects Variations

4.3.1 Phonological Variations

In Chapter 2, we detailed the phonological variations between Arabic dialects which these variations in the pronunciation of some Arabic consonants sometimes notice in written form.

4.3.2 Grammatical Variations

There are some differences between Arabic dialects and MSA in respect of morphology, word order, and sentence structure (Almeman *et al.* 2013). We noticed from the collected data that some grammatical changes happen to dialectal words which originate from MSA.

These changes may occur as a prefix or suffix; for example in the Egyptian dialect the MSA prefix (س) (s) meaning "will" used to express the future is converted to (ـه) (h) or (ح) (ħ). Furthermore, some Arabic dialects add (ش) (ʃ) as a suffix of negation. In addition, there are some changes which occur in stems, for example in Gulf, the MSA word (لك) (lk) which means "yours" the (ك) (k) is converted to (تش) (tʃ), (تس) (ts), or (ج) (dʒ).

4.4 The Arabic Dialects Corpora

The media sources of texts contain people’s opinions written in their dialects are Twitter, forums, Facebook, blogs, and online commentary. The following sections describe our method of collecting the Arabic dialect texts from an online newspaper’s comments section.

4.4.1 Online Newspapers Comments Corpus Creation

The readers’ comments of an online newspaper are another source of dialectal Arabic text. An online comments section was chosen as a resource to collect data because it is public, structured and formatted in a consistent way, which makes it easy to extract (Zaidan and Callison-Burch 2011). Furthermore, we can automatically collect large amounts of data as it is updated every day with new topics.

The readers’ comments were collected from 25 online Arabic newspapers, based on the country which issued each of the newspapers. For example, Ammon for Jordanian comments (LEV dialect), Hespress for Moroccan comments (NOR dialect), Alyoum Alsabe’ for Egyptian comments (EGY dialect), Almasalah for Iraqi comments (IRQ dialect), and Ajel for Saudi comments (GLF dialect). This step was done by exploring the web to search for famous online newspapers in the Arab countries, in addition to asking native speakers about the well-known newspapers in their country.

We endeavoured to make our dataset balanced in terms of sub-corpus size per dialect by collecting around 1000 comments for each dialect. Then, we classified texts and labelled each according to the country that issued the newspaper. In addition, to ensure that each comment belonged to the dialect for which it was labelled, we applied the Twitter seed filter to the newspaper comments: the comments were automatically reviewed against the list of seed words created to collect tweets, checking words in the comment to confirm it belonged to the assigned dialect. However, we encountered some difficulty with comments because lots of comments, especially from GLF sub-corpus, were actually written in MSA, which affected the results of automatic labelling; so we found that we also needed to review and sometimes re-label the comments manually using an annotation tool (Alshutayri and Atwell 2018a), see Chapter 6. The last step was cleaning the collected comments by removing repeated comments and any unwanted symbols or spaces.

Around 10K comments were collected by crawling the newspaper sites during a two month period. The total number of words was 309,994K words; these included 90,366K words from GLF, 31,374K from EGY, 43,468K from IRQ, 58,516K from LEV, and 86,270K from NOR. Figure 4.1 shows the distribution of words per dialect.

We planned to collect readers' comments from each country in the five groups of dialects. For example, comments from Saudi Arabian newspapers and comments from Kuwait newspapers covered the Gulf dialect and so on for all dialects, but in some countries such as Lebanon and Qatar we did not find a lot of comments.

Table 4.1 shows the number of comments from each country.

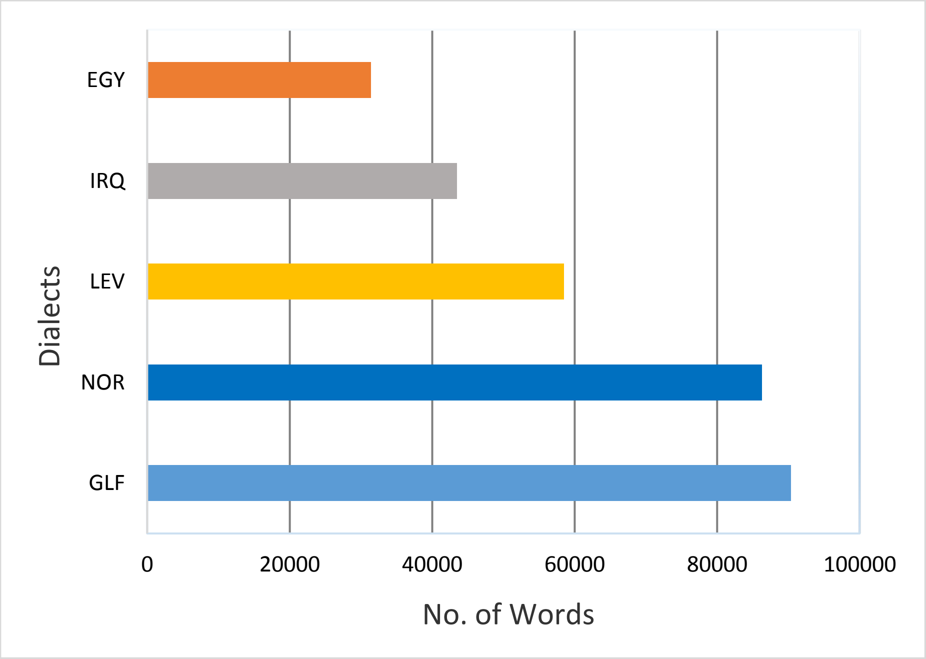


Figure 4.1 The number of words collected from comments on online newspaper for each dialect.

Table 4.1 Number of comments for each country

Dialect		No. of comments
EGY	Egypt	719
IRQ	Iraq	1029
GLF	Kuwait	1189
	Saudi Arabia	1020
	Bahrain	1018
	Emirates	221
LEV	Jordan	1176
	Syria	1034
	Palestine	63
NOR	Morocco	1190
	Algeria	1060
	Libya	313
	Tunisia	64

4.5 Conclusion

This chapter has explored using comments found in online newspapers as a reference for Arabic dialects. We divided the Arab countries into five groups, one for each of the five main dialects: Gulf, Iraqi, Egyptian, Levantine and North African.

We considered online comments in newspapers to be a good source of dialectal Arabic, especially if the article talks about things that are specifically interesting to the people of this particular country; for example articles about living conditions and high cost of living, art, or sport; if the topic of the article is about political news, many readers’ comments use MSA instead of their dialect, so a lot of comments mix MSA and dialect. The comments were classified based on the country that issued the newspaper.

Chapter 5
Extending the Arabic Dialect Corpus

This chapter is based on Creating an Arabic Dialect Text Corpus by Exploring Twitter, Facebook, and Online Newspapers and A Social Media Corpus of Arabic Dialect Text (Alshutayri and Atwell 2018b; Alshutayri and Atwell 2018c). It presents how we extended the Social Media Arabic Dialect Corpus (SMADC) by collecting more tweets from Twitter based on coordinate points, and scrape Facebook posts to collect users' comments on Facebook posts.

5.1 Tweets Based on Spatial Coordinate Points

Chapter 3 shows a method used to collect tweets based on seed terms. In this chapter we extend the Arabic dialect corpus to be sure that all dialectal text is covered, also examples with different terms not just the seed terms which were used to collect tweets previously. So, we used another method to collect tweets based on the spatial coordinate points of each country using the following steps:

1. Use the same app that was used in Chapter 3 to connect with the Twitter API[2] and access the Twitter data programmatically.
2. Use the query lang:ar which extracts all tweets written in the Arabic language.
3. Filter the extracted tweets by tracking the spatial coordinate points (longitude and latitude) for each dialect area using a website to find latitude and longitude (Zwiefelhofer 2008) to be sure that the extracted tweets belong to a specific dialect. We specified the spatial coordinate points for capital cities in north African countries, Gulf Arabian countries, Levantine countries, Egypt and Iraq. In addition we also used the spatial coordinate points of the big cities in each country:
    a. The spatial coordinate points of Rabat from Morocco, Algiers from Algeria, Tunis from Tunisia, and Tripoli from Libya. In addition to other cities, such as Casablanca, Marrakesh, and Agadir from Morocco, Oran, Annaba, and Ouargla from Algeria, Sfax, Sousse, and Al-Qayrawan from Tunisia, and Misrata,

[2] http://apps.twitter.com

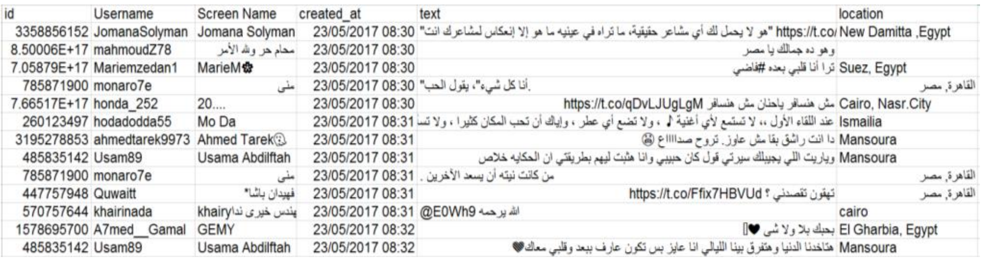
Benghazi, Sabha from Libya are used to cover NOR dialect.

b. The spatial coordinate points of Cairo, Alexandria, Port Said, Asyut, Sohag, Tanta, and Luxor are used to cover EGY dialect.
c. The spatial coordinate points of Baghdad, Ramadi, Karbala, Najaf, Kirkuk, Mosul, Erbil, Sulaymaniyah, Al-Falluujah, Nasiriyah, and Basrah are used to cover IRQ dialect.
d. The spatial coordinate points of Amman from Jordan, Damascus from Syria, Beirut from Lebanon, Jerusalem from Palestine. In addition to Irbid, Az-Zarqa, Jerash from Jordan, Aleppo, Hama, Homs, Latakia, Tartus from Syria, Tripoli, Byblos, Baalbek from Lebanon, and Gaza, Nablus, Ramallah, and Haifa from Palestine are used to cover LEV dialect,.
e. The spatial coordinate points of Riyadh from Saudi Arabia, Kuwait from Kuwait, Abu Dhabi from United Arab Emirates, Doha from Qatar, and Manama from Bahrain. In addition to Jeddah, Makkah, Medina, Dammam, Tabuk and Abha from Saudi Arabia, Dubai, and Ras al-Khaimah, from UAE, and Ar-Rayyan and Al Khor from Qatar are used to cover GLF dialect.

Appendix B contains a table shows the longitude and latitude were used to collect tweets from the specified areas for each city. These spatial coordinate points helped us to collect tweets from the specified arears but to collect tweets which have different subjects and contain different dialectal terms we ran the API at several different time periods to cover a wider variety of topics and events. Figure 5.1 shows the screenshot of the extracted tweets in .CSV file. In addition to tweets we extracted some meta data could help us in other research such as, the user’s name, id, screen name, and location if it written in the user’s profile, beside the date which we ran the API in it.

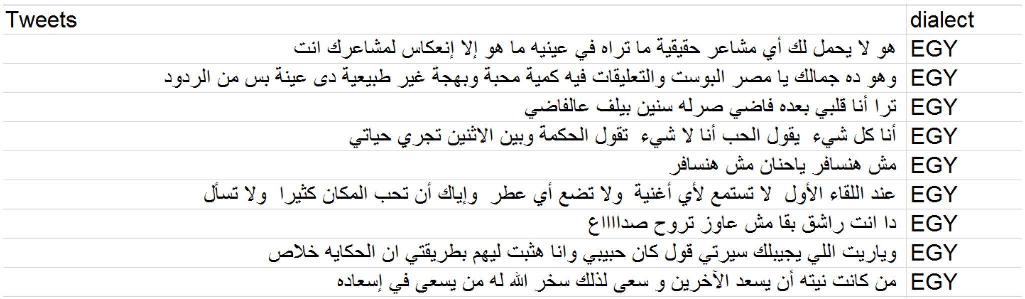
4. Finally, we extracted the users’ tweets from .CSV files to clean the tweets and delete all emojis, non-Arabic characters, all symbols such as (#, _, “), question mark, exclamation mark, and links using a script written in Python and created a new .CSV file for each dialect and label each tweet with its dialect based on the spatial coordinate points which

were used to collect this tweets, Figure 5.2 shows the screenshot of the result from this step.



id	Username	Screen Name	created_at	text	location
3358856152	JomanaSolyman	Jomana Solyman	23/05/2017 08:30	"هو لا يحمل لك أي مشاعر حقيقية، ما تراه في عينيه ما هو إلا إنعكاس لمشاعرك انت" https://t.co/	New Damitta ,Egypt
8.50006E+17	mahmoudZ78	محام حر وله الأمر	23/05/2017 08:30	وهو ده جمالك يا مصر	
7.05879E+17	Mariemzedan1	MarieM✿	23/05/2017 08:30	ترا أنا قلبي بعده #فاضي	Suez, Egypt
785871900	monaro7e	منى	23/05/2017 08:30	"أنا كل شيء"، يقول الحب.	القاهرة, مصر
7.66517E+17	honda_252	20....	23/05/2017 08:30	مش هنسافر ياحنان مش هنسافر https://t.co/qDvLJUgLgM	Cairo, Nasr.City
260123497	hodadodda55	Mo Da	23/05/2017 08:31	عند اللقاء الأول ،، لا تستمع لأي أغنية ♪ ، ولا تضع أي عطر ، وإياك أن تحب المكان كثيرا ، ولا تسا	Ismailia
3195278853	ahmedtarek9973	Ahmed Tarek☺	23/05/2017 08:31	دا انت راشق بقا مش عاوز. تروح صدااااع 😫	Mansoura
485835142	Usam89	Usama Abdilftah	23/05/2017 08:31	وياريت اللي يجيبلك سيرتي قول كان حبيبي وانا هثبت ليهم بطريقتي ان الحكايه خلاص	Mansoura
785871900	monaro7e	منى	23/05/2017 08:31	من كانت نيته أن يسعد الأخرين .	القاهرة, مصر
447757948	Quwaitt	فهيدان باشا*	23/05/2017 08:31	تهقون تقصدني ؟ https://t.co/Ffix7HBVUd	القاهرة, مصر
570757644	khairinada	khairyهندس خيرى ندا	23/05/2017 08:31	@E0Wh9 الله يرحمه	cairo
1578695700	A7med__Gamal	GEMY	23/05/2017 08:32	بحبك بلا ولا شى ♥▯	El Gharbia, Egypt
485835142	Usam89	Usama Abdilftah	23/05/2017 08:32	هتاخدنا الدنيا وهتفرق بينا الليالي انا عايز بس تكون عارف ببعد وقلبي معاك♥	Mansoura

Figure 5.1 Screenshot of the tweets .CSV file (Before pre-processing).



Tweets	dialect
هو لا يحمل لك أي مشاعر حقيقية ما تراه في عينيه ما هو إلا إنعكاس لمشاعرك انت	EGY
وهو ده جمالك يا مصر البوست والتعليقات فيه كمية محبة وبهجة غير طبيعية دى عينة بس من الردود	EGY
ترا أنا قلبي بعده فاضي صرله سنين بيلف عالفاضي	EGY
أنا كل شيء يقول الحب أنا لا شيء تقول الحكمة وبين الاثنين تجري حياتي	EGY
مش هنسافر ياحنان مش هنسافر	EGY
عند اللقاء الأول لا تستمع لأي أغنية ولا تضع أي عطر وإياك أن تحب المكان كثيرا ولا تسأل	EGY
دا انت راشق بقا مش عاوز تروح صدااااع	EGY
وياريت اللي يجيبلك سيرتي قول كان حبيبي وانا هثبت ليهم بطريقتي ان الحكايه خلاص	EGY
من كانت نيته أن يسعد الآخرين و سعى لذلك سخر الله له من يسعى في إسعاده	EGY

Figure 5.2 Screenshot of the tweets .CSV file (After pre-processing).

Using this method to collect tweets based on spatial coordinate points for one month, we obtained 112,321 tweets from different countries in the Arab world. We got 44,619 tweets from GLF dialect, 23,809 tweets from EGY dialect, 15,473 tweets from IRQ dialect, 14,790 tweets from LEV dialect, 13,630 tweets from NOR dialect. After the cleaning step and deletion of redundant tweets, we got 107,229 tweets, divided into 43,252 tweets from GLF dialect, 23,483 tweets from EGY dialect, 14,511 tweets from IRQ dialect, 12,944 tweets from LEV dialect, 13,039 tweets from NOR dialect. Figure 5.3 shows the distribution of tweets per dialect. We noticed that we can extract lots of tweets from the GLF dialect in comparison to LEV, IRQ, NOR and EGY. We speculate that this is because Twitter is not as popular in these dialects' countries as Facebook; and internal problems in some countries affected the ease of use of the Internet.

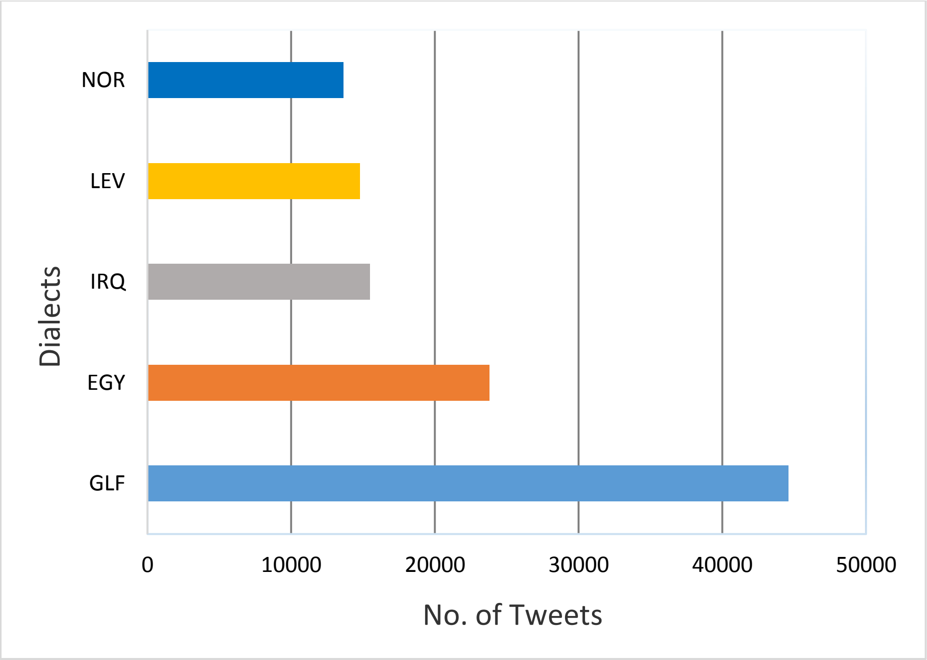


Figure 5.3 The distribution of tweets collected for each dialect.

5.2 Comments from Facebook

The text in Twitter does not exceed 140 characters (at the time of collecting the tweets), so we tried to explore another sources of text that contains more dialectal words without a limit. The other source of Arabic dialect texts is Facebook which is considered one of the famous social media applications in the Arab world. Lots of users write in Facebook using their dialects. We collected comments by following the steps below:

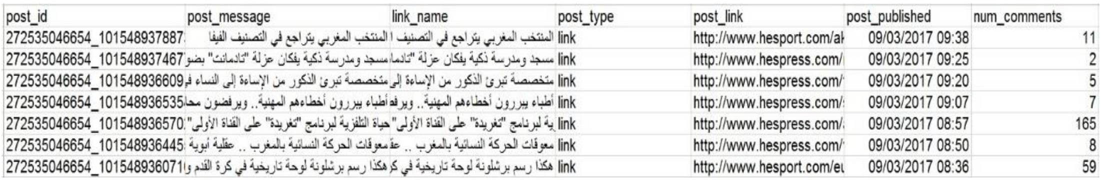
1. To collect the Facebook comments, the Facebook pages used to scrape timeline posts and comments were chosen by using Google to search about the most popular Arabic pages on Facebook in different domains such as, sport pages, comedy pages, channel and program pages, and news pages.
2. The result from the first step was a list of Arabic Facebook pages. We checked every page to confirm it had 50,000 or more followers, posts and comments, then we created a final list of pages to scrape posts.
3. We created an app which connects with the Facebook Graph API[3] to access and explorer the Facebook data programmatically. The app worked in steps:

[3] https://developers.facebook.com/

a. First, it collected all posts of the page starting from the date the page was established until the day that the app was executed. The result of this step was a file of type Comma Separated Values (CSV) for each page contained a list of post ids for each page which was used to scrape comments from each post, in addition to some metadata for each post: post type, post link, post published date, and the number of comments in each post. These metadata may help us in our research or other researchers, Figure 5.4 shows the screenshot of the result from this step.

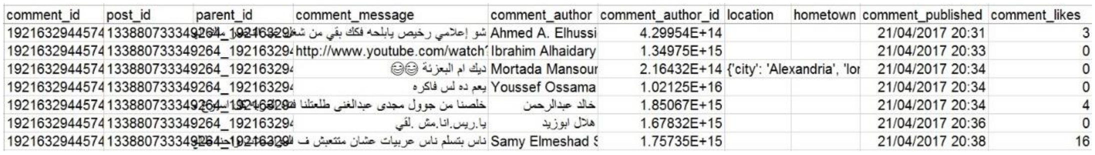
b. Then, the results of the previous step for each page were used to scrape comments for each post based on the post id. The result of this step was .CSV file contained a list of comment messages and metadata: comment id, post id, parent id of the comment if the comment is a reply to another comment, comment author name and id, comment location if the author added the location information in his/her page, comment published date, and the number of likes for each comment, Figure 5.5 shows the screenshot of the result from this step.

4. In the third step, the comment id and message extracted from the previous step was labelled with the dialect based on the country of the Facebook page which was used to collect the posts from it.
5. In the last step, a Python script was created to pre-process (clean) the comment message and delete all emojis, non-Arabic character, all symbols such as (#, _, “), question mark, exclamation mark, and links, Figure 5.6 shows the screenshot of the result from this step.



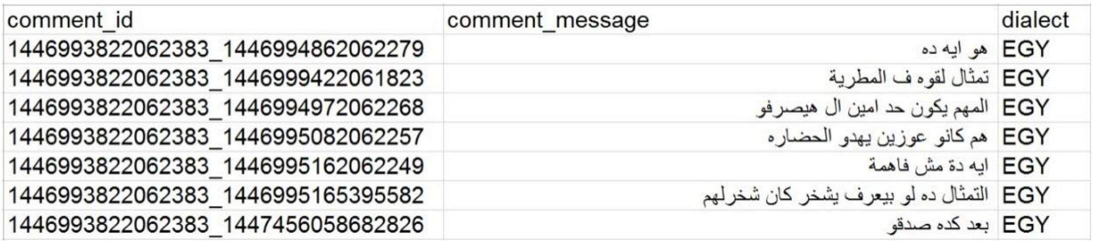
post_id	post_message	link_name	post_type	post_link	post_published	num_comments
272535046654_101548937887	المنتخب المغربي يتراجع في التصنيف الفيفا	المنتخب المغربي يتراجع في التصنيف ا	link	http://www.hesport.com/al	09/03/2017 09:38	11
272535046654_101548937467	مسجد ومدرسة ذكية يفكان عزلة "تادمانت" بضو	مسجد ومدرسة ذكية يفكان عزلة "تادما	link	http://www.hespress.com/	09/03/2017 09:25	2
272535046654_101548936609	متخصصة تبرئ الذكور من الإساءة إلى النساء فر	متخصصة تبرئ الذكور من الإساءة إلى	link	http://www.hespress.com/	09/03/2017 09:20	5
272535046654_101548936535	أطباء يبررون أخطاءهم المهنية.. ويرفضون محا	أطباء يبررون أخطاءهم المهنية.. ويرف	link	http://www.hespress.com/	09/03/2017 09:07	7
272535046654_101548936570	"حياة التلفزية لبرنامج "تغريدة" على القناة الأولى"	ية لبرنامج "تغريدة" على القناة الأولى"	link	http://www.hespress.com/	09/03/2017 08:57	165
272535046654_101548936445	معوقات الحركة النسائية بالمغرب .. عقلية أبوية	معوقات الحركة النسائية بالمغرب .. عق	link	http://www.hespress.com/	09/03/2017 08:50	8
272535046654_101548936071	هكذا رسم برشلونة لوحة تاريخية في كرة القدم وا	هكذا رسم برشلونة لوحة تاريخية في كر	link	http://www.hesport.com/eu	09/03/2017 08:36	59

Figure 5.4 Screenshot of the posts .CSV file.



comment_id	post_id	parent_id	comment_message	comment_author	comment_author_id	location	hometown	comment_published	comment_likes
1921632944574	133880733349264_19216329		شو إعلامي رخيص يابلحه فكك بقي من شغ	Ahmed A. Elhussi	4.29954E+14			21/04/2017 20:31	3
1921632944574	133880733349264_19216329		http://www.youtube.com/watch?	Ibrahim Alhaidary	1.34975E+15			21/04/2017 20:33	0
1921632944574	133880733349264_19216329		ديك ام البعزنة 😄😄	Mortada Mansour	2.16432E+14	{'city': 'Alexandria', 'lor		21/04/2017 20:34	0
1921632944574	133880733349264_19216329		يعم ده لس فاكره	Youssef Ossama	1.02125E+16			21/04/2017 20:34	0
1921632944574	133880733349264_19216329		خلصنا من جوول مجدى عبدالغنى طلعتلنا ف	خالد عبدالرحمن	1.85067E+15			21/04/2017 20:34	4
1921632944574	133880733349264_19216329		يا.ريس.انا.مش .لقي	هلال ابوزيد	1.67832E+15			21/04/2017 20:36	0
1921632944574	133880733349264_19216329		ناس بتسلم ناس عربيات عشان متتعبش ف	Samy Elmeshad S	1.75735E+15			21/04/2017 20:38	16

Figure 5.5 Screenshot of the comments .CSV file.



comment_id	comment_message	dialect
1446993822062383_1446994862062279	هو ايه ده	EGY
1446993822062383_1446999422061823	تمثال لقوه ف المطرية	EGY
1446993822062383_1446994972062268	المهم يكون حد امين ال هيصرفو	EGY
1446993822062383_1446995082062257	هم كانو عوزين يهدو الحضاره	EGY
1446993822062383_1446995162062249	ايه دة مش فاهمة	EGY
1446993822062383_1446995165395582	التمثال ده لو بيعرف يشخر كان شخرلهم	EGY
1446993822062383_1447456058682826	بعد كده صدقو	EGY

Figure 5.6 Screenshot of the final comments .CSV file for each dialect.

The extractor program connected to API to scrape Facebook and ran for one month. At the end, we had obtained a sufficiently large quantity of text to create an Arabic dialect corpus and use it for training and test data for Machine Learning classification purposes. The total number of collected posts was 422,070 and the total number of collected comments was 2,888,788. Our data comprised 488,607 comments from EGY dialect, 508,695 comments from NOR dialect, 125,495 comments from GLF dialect, 146,821 comments from IRQ dialect, 302,502 comments from LEV dialect, and 1,316,668 comments with a mix of dialects. After the cleaning step we kept 1,389,505 comments, divided into 263,596 comments from EGY dialect, 212,712 comments from NOR dialect, 106,590 comments from GLF dialect, 97,672 comments from IRQ dialect, 132,093 comments from LEV dialect, and 576,842 comments of mixed dialects.

Table 5.1 shows the number of posts and comments collected for each Facebook page.

We wanted to make SMADC balanced by collecting the same number of comments for each dialect, but we did not find Facebook pages rich with comment for some countries such as Kuwait, UAE, Qatar, and Bahrain. Figure 5.7 shows the number of comments collected for each dialect. We noticed that the number of comments in IRQ and GLF are smaller compared to other dialects. We speculate that this is due to a lower number of Facebook pages for some dialects due to unpopularity of Facebook in the Gulf area in comparison with Twitter, and due to the poor telecommunications network coverage in Iraq due to the impact of war. We collected a higher number of comments for NOR dialect because, similar to North African countries, Facebook is more popular than Twitter.

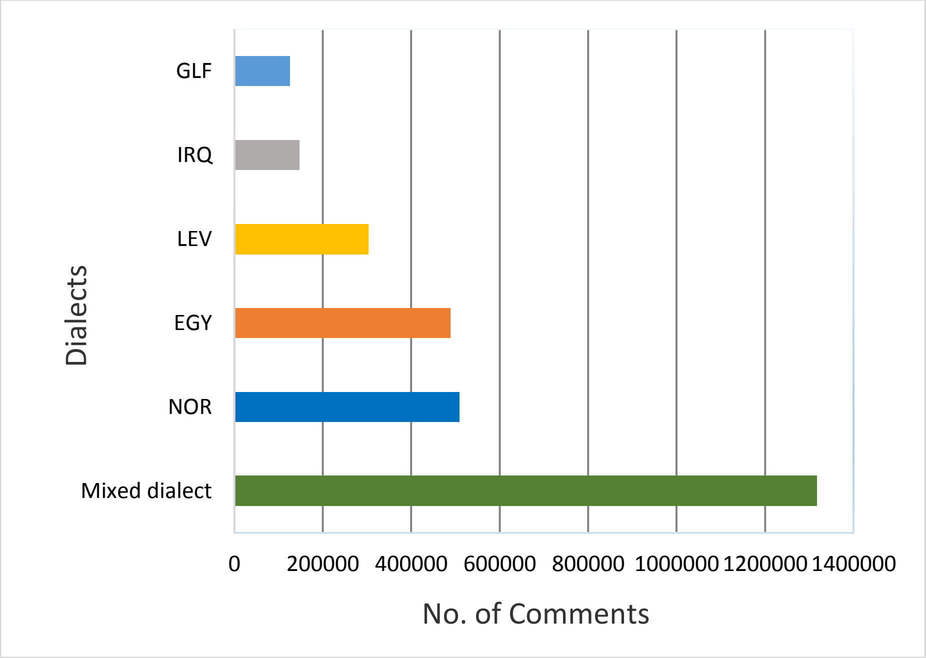


Figure 5.7 The distribution of Facebook comments collected for each dialect.

Table 5.1 The number of posts and comments collected from each country.

Dialect	Country	Facebook Page	Post Count	Comments
EGY	Egypt	asa7bess	3,204	163,557
		Vodafone.Egypt	4,774	35,256
		womenconfused	169	407
		Youm7	355,906	289,387
NOR	Algeria	123VivaDzcom	892	8,509
	Tunisia	Blid.Tounis	1,257	77,917
	Morocco	Hespress	5,013	420,149
	Libya	libyaakhbar	2,750	2,120
GLF	Saudi Arabia	ksauniv group	10,557	95,296
		ActionYaDawry	358	2,128
		AhmadAlShugairi	1,899	2,981
		BabRizq	2,023	22,729
		sabq.org	500	2,361
IRQ	Iraq	AJA.Iraq	292	9,813
		aliraqOfficiaal	328	13,607
		AR.SonGs	917	41,300
		iraqiajeeb	3,282	45,840
		IraqiProPlayers	300	36,261
LEV	Jordan	al.ordonn	3,831	73,959
	Palestine	lahza.blahza	4,510	110,058
	Lebanon	lebanonpic	989	117,635
	Syria	syriaalyom	3,902	850
All Dialects	Arab World	3ajeyeb	4,797	549,380
		ArabIdol	2,691	242,361
		arabsgottalent	3,364	338,611
		MBC.Group	294	6,642
		sadaalmalaeb	3,271	179,674

5.3 Conclusion

In this chapter we extended the corpus by collected a new tweets based on spatial coordinate points for each city in different countries. In addition to scrape Facebook posts and extracted all comments from these posts. In general we could say these two methods help us to collect more annotated dialectal texts in around 70% but still we noticed even with using the spatial coordinates points there are some overlap between these points and we need to annotate SMADC manually to be sure that the texts classified according to the text dialect.

Chapter 6
Arabic Dialect Texts Annotation

This chapter explores Arabic dialect annotation using an online game. It presents our method on crowdsourcing Arabic dialect annotation. In this chapter, the second section presents why the annotation process is important. The third section describes the method used to annotate the collected dataset to build a corpus of Arabic dialect texts. The fourth section shows how we evaluate the annotated results. The fifth section presents the result and the number of annotated documents. Finally, the last section presents the conclusion. This chapter is derived from the published paper under the title Arabic Dialects Annotation using an Online Game (Alshutayri and Atwell 2018a).

6.1 Annotation Tool

Some tweets were collected based on spatial coordinate points and some tweets were based on seed terms which are distinguished words that are very common in one dialect and not used in any other dialects, as explain in Chapter 3. The total number of tweets is 280K, and there are 2M comments from Facebook. In addition, 10K comments by trawling through newspaper websites over a period of two months. Table 6.1 shows the total number of words for each text source.

Table 6.1 The Total Number of Words from each text source.

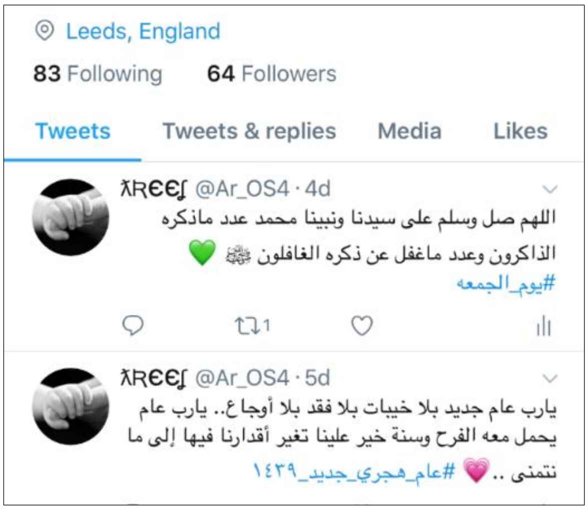
Source	Number of Words
Twitter	6,827,733
Facebook	7,056,812
Newspaper	3,318,717

To annotate each sentence with the correct dialect, we explored a novel approach to crowdsourcing corpus annotation. We developed the task of annotation as an online game, where players can test their dialect classification skills and receive a score representing the level of their knowledge.

6.1.1 Importance of the Annotation Tool

We participated in the VarDial2016 workshop at COLING 2016 Discriminating Similar Languages (DSL) 2016 shared task (Alshutayri *et al.* 2016). The shared task offered two tasks. The first task worked on the identification of very similar languages in newswire texts. The second task focused on Arabic dialect identification in speech transcripts (Malmasi *et al.* 2016). The Arabic dialect texts used for training and testing were developed using the QCRI Automatic Speech Recognition (ASR) QATS system (Khurana and Ali 2016) to label each document with a dialect (Ali *et al.* 2016). Some evidently mislabelled documents were found which affected the accuracy of classification; so, to avoid this problem, a new text corpus and labelling method were created.

In the first step of labelling the corpus, we initially assumed each tweet could be labelled based on the location that appears in the user's profile and the spatial coordinate points which we used to collect the tweets from Twitter. As for the comments, they were collected from online newspapers, and each comment was labelled based on the country in which the newspaper is published. Finally, for the comments collected from Facebook posts, each comment was labelled based on the country of the Facebook page and, if a famous public group or person owns it, depending on the nationality of the owner of the Facebook page. However, through the inspection of the corpus, we noticed some mislabelled documents due to disagreement between the locations of the users and their dialects. So, we needed to verify that the document is labelled with the correct dialect. Figure 6.1 gives an example of the confusion between the user location and their dialect.


Leeds, England

83 Following 64 Followers

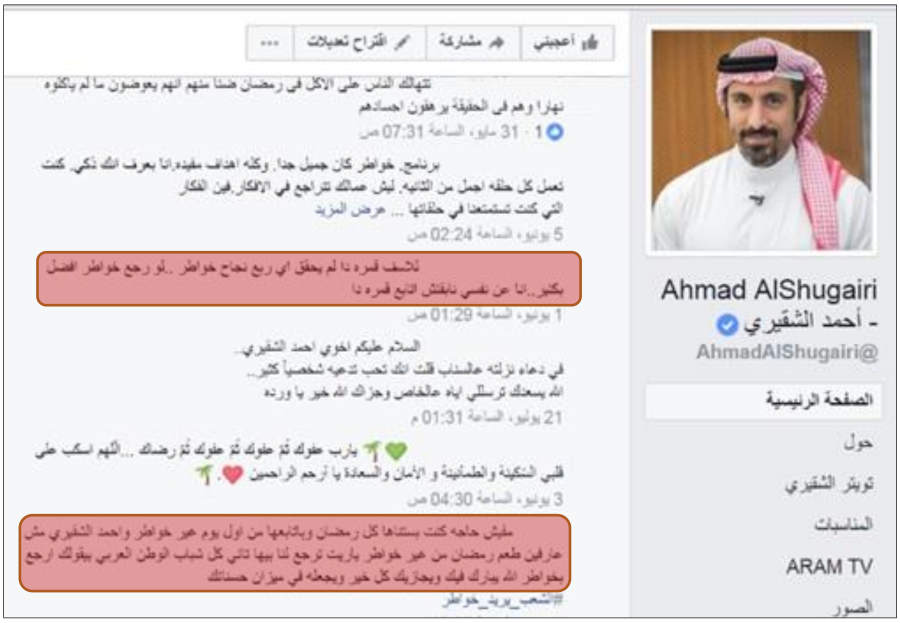
ℵREE∫ @Ar_OS4 · 4d
اللهم صل وسلم على سيدنا ونبينا محمد عدد ماذكره
الذاكرون وعدد ماغفل عن ذكره الغافلون ﷺ 💚
#يوم_الجمعه

1

ℵREE∫ @Ar_OS4 · 5d
يارب عام جديد بلا خيبات بلا فقد بلا أوجاع.. يارب عام
يحمل معه الفرح وسنة خير علينا تغير أقدارنا فيها إلى ما
نتمنى .. 💗 #عام_هجري_جديد_١٤٣٩

Figure 6.1 Example of user location and his tweets.

As shown in Figure 6.1 the user location is England while the tweets are written using Arabic, so in this case we should not label tweets based on spatial coordinate points. Similarly, for Facebook comments as shown in Figure 6.2, the Facebook page's country based on the nationality of the page owner is Saudi Arabia, but some comments were not written in GLF dialect as we supposed in our method of labelling, such as the highlighted comment in the Figure 6.2.



تتهالك الناس على الاكل في رمضان ظنا منهم انهم يعوضون ما لم ياكلوه
نهارا وهم في الحقيقة يرهقون اجسادهم
1 · 31 مايو، الساعة 07:31 ص

برنامج خواطر كان جميل جدا. وكله اهداف مفيدة،انا بعرف انك ذكي. كنت
تعمل كل حلقه اجمل من الثانيه. ليش صالك تتراجع في الافكار.فين الفكار
التي كنت تستمتعنا في حلقاتها ... عرض المزيد
5 يوليو، الساعة 02:24 ص

للاسف قمره دا لم يحقق اي ربع نجاح خواطر ..لو رجع خواطر افضل
بكتير..انا عن نفسي مابقتش اتابع قمره دا
1 يوليو، الساعة 01:29 ص

السلام عليكم اخوي احمد الشقيري..
في دعاء نزلته عالسناب قلت انك تحب تدعيه شخصياً كثير..
الله يسعدك ترسللي اياه عالخاص وجزاك الله خير يا ورده
21 يوليو، الساعة 01:31 م

💚🌴 يارب عفوك ثم عفوك ثم عفوك ثم رضاك ...اللهم اسكب على
قلبي السكينة والطمانينة و الأمان والسعادة يا أرحم الراحمين ❤️.🌴
3 يوليو، الساعة 04:30 ص

مفيش حاجه كنت بستناها كل رمضان وبتابعها من اول يوم غير خواطر واحمد الشقيري مش
عارفين طعم رمضان من غير خواطر ياريت ترجع لنا بيها تاني كل شباب الوطن العربي بيقولك ارجع
بخواطر الله يبارك فيك ويجازيك كل خير ويجعله في ميزان حسناتك
#الشعب_يريد_خواطر

Ahmad AlShugairi
أحمد الشقيري -
AhmadAlShugairi@

Figure 6.2 Example of the Facebook page's country and the users comments.

6.1.2 Description of the Annotation Tool

We used a Hypertext Markup Language (HTML) to create the website homepage and Java programing language to program the website, because Java helped us to connect to MYSQL database to select from the database or insert into the database. We did this by using Java Server Page (JSP), which is a technology helps to create dynamic web pages which can interact with a user (Oracle 2010), we also used Java Servlet to connect with the database and insert the texts annotated by a user into database (Tutorialspoint 2015). To annotate each document with the correct dialect, 100K documents were randomly selected from the corpus (tweets and comments), and an annotation tool was created and hosted a website.

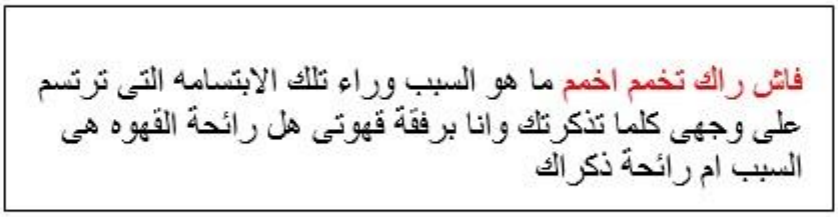
In the developed annotation tool, the player annotates 15 documents (tweets and comments) per screen. Each of these documents is labelled with four labels, so the player must read the document and make four judgments about this document. The first judgment is the level of dialectal content in the document. The second judgment is the type of dialect if the document is not MSA. The third judgment is the reason which makes the player select this dialect. Finally, if the reason selected in the third judgment is dialectal terms, then the fourth judgment requires the player to write the dialectal words found in the document.

The following list shows the options under each judgment to let the player choose one of them.

- The level of dialectal content
    - MSA (for document written in MSA)
    - Partial dialect (for document written in MSA where dialectical terms are less than 40% of the overall text, see Figure 6.3)
    - Mix of MSA and dialect (for document with approximately 50% MSA and 50% dialect code switching, see Figure 6.4)
    - Dialect (for a document written completely in dialect)
- The type of dialect if the document is not written in MSA
    - Egyptian
    - Gulf
    - Iraqi
    - Levantine
    - North Africa
    - Not sure
- The reason that makes this document dialectal.
    - Sentence structure

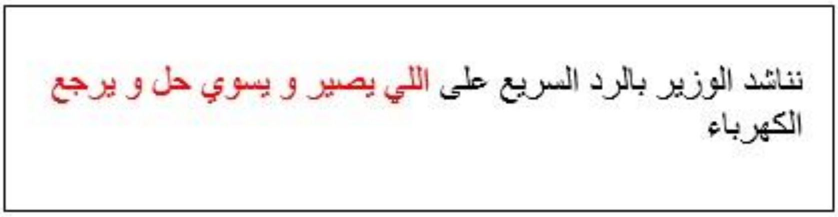
  - Dialectal terms
- The words which identify the dialect (we need to use these words as a dictionary for each dialect).

To annotate the collected data, we built an interface as a web page (http://www.alshutayri.com/index.jsp), to display a group of Arabic documents randomly selected from our collected dataset. Figure 6.5 shows the interface of the Annotation Tool.



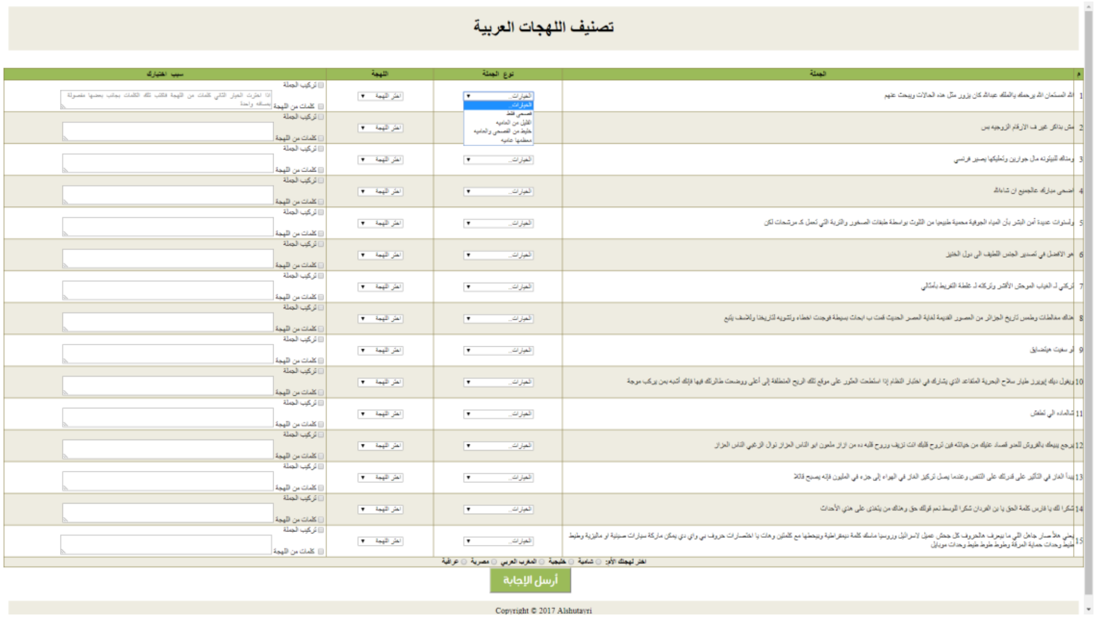
فاش راك تخمم اخمم ما هو السبب وراء تلك الابتسامه التى ترتسم
على وجهى كلما تذكرتك وانا برفقة قهوتى هل رائحة القهوه هى
السبب ام رائحة ذكراك

Figure 6.3 Example of document labelled as little bit of dialect.



نناشد الوزير بالرد السريع على اللي يصير و يسوي حل و يرجع
الكهرباء

Figure 6.4 Example of document labelled as mix of MSA and dialect.



تصنيف اللهجات العربية

سبب اختيارك	اللهجة	نوع الجملة	الجملة	م
☐ تركيب الجملة اذا اخترت الخيار الثاني كلمات من اللهجة فاكتب تلك الكلمات بجانب بعضها مفصولة بمسافة واحدة ☐ كلمات من اللهجة	اختر اللهجة ▼	الخيارات.. ▼	الله المستعان الله يرحمه والملك عبدالله كان يزور مثل هذه الحالات ويبحث عنهم	1
☐ تركيب الجملة ☐ كلمات من اللهجة	اختر اللهجة ▼		مش بذاكر غير ف الارقام الزوجيه بس	2
☐ تركيب الجملة ☐ كلمات من اللهجة	اختر اللهجة ▼	الخيارات.. ▼	ومناله للبيتونه مال جوارين وتحليكها يصير فرنسي	3
☐ تركيب الجملة ☐ كلمات من اللهجة	اختر اللهجة ▼	الخيارات.. ▼	اضحى مبارك عالجميع ان شاءالله	4
☐ تركيب الجملة ☐ كلمات من اللهجة	اختر اللهجة ▼	الخيارات.. ▼	ولسنوات عديدة آمن البشر بأن المياه الجوفية محمية طبيعيا من التلوث بواسطة طبقات الصخور والتربة التي تعمل كـ مرشحات لكن	5
☐ تركيب الجملة ☐ كلمات من اللهجة	اختر اللهجة ▼	الخيارات.. ▼	هو الافضل في تصدير الجنس اللطيف الى دول الخليج	6
☐ تركيب الجملة ☐ كلمات من اللهجة	اختر اللهجة ▼	الخيارات.. ▼	تركني لـ الغياب الموحش الأكثر وتركته لـ غلطة التفريط بأمثالي	7
☐ تركيب الجملة ☐ كلمات من اللهجة	اختر اللهجة ▼	الخيارات.. ▼	هناك مغالطات وطمس تاريخ الجزائر من العصور القديمة لغاية العصر الحديث قمت ب ابحاث بسيطة فوجدت اخطاء وتشويه لتاريخنا والاسف يتبع	8
☐ تركيب الجملة ☐ كلمات من اللهجة	اختر اللهجة ▼	الخيارات.. ▼	أو سفيت هيتضايق	9
☐ تركيب الجملة ☐ كلمات من اللهجة	اختر اللهجة ▼	الخيارات.. ▼	ويقول ديك إيريز طيار سلاح البحرية المتقاعد الذي يشارك في اختبار النظام إذا استطعت العثور على موقع تلك الريح المنطلقة إلى أعلى ووضعت طائرتك فيها فإنك أشبه بمن يركب موجة	10
☐ تركيب الجملة ☐ كلمات من اللهجة	اختر اللهجة ▼	الخيارات.. ▼	شالماده الي تطفش	11
☐ تركيب الجملة ☐ كلمات من اللهجة	اختر اللهجة ▼	الخيارات.. ▼	يرجع يبيعك بالقروش للعدو فساد عنيك من خيانته فين تروح قلبك انت تزيف وروح قلبه ده من ازاز ملعون ابو الناس الجزاز نول الزغبي الناس الجزاز	12
☐ تركيب الجملة ☐ كلمات من اللهجة	اختر اللهجة ▼	الخيارات.. ▼	يبدأ الغاز في التأثير على قدرتك على التنفس وعندما يصل تركيز الغاز في الهواء إلى جزء في المليون فإنه يصبح قاتلا	13
☐ تركيب الجملة ☐ كلمات من اللهجة	اختر اللهجة ▼	الخيارات.. ▼	شكرا لك يا فارس كلمة الحق يا بن الفردان شكرا للوسط نعم قولك حق وهناك من يتغذى على هذي الأحداث	14
☐ تركيب الجملة ☐ كلمات من اللهجة	اختر اللهجة ▼	الخيارات.. ▼	يطلع هلأ صار جاهل اللي ما بيعرف هالحروف كل جحش عميل لاسرائيل وروسيا ماسك كلمة ديمقراطية ويحطها مع كلمتين وهات يا اختصارات حروف بي واي دي يمكن ماركة سيارات صينية او ماليزية وطيط طيط وحدات حماية المرأة وطوط طوط طيط وحدات موبايل	15

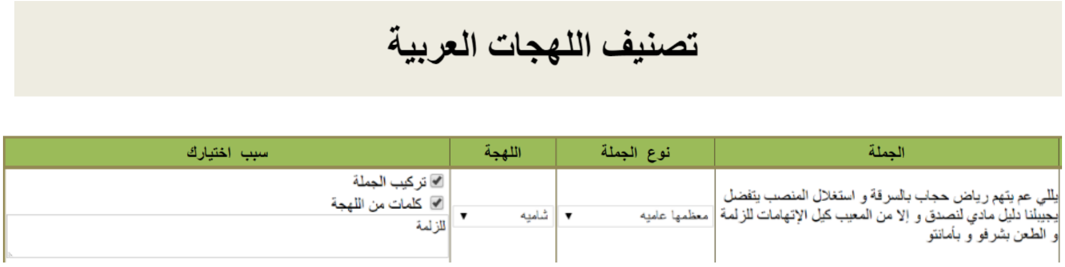
اختر لهجتك الأم: ○ شامية ○ خليجية ○ المغرب العربي ○ مصرية ○ عراقية

Copyright © 2017 Alshutayri

Figure 6.5 The annotation interface.

Each page displays 15 documents randomly selected from the dataset. As shown in Figure 6.6, the first label indicates the amount of dialectal content in the document to decide whether the document is MSA or contains dialectal content. If the document is MSA the other labels will be inactive, and the player needs to move to the next document. But, if the document is not MSA, then all labels are required and the player needs to move to the second label to specify the document dialect if it is one of the five dialects (EGY, GLF, LEV, IRQ, and NOR), or enter 'Not Sure' if the document is written using one dialect or a mix of dialects and is difficult to categorise exactly which dialect. The third and fourth labels are to explain the causes which led the player to choose the selected dialect. For example, the sentence structure if the words in the document are all MSA, but the structure of the sentence is not based on the MSA grammar rules, and/or the dialectal terms which help to identify the dialect.

In fact, there is no agreed standard for writing Arabic dialects because MSA is the formal standard form of written Arabic (Elfardy and Diab 2012); therefore, some documents apparently contain only MSA vocabulary but are annotated as dialect based on non-standard sentence structure.



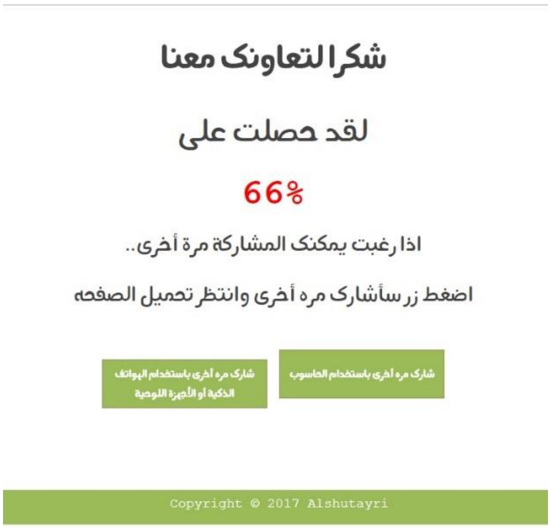
تصنيف اللهجات العربية

الجملة	نوع الجملة	اللهجة	سبب اختيارك
يللي عم يتهم رياض حجاب بالسرقة و استغلال المنصب يتفضل يجيبلنا دليل مادي لنصدق و إلا من المعيب كيل الإتهامات للزلمة و الطعن بشرفو و بأمانتو	معظمها عاميه ▾	شاميه ▾	☑ تركيب الجملة ☑ كلمات من اللهجة للزلمة

Figure 6.6 Example of the annotated document

At the end of the page, before submitting the annotated documents, the mother dialect must be chosen. This may help to decide which annotated document must be accepted if one document has different annotations. So, if in our dataset a document was selected from Gulf newspaper and the mother dialect for the player is Gulf that would give us a good sign to accept his/her annotation even if another player with a different mother dialect annotated the same document with a different dialect. Finally, the player needs to press the submit button to send his/her answers and get the score by comparing his/her labelling documents with our pre-labelled sample as shown in Figure 6.7.

As a control, to be sure that the player reads the document before selecting the options, three MSA documents collected from newspaper articles (Al-Sulaiti and Atwell 2004), were mixed with 12 documents selected from the dataset. These three MSA documents are used as a control because they must be labelled as MSA; if the player labels all the three MSA documents as dialect then the player's submitted documents are not counted in the annotated corpus. Furthermore, to verify the annotation process, each document is redundantly annotated three times by three players, by using a count starting from zero which and increases every time the document is annotated by a player and inserted into the corpus. Therefore, each document is selected randomly from the dataset no more than three times.



شكرا لتعاونك معنا

لقد حصلت على

66%

اذا رغبت يمكنك المشاركة مرة أخرى..

اضغط زر سأشارك مره أخرى وانتظر تحميل الصفحه

شارك مره أخرى باستخدام الهواتف الذكية أو الأجهزة اللوحية
شارك مره أخرى باستخدام الحاسوب

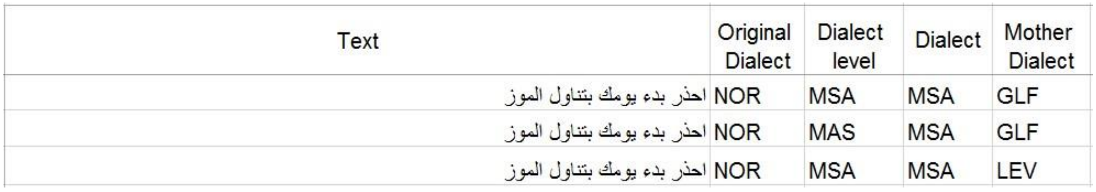
Copyright © 2017 Alshutayri

Figure 6.7 Example of the annotated document.

6.1.3 The Evaluation of the Annotation Tool

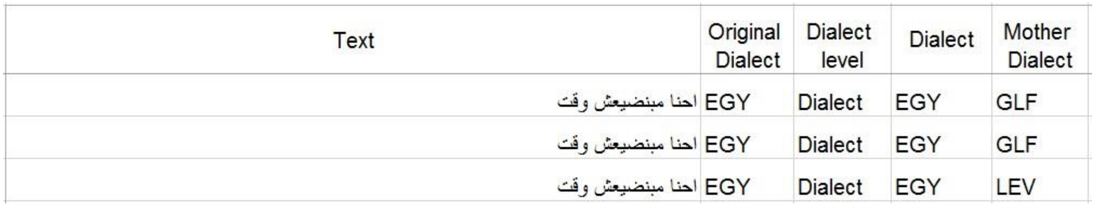
To ensure that each document received the correct label, each document was annotated by three players besides the gold standard, which is an initial label used to label each document based on the source of comments and tweets, as mentioned in Section 6.1.1. In addition, the mother dialect for each player helps to decide which label must be counted as being correct if players gave different labels for one document. The results of annotated documents was evaluated in two cases:

- Agreement between annotators: All the players label one document with same label as in Figure 6.8 and 6.9. The agreed label is considered to be correct, even if the agreed label is different from the original label because, as mentioned in Section 6.1.1, the initial label may not be correct.
- Disagreement between annotators: When some of the players label the document with different labels, as in Figure 6.10, the mother dialect could help to decide which label must be accepted as being correct for this document.



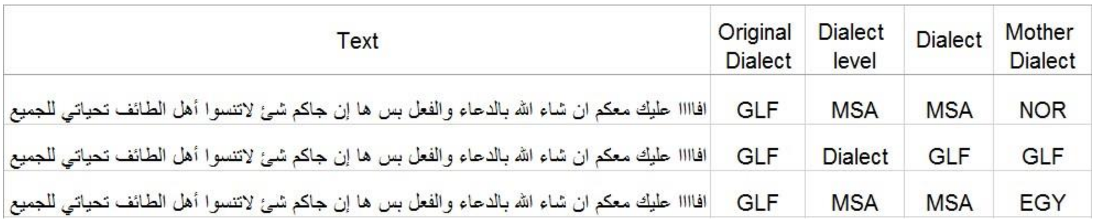
Text	Original Dialect	Dialect level	Dialect	Mother Dialect
احذر بدء يومك بتناول الموز	NOR	MSA	MSA	GLF
احذر بدء يومك بتناول الموز	NOR	MAS	MSA	GLF
احذر بدء يومك بتناول الموز	NOR	MSA	MSA	LEV

Figure 6.8 Example 1 of the agreement between annotators.



Text	Original Dialect	Dialect level	Dialect	Mother Dialect
احنا مبنضيعش وقت	EGY	Dialect	EGY	GLF
احنا مبنضيعش وقت	EGY	Dialect	EGY	GLF
احنا مبنضيعش وقت	EGY	Dialect	EGY	LEV

Figure 6.9 Example 2 of the agreement between annotators.



Text	Original Dialect	Dialect level	Dialect	Mother Dialect
افاااا عليك معكم ان شاء الله بالدعاء والفعل بس ها إن جاكم شئ لاتنسوا أهل الطائف تحياتي للجميع	GLF	MSA	MSA	NOR
افاااا عليك معكم ان شاء الله بالدعاء والفعل بس ها إن جاكم شئ لاتنسوا أهل الطائف تحياتي للجميع	GLF	Dialect	GLF	GLF
افاااا عليك معكم ان شاء الله بالدعاء والفعل بس ها إن جاكم شئ لاتنسوا أهل الطائف تحياتي للجميع	GLF	MSA	MSA	EGY

Figure 6.10 Example of the disagreement between annotators.

To evaluate the quality of the annotation, the inter-annotator agreement was calculated using Fleiss Kappa (Fleiss 1971) to calculate the annotator agreement for more than two annotators. The kappa κ can be defined as:

$$\kappa = \frac{\bar{P} - \bar{P}_e}{1 - \bar{P}_e} \qquad (6.1)$$

Where $\bar{P} - \bar{P}_e$ gives the max level of agreement, and $1 - \bar{P}_e$ gives the achieved level of agreement between annotators. κ varies between 1 and 0, κ=1 means a complete agreement between annotators, and $\kappa \leq 0$ means no agreement between annotators.

To calculate κ, we first need to calculate p_j for each category by taking the sum of all assignment for j and divided by sum of cells.

$$p_j = \frac{1}{Nn} \sum_{i=1}^{N} nij \qquad (6.2)$$

Then, calculate p_i which compute the agreement between annotators for each document.

$$p_i = \frac{1}{n(n-1)} [(\sum_{j=1}^{k} n_{ij}^2) - (n)] \qquad (6.3)$$

After that, $\bar{P}$ was calculated by dividing the summation of p_i for each document by the number of annotated documents.

$$\bar{P} = \frac{1}{N}\sum_{i=1}^{N} P_i \qquad (6.4)$$

Finally, $\bar{P_e}$ was calculated to go into κ formula.

$$\bar{P_e} = \sum_{j=1}^{k} p_j^2 \qquad (6.5)$$

The above equations were applied on the dataset to calculate the agreement between annotators.

N = 3966, N is the total number of documents, three annotators n=3, and seven categories k=7,

Sum of all cells = N * n = 3966 * 3 = 11898

By applying Equation (6.2) on each category to calculate p_j:

$$p_{MSA} = \frac{6329}{11898} = 0.531938$$

$$p_{GLF} = \frac{1628}{11898} = 0.13683$$

$$p_{IRQ} = \frac{406}{11898} = 0.034123$$

$$p_{LEV} = \frac{675}{11898} = 0.056732$$

$$p_{NOR} = \frac{534}{11898} = 0.044881$$

$$p_{EGY} = \frac{887}{11898} = 0.07455$$

$$p_{Not\_Sure} = \frac{1439}{11898} = 0.120945$$

Then apply Equation (6.3) to calculate p_i for each document:

$$p_1 = \frac{1}{3(3-1)}(1^2 + 2^2 + 0^2 + \ 0^2 + 0^2 + \ 0^2 + \ 0^2 - 3\,) = 0.3333$$

.

.

.

.

$$p_{3966} = \frac{1}{3(3-1)}(3^2 + 0^2 \ + 0^2 \ + 0^2 \ + 0^2 \ + 0^2 \ + 0^2 - 3) = 1$$

Then apply Equation (6.4) to calculate $\bar{P}$, by calculate the sum of P_i:

$$\sum_{i=1}^{N} P_i = 0.3333 + \ldots \ldots \ldots \ldots \ldots . + 1 = 3400$$

Sum of p_i= 3400

$$\bar{P} = \frac{1}{3966}(3400) = 0.857287$$

Applying Equation (6.5) to calculate $\bar{P_e}$:

$$\bar{P_e} = (0.531938)^2 + (0.13683)^2 + (0.034123)^2 + (0.056732)^2 + (0.044881)^2 + (0.07455)^2 + (0.120945)^2 = 0.328263$$

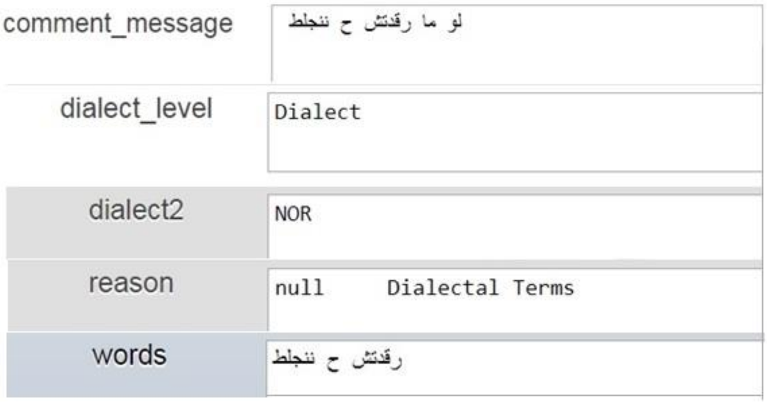
Finally, use Equation (6.1) to calculate κ.

$$\kappa = \frac{0.857287 - 0.328263}{1 - 0.328263} = 0.787$$

The result equals 0.787 around 79% which is substantial agreement according to (Landis and Koch 1977).

6.1.4 The Result from the Annotation Tool

The result of the annotation tool is a set of documents which are labelled with four labels: the first label is the dialect level, which is an option from three choices: Partial dialect, Mix of MSA and dialect, or Dialect. The second label is the specific dialect which is one of the five dialects: GLF, EGY, LEV, IRQ, or NOR. The third label shows the reasons that help to identify the document’s dialect. The last label shows the dialectal words which help to identify the document’s dialect. Figure 6.11 shows the result of one annotated document in the corpus.



comment_message	لو ما رقدتش ح نتجلط
dialect_level	Dialect
dialect2	NOR
reason	null Dialectal Terms
words	رقدتش ح نتجلط

Figure 6.11 Result of the annotated document.

We launched the website via Twitter and WhatsApp at the beginning of August 2017. At the time that this chapter was written, the annotation website has been running for around four months, and we have accumulated 24,060 annotated documents with a total numbers of words equal to 586,952. The distribution of dialectal content in the annotated documents is shown in Figure 6.12, where the documents with dialect content number 16239 which is divided between 10250 documents which had dialect content, 2447 documents which had partial dialect, 3542 documents had a mix of MSA and dialect, and 7821 documents had MSA content.

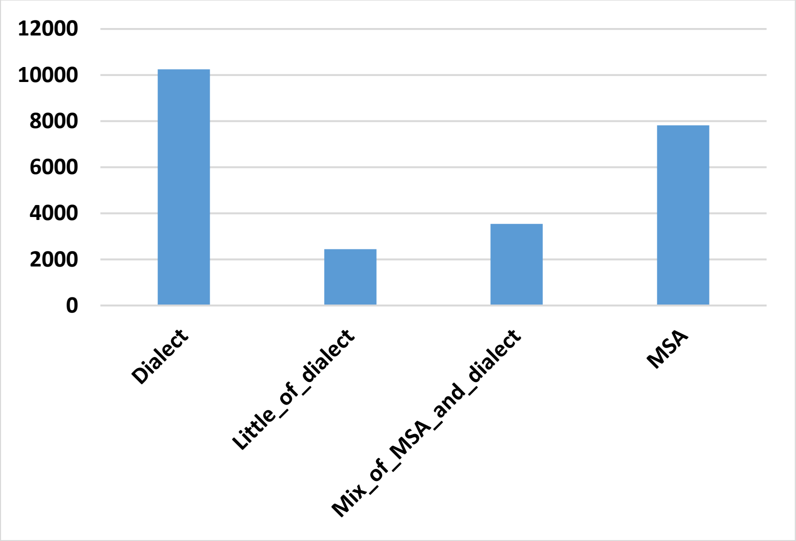


Figure 6.12 The result of the level of dialectal content in the annotated documents.

The distribution of dialects of the annotated corpus shown in Figure 6.13, where GLF dialect consists of 5K documents, EGY dialect 4K documents, NOR dialect 2K documents, LEV dialect 3K, and IRQ dialect 2K documents. The number of users (players) is 1,840 from different countries around the world. Figure 6.14 shows the distributions of users on the days and Figure 6.15 shows the percentage of the players from the top ten countries. For our immediate research on Arabic dialects classification, the annotated documents which we have already collected could be sufficient, but we decided to continue with this experiment to collect a larger annotated Arabic dialect text corpus and let the corpus be available for other research.

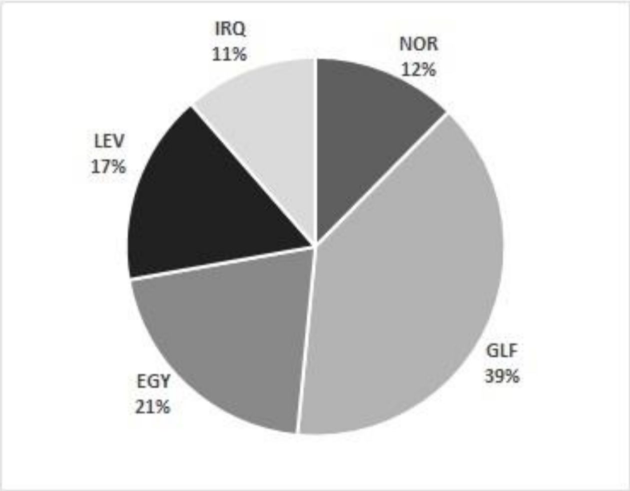


Figure 6.13 The distribution of labels (dialects) of the annotated corpus.

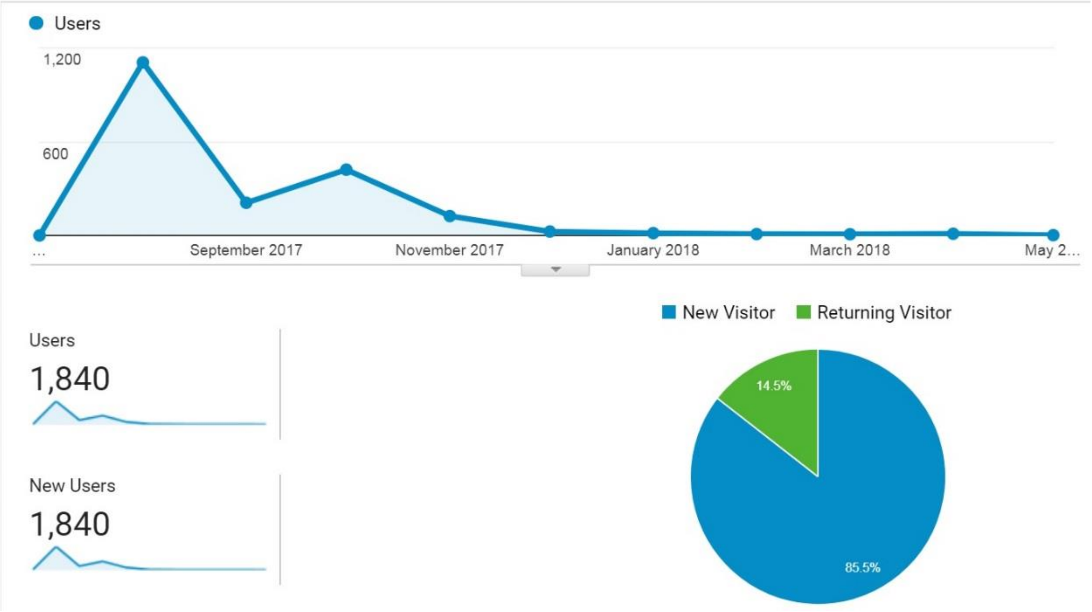
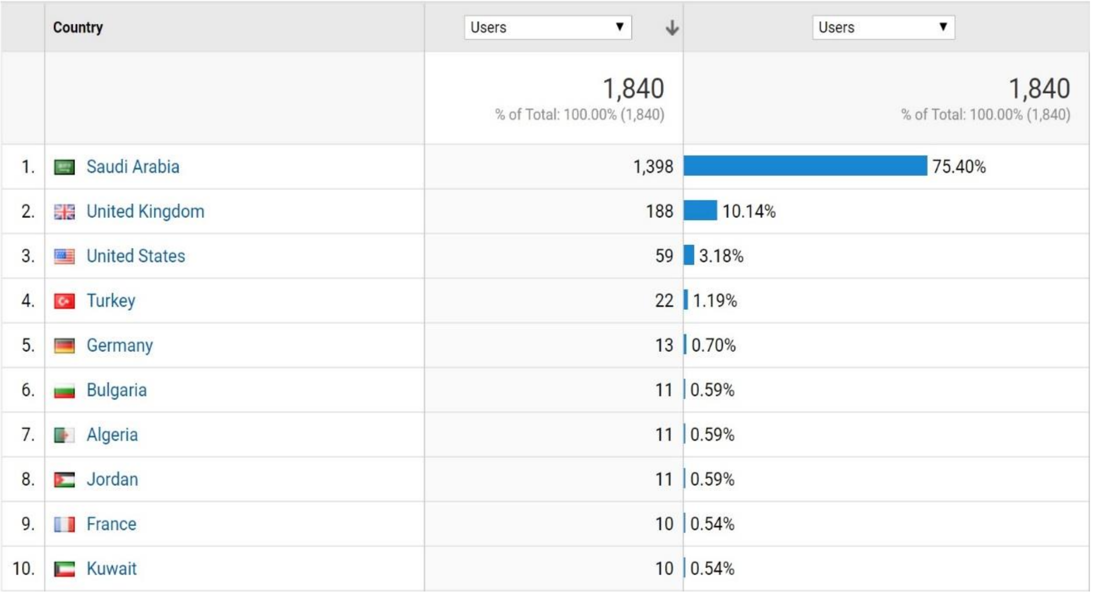


Figure 6.14 Distribution of the number of users during months.



	Country	Users ▼	Users ▼
		1,840 % of Total: 100.00% (1,840)	1,840 % of Total: 100.00% (1,840)
1.	Saudi Arabia	1,398	75.40%
2.	United Kingdom	188	10.14%
3.	United States	59	3.18%
4.	Turkey	22	1.19%
5.	Germany	13	0.70%
6.	Bulgaria	11	0.59%
7.	Algeria	11	0.59%
8.	Jordan	11	0.59%
9.	France	10	0.54%
10.	Kuwait	10	0.54%

Figure 6.15 Percentage of the players from the top 10 countries.

6.2 Conclusion

In this chapter, we presented a new approach to annotate the dataset collected from Twitter, Facebook, and Online Newspapers for the five main Arabic dialects: Gulf, Iraqi, Egyptian, Levantine and North African.

The annotation website was created as an online game to attract more users who talk different Arabic dialects as unpaid volunteers with no need to register in comparing with other crowdsourcing websites. This experiment is a new approach and helps to annotate the sufficient dataset for text research in Arabic dialect classification. The number of users has decreased now in comparison with the beginning because we need to distribute the website widely not just between our friends.

Chapter 7
Final Version of Corpus

In this chapter we present a description of the final version of the corpus that we collected. In the first section, we present the difficulties in using Social media as a source of corpus. We describe the process of collecting the corpus and the content and the size of the corpus. This chapter is based on A Social Media Corpus of Arabic Dialect Text (Alshutayri and Atwell 2018c).

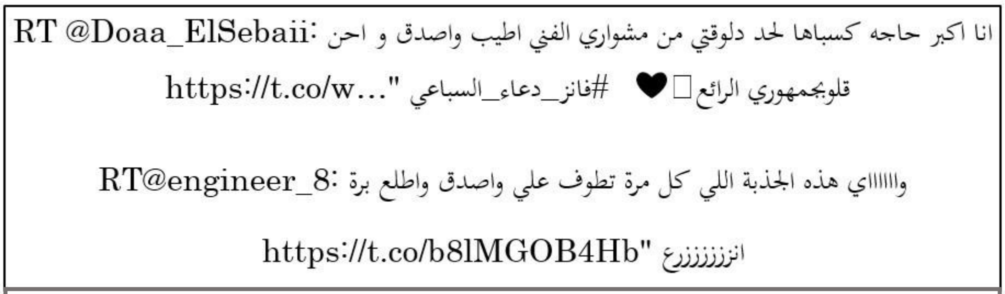
7.1 Social Media Arabic Dialect Corpus (SMADC)

7.1.1 The Difficulties in using Social Media as a Source of Corpora

Social media applications such as Twitter and Facebook are considered to be popular applications that Arabs use to have discussions or exchange views, writing in their dialects. However, the data extracted from Twitter and Facebook usually contains features which can be unhelpful "noise" for Machine Learning classifiers:

1) Words from one dialect found in tweets or comments from another dialect because of the TV industry, which has made some dialectal words popular in all Arab countries. That means one or more features can overlap between dialects (Lu and Mohamed, 2011).
2) Repeated characters and non-Arabic alphabetical characters such as #, @, URL if the text contained a web link or picture, emojis, or the user name in retweet or comment.
3) There are many copied and hence redundant texts in one dialect due to retweeting or copying.
4) Texts that have spelling mistakes, connect words without space or are incomplete texts.
5) Code-switching between MSA and dialect or sometimes between two dialects.

Figure 7.1 shows examples of noise.



RT @Doaa_ElSebaii: انا اكبر حاجه كسباها لحد دلوقتي من مشواري الفني اطيب واصدق و احن

https://t.co/w..." #فانز_دعاء_السباعي ♥□ قلوبجمهوري الرائع

RT@engineer_8: واااااي هذه الجذبة اللي كل مرة تطوف علي واصدق واطلع برة

https://t.co/b8lMGOB4Hb" انززززززع

RT @Doaa_ElSebaii: The biggest thing that I earned until now from my art career is the best and truest and most compassionate hearts of my wonderful audience #Fanz_Duaa_Sibai

RT@engineer_8: Oh this lie that every time I believe and go to wait outside.

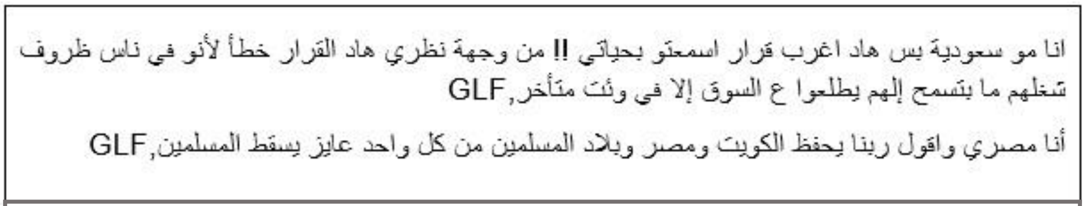
Figure 7.1 Dialectal Examples of noise in Twitter data.

Arabic Dialect classification based on Arabic text for that any other non-Arabic characters were considered as a noise. Therefore, data pre-processing is needed to remove noise and improve the accuracy of the classification. In this research the pre-processing step works to delete non-Arabic alphabetical characters such as #, @, URL if the text contained a web link or picture, emojis, or the user name in retweet or comment. In addition to noise in tweets, extracting tweets in a short time period produces many tweets that focus on recent topics. Hence, the number of words that are used in tweets is limited, and the classifier might train topic classification instead of dialect classification (Lu and Mohamed 2011). Furthermore, the difference between the amount of extracted tweets from one dialect and from another may produce an unbalanced dataset for the training process. To solve these problems, we ran our Twitter's extractor program, using different periods and different times for each dialect, to create a balanced tweet corpus with various topics.

7.1.2 Process of Collection

The corpus covers five Arabic dialects: GLF, EGY, NOR, LEV, and IRQ. It consists of tweets from Twitter, Comments from online Newspaper, and comments from Facebook. The tweets were collected using two methods: one based on seeds terms as presented in Chapter 3, and one based on spatial coordinate points, see Chapter 5. The comments from Facebook were collected based on the country of the Facebook page, see Chapter 5. The comments from Newspapers were based on the country that issued the newspaper, see Chapter 4. After the collection step, the texts from the three different sources were reviewed and processed based on the following criteria:

- Exclude any documents if the writer of the tweet or comment write a nationality that is in conflict with the label of the document based on the method which was used to collect this document, see Figure 7.2.
- Exclude any duplicated documents which appear frequently, especially in tweets due to retweeting or copying.
- Record the length for each document as written.



انا مو سعودية بس هاد اغرب قرار اسمعتو بحياتي !! من وجهة نظري هاد القرار خطأ لأنو في ناس ظروف شغلهم ما بتسمح إلهم يطلعوا ع السوق إلا في وئت متأخر,GLF

أنا مصري واقول ربنا يحفظ الكويت ومصر وبلاد المسلمين من كل واحد عايز يسقط المسلمين,GLF

I am not Saudi but this the most strange decision that I heard ever!! From my opinion this decision is wrong because for some people, their working conditions do not allow them to go to the market until late time, GLF

I am Egyptian and I say our lord saves Kuwait, Egypt, and the Muslim countries from everyone who wants Muslims to fall, GLF

Figure 7.2 Example of the excluding documents from the corpus.

7.1.3 Contents and Size of the Corpus

The final version of the corpus after applying the previous criteria in Section 7.1.2, contains 1,088,578 documents; they include 812,849 Facebook comments, 9,440 online newspaper comments, and 266,289 Twitter tweets; 180,282 based on seed terms, and 86,007 based on spatial coordinate points. According to these numbers, we found that Facebook provided more comments in comparison to Twitter and online newspaper, because using Facebook to scrape all posts for a specific Facebook page got all posts from the beginning of the page creation, so for each post lots of comments are collected from different users with a good amount of different words. In contrast, on Twitter it is difficult to recognize a specific account to collect all that account’s tweets, and furthermore we want to cover a large number of users with different tweets topics and dialect. So, the program worked every day for a specific period ranging from 4-6 hours to collect all matching tweets written at this time.

Table 7.1 shows the number of documents for each dialect from different sources and Figure 7.3 presents the distribution of the documents per dialect.

Table 7.1 The number of documents in each dialect.

Dialect	**Tweets Based on**		**Comments from**		**Total**
	Seed Terms	**Spatial Coordinate Points**	**Online Newspaper**	**Facebook**	
GLF	33,024	34,188	3,208	106,599	177,019
EGY	27,049	19,297	716	263,636	310,698
NOR	29,843	9,251	2,411	212,777	254,282
LEV	46,518	12,712	2,192	132,103	193,525
IRQ	43,848	10,559	913	97,734	153,054
Total	180,282	86,007	9,440	812,849	1,088,578

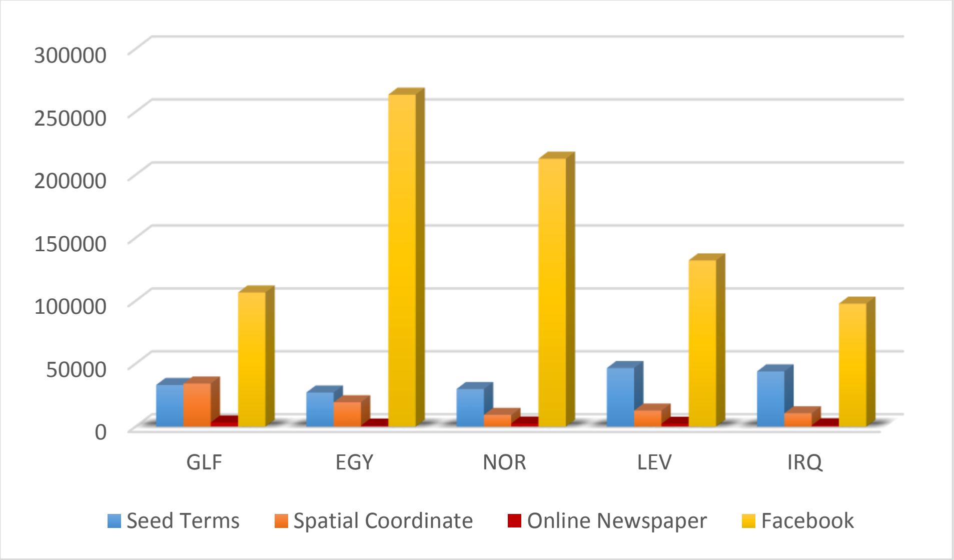


Figure 7.3 Distribution of the documents from different sources for each dialect.

The total number of word types was 1,675,026 word types, and the total number of word tokens was 13,876,504 word tokens, as shown in Table 7.2 and 7.3. Figure 7.4 and 7.5 show the distribution of the word tokens and types per dialect.

Table 7.2 The number of word types in each dialect in different sources.

	Tweets Based on Seed Terms	Tweets Based on Coordinate Points	Comments from Newspaper	Facebook Comments	TOTAL
GLF	51,527	77,302	28,949	153,146	310,924
EGY	40,956	48,230	12,654	211,891	313,731
NOR	43,555	96,901	27,585	346,298	514,339
LEV	62,463	38,705	20,869	175,216	297,253
IRQ	56,429	35,901	14,907	131,542	238,779
Total	254,930	297,039	104,964	1,018,093	1,675,026

Table 7.3 The number of word tokens in each dialect in different sources.

Dialect	Tweets Based on Seed Terms	Tweets Based on Coordinate Points	Comments from Newspaper	Facebook Comments	TOTAL
GLF	411,836	365,319	90,366	2,352,838	3,220,359
EGY	367,247	194,656	31,374	2,250,456	2,843,733
NOR	414,368	30,844	86,270	3,390,410	3,921,892
LEV	594,063	137,181	58,516	1,398,857	2,188,617
IRQ	644,902	118,314	43,468	895,219	1,701,903
Total	2,432,416	846,314	309,994	10,287,780	13,876,504

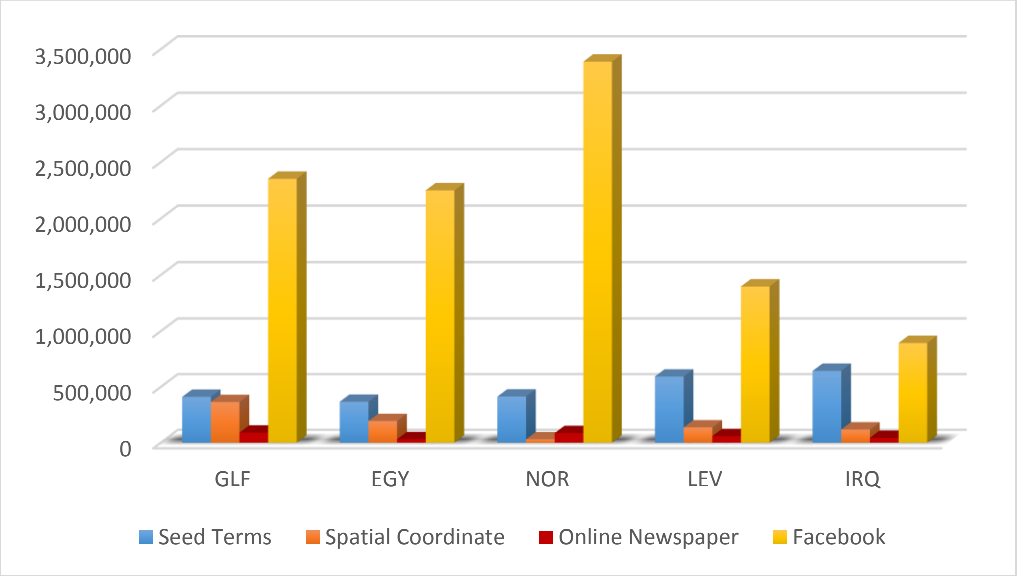


Figure 7.4 Distribution of word tokens in each dialect in different sources.

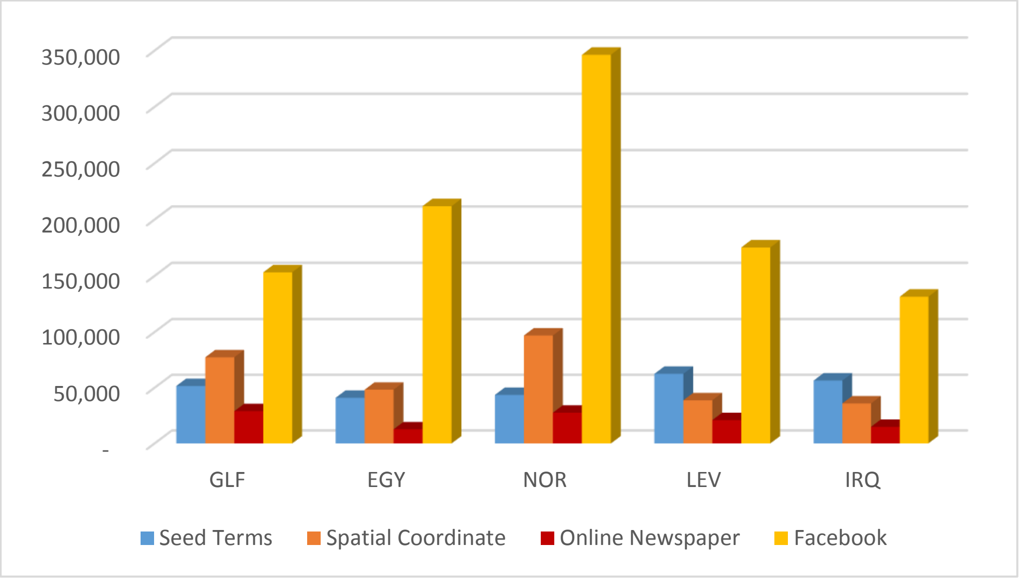


Figure 7.5 Distribution of word types in each dialect in different sources.

The Social Media Dialect Corpus (SMADC) was explored to produce the most frequent words in each dialect from the different source of Arabic dialect text. Tables 7.4, 7.5, 7.6, 7.7, and 7.8 present the twenty frequent words in each dialect and figure 7.6 shows the distribution of 100 words for each dialect.

Table 7.4 The most frequent words for EGY dialect found in SMADC.

Word	IPA	Frequency	Translation
مش	mʃ	29944	Not
ده	dh	21430	This/That
ايه	ajh	18510	What
اللي	alli:	16512	Which/Who
بس	bs	15945	But
دي	di:	11330	This/That
مفيش	mfi:ʃ	10049	There is nothing
كده	kdh	10330	Like this
مصر	msˤr	10049	Egypt
عايز	ʕa:jz	9092	I want
عشان	ʕʃa:n	8307	Because
دا	da:	8278	This/That
حد	ħd	7962	someone
احنا	aħna:	7586	We
دلوقتي	dlwʔti:	7263	Now
ليه	ljh	7132	Why
دى	dj	5687	This/That
علشان	ʕlʃa:n	5272	In order to
فين	fjn	4380	Where
عاوز	ʕa:wz	4209	I want

Table 7.5 The most frequent words for GLF dialect found in SMADC.

Word	IPA	Frequency	Translation
الحين	alħi:n	18057	Now
اللي	allj	15420	Which/Who
بس	bs	13914	Enough/But
شوف	ʃu:f	10250	Look
عشان	ʕʃa:n	10020	Because
وش	wʃ	9954	What
وشلون	wʃlu:n	9850	How
اخوي	axu:j	9627	My Brother
هذي	hði:	9097	This
كذا	kða:	9069	So
ايش	ajʃ	8825	What
ليش	ljʃ	8756	Why
زي	zj	8532	Like
وين	wjn	8493	Where
ابي	abj	8004	I want
خلاص	xla:sˤ	7900	Enough
أخوي	axu:j	7778	My Brother
لبيه	lbjh	7768	Yes
عندنا	ʕndna:	7631	We have
كمان	kma:n	7561	Also

Table 7.6 The most frequent words for LEV dialect found in SMADC.

Word	IPA	Frequency	Translation
بس	bs	11914	But
الحكي	alħki:	8814	The Story
مش	mʃ	8687	Not
شو	ʃu:	7748	What
هيك	hjk	7181	Like this
بدي	bdi:	5390	I want
منيح	mni:ħ	5255	Good
مشان	mʃa:n	4807	In order to
انو	anu:	3038	It is a
عم	ʕm	2813	Express of present continuance
مو	mu:	2555	Is not it
هاد	ha:d	2512	That
هاي	ha:j	2465	This
كرمال	krma:l	2319	Because of
هسا	hsa:	2309	Now
هلق	hlq	2287	Now
كتير	kti:r	2034	much
حدا	ħda:	1914	Someone
بدك	bdk	1810	You want
ياك	ja:k	1668	You

Table 7.7 The most frequent words for IRQ dialect found in SMADC.

Word	IPA	Frequency	Translation
بس	bs	9777	But
مو	mu:	6110	Not
جم	dʒm	6090	How many
هاي	ha:j	5024	This
اني	ani:	3938	I/me
شي	ʃi:	3534	Something
العراق	alʕra:q	3290	Iraq
شنو	ʃnu:	2892	What
ليش	ljʃ	2890	Why
هيج	hjdʒ	2256	Like
شكد	ʃkd	1993	How many
يابا	ja:ba:	1706	To call someone
هسه	hsh	1425	Now
اكو	aku:	1374	Exist
ماكو	ma:ku:	1326	Nothing
شلون	ʃlu:n	1218	How
جان	dʒa:n	1145	It was
منو	mnu:	1126	Who
النا	alna:	1004	Ours
لعد	lʕd	890	So

Table 7.8 The most frequent words for NOR dialect found in SMADC.

Word	IPA	Frequency	Translation
لي	li:	11766	The
بزاف	bza:f	8088	many
واش	wa:ʃ	7595	Do you
هاد	ha:d	7077	This
علاش	ʕla:ʃ	5960	Why
ديالي	dja:li:	5572	That’s mine
راه	ra:h	5485	To notice
غادي	ɣa:di:	5120	Going
ديال	dja:l	5029	Related
باش	ba:ʃ	4704	Because
الجزائر	aldʒza:ʔr	4499	Algeria
المغاربة	almɣa:rbh	4031	Moroccans
اش	aʃ	1984	What
بس	bs	1947	But
برشا	brʃa:	1850	much
شكون	ʃku:n	1836	Who
كيفاش	kjfa:ʃ	1782	How
كاين	ka:jn	1749	exist
مزيان	mzja:n	1650	Beautiful
ديما	djma:	1468	Always

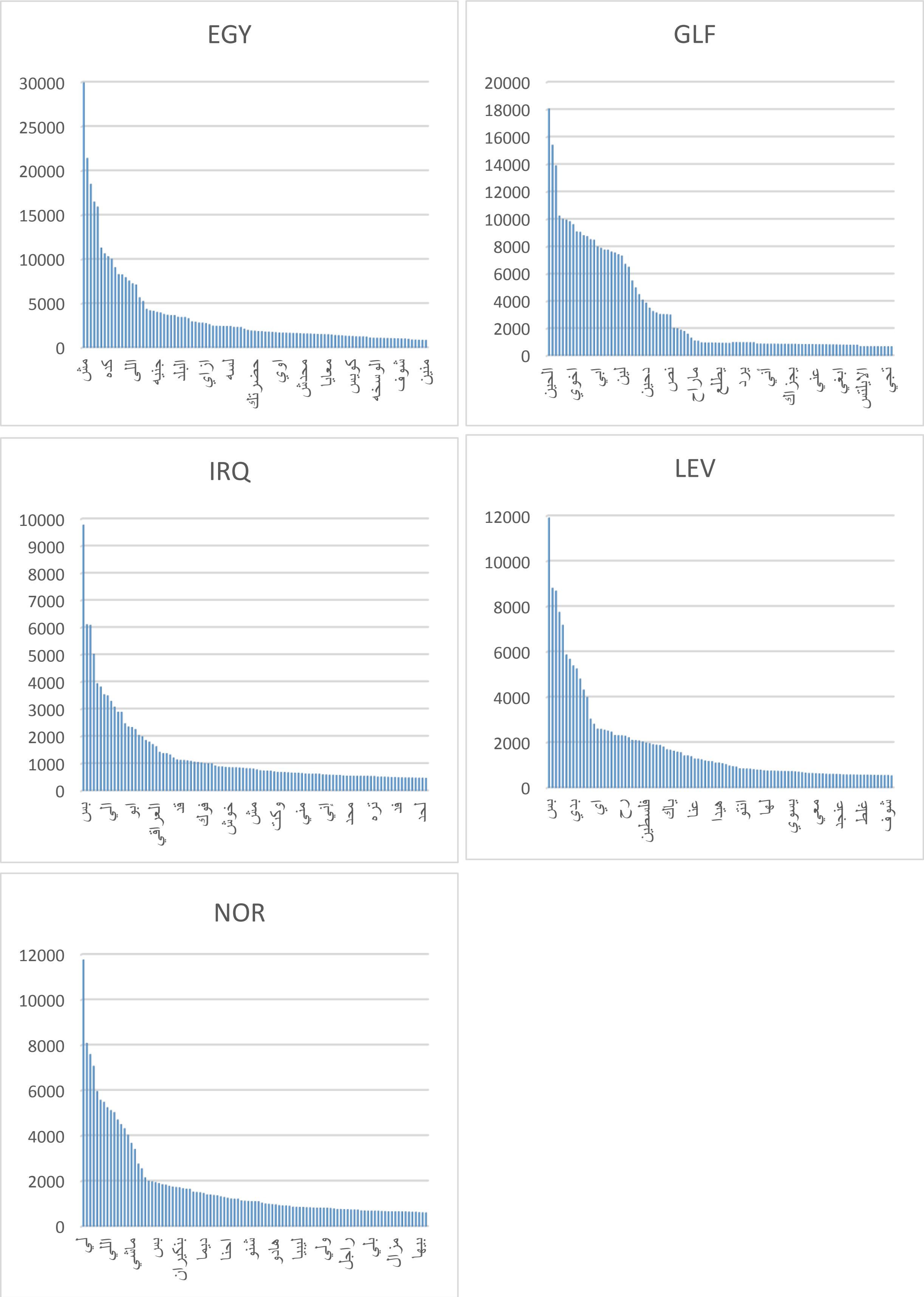


Figure 7.6 The distribution of the most frequent words in each dialect.

To show the significant differences in word frequencies between the dialects, we used two methods.

The first method based on using a statistical measure called chi-squared test, also written as χ^2, Equation (7.1) show how to calculate χ^2. First, the top frequent word for each dialect is chosen with its frequency to apply χ^2 test. Table 7.9 shows the words used in this test and their frequencies. Second, the expected frequency was calculated using Equation (7.2). Where $\sum O_i$ is the total of the observed frequency times the total of a row frequencies $\sum O_j$ divided by the summation of the total rows frequencies $\sum\sum O_j$, one example applied to show how to calculate the expected frequency and Table 7.10 shows the result of applying Equation (7.2) on the frequencies shown in Table 7.9.

$$\chi^2 = \sum \frac{(O_{ij} - E_{ij})^2}{E_{ij}} \qquad (7.1)$$

$$E_{ij} = \frac{\sum O_i \times \sum O_j}{\sum\sum O_j} \qquad (7.2)$$

$$E_{11} = \frac{18579 \times 25583}{94380} = 5036.09$$

Table 7.9 The frequency of the top frequent word from each dialect in SMADC.

	الحين	مش	الحكي	مو	بزاف	Total
GLF	18057	7442	71	6	7	**25607**
EGY	32	29944	10	211	44	**30241**
LEV	224	8687	8814	2555	6	**20290**
IRQ	200	809	18	6110	99	**7236**
NOR	66	2154	53	673	8088	**11034**
	18579	**49036**	**8966**	**9555**	**8272**	**94408**

Table 7.10 The expected frequency for the top frequent word from each dialect in SMADC.

	الحين	مش	الحكي	مو	بزاف
GLF	5036.0940	13291.883	2430.3578	2590.0114	2234.6498
EGY	5953.0360	15711.990	2872.8629	3061.5888	2641.5215
LEV	3993.3629	10539.778	1927.1485	2053.7479	1771.9621
IRQ	1424.4293	3759.5305	687.41233	732.57024	632.05747
NOR	2172.0776	5732.8165	1048.2183	1117.0785	963.80902

After that, the chi-squared test is calculated using Equation (7.1). Table 7.11 shows the result of applying the first part of Equation (7.1) by dividing the power of the subtraction of the expected frequency E_{ij} from the observed frequency O_{ij} by the expected frequency E_{ij}.

$$E_{11} = \frac{(18057 - 5036.09)^2}{5036.09} = 33665.77$$

Table 7.11 The individual χ^2 values for the top frequent word from each dialect in SMADC.

	الحين	مش	الحكي	مو	بزاف
GLF	33665.7715	2574.58916	2290.43207	2578.02836	2220.67177
EGY	5889.20803	12891.434	2852.89777	2654.13064	2554.25446
LEV	3557.92779	325.698326	24610.8296	122.33908	1759.98243
IRQ	1052.51079	2315.61641	651.883666	39473.0073	449.563971
NOR	2042.08308	2234.14227	944.898094	176.537032	52659.9104

Then, the summation was applied on the results on Table 7.11 to calculate the value of chi-squared test which equal to 202548.34. The chi-squared value is positive which show that there is a significant differences in word frequencies between dialects.

The second method based on creating a table of words by extracting the top five frequent words from each dialect then the frequency of each word in each dialect is written. As shown in Table 7.12 some words are use in all dialect but they are more frequent in specific dialect than other dialects. In terms of EGY the top five words are frequent in EGY dialect by 100% as well as NOR. However, in GLF and LEV the top five words are frequent by 60% because two words are also frequently use in EGY dialect, while in IRQ dialect the top five words are frequent use by 80%.

The first column in Table 7.12 shows the dialects covered in SMADC while the second column shows the top five words extracted from the dictionary of each dialect. The frequency of each word written in the remaining columns.

Table 7.12 The most significant differences in word frequencies in SMADC.

Dialect	Word	GLF	EGY	LEV	IRQ	NOR
GLF	الحين	**18057**	32	224	200	66
	اللي	15420	**16512**	5872	2350	5244
	بس	13914	**15945**	11914	9777	1947
	شوف	**10250**	816	1005	557	1003
	عشان	**10020**	8307	1960	658	331
EGY	مش	7442	**29944**	8687	809	2154
	ده	264	**21430**	340	308	471
	ايه	6736	**18510**	609	261	642
	اللي	15420	**16512**	5872	2350	5244
	بس	13914	**15945**	11914	9777	1947
LEV	بس	13914	**15945**	11914	9777	1947
	الحكي	71	10	**8814**	18	53
	مش	7442	**29944**	8687	809	2154
	شو	320	420	**7748**	516	687
	هيك	62	146	**7181**	90	741

IRQ	بس	13914	**15945**	11914	9777	1947
	مو	6	211	2555	**6110**	673
	جم	58	27	1	**6090**	10
	هاي	159	254	2465	**5024**	642
	اني	0	0	1248	**3938**	786
NOR	لي	0	0	0	0	**11766**
	بزاف	7	44	6	99	**8088**
	واش	31	1774	241	19	**7595**
	هاد	53	148	2512	37	**7077**
	علاش	0	53	94	4	**5960**

7.2 Conclusion

This chapter has explored social media as a resource for Arabic dialects text, for use in research in Arabic text analytics and Arabic corpus linguistics (Atwell 2018a; Atwell 2018b). We divided the Arab countries into five groups, one for each of the five main dialects: Gulf, Iraqi, Egyptian, Levantine, and North African.

The texts' dialect collected from Twitter was classified based on the seed words that are used in only one dialect but not in the other dialects. Additionally, we used the user's location to enhance dialect classification and specified via spatial coordinates the country and dialect to which each tweet belongs.

We scraped Facebook posts and extracted comments from these posts, extracting from well-known Facebook pages in Arab countries. The extracted comments were classified based on the nationality of the Facebook page owner.

In general, social media can be used to collect an Arabic dialect text corpus. To make SMADC balanced we had to run the extractor for different durations for each dialect; for example we noticed that Twitter is more popular in Arabian Gulf area which help us to collect lots of tweets for GLF dialect whereas there were fewer tweets from North African countries and Iraq. In comparison with Twitter, Facebook is more popular in North Africa.

By combining texts from this range of sources, we were able to build an Arabic dialect text corpus with a more balance distribution of dialects than other Arabic dialect corpora discussed in Chapter 2. We plan to make the Social Media Arabic Dialect Corpus (SMADC) available to other researchers, in 2 formats (raw and cleaned) and with a range of metadata.

Part III

Arabic Dialect Texts Classification

Chapter 8
Initial Experiment in Classification

This chapter is based on Arabic Language WEKA-Based Dialect Classifier for Arabic Automatic Speech Recognition Transcripts. It describes an Arabic dialect identification system which we developed to participate in the VarDial2016 workshop at COLING 2016 Discriminating Similar Languages (DSL) 2016 shared task (Alshutayri *et al.* 2016). We classified Arabic dialects by using the Waikato Environment for Knowledge Analysis (WEKA) data analytic tool which contains many alternative filters and classifiers for machine learning. We experimented with several classifiers and the best accuracy was achieved using the Sequential Minimal Optimization (SMO) algorithm for training and testing process, set to three different feature-sets for each testing process. Our approach achieved an accuracy equal to 42.85% which is considerably worse in comparison to the evaluation scores on the training set of 80-90% and with training set 60:40 percentage split which achieved accuracy around 50%. We observed that Buckwalter transcripts were developed using the QCRI Automatic Speech Recognition (ASR) QATS system (Khurana and Ali 2016) are given without short vowels, though the Buckwalter system has notation for these. We elaborate such observations, describe our methods and analyse the training dataset.

8.1 Introduction

Arabic spoken dialect includes local words, phrases and even local variant morphology and grammar. With the spread of informal writing, for example on social networks and in local-dialect blogs, news and other online sources, Arabs are starting to write in their dialects. Because of the dominance of the MSA standard, there are no official writing standards for Arabic dialects, so spelling, morphology, lexis and grammar can be subject to individual transcription choice; it is up to a dialect speaker to decide how to write down their text.

Dialect speakers have been taught from school to write down everything in MSA, so they may well normalise or translate into MSA rather than phonetically transcribe words and utterances. Pronunciation of vowels in words constitute one of the key differences between Arabic dialects; but in written MSA, most vowels are omitted, leaving few clues to distinguish the source dialect.

All this makes it challenging to collect an Arabic dialects texts corpus. Previous DSL shared tasks (Zampieri *et al.* 2015) were based on officially recognised and differentiated languages (Bosnian v Croatian v Serbian, Malay v Indonesian etc.) with readily-available published sources; each example is a short text excerpt of 20100 tokens, sampled from journalistic texts. Local and national Arabic news sources and other journalistic text may include some local words but are still permeated and dominated by MSA, so a DSL Arabic dialects journalistic texts data-set would be contaminated with MSA/dialect code-switching, and blocks of MSA. The DSL organisers tried instead to gather dialect data more directly from dialect speakers, and tried to avoid the problem of translation into MSA by using Automatic Speech Recognition rather than human scribes. However, these texts were often much shorter than 20-100 words, sometimes only 1 or 2 word utterances; and these short utterances could be common to two or more dialects, with no further indicators for differentiation. Arabic linguistics experts in our team found clear evidence of MSA in numerous dialect texts, possibly introduced by the ASR transcription method; and numerous short utterance instances which had no linguistic evidence of a specific Arabic dialect.

The DSL shared task (Malmasi *et al.* 2016) was to identify Arabic dialects in texts in five classes: EGY, GLF, LEV, NOR, and MSA; in utterance/phrase level identification which is more challenging than document dialect identification, since short texts have fewer identifying features.

In this chapter we describe our method for defining features and choosing the best combination of classifier and feature-set for this task. We show the results of different variants of SMO with different feature-tokenizers. Finally, we conclude the chapter by discussing the limitations that affected our results.

8.2 Related Work

There have been many studies about Arabic dialect identification. One of these studies, presented by Zaidan and Callison-Burch, was described in Chapter 2. The authors classified dialect using a Naïve Bayes classifier with wordGram and charcterNGram as features and trained the classifier using unigram, bigram, and trigram models for word, and unigram, trigram, and 5-gram for character model. Based on the dataset they used in the training process they found that a unigram word model achieved best accuracy when examining the classifier using 10-fold cross validation (Zaidan and Callison-Burch 2014).

Another study which was explained in detail in Chapter 2, classifies Arabic dialects used a sentence-level approach to classify whether the sentence was MSA or Egyptian dialect (Elfardy and Diab 2013). They based the study on a supervised approach using Naïve Bayes classifier which was trained on labelled sentences with two types of features: Core Features to indicate if the given sentence is dialectal or non-dialectal. Meta Features to estimate whether the sentence is informal or not. The system accuracy was about 85.5%.

8.3 Data

The data for the shared task provided from the DSL Corpus Collection (Ali *et al.* 2016) is a dataset containing ASR transcripts of utterances by Arabic dialect speakers; there was no guarantee that each utterance was unique to a dialect. The task is performed at the utterance-level and they provided us with two sets. The first set is for training and contains 7,619 utterances labelled and divided unevenly between 5 classes that cover four Arabic dialects (EGY, GLF, LEV, NOR), and MSA (it is not clear how MSA speakers were procured as MSA is not a spoken dialect). Table 8.1 shows the number of utterances for each class. The second set is for testing, consisting of 1,540 unlabelled utterances. The utterance length ranged from one word to 3305 words with an average of 40 words/utterance and standard deviation = 60.

The number of utterances with word count less than 10 words is 1761 = 23.1%. Figure 8.1 shows the utterances distribution over utterance length.

Table 8.1 The number of utterances for each class

Classes	Number of Utterances
EGY	1578
GLF	1672
LEV	1758
NOR	1612
MSA	999

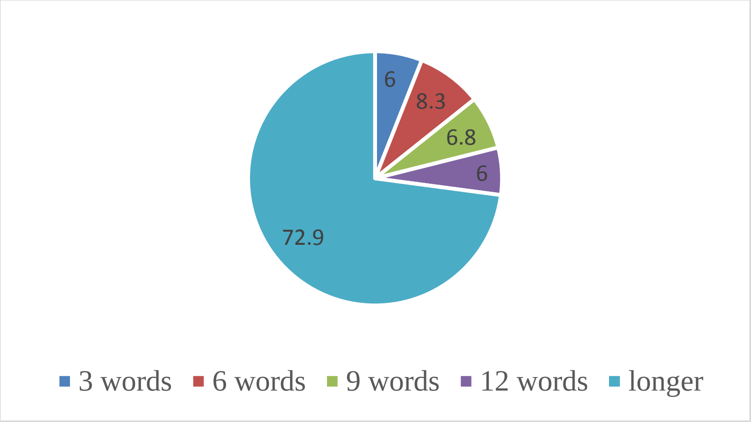


Figure 8.1 The sentence distribution over sentence length.

8.4 Method

At the beginning, we tried to choose the best classifier for the Arabic Dialects Identification (ADI) task from a set of classifiers provided by WEKA (Hall *et al.* 2009). This was done by measuring the performance of several classifiers on testing with the training dataset, 10-fold cross-validation, and by percentage split which divides the training set into 60% for training and 40% for testing. Table 8.2 reports results for a range of classifiers that we tried, using the WEKA StringToWordVector filter with WordTokenizer to extract words as features from utterance-strings. SMO was the best performing classifier. Table 8.3 shows the results of SMO using CharacterNGram Tokenizer with Max=3 and Min=1. The Word Tokenizer method, also known as Bag of Words, is a filter that converts the utterances into a set of attributes that represents the occurrence of words (delimited by space, comma, etc.) from the training set. It is designed to keep the n (which we set to 1000) top words per class. NGramWord Tokenizer is similar to Word Tokenizer with the exception that it also has the ability to include word-sequences with the maximum and minimum number of words; while CharacterNGram Tokenizer counts 1-2- and/or 3-character n-grams in the utterance-string.

The second column in Table 8.2 shows the results of the same (dialect-labelled) data as those used to train the classifier. The third column represents the results of 10-fold cross-validation. The fourth column shows the results of a randomly selected 40% of original training data for test of classifiers trained on the other 60%. After running the experiments in Table 8.2, we realised that 10-fold cross-validation is very time consuming (at least 10 times the duration of evaluation on training set or 60:40 percentage split) but produces the same classifier ranking, so we did not repeat the 10-fold cross-validation for Table 8.3.

Table 8.2 The accuracy of different classifiers (wordTokenizer)

Classifier	Evaluate on training set	10-fold cross-validation	60% train, 40% test
NaiveBayes	47.09	45.01	43.93
SMO	**89.29**	**52.82**	**50.13**
J48	72.28	43.26	41.5
ZeroR	23.07	23.07	22.41
JRip	35.67	32.76	32.51

Table 8.3 The accuracy of different classifiers (CharacterNGramTokenizer)

Classifier	Evaluate on training set	60% train, 40% test
SMO	**94.46**	**53.08**
J48	88.36	37.53
REPTree	53.71	35.56
JRip	41.62	36.35

```
inst#     actual  predicted        error prediction
4              2:GLF                    3:NOR
"$Ahd AlgrAfyk tfAqmh    Q          GLF"
"شاهد الغرافيك تفاقمه             Q   GLF"


15              2:GLF                    4:LEV
"$Ark wEqb Eqdyn llywm Em byEtrDwA mEkm lkn   Q     GLF"
"شارك وعقب عقدين لليوم عم بيعترضوا معكم لكن        Q        GLF"
```

Figure 8.2 Example of misclassified sentences.

Looking at Table 8.2, we noticed that by using SMO we got 6803 utterances correctly classified and 816 utterances misclassified. To improve the identification results we output the misclassified utterances and converted the text from Buckwalter to normal readable Arabic script because looking at the Buckwalter texts is difficult even if you know the Buckwalter transliteration system (Buckwalter 2002). Then, we asked our Arabic linguistic experts to examine some of the texts which were misclassified, and try to find features which might correctly predict the dialect. Figure 8.2 shows example of misclassified utterances.

The example above shows that instance 4 is actually labelled class 2:GLF but the classifier made an error and predicted class 3:NOR.

The Arabic linguistics experts analysed the shortcomings in the misclassified utterances from the training data. They found that numerous texts are too short to say anything about their dialect origins, for example: $Ark is a short one-word text which appears unchanged labelled as different dialects. Some of the utterance seem to be entirely MSA despite having dialect labels, possibly due to the Automatic Speech Recognition method used; and a lot of the utterance have at least some MSA in them. Some utterances that have recognisable dialect words often have words which are shared between two or more dialects. They even found some utterances labelled as one dialect but evidently containing words not from that dialect; for example utterance 254 below is labelled as LEV in the training set, but contains a non-LEV lexical item, see Figure 8.3.

This analysis led us to conclude that it is impossible in principle for WEKA to classify all instances correctly. There is a proportion of texts that cannot be classified, and this sets a ceiling on accuracy that it is possible to achieve approximate to 90-91%.

```
inst#      actual  predicted        error prediction
254             4:LEV                   2:GLF
"<ElAmy h*A Hqh ly$    Q    LEV"
"إعلامي هذا حقه ليش             Q    LEV"   This is not LEV, Hqh ly$ is not LEV
```

Figure 8.3 Example of LEV misclassified sentences.

8.4.1 Term Frequency (TF)

Term Frequency represents the frequency of particular word in a text (Gebre *et al.* 2013). Based on our task, we found some words are used more frequently in a particular dialect than in other dialects. We used the weight of TF to indicate the importance of a word in text.

8.4.2 Inverse Document Frequency (IDF)

Inverse Document Frequency was used to scale the weight of frequent words which appear in different texts (of more than one dialect); a word which appears in many dialects cannot be used as feature (Gebre *et al.* 2013).

8.5 Features

The first experiments to choose the best classifier to identify Arabic dialects showed that SMO is the best machine learning classifier algorithm, but we may increase accuracy by adjusting parameters and features taken into account.

The WordTokenizer setting assumes features are words or character-strings between spaces while the CharacterNGramTokenizer assumes features are 1/2/3-character sequences. We used the WEKA StringToWordVector filter with WordTokeniser which splits the text into words between delimiters: (full stop, comma, semi-colon, colon, parenthesis, question, quotation and exclamation mark). After that, we decided to use

SMO, but we suggested trying character n-grams as units, instead of words as units. We used CharacterNGramTokenizer to splits a string into an n-gram with min and max gram. We set Max and Min both to 1 which gives a model based on single characters; max and min both to 2 which is a char-bigram model; max and min both to 3 gives us a trigram model; max and min to 4 gives a 4-gram model. Table 8.4 shows the results of different gram values when evaluating with the training set and a 60:40 percentage split of the training set. Table 8.4 suggests that 4-gram model may be inappropriate as the training data is not sufficiently large.

Table 8.4 The accuracy of SMO classifier with CharacterNGram

Features	**Evaluate on training set**	**60% train, 40% test**
Character UniGram	43.23	41.11
Character BiGram	78.08	**52.4**
Character TriGram	**94.62**	49.87
Character QuadGram	85.01	50.39

In addition, in order to improve performance we replaced the dimensions of the feature vector with their IDF and TF weight which is a standard method from Information Retrieval (Robertson 2004). We changed values of TF/IDF, and Word Count (WC) between True and False each time to see which combination of settings gives best accuracy using the training set and 60:40 percentage split. Tables 8.5, and 8.6 show the results of variants combinations by using the SMO classifier with different tokenizers which are: WordTokenizer, NGramTokinizer, and CharacterNGram. The accuracy in Table 8.5 results from using same training set, while in Table 8.6 it was achieved by using 40% from the training dataset for testing and 60% for training.

Table 8.5 The accuracy of SMO classifier using random data from the training dataset.

		TF		-TF	
		IDF	**-IDF**	**IDF**	**-IDF**
WordTokenizer	**WC**	83.69	83.69	83.69	79.1
	-WC	83.69	89.29	83.69	89.29
NGramTokinizer (max=3, min=1)	**WC**	83.33	83.33	78.83	78.82
	-WC	88.97	88.97	88.97	88.97
CharacterNGram (max=3, min=1)	**WC**	84.87	84.88	71.64	71.65
	-WC	94.4	94.46	94.4	94.46
CharacterNGram (max=3, min=3)	**WC**	86.38	86.38	76.02	76.01
	-WC	**94.62**	**94.62**	**94.62**	**94.62**

Table 8.6 The accuracy of SMO classifier using 40% from the training data.

		TF		-TF	
		IDF	**-IDF**	**IDF**	**-IDF**
WordTokenizer	**WC**	51.05	51.02	51.05	49.44
	-WC	51.05	50.26	51.05	5026
NGramTokinizer (max=3, min=1)	**WC**	50.89	50.89	49.48	49.48
	-WC	49.7	49.7	49.7	49.7
CharacterNGram (max=3, min=1)	**WC**	**53.12**	53.05	52.33	52.3
	-WC	47.67	47.64	47.67	47.64
CharacterNGram (max=3, min=3)	**WC**	**53.12**	**53.12**	51.9	51.87
	-WC	49.87	49.87	49.87	49.87

According to the above tables, the best results are achieved using SMO with CharacterNGram (Max=3, Min=1, IDF=True, TF=True, WC=True) which gets the same score as CharacterNGram (Max=3, Min=3, IDF=True, TF=True, WC=True) in testing “60:40” percentage spilt equal to 53.12%, but Max=3, Min=3 scores higher on Training set equal to 86.38%. We supposed the models were very similar: (3-1) has all the trigrams of (3-3) and also some bigrams and unigrams but these probably are common to all or most dialects and so do not help in discrimination.

However, the task rules stated that we were restricted to trying our three best classifiers, so at this stage we had to choose three ”best” results. Sometimes the training set score is high, but the 60:40 percentage split score is low; and sometimes the 60:40 percentage split score is high but the training set score is poor. So, we decided to use 60:40 percentage split as our guide to choose the best combination, because using the training set for training as well as evaluation may over-fit the training set. Furthermore, we noticed that the best combination of TF/IDF and WC values is when all values are True. Figure 8.4 below shows the chart that summarises the results for different combinations of TF/IDF and WC values with SMO classifier.

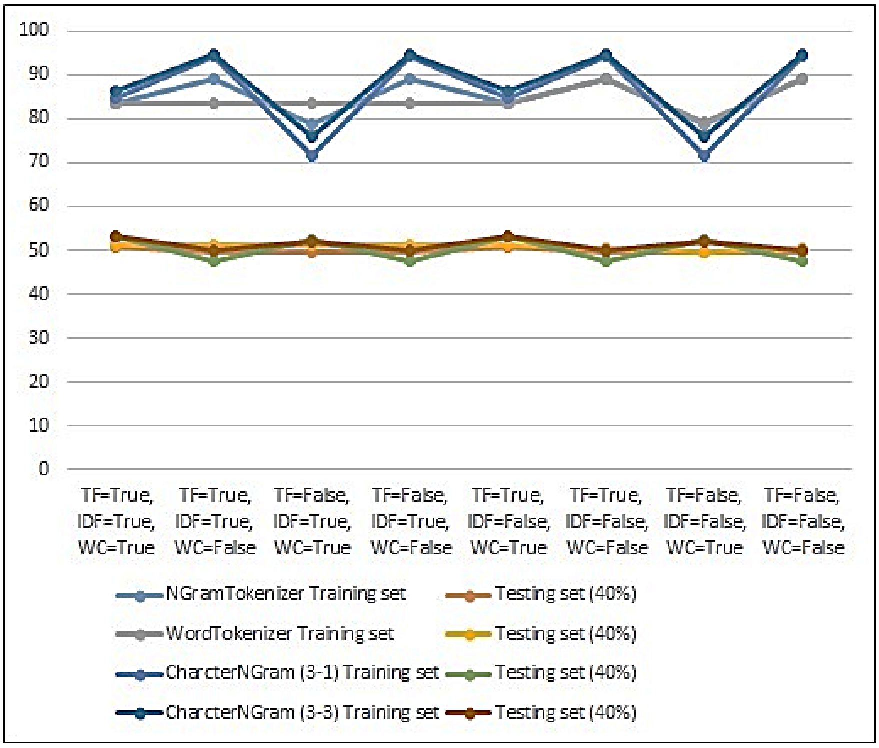


Figure 8.4 Summary of different combinations of TF/IDF and WC values with SMO classifier.

8.6 Results

We finally evaluated our system using the supplied separate test data set and submitted three different results using the SMO classifier with three different features-sets:

Run1 is obtained by using CharacterNGram, Max=3, Min=3, IDF=True, TF=True, WC=True. This achieved an accuracy of around 42%.

Run2 is obtained by using WordTokenizer, IDF=True, TF=True, WC=True, we removed ’ delimiter because it is used as a letter in the Buckwalter transcription. The performance of this model equals 37%.

Run3 is obtained by using NGramTokenizer, Max=3, Min=1, IDF=True, TF=True, WC=True, also we removed ’ delimiter as in Run2. This achieved an accuracy equalling 38%. Table 8.7 shows the results of the three runs.

Table 8.7 The result of the three classifiers

Run	Accuracy	F1 (weighted)
1	42.86	43.49
2	37.92	38.41
3	38.25	38.71

8.7 Conclusion

We built systems that classify Arabic dialects in shared tasks by using the WEKA data analytic tool and SMO machine learning algorithm after testing variants of SMO with different tokenizers; IDF, TF and WC values, and comparing the results by testing on a training set (around 80-90% correct) against using 60% to train and separate 40% for test (around 50% correct). By testing our system on the testing data set, we got an average accuracy of 42.85%. We think that this low accuracy was due to ASR transcription because most of the misclassified instances are not readily classifiable even by three human Arabic Linguistic experts, which provides strong evidence that a Machine Learning classifier can do no better. Clearly if the training data contains inappropriately-transcribed text and mislabelled instances, this will reduce the ceiling of accuracy that any classifier can achieve.

Chapter 9
Classifying Arabic Dialects in Three Different Corpora Using Ensemble Classifier

This chapter is based on Classifying Arabic Dialects in Three Different Corpora Using Ensemble Classifier (Alshutayri and Atwell, in preparation). It describes the method that we used to classify a text as belonging to a certain Arabic dialect and presents the comparison between three different data sets to explore which is the best source of written Arabic dialects. The three data sets used in this experiment were: the data set provided for the Discriminating Similar Languages (DSL) 2016 shared task, some tweets collected from Twitter, and readers' comments collected from an online newspaper. We classified Arabic dialects by using the ensemble method by combining Sequential Minimal Optimization (SMO) algorithm with multinomial Naive Bayes (MNB). To apply our approach we used Waikato Environment for Knowledge Analysis (WEKA) data analytic tool which contains many alternative filters and classifiers for machine learning. Our approach achieved an accuracy of 60.68% using a combination of the three sources of data sets for training and testing processes, and 50.17% when testing the system trained in one source of data set using a combination of the three sources of testing data sets.

In this chapter, we present a comparison between three different sources of data by applying SMO and MNB classifiers in each data set with three different tokenizers. In addition, we describe our method for applying the ensemble classifier. Finally, we conclude the chapter by discussing the limitations that have affected our results.

9.1 Data

In this experiment we used three different sources of data to compare the accuracy of the results and to check which is the best source of written Arabic dialects.

The three data sets are:

- The first data set was transcripts of utterances by Arabic dialect speakers using Automatic Speech Recognition (ASR) provided from the DSL shared Task 2016 (Ali *et al.* 2016). The dataset containing two sets. The first set is for training and contains 7,619 utterances labelled and divided unevenly between 5 classes that cover four Arabic dialects (EGY, GLF, LEV, NOR), and MSA. As we noticed in the training data set, there was no guarantee that each utterance was unique to a dialect. The second set is for testing, consisting of 1,540 labelled utterances.

- The second data set was tweets (sentences) we collected from Twitter for five country groups to cover five Arabic dialects (EGY, GLF, LEV, NOR, IRQ), and MSA. The data set divided into two sets: The first set contains 8,407 labelled tweets used for training and divided unequally between the Arabic dialects. The second set is for testing, and contains 1,764 labelled tweets. We wrote a paper that describes our method in detail in exploring Twitter as a source of an Arabic dialect corpus (Alshutayri and Atwell 2017).
- The third data set was readers' comments (sentences) we collected from an online newspaper that issues from different countries in the Arab world to cover five Arabic dialects (EGY, GLF, LEV, NOR, IRQ), and MSA. As well as the previous two sources of data sets, the comments data set is divided into two sets: the first consists of 6,790 labelled comments used for training and divided unequally between the Arabic dialects, whereas the second set is for testing, and contains 2,309 labelled comments.

Table 9.1 shows the number of utterances-sentences for each class in each data set that was used in the training process and Table 9.2 shows the number of utterances-sentences for each class in each data set that used in the testing process.

Table 9.1 The number of sentences per class for each data set (Training data).

Data Set	MSA	GLF	EGY	NOR	LEV	IRQ
DSL	999	1672	1578	1612	1758	0
Twitter	317	2152	1541	1585	1533	1279
Newspaper Comments	3861	967	524	641	672	125

Table 9.2 The number of sentences per class for each data set (Testing data).

Data Set	MSA	GLF	EGY	NOR	LEV	IRQ
DSL	274	256	315	351	344	0
Twitter	102	450	326	377	286	223
Newspaper Comments	845	700	316	145	222	81

9.2 Method

The task of classifying is performed at the utterance-sentence level using Weka (Hall *et al.* 2009). We used the SMO algorithm which gave us good results in DSL2016 compared to other algorithms. In addition, we tried to find another classifier that may improve the accuracy of classification when the ensemble method is used. For this we measured the performance of some classifiers such as Naive Bayes, K-nearest neighbours (KNN) and Multinomial Naive Bayes (MNB) for testing with the percentage split which divides the training set into 60% for training and 40% for testing. We found that MNB is the best classifier that can be used in text classification besides SMO classifier.

9.3 Features

Good feature selection may increase the accuracy of classification, so we adjusted some parameters and features taken into account. We used WEKA StringToWordVector filter with three different tokenizers: WordTokenizer to extract words between spaces or any other delimiters such as full-stop, comma, semi-colon, colon, parenthesis, question, quotation and exclamation mark; CharacterNGramTokenizer to extract a sequence of characters based on the number of grams; and NGramTokenizer to extract a sequence of words with maximum and minimum number of words. In our experiment we decided to use the SMO and MNB with the three different tokenizers in order to choose the feature that most accurately distinguishes between Arabic dialects. In both NGramTokenizer and CharacterNGramTokenizer we set Max and Min to 3 which gave us a best accuracy according to our experiment in DSL. In addition, checked the effect of replacing the dimensions of the feature vector with their IDF and TF weight on the performance of the classification, but we found that the use of TFIDF may improve the accuracy based on the feature and data used. Table 9.3 summarises the results of the classification process using a 60:40 percentage split of the training set with different tokenizers to extract words as features from utterance-sentences.

We found that WordTokenizer is the most accurate feature in classifying dialects with SMO and MNB classifiers. According to the results shown in Table 9.3, we will use an ensemble method because SMO was best to classify newspaper Comments while MNB was best classifier to classify DSL and Tweets. However, TF-IDF will not improve the accuracy with WordTokenizer and DSL data so we decided to not use it.

Table 9.3 Comparison of SMO and MNB with different features.

Data Set	Tokenizer	SMO-TFIDF	SMO	MNB-TFIDF	MNB
DSL	WordTokenizer	51.21	50.36	60.2	**61.48**
Twitter		93.1	93.22	88.64	**93.69**
Newspaper Comments		72.82	**93.69**	75.77	71.61
DSL	NGramTokenizer	50.24	47.6	54.29	**55.74**
Twitter		92.53	**92.56**	87.83	91.97
Newspaper Comments		72.34	72.05	75.25	**75.92**
DSL	CharcterNGramTokenizer	53.24	47.6	54.29	**55.74**
Twitter		**89.32**	88.61	87.36	88.73
Newspaper Comments		**69.91**	66.27	66.89	66.2

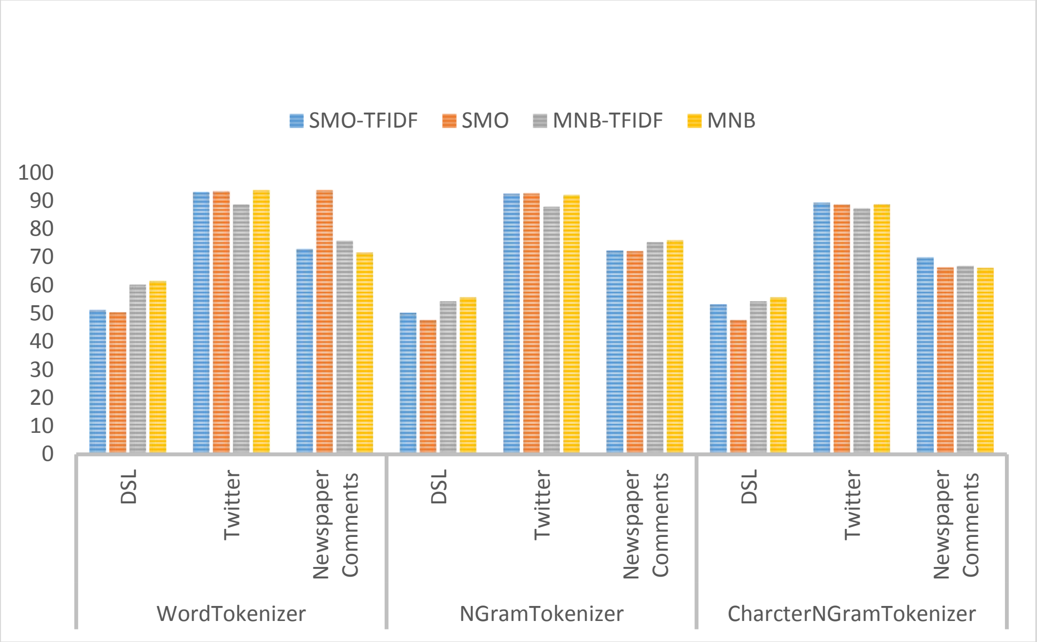


Figure 9.1 Summary of different tokenizers and combinations of TF/IDF with SMO and MNB.

9.4 Ensemble Classifier

Nowadays, in machine learning problems, it has become very popular to use an ensemble classifier instead of a single classifier. It works to combine different classifiers to classify instances instead of one classification algorithm to improve overall accuracy through enhanced decision making (Malmasi and Dras 2018). So, combining multiple classifiers will be more reliable and more sophisticated to identify or classify documents instead of relying on decision by one classifier.

9.5 Results

We did four experiments using ensemble classifiers which consists of two classifiers; SMO and MNB with WordTokenizer.

- **First experiment:** the system trained using a combination of the three sources of data (Training dataset) then tested each source of data set separately.
- **Second experiment:** the system trained using a combination of the three sources of data (Training dataset) then tested a combination of the three sources of data set (Testing dataset).
- **Third experiment:** the system trained using a single source of data sets, then tested each source of data set separately.

- **Fourth experiment:** the system trained using a single source of data sets, then tested a combination of the three sources of data.

Table 9.4 reports the results of the fourth experiment. Table 9.5 shows the results of the three experiments.

Table 9.4 The results of the three experiments.

Test Set	First experiment	Second experiment	Third experiment
DSL	48.7	60.68	49.67
Twitter	**69.78**		**76.95**
Newspaper Comments	66.86		62.32

Table 9.5 The results of the fourth experiments.

Training Data Set	Accuracy
DSL	39.66
Twitter	46.8
Newspaper Comments	50.17

9.6 Conclusion

We built systems that classify Arabic dialects generated from three different sources of text data using the WEKA data analytic tool and ensemble classifier consisting of SMO and MNB machine learning algorithms after testing variants of SMO and MNB with different tokenizers, IDF and TF values. Then we compared the results tested in the training set using 60:40 percentage split as 60% to train and separate 40% for test. We did four experiments to distinguish which is the best source of Arabic dialect texts and we found the best accuracy equal to 50.17% when using text from newspaper comments. In addition, we achieved a high accuracy equal to 69.78% when testing a Twitter data set in the system trained in a combination of all sources of data sets; we think that happened because many Twitter users write in their dialect and the text does not exceed 140 characters (at the time of the data collected). The problem with newspapers comments in the other experiments is because many readers comment using MSA instead of their dialect, especially in political news, so many comments mix MSA and dialect together. The problem with the DSL data set is that it contains inappropriate-transcribed text, mislabelled instances, and some of the same utterances had more different labels. Because of this, we decided to create new corpus and label it using the crowdsource method to build our classification model using an appropriate dataset.

Chapter 10
Automatic Dialect Classification

Text classification is identifying a predefined class or category for a written document by exploring its characteristics or features (Ikonomakis *et al.* 2005; Sababa 2018). A machine learning algorithm works to identify the class for each document based on a model trained on a set of labelled documents; this is known as supervised learning.

This chapter describes the methods used to classify Arabic dialect texts and presents the achieved results, in addition to the techniques used to improve the accuracy.

The dataset used in this chapter is a subset of Social Media Arabic Dialect Corpus (SMADC) which was collected using Twitter, Facebook and comments from online newspapers as described in Chapters 3, 4, and 5, in addition to other available Arabic dialect corpora described in Chapter 2.

10.1 Lexicon Based Methods

10.1.1 The Datasets used in Lexicon Based Methods

Chapter 6 presented the annotation system or tool[4] 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 are labelled either dialectal documents or MSA documents.

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 produced by translating the English list of seed word (Sharoff 2006). The final MSA word list contains 29674 words, one word per line in a .txt file, divided into 15196 MSA words extracted from MSA documents in our labelled corpus, 13015 words as stop words extracted from Arabic web pages, 1000 words extracted from Sketch Engine, and 463 words as seed words. This word list is called “StopWords1” and was used in deleting all MSA words from dialect

[4] www.alshutayri.com

documents, as these may contain some MSA words due to the code switching between MSA and dialect.

The annotated 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. 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 train the classifier and 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.

In addition to SMADC, other Arabic dialect corpora were used to evaluate the performance of the system. The first corpus called Arabic Multi Dialect Written Corpora (AMDWC), created by Almeman and Lee (2013) covered four Arabic dialects: GLF, EGY, LEV, and NOR. The second corpus called Arabic Online Commentary Dataset (AOCD), created by Zaidan and Callison-Burch (2011) covered three Arabic dialects: GLF, EGY, and LEV. The third corpus called Arabic Dialect Dataset (ADD) created by El-Haj *et al.* (2018) covered four Arabic dialects: GLF, EGY, LEV, and NOR. All these corpora were described in detail in Chapter 2. Table 10.1 shows the total number of dictionary word-types in each dialect in each corpus.

Table 10.1 Number of words in each dictionary created using each corpus.

Corpus	GLF	EGY	LEV	IRQ	NOR
SMADC	3472	2032	2028	1889	1436
AMDWC	956687	793018	786167	0	740072
AOCD	57868	58910	45262	0	0
ADD	17842	31074	19198	0	20190

To classify the Arabic dialect text using the Lexicon based method, we used a range of different methods of classification and conducted five experiments, all of which used a dictionary for each dialect. The difference between the five experiments is the size of the dictionary used in each model. The following sections describe the difference between the experiments conducted, and the result of each experiment.

10.1.2 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 dialectal dictionaries are .txt files containing one word per line. 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 10.1, to classify each document as being a specific dialect, the system follows four steps:

1. 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.
2. The result from the first step is a document containing only dialectal words.
3. Detect the dialect for each word in the document by comparing each word with the words in the dictionaries created for each dialect.
4. Identify dialect.

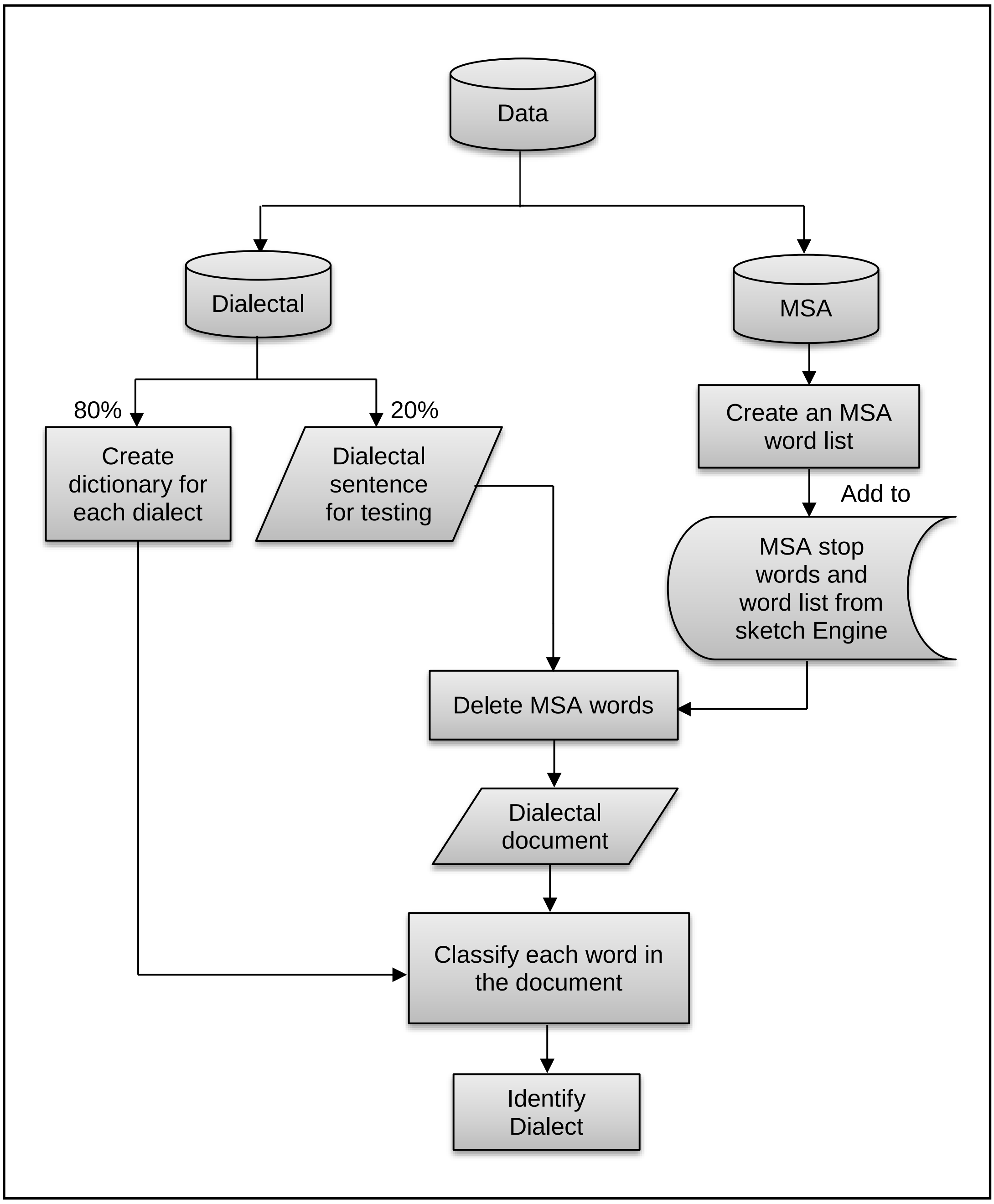

Data
Dialectal
MSA
80%
20%
Create dictionary for each dialect
Dialectal sentence for testing
Create an MSA word list
Add to
MSA stop words and word list from sketch Engine
Delete MSA words
Dialectal document
Classify each word in the document
Identify Dialect

Figure 10.1 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 10.2 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. As shown in Table 10.2 the dialectal terms method scored 56.91% which indicate that using this method is not effective in dealing with ambiguous words, because it ignores the context of words, and as is known, context is the main means of ambiguity resolution (Adouane and Dobnik 2017).

' ماشاء الله حلو كتير ' , LEV

Unclassified

Figure 10.2 Example of unclassified document.

Table 10.2 shows the accuracies achieved by applying the dialectal terms method on the testing set using the dictionaries created using the dialectal terms written by the annotators. The first column represents the MSA words list used to delete MSA words from documents before classification, and the second column represents the achieved accuracies based on using the dialectal terms to create dictionaries. The best accuracy is 56.91 with 140 documents correctly classified using StopWords1. Based on this method, 85 documents were unclassified to a specific dialect because they consist of some ambiguous terms which are used in more than one dialect, as in the example of Figure 10.2. As a solution to this problem, a voting method is used and another way is using a frequent term method which described in Section 10.1.4.

Table 10.2 The result of using the dialectal terms method.

MSA	SMADC
StopWords1	**56.91%** 140/250
StopWords1 and ADD MSA Documents	55.14% 134/250
StopWords1, ADD MSA Documents, and AOCD MSA Documents	48.34% 102/246
Without delete MSA Words	55.60% 139/250

10.1.3 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 similar to dialectal term method presented in Section 10.1.2. The classification starts at the word level based on the dictionaries created from the 80% training set of documents described in Section 10.1.1. 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 A. The size of the matrix is $A_{|n| \times |5|}$, where n is the number of words in each document. n varies from document to another according to the number of words in each document, and 5 is the number of dialects (EGY, NOR, GLF, LEV, and IRQ).

10.1.3.1 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 (10.1).

$$a_{ij} = \begin{cases} 1\ if\ word \in dialect \\ 0\ otherwise \end{cases} \qquad (10.1)$$

The following illustrates the method. We apply Equation (10.1) on the following document **A** labelled as IRQ dialect:

يعجبني اغرد عن كلشي يخطر بالي ,IRQ

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

The proposed model is an extension of the dialectal terms method with a voting method to deal with an ambiguity. The model used the dictionaries created using SMADC to classify the document by looking in the dictionaries for each word in the document. The result of classification is IRQ according to Table 10.3; 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.

Table 10.3 The matrix representation of document A with simple voting.

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
بالي	0	1	0	0	0
Total	**2**	**2**	**4**	**1**	**1**

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**:

هههههههههههه 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.

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

هههههههههههه خليتني ليش بهالطريقة بس للحين بهالشكل

According to the result in Table 10.4 the document is unclassified because more than one dialect has the same number of words.

Table 10.4 The matrix representation of document B with simple voting.

Words	NOR	EGY	IRQ	LEV	GLF
ههههههههههه	0	1	1	1	0
خليتني	0	0	0	0	0
ليش	1	0	1	1	1
بهالطريقة	0	0	0	0	0
بس	1	1	1	1	1
للحين	0	0	0	0	1
بهالشكل	0	0	0	0	0
Total	**2**	**2**	**3**	**3**	**3**

10.1.3.2 Weighted Voting Method

This method is used to solve the problem of unclassified documents in Section 10.1.3.1. 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 (10.2). If a word can belong to more than one dialect, its vote is shared between the dialects.

$$a_{ij} = \begin{cases} \frac{1}{m} \; if \; word \in dialect \\ 0 \; otherwise \end{cases} \qquad (10.2)$$

$\frac{1}{m}$ 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 10.5.

Table 10.5 The matrix representation of document B with weighted voting.

Words	NOR	EGY	IRQ	LEV	GLF
ههههههههههه	0	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{3}$	0
خليتني	0	0	0	0	0
ليش	$\frac{1}{4}$	0	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$
بهالطريقة	0	0	0	0	0
بس	$\frac{1}{5}$	$\frac{1}{5}$	$\frac{1}{5}$	$\frac{1}{5}$	$\frac{1}{5}$
للحين	0	0	0	0	1
بهالشكل	0	0	0	0	0
Total	**0.45**	**0.5333**	**0.7833**	**0.7833**	**1.45**

10.1.3.3 Results of Voting Method

This section presents the results of the testing dataset with the different dialect dictionaries. The description of the dataset used to test the model and to create the dictionaries is presented in Section 10.1.1. This method is focused on the existence of the word in the dictionary. The dictionaries consist of the words found in the text, one word per line. So, the frequency of the word is ignored, unlike the frequent term method which described in Section 10.1.4.

In this section the testing dataset is the same in all the following sections. The dictionaries were created using four corpora: Social Media Arabic Dialect Corpus (SMADC), Arabic Multi Dialect Written Corpora (AMDWC), Arabic Online Commentary Dataset (AOCD), and Arabic Dialect Dataset (ADD).

Each dictionary created using the documents were labelled as dialect but due to code switching, there are some MSA words in each dictionary extracted from the dialect documents.

In the first experiment, the documents resulting from the annotation tool as mentioned in Section 10.1.1 were used to create dialect dictionaries. In the second experiment, we used AMDWC (Almeman and Lee 2013), the third experiment used AOCD (Zaidan and Callison-Burch 2011), and in the fourth experiment the dialect dictionaries were created using ADD (El-Haj *et al.* 2018). In the last experiment all dictionaries from the corpora used in the previous experiments are combined together.

The MSA word list used in the first experiment is StopWords1 as described in Section 10.1.1, but we increased the size of MSA StopWords1 list using the MSA documents in AOCD and MSA documents in ADD. The MSA list created using StopWords1 and ADD consists of 43428 words, and the MSA list created using StopWords1, ADD and AOCD consists of 178979 words.

10.1.3.4 Results of Voting Method using Social Media Arabic Dialect Corpus (SMADC)

The model in this experiment uses dialect dictionaries based on the texts collected by Alshutayri and Atwell (2017), Alshutayri and Atwell (2018b), and Alshutayri and Atwell (2018c) to create the dialect dictionaries using Social Media Arabic Dialect Corpus (SMADC) to classify each word in the document. This corpus covers five Arabic dialects: EGY, GLF, LEV, IRQ, and NOR. Therefore, five dictionaries are created to cover EGY dialect, GLF dialect, LEV dialect, IRQ dialect, and NOR dialect.

The model was tested using the testing dataset described in Section 10.1.1. The highest accuracy achieved is 74.0% without deleting MSA words from the classified document. The lowest accuracy is 55.28% when deleting MSA words using combination of StopWords1, ADD MSA documents, and AOCD MSA documents. 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 10.1.3.1. Table 10.6 shows the different accuracies achieved using SMADC.

The first column in Table 10.6 shows the list of MSA stop words used to delete MSA words from each document before classifying the document based on the dictionaries. The second column overhead represents the name of the corpus used to create dictionaries, and the second and the third columns below represent the methods used to classify documents. The cells inside the second and third columns present the achieved accuracies using these methods and the number of correctly classified documents divided by the number of whole test set.

Table 10.6 The result of using voting methods based on the dictionary created from SMADC.

MSA	**SMADC**	
	Simple Vote	**Weighted Vote**
StopWords1	69.19% 173/250	72.0% 180/250
StopWords1 and ADD MSA Documents	69.19% 173/250	72.8% 182/250
StopWords1, ADD MSA Documents, and AOCD Documents	54.06% 133/246	55.28% 136/246
Without deleting MSA Words	65.60% 164/250	**74.0%** 185/250

10.1.3.5 Results of Voting Method using Arabic Multi Dialect Written Corpora (AMDWC)

The dialect dictionaries used in this model were created using the texts collected by Almeman and Lee (2013). The Arabic Multi Dialect Written Corpus (AMDWC) covers four Arabic dialects: EGY, GLF, LEV, and NOR. So, four dictionaries were created to cover each dialect. As the IRQ dialect is not covered in this corpus, the IRQ dictionary was created from SMADC was used in this experiment, to make the experiment cover all five Arabic dialects.

Using the same testing dataset the model showed low accuracies ranging between 22%-26% due to the noise in the dictionaries, MSA words appearing in the dialect corpus and similar dialect words found in more than one dictionary. Table 10.7 shows the different accuracies achieved using the Arabic multi dialect written corpora.

Table 10.7 The result of using voting methods based on the AMDWC.

MSA	**AMDWC**	
	Simple Vote	**Weighted Vote**
StopWords1	22.40% 56/250	25.6% 64/250
StopWords1 and ADD MSA Documents	22.40% 56/250	25.6% 64/250
StopWords1, ADD MSA Documents, and AOCD Documents	24.39% 60/246	**26.01%** 64/246
Without deleting MSA Words	22.8% 57/250	25.6% 64/250

10.1.3.6 Results of Voting Method using Arabic Online Commentary Dataset (AOCD)

In this experiment, the dictionaries were created using the texts collected by Zaidan and Callison-Burch (2011) to create three dictionaries to cover EGY dialect, GLF dialect, and LEV dialect. IRQ and NOR dialect dictionaries were created from SMADC, to make the experiment cover all five Arabic dialects.

The model showed good accuracies using the weighted voting method in comparison to simple voting method. The highest accuracy achieved using this model is 56.39% based on MSA StopWords1 and ADD MSA list. The lowest accuracy was when the model was tested without deleting MSA words which gave an accuracy of 50.0%. Table 10.8 shows the different accuracies achieved using the AOCD corpora.

Table 10.8 The results using voting methods based on the AOCD.

MSA	**AOCD**	
	Simple Vote	**Weighted Vote**
StopWords1	45.6% 114/250	53.6% 134/250
StopWords1 and ADD MSA Documents	48.0% 120/250	**56.39%** 141/250
StopWords1, ADD MSA Documents, and AOCD Documents	50.81% 125/246	54.06% 133/246
Without deleting MSA Words	43.2% 108/250	50.0% 127/250

10.1.3.7 Results of Voting Method using Arabic Dialect Dataset (ADD)

The model in this experiment uses the dictionaries created using the texts collected by El-Haj *et al.* (2018). Four dictionaries were created to cover EGY dialect, GLF dialect, LEV dialect, and NOR dialect, and the dictionary created from SMADC was used as IRQ dictionary, to make the experiment cover all five Arabic dialects.

The accuracy achieved ranged between 38%-44%. The highest accuracy is 44.80%, achieved when the system tested used the StopWord1 and ADD MSA list based on weighted voting method. The lowest accuracy 38.4% is without deleting MSA words. Table 10.9 shows the different accuracies achieved using ADD.

Table 10.9 The results using voting methods based on the ADD.

MSA	ADD	
	Simple Vote	**Weighted Vote**
StopWords1	40.40% 101/250	**44.80%** 112/250
StopWords1 and ADD MSA Documents	38.80% 97/250	**44.80%** 112/250
StopWords1, ADD MSA Documents, and AOCD Documents	40.65% 100/246	41.86% 103/246
Without deleting MSA Words	34.8% 87/250	38.4% 96/250

10.1.3.8 Results of Voting Method using Combination of Different Arabic Corpora

This model combines all dictionaries used in the previous experiments and creates one dictionary for each dialect. So, five dictionaries were created to cover EGY dialect, GLF dialect, LEV dialect, IRQ dialect, and NOR dialect. Each dictionary consists of the words found in all the corpora, one word per line.

The best accuracy achieved using this model is 27.23% using the weighted voting method and MSA StopWords1 list, ADD corpus, and AOCD corpus. The lowest accuracy 26.40% occurs without deleting MSA words. Table 10.10 shows the different accuracies achieved using combination of Arabic dialect corpora.

Table 10.10 The result of using voting methods based on combination of Arabic Dialects corpora.

MSA	**Combination of Arabic Dialect Corpora**	
	Simple Vote	**Weighted Vote**
StopWords1	25.2% 63/250	26.8% 67/250
StopWords1 and ADD MSA Documents	25.2% 63/250	27.20% 68/250
StopWords1, ADD MSA Documents, and AOCD Documents	26.42% 65/246	**27.23%** 67/246
Without deleting MSA Words	24.4% 61/250	26.40% 66/250

According to the previous result, the weighted voting method showed good results in comparison with the simple voting method. Table 10.11 shows a summary of all accuracies achieved using different Arabic dialect corpora with different MSA word lists based on weighted voting method. The first column in Table 10.11 shows the list of MSA stop words used to delete MSA words from each document. The columns from 2 to 6 represent the accuracies scored based on different Arabic dialect corpora. The highest accuracy is 74% based on the dictionaries created using SMADC without deleting MSA words, then 56.39% using AOCD corpus with StopWords1 and ADD MSA list, 44.80% based on ADD corpus with the StopWords1, and ADD MSA list. The dictionaries created using combination of all Arabic dialect corpora scored 27.23%, and the lowest accuracy is 26.01% based on AMDWC corpus.

Table 10.11 Summary of results achieved using Weighted Voting Method with different Arabic Dialects corpora.

MSA	**SMADC**	**AMDWC**	**AOCD**	**ADD**	**Combined of ADC**
StopWords1	72.0%	25.6%	53.6%	**44.80%**	26.8%
StopWords1 and ADD MSA	72.8%	25.6%	**56.39%**	**44.80%**	27.20%
StopWords1, ADD MSA, and AOCD MSA	55.28%	**26.01%**	54.06%	41.86%	**27.23%**
Without deleting MSA words	**74.0%**	25.6%	50.0%	38.4%	26.40%

10.1.4 Frequent Terms Methods

Another method presents in this section to solve the problem shown in the dialectal terms method described in Section 10.1.1 and to improve the accuracy of classification achieved using the voting method. In the 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 10.1 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:

$$W(word, dict) = \frac{F(word)}{L(dict)} \qquad (10.3)$$

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 (10.3) 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.

10.1.4.1 Weight Average Method (WAM)

This method based on calculating the average of the word weights for each document. Table 10.12 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:

$$Avg_{dialect} = \frac{\sum_{dialect} W}{L(document)} \qquad (10.4)$$

Table 10.12 The value of the weight of each word.

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

By calculating the average for the dialect vectors using the Equation (10.4), the model classified the document as LEV dialect, after comparing the results of the average obtained from the following equations.

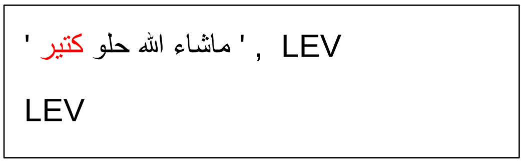
$$Avg_{EGY} = \frac{\sum_{EGY} W}{L(document)} = \frac{0 + 0.00049212 + 0.00147637}{3} = \frac{0.00196849}{3} = 0.00065616$$

$$Avg_{LEV} = \frac{\sum_{LEV} W}{L(document)} = \frac{0.00049309 + 0.00295857 + 0.00394477}{3} = \frac{0.00739643}{3} = 0.00246547$$

$$Avg_{GLF} = \frac{\sum_{GLF} W}{L(document)} = \frac{0.00026143 + 0.00026143 + 0}{3} = \frac{0.00052286}{3}$$
$$= 0.00017428$$

$$Avg_{IRQ} = \frac{\sum_{IRQ} W}{L(document)} = \frac{0 + 0.00053304 + 0}{3} = \frac{0.00053304}{3} = 0.00017768$$

By applying the proposed model on the same unclassified example in Figure 10.2, we found that the model classified the document correctly as in Figure 10.3.



' ماشاء الله حلو كتير ' , LEV

LEV

Figure 10.3 Example of correctly classified document.

10.1.4.2 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.

$$P(doc|c) = \prod W(word, dict) \qquad (10.5)$$

We applied Equation (10.5) on the weights in Table 10.12. 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 10.12 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 (10.5) the document is classified as IRQ dialect, which is a wrong prediction.

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

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

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

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

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 (10.5) the document is correctly classified as LEV dialect.

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

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

$$\begin{aligned} P_{GLF} = \prod W(word|GLF) &= 0.00026143 \times 0.00026143 \times \frac{1}{L(dic_{GLF})} \\ &= 0.00026143 \times 0.00026143 \times \frac{1}{3472} \\ &= 0.00026143 \times 0.00026143 \times 0.00028801 \\ &= 0.0000000000196 \end{aligned}$$

$$
\begin{aligned}
P_{IRQ} = \prod W(word|IRQ) &= \frac{1}{L(dic_{IRQ})} \times 0.00053304 \times \frac{1}{L(dic_{IRQ})} \\
&= \frac{1}{1889} \times 0.00053304 \times \frac{1}{1889} \\
&= 0.00005293 \times 0.00053304 \times 0.00005293 \\
&= 0.0000000000149
\end{aligned}
$$

$$
\begin{aligned}
P_{NOR} = \prod W(word|NOR) &= \frac{1}{L(dic_{NOR})} \times \frac{1}{L(dic_{NOR})} \times \frac{1}{L(dic_{NOR})} \\
&= \frac{1}{1436} \times \frac{1}{1436} \times \frac{1}{1436} \\
&= 0.00069637 \times 0.00069637 \times 0.00069637 \\
&= 0.0000000003376
\end{aligned}
$$

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.

10.1.4.3 Result of Frequent Terms Method

According to Section 10.1.4, 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.

To compare between the frequent term method and the weighted voting method, the same experiments were conducted with the same corpora, but dictionaries were used in the frequent term method consist of the word’s frequency.

The dataset used in the first experiment was the documents classified using the annotation tool as mentioned in Section 10.1.1. The second experiment used the Arabic Multi Dialect Written Corpora (AMDWC) (Almeman and Lee 2013), the third experiment used Arabic Online Commentary Dataset (AOCD) (Zaidan and Callison-Burch 2011), and the fourth experiment used Arabic Dialects Dataset (ADD) (El-Haj *et al.* 2018). Finally, the fifth experiment combined all dictionaries from these corpora with the dictionaries created from SMADC.

The MSA word list starts with StopWords1 as described in Section 10.1.1, then the MSA list is increased using the MSA documents in AOCD and ADD to consist of 178979 words.

10.1.4.4 Results of Frequent Terms using Social Media Arabic Dialect Corpus (SMADC)

In this experiment, the model is based on using the texts collected by Alshutayri and Atwell (2017), Alshutayri and Atwell (2018b), and Alshutayri and Atwell (2018c) to create five dictionaries to cover EGY dialect, GLF dialect, LEV dialect, IRQ dialect, and NOR dialect.

The model was tested using the test dataset described in Section 10.1.1. Based on the average method, the model achieved 88% accuracy using the MSA StopWords1 list, however, a low level of accuracy was noticed is 58.53% when the model used combination of StopWords1, ADD MSA documents, and AOCD MSA documents. Moreover, using the multiply method achieves low accuracy due to replacing zero with one when the word does not exist in the dictionary, as described in Section 10.1.4.2. Table 10.13 reports the different accuracies achieved using SMADC based on using one to represent when the word is not found in the dictionary.

The first column in Table 10.13 shows the list of MSA stop words used to delete MSA words from each document before classifying documents based on dictionaries. The second column overhead represents the name of the corpus used to create the dictionaries, and the second and third columns below represent the methods used to classify documents.

Table 10.13 The result of using frequent terms method based on SMADC (one instead of zero).

MSA	SMADC	
	WMM	**WAM**
StopWords1	17.59% 44/250	**88.0%** 220/250
StopWords1 and Lancaster MSA Documents	21.2% 53/250	83.2% 208/250
StopWords1, Lancaster MSA Documents, and AOCD Documents	**46.34%** 114/246	58.53% 144/246
Without deleting MSA Words	6.0% 15/250	64.0% 160/250

Table 10.14 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.

Table 10.14 The result of using frequent terms method based on SMADC (one/number of words in the dictionary instead of zero).

MSA	SMADC	
	WMM	**WAM**
StopWords1	**55.60%** 139/250	**88.0%** 220/250
StopWords1 and ADD MSA Documents	48.4% 121/250	83.2% 208/250
StopWords1, ADD MSA Documents, and AOCD MSA Documents	33.33% 82/246	58.53% 144/246
Without deleting MSA Words	43.6 109/250	64.0% 160/250

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.

10.1.4.5 Results of Frequent Terms using Arabic Multi Dialect Written Corpora (AMDWC)

In this experiment, the model used the texts collected by Almeman and Lee (2013) to create the dialect dictionaries. Four dictionaries were created to cover EGY dialect, GLF dialect, LEV dialect, and NOR dialect, and the IRQ dictionary created from SMADC was used in this experiment, to make the experiment cover all five Arabic dialects. The dictionaries consist of the words found in the text and their frequency (the number of times the word appears in the texts).

The model was tested using the same testing dataset used in the previous section and the accuracy achieved is 76.42% using the average method and MSA StopWords1 list, ADD corpus, and AOCD corpus. Furthermore, the model tested without deleting MSA words to present the effect of deleting MSA words on the accuracy of classification gave a low accuracy, equal to 30% using the average method. Table 10.15 shows the different accuracies achieved using the AMDWC.

Table 10.15 The result of using frequent terms method based on the AMDWC.

MSA	**AMDWC**	
	WMM	**WAM**
StopWords1	22.0% 55/250	72.8% 182/250
StopWords1 and ADD MSA Documents	21.2% 53/250	73.6% 184/250
StopWords1, ADD MSA Documents, and AOCD MSA Documents	20.32% 50/246	**76.42%** 188/246
Without deleting MSA Words	**35.19%** 88/250	30.0% 75/250

10.1.4.6 Results of Frequent Terms using Arabic Online Commentary Dataset (AOCD)

The model in this experiment is based on using the texts collected by Zaidan and Callison-Burch (2011) to create the dialect dictionaries. The AOCD corpus covers three Arabic dialects: EGY, GLF, and LEV, in addition to MSA documents which were used to extend the list of StopWords1. So, three dictionaries were created to cover EGY dialect, GLF dialect, and LEV dialect. Each dictionary consists of the words found in the text and their frequency. Since this does not include IRQ and NOR dialects, the dictionaries created from SMADC were used as IRQ and NOR dictionaries, to make the experiment cover all five Arabic dialects.

The model achieved an accuracy equal to 81.2% using the average method and MSA StopWords1 list. A low level of accuracy was noticed when the model was tested without deleting MSA words which gave an accuracy equal to 26.40% used the multiply method. Table 10.16 shows the different accuracies achieved using the AOCD corpora.

Table 10.16 The result of using frequent terms method based on AOCD.

MSA	**AOCD**	
	WMM	**WAM**
StopWords1	**31.6%** 79/250	**81.2%** 203/250
StopWords1 and ADD MSA Documents	29.59% 74/250	80.80% 202/250
StopWords1, ADD MSA Documents, and AOCD Documents	26.42% 65/246	72.35% 178/246
Without deleting MSA Words	26.40% 66/250	45.6% 114/250

10.1.4.7 Results of Frequent Terms using Arabic Dialects Dataset (ADD)

In this experiment, the dialect dictionaries were created using the texts collected by El-Haj *et al.* (2018). The corpus covers four Arabic dialects: EGY, GLF, LEV, and NOR, in addition to MSA which was used to extend the list of StopWords1. Therefore, four dictionaries were created to cover EGY dialect, GLF dialect, LEV dialect, and NOR dialect. Each dictionary consists of the words found in the text and their frequency. Since this does not include IRQ, dialect, the dictionary created from SMADC was used to as the IRQ dictionary, so that the experiment covers all five Arabic dialects.

The model achieved an accuracy of around 65.2% using the average method and MSA StopWords1 list. The low accuracy 20% used the multiply method without deleting MSA words. Table 10.17 shows the different accuracies achieved using ADD.

Table 10.17 The result of using frequent terms method based on ADD.

MSA	ADD	
	WMM	**WAM**
StopWords1	**22.40%** 56/250	**65.2%** 163/250
StopWords1 and ADD MSA Documents	20.8% 52/250	57.59% 144/250
StopWords1, ADD MSA Documents, and AOCD MSA Documents	20.32% 50/246	56.50% 139/246
Without deleting MSA Words	20.0% 50/250	39.2% 98/250

10.1.4.8 Results of Frequent Terms using Combination of Different Arabic Corpora

In this experiment, the dialect dictionaries are combinations of all of the dictionaries from other corpora with the dictionaries created from SMADC. So, five dictionaries were created to cover EGY dialect, GLF dialect, LEV dialect, IRQ dialect, and NOR dialect. Each dictionary consists of the words found in the all dictionaries and the total of their frequencies. The number of words in each dialect shown in Table 10.18.

The accuracy of this model is 71.95% using the average method and MSA StopWords1 list, ADD corpus, and AOCD corpus. The low accuracy 20.32% used the multiply method and StopWords1, ADD MSA words, and AOCD MSA words. Table 10.19 shows the different accuracies achieved using combination of Arabic dialect corpora.

Table 10.18 Number of words in each dictionary created using all Arabic dialects corpora.

Dialect	Number of words
GLF	966001
EGY	812113
LEV	796213
IRQ	1889
NOR	740745

Table 10.19 The result of using frequent terms method based on combination of Arabic Dialects corpora.

MSA	Combination of Arabic Dialect Corpora	
	WMM	**WAM**
StopWords1	20.8% 52/250	71.2% 178/250
StopWords1 and ADD MSA Documents	20.4% 51/250	70.8% 177/250
StopWords1, ADD MSA Documents, and AOCD Documents	20.32% 50/246	**71.95%** 177/246
Without deleting MSA Words	**30.0%** 75/250	34.4% 86/250

Table 10.20 shows a summary of all accuracies achieved using different Arabic dialect corpora with different MSA word lists based on the Weight Average Method (WAM) which obtained better results than the Weight Multiplied Method (WMM). The first column in Table 10.20 shows the list of MSA stop words used to delete MSA words from each document. The columns from 2 to 6 represent the accuracies scored based on different Arabic dialect corpora.

The highest accuracy is 88% based on the dictionaries created using SMADC with the StopWords1, then 81.2% using AOCD corpus with StopWords1, 76.42% based on AMDWC corpus with the StopWords1, ADD MSA list, and AOCD MSA list. The dictionaries created using combination of all Arabic dialect corpora scored 71.95%, and the lowest accuracy 65.2% is based on ADD corpus.

Figure 10.4 presents a graph that compares all results achieved using different Arabic dialect corpora based on Weight Average Method and Weighted Voting Method.

Table 10.20 Summary of results achieved using Weight Average Method with different Arabic Dialect corpora.

MSA	SMADC	AMDWC	AOCD	ADD	Combined of ADC
StopWords1	**88.0%**	72.8%	**81.2%**	**65.2%**	71.2%
StopWords1 and LMSA	83.2%	73.6%	80.80%	57.59%	70.8%
StopWords1, LMSA, and AOCD MSA	58.53%	**76.42%**	72.35%	56.50%	**71.95%**
Without deleting MSA words	64.0%	30.0%	45.6%	39.2%	34.4%

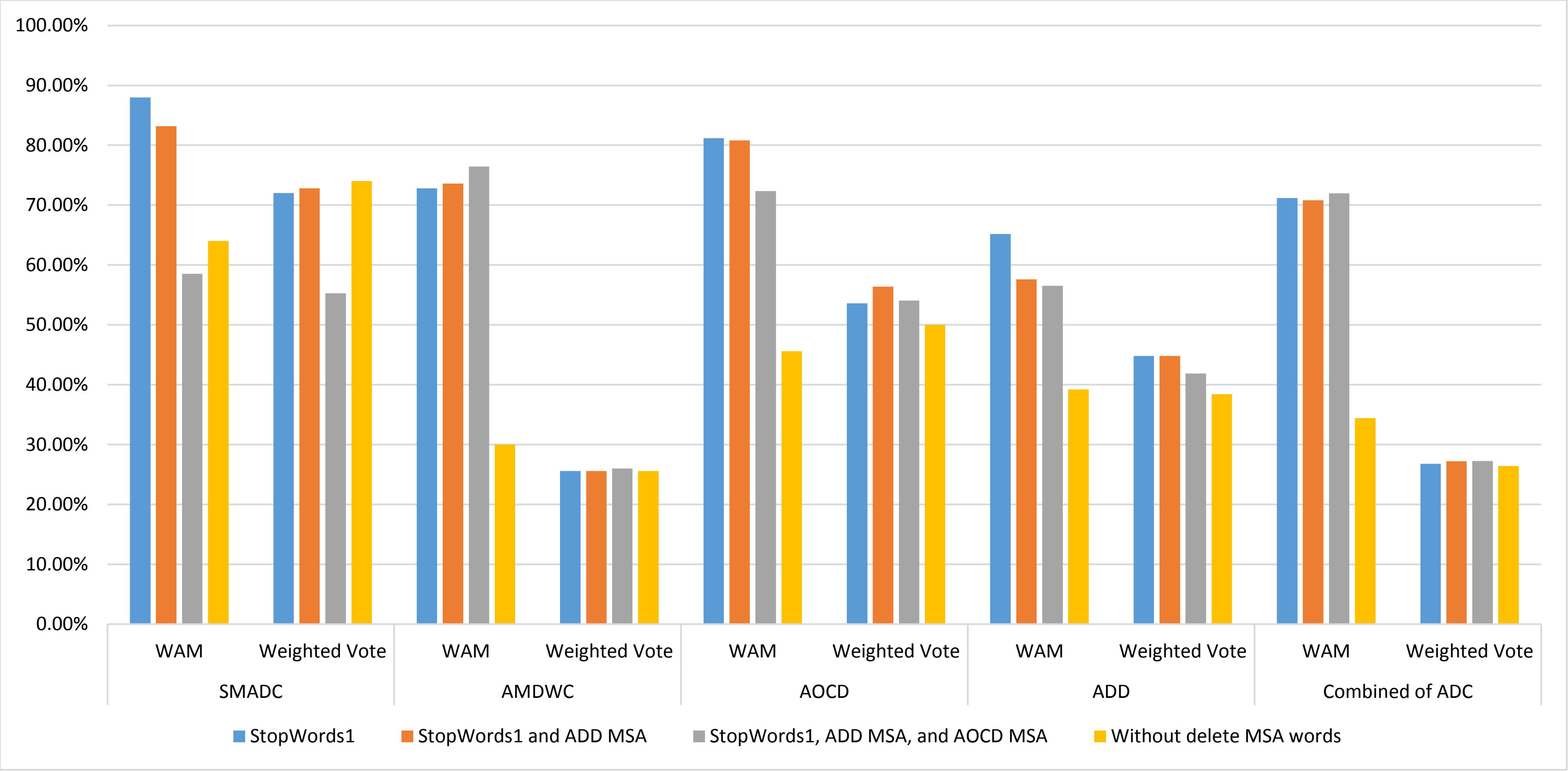


Figure 10.4 The achieved results using Weight Average Method (WAM) and Weighted Voting Method with different Arabic dialect corpora.

10.2 Methods Result and Discussion

This section analyses the results achieved so far, using a lexicon based method with different corpora used to create dictionaries. The purpose of this analysis is to improve the accuracy of classification by exploring the causes of low accuracy and fixing this, if possible .

According to all the previous experiments, deleting MSA words from the testing documents increased the accuracy of classification because all documents consist of dialectal words in addition to MSA words, such as prepositions and proper nouns. These MSA words are used in all Arabic dialects and can be considered as noise which must be deleted from each document before classifying it to the appropriate dialect.

For the lexicon based method, in the first experiment, the StopWords1 list was used and scored 88%; then we proposed to increase the size of the MSA list to cover new MSA words and delete all of the noise from each document before the classification process. In the second experiment, a new MSA list generated from MSA documents in the ADD dataset was added to MSA StopWord1. However, the accuracy 83.2% was lower than the accuracy achieved using StopWords1. In the third experiment, the MSA documents in AOCD were used to create a new MSA words list and to add this new list to the previous MSA lists with the intention of covering new MSA words not covered in the previous list. The accuracy achieved in the third experiment was 58.53% which was lower than the previously achieved accuracy.

By examining the MSA documents in the ADD dataset and AOCD, some mislabelled documents were uncovered. These documents contain dialectal words in addition to MSA words but are labelled as MSA documents. This mislabelling affected the accuracy of classification because the new MSA list created from these documents contains dialectal terms and the step of deleting MSA words from each document before the classification process deleted some dialectal words from the testing documents as they were considered as noise according to the new MSA list. Figure 10.5 shows examples of documents labelled as MSA while they contain dialectal words.

ومش قادر لغاية الان ابدله كل مالها الحياه **لورا** والوضع بضيق على المواطن من وين يجيب المواطن **مصاري** MSA,

And I am still unable to replace it as life is getting worse and citizens are finding it harder to make ends meet; from where would a citizen get money?

ومنذ البداية مع الثورة وتاريخيا **إحنا** ضد كل الممارسات الخاطئة MSA,

And, from the beginning, we were with the revolution but we are against all the wrong practices.

Figure 10.5 Example of MSA mislabelled document in ADD and AOCD.

In the previous experiments, we assumed that if we increase the size of the dictionary and enrich it with new words that will increase the accuracy of classification. However, in Section 10.1.4 by comparing the results achieved in the first experiment using SMADC with the results achieved in the last experiment using combination of all corpora, we noticed that the accuracy decreased by increasing the size of dictionary, due to the noise found in these corpora such as the mislabelling of some documents which affected the quality of the dictionaries extracted.

Another problem is due to mislabelled dialect documents. Figure 10.6 shows examples of mislabelled documents found through an examination of the dialect corpora. The first document labelled as LEV dialect contains an Egyptian term إزاي /ai:za:j/, which means how. The second document is labelled as EGY dialect while the structure of the document is GLF dialect and contains the Gulf dialect terms وش /wʃ/, which means what, and سالفه /sa:lfh/ which means story. The third document is labelled as NOR but it is written using Levantine structure and the terms: بدى /bdi:/ which means I want, اجلي /adʒli:/ which means I want to wash, and الجلي /aldʒli:/ which means utensils. The fourth document is labelled as LEV but it written using Gulf terms: مب /mb/ which means not, and الحين /alħi:n/, which means now.

متربي على أسس التنمية متربي على أسس تطوير المجتمع **إزاي** سترفع على التنويه لذا يستغرب
فعمل اخترع صحيح LEV,

Trained on the basis of development, of community development; how will the mention be raised? he will be surprised; he really worked and invented.

واثق من نفسه ومن فريقه الرجال انت **وش** دخلك **بسالفه** وفايزين عليكم يا برشلونه EGY,

The man is confident of himself and his team, so, it is not your business, and we are winning against Barcelona.

احنا عنا اعملو شاي اشربو الشاي يلا خلصو **بدى** **اجلى** الكاسات و اخلص من **الجلى** NOR,

In our situation, “Make tea, Drink tea, finish it quickly because I want to wash the cups and finish washing the utensils.

وانا **الحين** البسه من شومارت يعني **مب** طبي وما شفت شي والاشتكى من شي LEV,

And, now, I am let him wearing shoes from Shoemart which means they are not medical and it did not hurt him, and nothing happened to him.

Figure 10.6 Examples of dialect mislabelled documents in Arabic dialect corpora.

10.3 Enhancing the Frequency Method by Cleaning the Dictionaries

This section presents the steps followed to improve the accuracies by fixing the problem shown in the previous section. In order to solve the problem of mislabelled MSA documents in other corpora used to create lists of MSA words, we did the following:

1. Extract the unclassified and the misclassified documents from the testing set.

2. Test these documents again using WAM based on the dictionaries created using SMADC, to extract the list of words deleted from each document based on the list of MSA words created from StopWords1, ADD MSA, and AOCD MSA.
3. Revise the deleted words collected from the previous step and check whether it contains dialectal words in order to delete these words from the MSA words list.
4. Use the lists of dialectal words to delete all dialectal words from the MSA word list and create a new cleaned MSA words list.

The cleaned MSA word list contains 148,501 words after deleting all dialectal words and also duplicated words. After following the above steps the accuracy improved to 90% using SMADC. Table 10.21 and Table 10.22 show the accuracy using frequent terms method and voting method after cleaning the MSA words list. The first column in Table 10.21 shows the list of Arabic dialect corpora. The second column overhead represents the cleaned list of MSA words to clean documents before classification. The second and third columns below represent the methods used to classify documents.

Table 10.21 Improved results after deleting dialectal words from MSA words list (Frequent Terms Method).

Corpus	Cleaned MSA List	
	WMM	WAM
SMADC	**64.4%** 161/250	**90.0%** 225/250
AMDWC	26.40% 66/250	70.0% 175/250
AOCD	38.80% 97/250	79.2% 198/250
ADD	24.8% 62/250	64.8% 162/250

Table 10.22 The improved results after deleting dialectal words from MSA words list (Voting method).

Corpus	Cleaned MSA list	
	Simple Vote	**Weighted Vote**
SMADC	**76.0%** 190/250	**77.60%** 194/250
AMDWC	24.0% 60/250	25.6% 64/250
AOCD	52.40% 131/246	57.99% 145/246
ADD	47.59% 119/250	50.0% 125/250

To improve the accuracy in the last experiment using a combination of all corpora to create the dictionaries, the following steps were implemented:

1. Delete all MSA words from each dictionary using the cleaned MSA word list.
2. Analyse the misclassified document to check each word in the document and decide which dictionary each word must belong to, based on the seed words used to collect tweets and the frequencies of words in our dictionaries in addition to our knowledge of Arabic dialect.
3. According to the previous step some words were deleted from some dictionaries or moved to other dictionaries.

The model was tested again after cleaning the combined Arabic corpora dictionaries and the best accuracy is 82.39% using the average method and StopWords1 as other MSA word lists still contain dialectal words due to the mislabelled MSA documents. Table 10.23 shows the number of words in each dictionary after cleaning process. Table 10.24 and Table 10.25 show the accuracies achieved using frequent terms method and voting method after cleaning the combined dictionary of all Arabic dialect corpora with different stop word lists.

Table 10.23 Number of words in each dictionary created using a combination of Arabic dialects corpora (after cleaned dictionaries).

Dialect	**Number of words**
GLF	867818
EGY	699256
LEV	699451
IRQ	607
NOR	647680

Table 10.24 The improved results after deleting MSA words from the combined dictionary (Frequent Terms Method).

MSA	**Cleaned of combined Arabic Dialect Corpora**	
	WMM	**WAM**
StopWords1	20.0% 50/250	**82.39%** 206/250
StopWords1 and ADD MSA Documents	20.0% 50/250	76.4% 191/250
StopWords1, ADD MSA Documents, and AOCD Documents	**20.32%** 50/246	71.54% 176/246
Without deleting MSA Words	16.40% 41/250	72.39% 181/250

Table 10.25 The result of using cleaned combination of Arabic Dialects corpora (Voting method).

MSA	**Cleaned of combined Arabic Dialect Corpora**	
	Simple Vote	**Weighted Vote**
StopWords1	21.6% 54/250	26.40% 66/250
StopWords1 and ADD MSA Documents	21.6% 54/250	26.40% 66/250
StopWords1, ADD MSA Documents, and AOCD MSA Documents	**26.42%** 65/246	**27.23%** 67/246
Without delete MSA Words	21.6% 54/250	26.40% 66/250

Tables 10.26 and 10.27 summarise the accuracies after analysing the results in Section 10.1.3 and 10.1.4 to improve the accuracy of classification. According to Table 10.28 the best accuracy using cleaned combination Arabic dialect corpora is 82.39% using StopWords1 based on weighted average method. The weighted voting method show low accuracies in comparison to weighted average method with accuracies ranging between 26.40%-27.23%.

Table 10.27 shows the accuracies of classification after cleaning the MSA words list from dialectal terms and testing the dataset based on different dictionaries. The best accuracy is 90% using weighted average method and based on SMADC followed by 79.2% based on AOCD corpus. The results using the weighted voting method are 77.60% based on dictionaries created using SMADC and 57.99% based on AOCD dictionaries.

Table 10.26 Summary of results achieved using the cleaned combination of Arabic Dialects Corpora (ADC).

MSA	Cleaned Combination of ADC	
	Weighted Average	Weighted Vote
StopWords1	**82.39%**	26.40%
StopWords1 and ADD MSA	76.4%	26.40%
StopWords1, ADD MSA, and AOCD MSA	71.54%	**27.23%**
Without delete MSA words	72.39%	26.40%

Table 10.27 Summary of improved results after deleting dialectal words from MSA words list.

Corpus	Cleaned MSA list	
	Weighted Average	Weighted Vote
SMADC	**90.0%**	**77.60%**
AMDWC	70.0%	25.6%
AOCD	79.2%	57.99%
LADD	64.8%	50.0%

In the previous sections, the SMADC data set which were used to create the dictionaries was a small set of the annotated documents that resulted from the annotation tool (see Chapter 6) as described in Section 10.1.1. The total number of documents is 12130, divided between 4507 GLF documents, 1620 NOR documents, 2533 EGY documents, 2002 LEV documents, and 1468 IRQ documents. The total number of tokens in all documents is 486,147. Table 10.28 shows the number of types in each dictionary. Tables 10.29 and 10.30 show the achieved accuracy of classification using all annotated documents in SMADC based in frequent terms methods and voting methods.

Table 10.28 Number of words in each dictionary created using all annotated documents in SMADC.

Dialect	Number of words
GLF	20252
EGY	11868
LEV	11631
IRQ	9732
NOR	11725

Table 10.29 The result of using all annotated documents (Frequent Terms).

MSA	SMADC	
	WMM	**WAM**
StopWords1	**74.0%** 185/250	80.0% 200/250
StopWords1 and ADD MSA Documents	69.19% 173/250	**85.2%** 213/250
StopWords1, ADD MSA Documents, and AOCD Documents	56.50% 139/246	70.73% 174/246
Without delete MSA Words	65.2% 163/250	82.8% 207/250

Table 10.30 The result of using all annotated documents (Voting method).

MSA	SMADC	
	Simple Vote	**Weighted Vote**
StopWords1	68.8% 172/250	72.39% 181/250
StopWords1 and ADD MSA Documents	67.2% 168/250	71.2% 178/250
StopWords1, ADD MSA Documents, and AOCD MSA Documents	55.69% 137/246	58.13% 143/246
Without delete MSA Words	**69.6 %** 174/250	**73.6%** 184/250

10.4 Machine Learning Method

As mentioned in Chapter 2, there are some popular machine learning algorithms (classifiers) used in text classification including Naive Bayes (NB), Decision Tree (DT), K-Nearest Neighbour (KNN), Logistic Regression (LR), and Support Vector Classifier (SVM) (Wang *et al.* 2018).

To decide which is the best classifier of Arabic dialect texts, in Chapters 3, 8, and 9, some classifiers were examined to classify Arabic dialect texts and the best results were found when using SMO and MNB classifiers.

Figure 10.7 shows the architecture of the proposed classification model using a machine learning algorithm. At the beginning, the dataset is divided into two sets. The training set consisting of 80% of the labelled documents was used to train the classifier, and a testing set consisting of 20% of the labelled documents was used to evaluate the classifier’s performance. The next step is feature extraction to create a feature vector. Then, a machine learning algorithm was chosen to train the model and build a classifier. The architecture will be discussed in detail in the following sections.

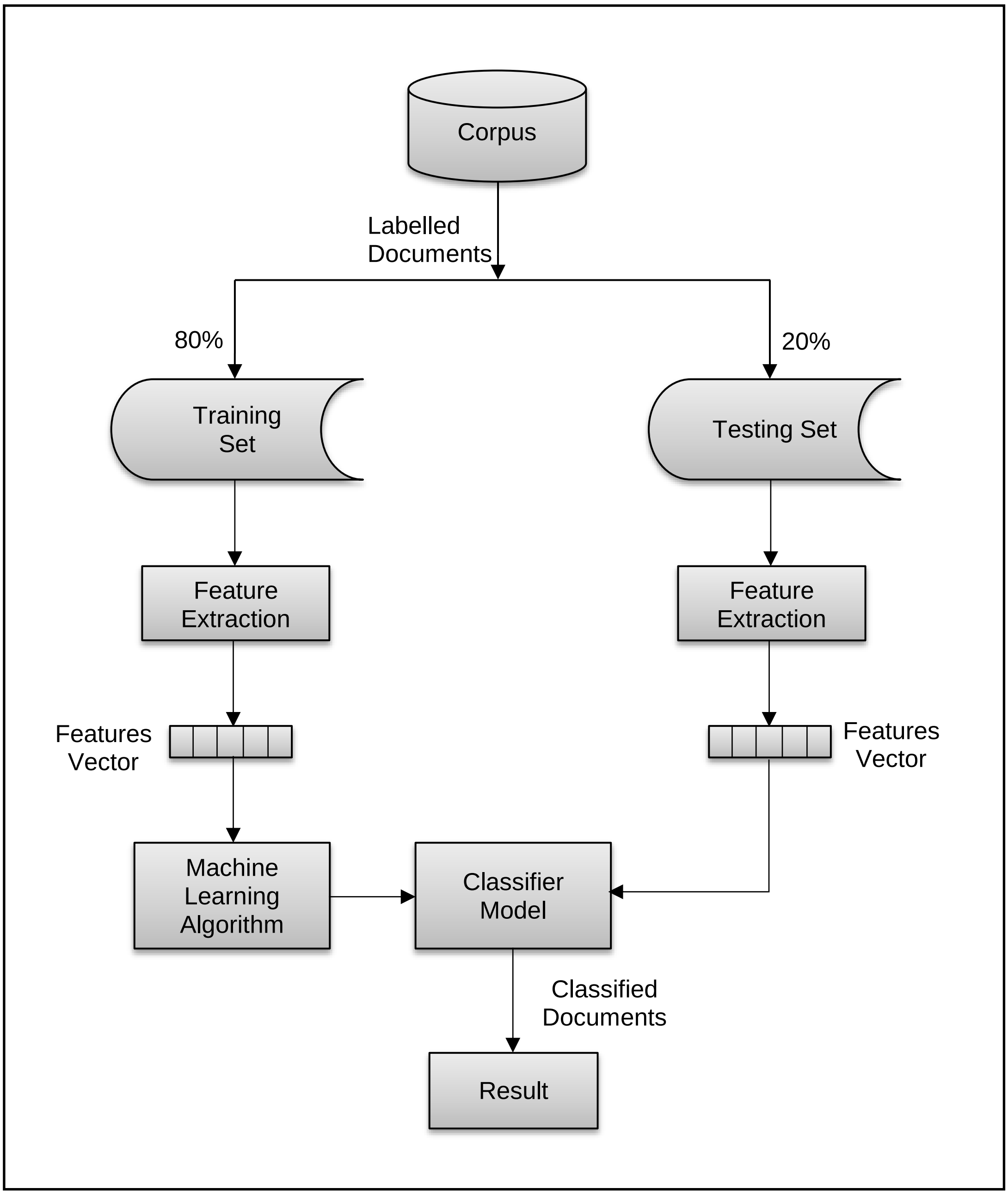


Figure 10.7 The architecture of classification process using machine learning.

10.4.1 Feature Extraction

In this step, the set of documents will be transformed into feature vectors by extracting the characteristics of each document. The features used to describe each document are: N-gram, and TF-IDF. These features were selected based on the experiments in Chapters 3, 8, and 9.

10.4.1.1 N-gram Features

According to Cavnar and Trenkle (1994), Muntsa and Llu´ıs (2004), and Mahedero *et al.* (2005), an N-gram based approach in language text classification achieves best accuracy ranging from 90% to 99%. An N-gram is a continuous sequence of character segment of a given text (Cavnar and Trenkle 1994; Sababa 2018). The size of n-gram could vary: 1-gram or unigram; 2-gram or bigram; 3-gram or trigrams and size four and five and so on. The following examples show the difference between character gram and word gram.

For example, character n-grams of the word “Text” could be:

unigram: T, e, x, t

bigrams: _T, Te, ex, xt, t_

trigrams: _Te, Tex, ext, xt_

Word, n-gram of the sentence “This is a text” could be:

unigram: This, is, a, text

bigrams: This is, is a, a text

trigrams: This is a, is a text

In this research, the N-gram features are characters and words as in the experiment in (Alshutayri *et al.* 2016). According to Section 2.4, there are lexical, orthographical, and phonological variations between Arabic dialects which can be used as features to describe each dialect. Therefore, the word unigram and bigram are used to extract word-based features from the text to cover lexical variations between dialects. Furthermore, character unigram and bigram are used to cover the morphological variations between dialects by extracting the prefix and suffix of words; as mentioned in Section 2.4.2 some dialects could be distinguished from each other by looking at the prefixes and suffixes which are added to the verbs to express time.

The result of this step is a matrix of feature vectors consisting of rows corresponding to the documents and columns corresponding to the feature counts for each feature in that document.

10.4.1.2 Term Frequency (TF) and Inverse Document Frequency (IDF)

TF-IDF is a numerical statistic used as a function in text classification (Joachims 1997; Abu-Errub 2014; Yun-tao *et al.* 2005) to calculate the weight of a word to represent the importance of a word in a document in a dataset. Term Frequency (TF) is the frequency of a term in a document and is calculated as the number of occurrences of the term in a document divided by the total number of terms in the document (Roul *et al.* 2014). Equation (10.6) is used to calculate the TF where t is the term and d is the document and t' is all other terms in document.

$$TF_{t,d} = \frac{f_{t,d}}{\sum_{t' \in d} f_{t',d}} \qquad (10.6)$$

Inverse Document Frequency (IDF) is used to increase the weight of terms that occur rarely in the dataset. The IDF of a term t is calculated by taking the logarithm of the total number of documents in the dataset divided by the number of documents containing the term (Gerard and Christopher 1988). Equation (10.7) is used to calculate the IDF where N is the total number of documents, and DF is number of documents contain term t (Roul *et al.* 2014).

$$IDF = \log(\frac{N}{DF}) \qquad (10.7)$$

TF-IDF is a composite weight for each term produced by combined TF and IDF as in Equation (10.8).

$$TF - IDF = TF * IDF \qquad (10.8)$$

In this research, the TF-IDF feature was used with the N-gram word feature as in (Alshutayri *et al.* 2016) to give a high weight to the important words in the document because some high-frequency words have low content discriminating power and are found in all documents.

10.4.2 Machine Learning Algorithms

10.4.2.1 Sequential Minimal Optimization (SMO)

A specific version of Support Vector Machine (SVM) is implemented as SMO in WEKA, SMO is an efficient version of SVM which based on finding the optimal separating hyper-plane between classes by analysing the training set to detect the critical boundary instances called support vectors for each class and creating a discriminant function which splits them as widely as possible (Witten and Frank 2005). SVM is a linear classifier and most text categorization problems are linearly separable (Joachims 1998).

10.4.2.2 Multinomial Naïve Bayes (MNB)

Multinomial Naïve Bayes is appropriate for text classification using a word frequencies technique. MNB performs better than Naïve Bayes (NB) because NB is based on creating a bag of words for each document while MNB adds the word frequencies to the bag of words by counting the number of times that every word occurs in the document (Witten and Frank 2005). Multinomial Naïve Bayes works well in text classification based on the independency between features, assuming that every feature is independent of the others (Wang *et al.* 2018; Huang 2017).

Using Equation (10.9) to calculate the probability for each class in the training set, N_c is the number of documents in class c; N is the total number of documents (Jurafsky 2011).

$$P(c) = \frac{N_c}{N} \qquad (10.9)$$

The next step is calculating the conditional probabilities for each word in the tested document using Equation (10.10), where $count(w,c)$, is the frequency of the word w in class c, $count(c)$ is the count of words in class c, and $|V|$ is the number of types in all classes (Jurafsky 2011).

$$P(w|c) = \frac{count(w,c)+1}{count(c)+|V|} \qquad (10.10)$$

Finally to choose the correct class; Equation (10.11) was used to multiply the probability resulting from Equation (10.10) for each word in the tested document by the priors probability for each class resulted from Equation (10.9) (Jurafsky 2011).

$$P(d|c) = P(c) \prod P(w|c) \qquad (10.11)$$

10.4.3 Machine Leaning Results

This section presents the results of classification task. According to the experiment results achieved in Section 10.1, we decided to use SMADC to train machine learning models because SMADC scored high levels of accuracy in comparison with other Arabic dialect corpora. The experiments conducted used five training datasets described in 10.4.3.1 and, to evaluate the model, we used the testing set described in Section 10.1.1 which is the same testing set used to evaluate the models in Sections 10.1.

10.4.3.1 The Datasets used in the Machine Learning

There are five datasets used in the machine learning based model for the training process. The first dataset contains 1,383 documents (18,697 tokens) to train the model, this dataset also will be used in the lexicon methods to create the dialect dictionaries. The second dataset consists of 3,000 documents (42,820 tokens). The third dataset consists of 10,531 documents (154,260 tokens). None of the documents in these training datasets are duplicated, and, to check the effects of the duplicated documents in the training process, the fourth dataset was created with duplicated documents from all of the annotated documents which resulted from the annotation tool (see Chapter 6). The fourth dataset consists of 12,046 documents (176,879 tokens). The allCorpus dataset consists of 1,088,578 documents (13,876,504 tokens). Table 10.31 shows the number of documents in each dialect used in each experiments.

The testing dataset to test all models and evaluate the classification algorithm contains 250 documents (7,341 tokens). This is used in all of the experiments presented in this chapter.

Table 10.31 The number of documents in each dialect for training the classification model.

Dialect	First Dataset	Second Dataset	Third Dataset	Fourth Dataset	AllCorpus Dataset
GLF	353	878	3897	4405	177019
EGY	342	684	2214	2565	310698
LEV	237	534	1735	2004	193525
IRQ	240	471	1301	1472	153054
NOR	211	433	1384	1600	254282

10.4.3.2 The Results using Multinomial Naïve Bayes (MNB)

In this section, we created two models using Multinomial Naïve Bayes (MNB) with different features extracted from five different training set size of the training set to explore the effects of the training set size on the accuracy of classification. The extracted features are Bag of words using word tokenizer, NGram tokenizer to extract words ranges between one and three, and CharNGram tokenizer to extract letter rangers between one and three. The first model used TF-IDF described in Section 10.4.1.2. The second model not use TF-IDF. Table 10.32 illustrates the results using two different models trained with four differently sized datasets. The first column represents the dataset, the second column the extracted features, the third column the model based on MNB with TF-IDF, and the fourth column the model based on MNB only.

Table 10.32 The result of using MNB with differently sized training datasets and tokenizers.

Data Set	**Features**	**MNB-TFIDF**	**MNB**
First	Word	**75.2**	71.2
	NGram(1-3)	**54**	44.8
	CharNGram(1-3)	**51.6**	48.4
Second	Word	**74.8**	72.8
	NGram(1-3)	**55.6**	50.4
	CharNGram(1-3)	**56**	53.2
Third	Word	91.2	**92**
	NGram(1-3)	89.6	**90.8**
	CharNGram(1-3)	59.2	59.2
Fourth	Word	**92**	90
	NGram(1-3)	**89.2**	88.4
	CharNGram(1-3)	58.4	**60**
All Corpus	Word	**88**	87.6
	NGram(1-3)	**82.4**	81.3
	CharNGram(1-3)	**73.2**	70.4

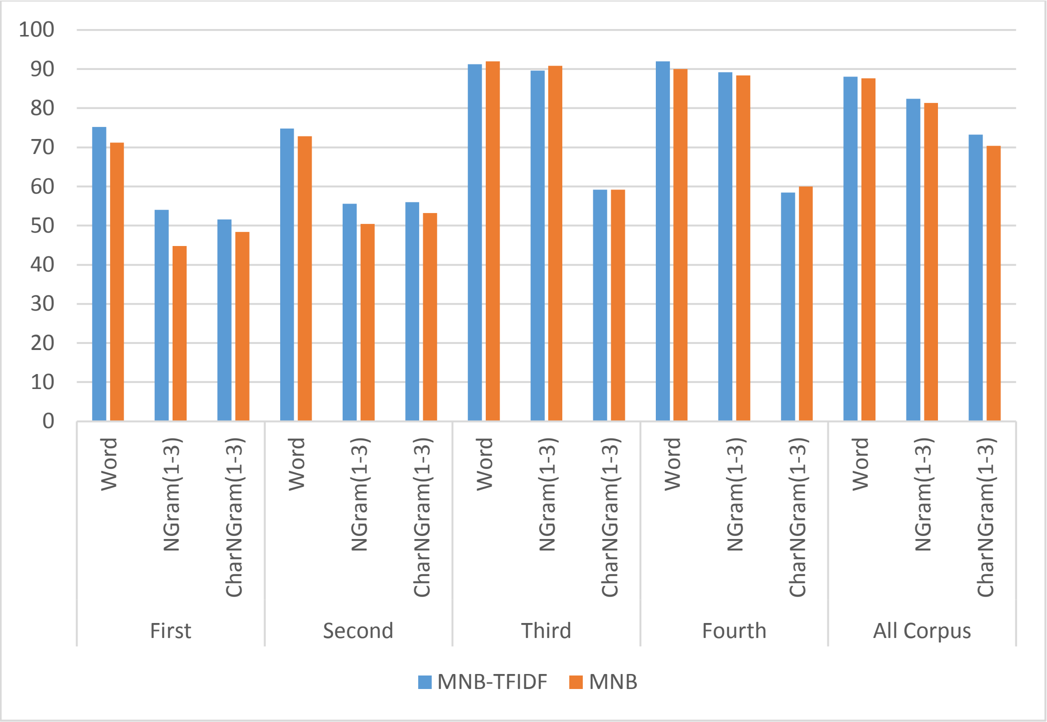


Figure 10.8 The accuracies using MNB with different training dataset sizes.

10.4.3.3 The results using Sequential Minimal Optimization (SMO)

This section uses the same features as were used in Section 10.4.3.2 with Sequential Minimal Optimization (SMO) algorithm. Table 10.33 shows the results using two different models trained with five differently sized datasets. The first column represents the dataset, the second column the extracted features, the third column the model based on SMO with TF-IDF, and the fourth column the model based on SMO only.

Table 10.33 The result of using SMO with differently sized training datasets and tokenizers.

Data Set	Tokenizer	SMO-TFIDF	SMO
First	Word	65.6	65.6
	NGram(1-3)	48.8	48.8
	CharNGram(1-3)	47.2	47.2
Second	Word	68.4	68.4
	NGram(1-3)	62.4	62.4
	CharNGram(1-3)	52.4	52.4
Third	Word	82	82
	NGram(1-3)	80.4	80.4
	CharNGram(1-3)	67.6	67.6
Fourth	Word	80.4	80.4
	NGram(1-3)	80.4	80.4
	CharNGram(1-3)	67.6	67.6
All Corpus	Word	82.3	82.3
	NGram(1-3)	80.1	80.1
	CharNGram(1-3)	73.2	73.2

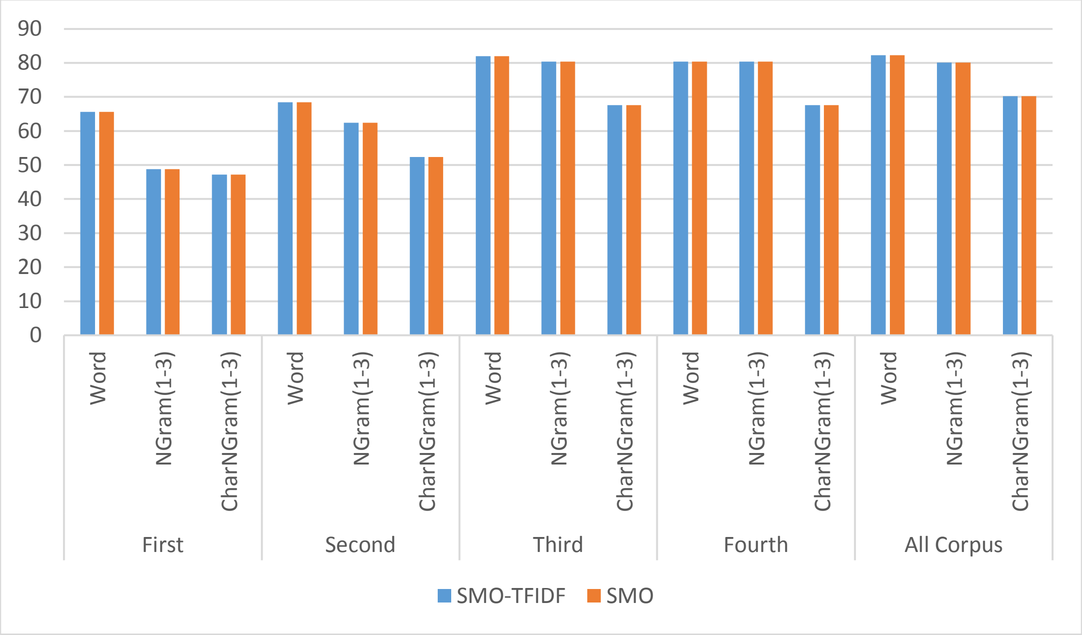


Figure 10.9 The accuracies using SMO with different size of training dataset.

10.4.3.4 The results using Naïve Bayes (NB)

The third classifier was used in machine learning methods is Naïve Bayes (NB). We created two models with different features extracted from five different training set size of the training set to explore the effects of the training set size on the accuracy of classification. This section uses the same features as were used in Section 10.4.3.2. Table 10.34 shows the results using two different models trained with four differently sized datasets. The first column represents the dataset, the second column the extracted features, the third column the model based on NB with TF-IDF, and the fourth column the model based on NB only.

Table 10.34 The result of using NB with differently sized training datasets and tokenizers.

Data Set	Tokenizer	NB-TFIDF	NB
First	Word	55.6	55.6
	NGram(1-3)	47.2	47.2
	CharNGram(1-3)	31.6	31.6
Second	Word	60.4	60.4
	NGram(1-3)	61.6	61.6
	CharNGram(1-3)	37.2	37.2
Third	Word	63.2	63.2
	NGram(1-3)	63.2	63.2
	CharNGram(1-3)	53.2	53.2
Fourth	Word	60.4	60.4
	NGram(1-3)	60.8	60.8
	CharNGram(1-3)	56	56
All Corpus	Word	48	48
	NGram(1-3)	55.4	55.4
	CharNGram(1-3)	52.7	52.7

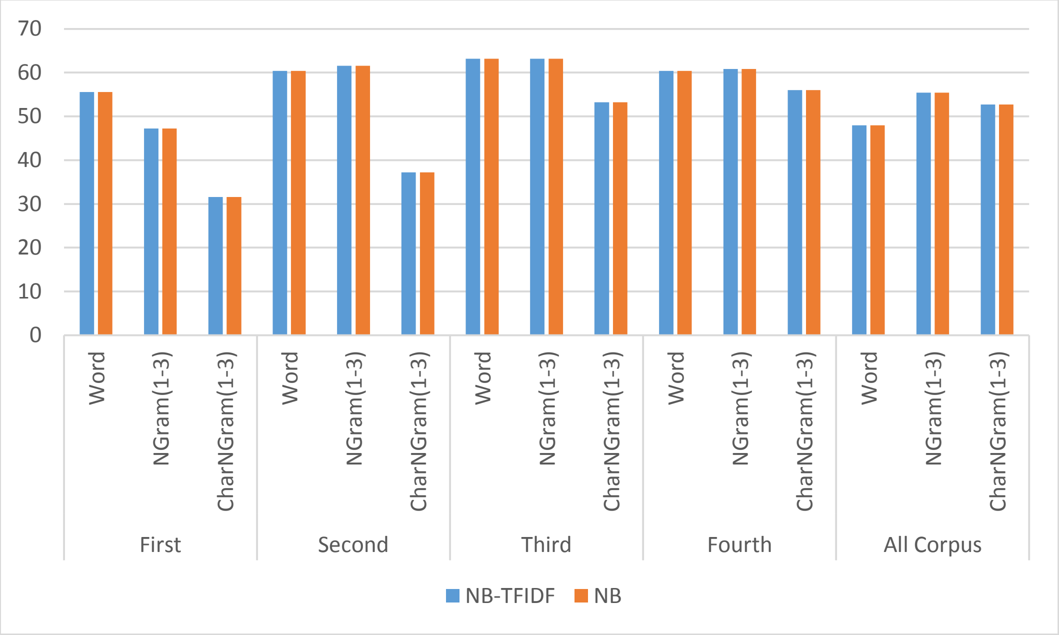


Figure 10.10 The accuracies using NB with different size of training dataset.

10.4.3.5 The Best Model of Classification

According to the results in Tables 10.32, 10.33, and 10.34 MNB provides the best accuracy in classifying Arabic dialect texts based on word as a feature. By comparing the achieved accuracies in Tables 10.32, 10.33, and 10.34, we found that the accuracies vary between classifiers based on the different features were used. The classification models created using three different classifiers (MNB-SMO-NB) and trained in word feature with TF-IDF using the first dataset scored accuracies ranging between 55.6%-75.2%, while the same models trained using the second dataset scored accuracies ranging between 60.4%-74.8%. Then, the models trained on the third data set scored accuracies ranging between 63.2%-91.2%. The models trained on the fourth dataset achieved accuracies ranging between 60.4%-92%. Finally the same models trained in all SMADC achieved accuracies ranging between 48%-88%. As described in Section 10.4.3.1, the first, second, third, and fourth datasets all are resulted from the annotation tool, while all corpus dataset is not certain annotated with the correct labels. It is also clear from the tables that, whenever the size of the training set increases, the accuracy also increases.

The same experiment was repeated using word feature without TF-IDF. The accuracies ranging between 55.6%-71.2% using the first dataset. When the models trained using the second dataset, the accuracies ranging between 60.4%-72.8%. Then, the models trained on the third dataset scored accuracies ranging between 63.2%-92%. The models trained on the fourth dataset achieved accuracies ranging between 60.4%-90%. Finally the same models trained in all SMADC achieved accuracies ranging between 48%-87.6%.

The second experiment based on using wordGram as a feature with minimum 1 word and maximum three words with TF-IDF. First, the models trained on the first dataset and scored accuracies ranging between 47.2-54%. Then, the models trained on the second dataset achieved accuracies ranging between 61.6%-62.4.6%, in this model, SMO classifiers achieves higher accuracy than MNB and NB. The third experiment based on using the third data set to train the models and the accuracies ranging between 63.2%-89.6%. The fourth experiment based on using the fourth data set to train the models and the accuracies ranging between 63.8%-89.2%. The last experiment based on using the allCorpus data set to train the models and the accuracies ranging between 55.4%-82.4%.

The experiment repeated without using TF-IDF, SMO and NB did not show any difference in the accuracy if classification using TF-IDF or without using it. The accuracies ranging between 44.8-%48.8% using the first dataset. When the models trained using the second dataset, the accuracies ranging between 61.6%-62.4%. Then, the models trained on the third data set scored accuracies ranging between 63.2%-90.8%. The models trained on the fourth dataset achieved accuracies ranging between 60.4%-884%. Finally the same models trained in all SMADC achieved accuracies ranging between 55.4%-81.3%.

The third experiment based on using CharacterGram as a feature with minimum 1 word and maximum three words with TF-IDF. First, the models trained on the first dataset and scored accuracies ranging between 31.6%-51.6%. Then, the models trained on the second dataset achieved accuracies ranging between 37.2%-56%, in this model, SMO classifiers achieves higher accuracy than MNB and NB. The third experiment based on using the third data set to train the models and the accuracies ranging between 53.2%-67.6%. The fourth experiment based on using the fourth data set to train the models and the accuracies ranging between 56%-67.6%. The last experiment based on using the allCorpus data set to train the models and the accuracies ranging between 52.7%-73.2%.

The experiment repeated without using TF-IDF. The accuracies ranging between 31.6%-48.4% using the first dataset. When the models trained using the second dataset, the accuracies ranging between 37.2%-53.2%. Then, the models trained on the third data set scored accuracies ranging between 53.2%-67.6%. The models trained on the fourth dataset achieved accuracies ranging between 56%-76.6%. Finally the same models trained in all SMADC achieved accuracies ranging between 52.7%-70.4%.

Figure 10.11 presents a graph that compares all results achieved using different classifiers and features.

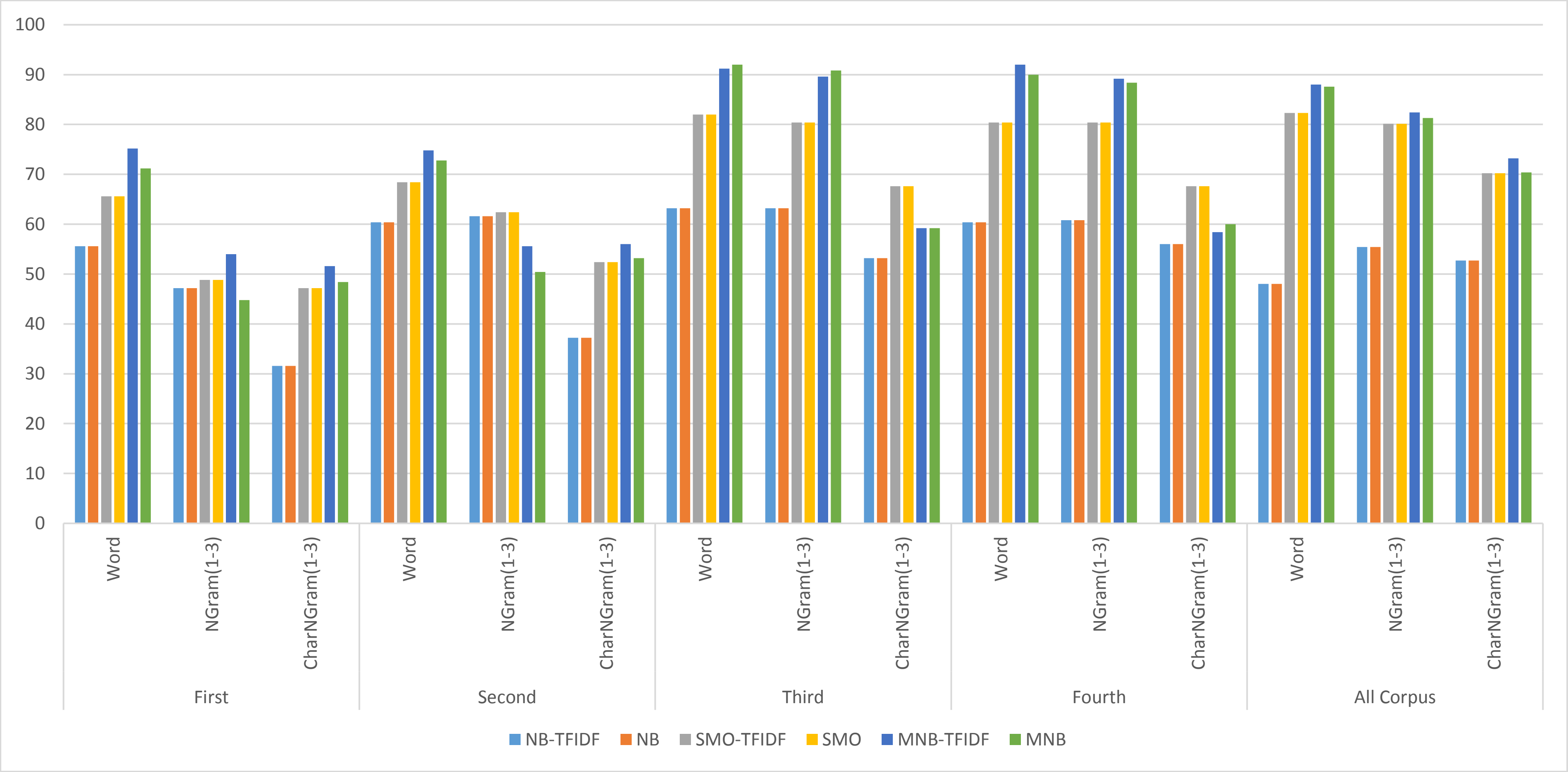


Figure 10.11 The achieved results using MNB, SMO, and NB with different features.

In the all experiments with different datasets the MNB classifier shows the good accuracies in classifying Arabic dialect texts and scoring 92% based on using word as a feature with TF-IDF. Figures 10.12 and 10.13 show the WEKA output which is the summary result of the MNB classification model and the confusion matrix to show the predicted labels and the actual labels.

```
=== Summary ===

Correctly Classified Instances         230               92      %
Incorrectly Classified Instances        20                8      %
Kappa statistic                        0.9
Mean absolute error                      0.0361
Root mean squared error                    0.1731
Relative absolute error                 11.2803 %
Root relative squared error               42.2414 %
Total Number of Instances                250

=== Detailed Accuracy By Class ===

TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
0.980    0.025     0.907    0.980     0.942     0.928      0.988     0.984     EGY
0.880    0.035     0.863    0.880     0.871     0.839      0.971     0.855     GLF
0.880    0.010     0.957    0.880     0.917     0.898      0.971     0.948     IRQ
0.940    0.025     0.904    0.940     0.922     0.902      0.993     0.973     LEV
0.920    0.005     0.979    0.920     0.948     0.937      0.981     0.966     NOR
Avg.0.920     0.020      0.922    0.920     0.920     0.901      0.981     0.945

=== Confusion Matrix ===

a  b  c  d  e   <-- classified as
49  1  0  0  0 |  a = EGY
1 44  2  3  0 |  b = GLF
0  5 44  0  1 |  c = IRQ
2  1  0 47  0 |  d = LEV
2  0  0  2 46 |  e = NOR
```

Figure 10.12 Summary result for the best model.

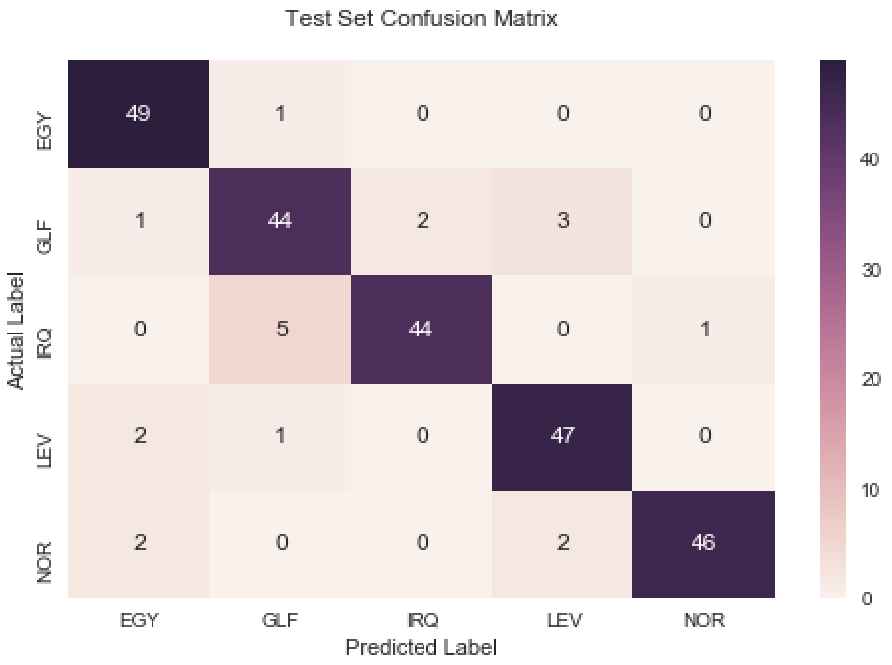


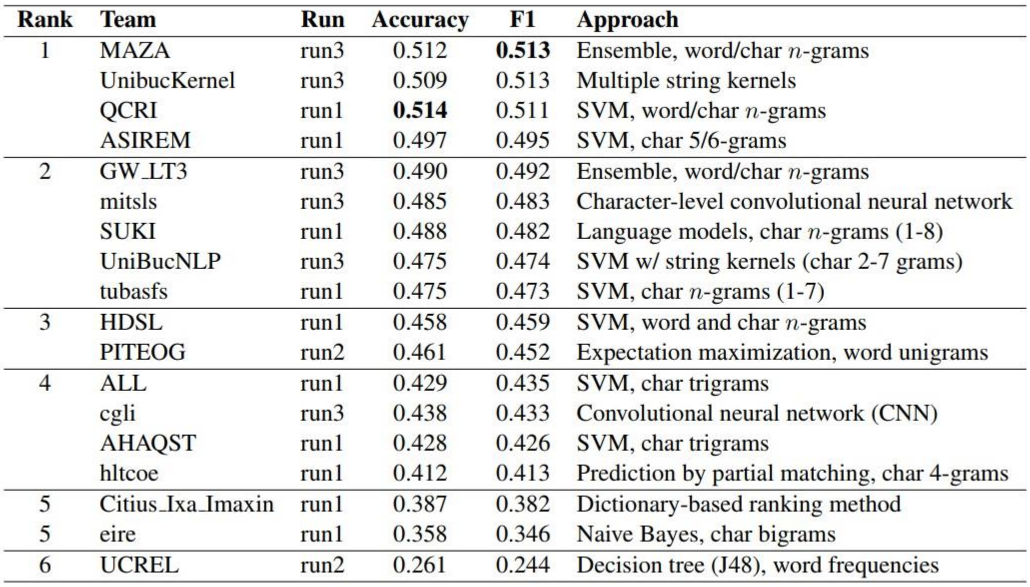
Figure 10.13 Confusion matrix for the best model.

10.4.3.6 The Achieved result in DSL2016

Chapter 8 described the first experiment conducted to classify Arabic dialect text in the VarDial2016 workshop at COLING 2016 Discriminating Similar Languages (DSL) 2016 shared task (Alshutayri *et al.* 2016). The shared task offered a task focused on Arabic dialect identification in speech transcripts (Malmasi *et al.* 2016). The Arabic dialect texts used for training and testing were developed using the QCRI Automatic Speech Recognition (ASR) QATS system (Khurana and Ali 2016) to label each document with a dialect (Ali *et al.* 2016). The number of teams participated in this task were 18 teams. Table 10.35 showed the achieved results in this task using different models. The best accuracy in the VarDial2016 shared task was 51.4%, which was achieved using an SVM classifier and character bigrams, trigrams, 4-grams and 5-grams (Eldesouki *et al.* 2016). The worst accuracy was 26.1% using Decision tree classifier and word frequencies as a feature. All other models scored accuracy between 51%- 35%. Seven models used SVM algorithm and two teams used Convolutional Neural Network (CNN). Our model used an SMO classifier with Character TriGram and scored 42.9% (Alshutayri *et al.* 2016).

In this thesis, we found that a classification model trained in word feature with TF-IDF using the MNB classifier is the best model to classify Arabic dialect text, scoring 92% by comparing the achieved accuracy in the VarDial2016 shared task with the new accuracy achieved using an MNB classifier.

Table 10.35 The result of DSL2016 shared task. Adopted from (Malmasi *et al.* 2016).



Rank	Team	Run	Accuracy	F1	Approach
1	MAZA	run3	0.512	**0.513**	Ensemble, word/char n-grams
	UnibucKernel	run3	0.509	0.513	Multiple string kernels
	QCRI	run1	**0.514**	0.511	SVM, word/char n-grams
	ASIREM	run1	0.497	0.495	SVM, char 5/6-grams
2	GW_LT3	run3	0.490	0.492	Ensemble, word/char n-grams
	mitsls	run3	0.485	0.483	Character-level convolutional neural network
	SUKI	run1	0.488	0.482	Language models, char n-grams (1-8)
	UniBucNLP	run3	0.475	0.474	SVM w/ string kernels (char 2-7 grams)
	tubasfs	run1	0.475	0.473	SVM, char n-grams (1-7)
3	HDSL	run1	0.458	0.459	SVM, word and char n-grams
	PITEOG	run2	0.461	0.452	Expectation maximization, word unigrams
4	ALL	run1	0.429	0.435	SVM, char trigrams
	cgli	run3	0.438	0.433	Convolutional neural network (CNN)
	AHAQST	run1	0.428	0.426	SVM, char trigrams
	hltcoe	run1	0.412	0.413	Prediction by partial matching, char 4-grams
5	Citius_Ixa_Imaxin	run1	0.387	0.382	Dictionary-based ranking method
5	eire	run1	0.358	0.346	Naive Bayes, char bigrams
6	UCREL	run2	0.261	0.244	Decision tree (J48), word frequencies

10.4.3.7 Initial experiment using Deep Learning models

In this thesis, we focused on classifying Arabic dialect texts using Lexicon and Machine learning methods, but recently as described in Section 2.4.2 some research started to use deep learning models for classification Arabic dialect text. So, we did last experiment using deep learning models on classification of Arabic dialectal text using the whole SMADC corpus. We used three different deep neural network models to classify Arabic dialect which are Long-Short Term Memory (LSTM), Bidirectional LSTM (BLSTM), and Convolutional LSTM (CLSTM). The models achieved different accuracies ranging between 455.73% and 64.54%, the highest accuracy scored 64.54% using BLSTM, followed by LSTM with a score of 61.47%, then CLSTM with a score of 55.73%. By comparing the achieved accuracies using deep learning models with the achieved accuracies using the machine learning model we found that machine learning scored 92%, which is better that the result scored by deep learning models in our experiment and other experiments described in Section 2.4.2 which ranging between 71.4% and 87.65%.

10.5 Conclusion

The classification of Arabic dialect text is a new 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 Al-Ameri 2015). In this chapter, we classified Arabic dialects text using two different methods: the first method is lexicon based method divided into Frequent terms methods including weight average method and weight multiplied method, and Voting based method including simple voting method and weighted voting method, and the second method is Machine Learning based method using two classifiers SMO and MNB.

The lexicon methods based on using dictionaries were created for each dialect from different Arabic dialect corpora. The classification process was used in these methods based on deleting all MSA words from the document then checking that each word in the document belongs to which dialect by searching the dialect dictionaries. The frequent terms method scored 88% using the weight average method 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. 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 machine learning using three classifiers SMO, NB and MNB based on the results in Chapter 8 which presented the first experiment on classifying Arabic dialect text and shows good accuracy using the SMO classifier, and Chapter 9 which classified three different datasets from three sources and shows that MNB can work with SMO to improve accuracy. The accuracy achieved using Machine Learning scored 92% based on using word as a feature with TF-IDF to produce a weighted vector for each word.

Part VII

Conclusion and Future Work

Chapter 11
Conclusions and Future Work

11.1 Overview

This thesis is split into seven parts with 11 chapters as shown in the following:

- **Part I** included two chapters: introduction, and literature review
  - **Chapter 1** provided background information about Arabic language and its dialects, the objectives of this research and the contributions.
  - **Chapter 2** covered the current and past work within the area of Arabic dialect corpora, classification of Arabic dialect and information about machine learning.
- **Part II** included five chapters: Exploring Twitter as a source of an Arabic dialect corpus, Creating an Arabic dialect texts corpus by exploring online newspapers, Extending the Arabic Dialects Corpus, Arabic dialect texts annotation, and the final version of corpus.
  - **Chapter 3** explored Twitter as a source of Arabic dialect texts to create written corpus of Arabic dialects. It described the method was used to extract tweets based on the seed words that are spoken in one dialect and not in the other dialects. In addition, to the user location to enhance dialect classification and specify the country and dialect to which each tweet belongs.
  - **Chapter 4** explored an online comments in electronic Arabic newspaper as a another source of Arabic dialect texts to create a corpus of dialectal Arabic by extracting comments from the famous electronic newspaper in each country in the Arab world.
  - **Chapter 5** extended the Arabic dialect texts corpus by collecting new tweets based on spatial coordinate points for each city in different countries in the Arab world. In addition to scrape Facebook posts and extracted all comments from these posts.
  - **Chapter 6** introduces a new approach to annotate the dataset were collected from Twitter, Facebook, and online newspaper by creating a website used for annotation process as an online game to attract more users who talk different Arabic dialects as unpaid volunteers with no need to register in comparing with other crowdsourcing websites.

  - **Chapter 7** presents a description of the final version of written Arabic dialects texts corpus that were collected from Twitter based on the seed words and spatial coordinates, Facebook based on famous Facebook pages in Arab countries, and online newspaper based on famous electronic newspaper in Arab countries. The Arab countries were divided into five groups, one for each of the five main dialects: Gulf, Iraqi, Egyptian, Levantine and North African.
- **Part III** included three chapters: Initial experiment in classification, Classifying Arabic dialects in three different corpora using ensemble classifier, and Automatic dialect classification.
  - **Chapter 8** described the systems were built to classify Arabic dialects in Discriminating Similar Languages (DSL) 2016 shared task by using WEKA data analytic tool and SMO machine learning algorithm after testing variants of SMO with different tokenizers, IDF, TF, WC values.
  - **Chapter 9** described the systems were built to classify Arabic dialects generated from three different sources of text data using WEKA data analytic tool and ensemble classifier consists of SMO and MNB machine learning algorithms.
  - **Chapter 10** introduces the methods were used to classify Arabic dialect texts and the achieved accuracies using these different methods.
- **Part IV** included two chapters: Conclusion and future work.
  - **Chapter 11** summarizes the thesis achievements, limitations, conclusion and future work.

11.2 Conclusions

In this thesis, we have classified Arabic dialect texts were collected from social media. The objective of this work was create an Arabic dialect text corpus and use this text to classify Arabic dialect using lexicon based methods and machine learning algorithms.

Chapter 1 provided a concise introduction of the research domain and Arabic language also included the objectives of this research and the contributions. In addition to overview of the thesis chapters.

In Chapter 2 background information about Arabic dialects and Arabic dialect corpora are presented. The research focused on five Arabic dialects divided based on the geographical locations: Gulf, Egyptian, Levantine, Iraqi, and North African. The variation between Arabic dialect are discussed: lexical, phonological, and orthographical variations. The related work focused on the previous research on creating Arabic dialect corpora and dialects classification. The machine learning algorithms and feature selection methods.

In Chapter 3 Twitter was explored as a source of Arabic dialect corpus using the list of seed words. Seed words are distinguishing words that are very common in one dialect and not used in any other dialects. By running the Twitter extractor for 144 hours, we collected 210,915K tweets with the total number of words equal to 3,627,733 words. The accuracy of classification increased from 42% in Chapter 8 to 79% in Chapter 3 using the new Twitter dataset.

Chapter 4 explored online newspaper as another source of Arabic dialect texts to cover long dialect texts as at that time when this source used Twitter was limit the text in 140 characters. The comments from electronic newspaper were extracted from 25 different Arabic electronic newspaper and classified based on the country which issued each of the newspapers.

In Chapter 5, the Arabic dialect texts corpus was extended by exploring Twitter based on the spatial coordinate points and scrape Facebook to collect users' comments on Facebook posts. The spatial coordinate points for capital, famous and big cities were specified to extract tweets based on location. This method collected 112,321 tweets from different countries in the Arab world. The total number of comments is 2,888,788 comments collected from most popular Arabic pages on Facebook in different domains such as, sport pages, comedy pages, channels and programs pages, and news pages.

The collected texts in Chapter 3, 4, and 5 were labelled based on: the location that appears in the user's profile, the spatial coordinate points, the country where the newspaper is published, and the country of the Facebook page depended on the nationality of the owner of the Facebook page. But this method produced some mislabelled documents, so in Chapter 6 the method on crowdsourcing Arabic dialect annotation was developed as an online annotation tool to label each document with the correct dialect.

Chapter 7 presented the difficulties in using social media as a source of Arabic dialect text, and the description of the final version of the corpus after applying the criteria in Section 7.1.2, contains 1,088,578 documents; they include 812,849 Facebook comments, 9,440 online newspaper comments, and 266,289 Twitter tweets; 180,282 based on seed terms, and 86,007 based on spatial coordinate points.

Chapter 8 showed the first experiment in classifying Arabic dialect which published in the VarDial2016 workshop at COLING 2016 Discriminating Similar Languages (DSL) 2016 shared task (Alshutayri *et al.* 2016). The shared task offered two tasks: first task worked on identification of very similar languages in newswire texts. The second task focused on Arabic dialect identification in speech transcripts (Malmasi *et al.* 2016) using the dataset were developed using the QCRI Automatic Speech Recognition (ASR) QATS system (Khurana and Ali 2016) to label each documents with a dialect (Ali *et al.* 2016). The result achieved in Chapter 8 showed the importance of creating an Arabic dialect texts corpus to improve the accuracy of classification.

Chapter 9 used an ensemble classifier method to combining Sequential Minimal Optimization (SMO) algorithm with Multinomial Naive Bayes (MNB) to classify Arabic dialect texts in three different corpora: transcripts of utterances by Arabic dialect speakers, texts (tweets) collected from Twitter, and readers' comments collected from electronic newspapers.

Chapter 10 introduced a new approach for classifying Arabic dialect text by building two models. The first model based on lexicon classifier, using different methods of classification based on dictionaries. The second method using Machine Learning algorithms.

11.3 Achieved Contributions

In this research the contributions are:

- The construction of a large multi-dialect corpus of Arabic.
- An exploration of how to extract geolocation sensitive text from various social and internet media.
- The use of gamification for corpus annotation.
- Identification and extraction of new linguistic features to classify Arabic dialect text which can be tested in different classifiers.
- Creation of dictionaries for each dialect.
- The use of ML and dictionary based approaches to automatically classify dialects.

11.4 Future Work

This research opens possibilities for other studies work on Arabic language and its dialect or any other language, especially with written text as many studies was in spoken Arabic. The following points are a potential avenues for future work.

- Exploring other sources of informal Arabic dialect text such as WhatsApp and Instagram applications, blogs or YouTube comments to cover most of the sources, in addition to using speech recognition on spoken Arabic dialects to extend SMADC and to build a corpus including different sources of the Arabic dialect text.

- Comparing Arabic dialect texts against other variants of Arabic, such as Classical Arabic of the Quran (Alrabiah *et al.* 2014a; Alrabiah *et al.* 2014b).

- Improving the result of classification by extracting the misclassified documents and find the reason of the misclassification.

- Combining WordTokenizer and CharacterNGram as a features to improve the results using an ensemble method.

- Modifying the interface of the annotation tool to be more attractive and easier to explore. In addition, we could make this annotation tool as an application which can be downloaded on to smart phones and tablets.

- Using deep learning models (Elaraby and Abdul-Mageed 2018) and word embedding classifiers to compare the results with WEKA classifiers as well as other tasks such as checking the similarity of Arabic sentences (Nagoudi and Schwab 2017).

- Extending this research to other language dialects such as Greek. Greek is spoken and written mainly in Greece and in Cyprus, and Cyprus has a slightly different dialect (Sababa 2018).

List of References

Abu-Errub, A. 2014. Arabic Text Classification Algorithm using TFIDF and Chi Square Measurements. *International Journal of Computer Applications (IJCA),* **93**(6), pp.40-45.

Adouane, W. and S. Dobnik. 2017. Identification of Languages in Algerian Arabic Multilingual Documents. *In: Proceedings of The Third Arabic Natural Language Processing Workshop (WANLP), Valencia, Spain*. Association for Computational Linguistics, pp.1-8.

Aguade, J. 2015. The Arabic Dialect of Tangier Across a Century. *In: 11th International Conference of AIDA, Bucharest*. pp.21-27.

Al-Badrashiny, M., H. Elfardy and M. Diab. 2015. AIDA2: A Hybrid Approach for Token and Sentence Level Dialect Identification in Arabic. *In: Proceedings of the 19th Conference on Computational Language Learning*, *30-31 July, Beijing, China*. pp.42–51.

Al-Harbi, W. A. and A. Emam. 2015. Effect of Saudi Dialect Preprocessing on Arabic Sentiment Analysis. *International Journal of Advanced Computer Technology (IJACT),* **4**(6), pp.91-99.

Al-Sabbagh, R., J. Diesner and R. Girju. 2013. Using the Semantic-Syntactic Interface for Reliable Arabic Modality Annotation. *In: International Joint Conference on Natural Language Processing*, *14-18 October, Nagoya, Japan*. pp.410–418.

Al-Sulaiti, L. and E. Atwell. 2004. Designing and Developing a Corpus of Contemporary Arabic. *In: Proceedings of TALC 2004: the sixthTeaching and Language Corpora conference, Granada, Spain*. pp.92-93.

AL-Walaie, M. A. and M. B. Khan. 2017. Arabic dialects classification using text mining techniques *In: International Conference on Computer and Applications (ICCA)*. IEEE, pp.325-329.

Ali, A., P. Bell and S. Renals. 2015. Automatic Dialect Detection in Arabic Broadcast Speech. *arXiv:1509.06928v1 Computation and Language (cs.CL)*.

Ali, A., N. Dehak, P. Cardinal, S. Khurana, S. H. Yella, J. Glass, P. B. and and S. Renals. 2016. Automatic Dialect Detection in Arabic Broadcast Speech. *Interspeech2016*, pp.2934-2938.

Ali, A., H. Mubarak and S. Vogel. 2014. Advances in Dialectal Arabic Speech Recognition: A Study Using Twitter to Improve Egyptian ASR. *In: International Workshop on Spoken Language Translation (IWSLT)*, *4-5 December, Lake Tahoe, US*.

Almeman, K. and M. Lee. 2013. Automatic Building of Arabic Multi Dialect Text Corpora by Bootstrapping Dialect Words. *In: Communications,* *Signal Processing, and their Applications (ICCSPA), 1st International Conference, Sharjah, UAE*. IEEE.

Almeman, K., M. Lee and A. A. Almiman. 2013. Multi Dialect Arabic Speech Parallel Corpora. *In: Communications, Signal Processing, and their Applications (ICCSPA), 1st International Conference, Sharjah, UAE*. IEEE.

Alorifi, F. S. 2008. *Automatic Identification of Arabic Dialects Using Hidden Markov Models*. Doctor of Philosophy thesis, University of Pittsburgh.

Alrabiah, M., N. Alhelewh, A. Al-salman and E. Atwell. 2014a. An Empirical Study On The Holy Quran Based On A Large Classical Arabic Corpus. *International Journal of Comutational Linguistics,* **5**(1), pp.1-13.

Alrabiah, M. S., A. Al-Salman, E. Atwell and N. Alhelewh. 2014b. KSUCCA: A key To Exploring Arabic Historical Linguistics. *International Journal of Computational Linguistics (IJCL),* **5**(2), pp.27-36.

Alshutayri, A. and E. Atwell. 2017. Exploring Twitter as a Source of an Arabic Dialect Corpus. *International Journal of Computational Linguistics (IJCL),* **8**(2), pp.37-44.

Alshutayri, A. and E. Atwell. 2018a. Arabic Dialects Annotation using an Online Game. *In: 2nd International Conference on Natural Language and Speech Processing (ICNLSP), Algiers, Algeria*. IEEE.

Alshutayri, A. and E. Atwell. 2018b. Creating an Arabic Dialect Text Corpus by Exploring Twitterr, Facebook, and Online Newspapers. *In: Proceedings of OSACT'2018 Open-Source Arabic Corpora and Processing Tools*, *07-12 May, Miyazaki, Japan*. pp.1-13.

Alshutayri, A. and E. Atwell. 2018c. *A Social Media Corpus of Arabic Dialect Text* [online]. Clermont-Ferrand: Presses universitaires Blaise Pascal.

Alshutayri, A., E. Atwell, A. Alosaimy, J. Dickins, M. Ingleby and J. Watson. 2016. Arabic Language WEKA-Based Dialect Classifier for Arabic Automatic Speech Recognition Transcripts. *In: Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects*, *12 December, Osaka, Japan*. pp.204-211.

Alsulaiti, L. and E. Atwell. 2005. Extending the Corpus of Contemporary Arabic. *In: CL 2005 International Conference on Corpus Linguistics, University of Birmingham, UK*.

Atwell, E. 2018a. *Classical and Modern Arabic corpora: genre and language change* [online]. John Benjamins.

Atwell, E. 2018b. *Using the Web to model Modern and Quranic Arabic* [online]. Edinburgh University Press.

Ayodele, T. O. 2010. Types of Machine Learning Algorithms. *In:* Y. ZHANG, ed. *New Advances in Machine Learning, Rijeka, Croatia*.

Belgacem, M., G. Antoniadis and L. Besacier. 2010. Automatic Identification of Arabic Dialects. *In: International Conference on Language Resources and Evaluation*, *May*. pp.3437-3441.

Biadsy, F., J. Hirschberg and N. Habash. 2009. Spoken Arabic Dialect Identification Using Phonotactic Modeling. *In: Proceedings of the EACL Workshop on Computational Approaches to Semitic Languages*, *31 March, Athens, Greece*. Association for Computational Linguistics, pp.53-61.

Bouamor, H., N. Habash and K. Oflazer. 2014. A Multidialectal Parallel Corpus of Arabic. *In: Proceeding of the Ninth International Conference on Language Resources and Evaluation (LREC), Iceland*. pp.1240-1245.

Buckwalter, T. 2002. *Arabic Transliteration* [online]. [Accessed]. Available from: http://www.qamus.org/transliteration.htm.

Cavnar, W. B. and J. M. Trenkle. 1994. N-Gram-Based Text Categorization. *In: In Proceedings of SDAIR-94, 3rd Annual Symposium on Docu-ment Analysis and Information Retrieval Las Vegas*. pp.161-175.

Duwairi, R. M. 2015. Sentiment Analysis for Dialectical Arabic. *In: 6th International Conference on Information and Communication Systems (ICICS)*. IEEE, pp.166-170.

El-Haj, M., P. Rayson and M. Aboelezz. 2018. Arabic Dialect Identification in the Context of Bivalency and Code-Switching. *In: 11th edition of the Language Resources and Evaluation Conference (LREC'18)*, *May, Miyazaki, Japan*.

Elaraby, M. and M. Abdul-Mageed. 2018. Deep Models for Arabic Dialect Identification on Benchmarked Data. *In: Proceedings of the Fifth Workshop on NLP for Similar Languages, Varieties and Dialects*, *20 August, Santa Fe, New Mexico, USA*. pp.263–274.

Eldesouki, M., F. Dalvi, H. Sajjad and K. Darwish. 2016. QCRI @ DSL 2016: Spoken Arabic Dialect Identification Using Textual Features. *In: Proceedings of the Third Workshop on NLP for Similar Languages, Varieties and Dialects*, *12 December, Osaka, Japan*. pp.221-226.

Elfardy, H. and M. Diab. 2012. Token Level Identification of Linguistic Code Switching. *In: Proceedings of the 24th International Conference on Computational Linguistics (COLING)*, *December, Mumbai*. pp.287–296.

Elfardy, H. and M. Diab. 2013. Sentence Level Dialect Identification in Arabic. *In: Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, *4-9 August, Sofia, Bulgaria*. pp.456–461.

Elmahdy, M., R. Gruhn, W. Minker and S. Abdennadher. 2010. Cross-Lingual Acoustic modeling for Dialectal Arabic Speech Recognition. *In: In*

Eleventh Annual Conference of the International Speech Communication Association. pp.873–876.

Fleiss, J. L. 1971. Measuring nominal scale agreement among many raters. *Psychological Bulletin,* **76**(5), pp.378–382.

Gairdner, W. H. T. 1925. *The Phonetics of Arabic.* Oxford University Press.

Gebre, B. G., M. Zampieri, P. Wittenburg and T. Heskes. 2013. Improving Native Language Identification with TF-IDFWeighting. *In: Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications*, *13 June, Atlanta, Georgia*. Association for Computational Linguistics, pp.216-223.

Gerard, S. and B. Christopher. 1988. Term-weighting approaches in automatic text retrieval. *Information Processing & Management,* **24**(5), pp.513-523.

Grigore, G. and G. Bițună. 2015. Arabic Varieties: Far and Wide. *In: Proceedings of the 11th International Conference of AIDA,* Bucharest. p.588.

Guellil, I. and F. Azouaou. 2016. Arabic Dialect Identification with an unsupervised learning (based on a lexicon) Application case: ALGERIAN Dialect *In: IEEE International Conference on Computational Science and Engineering, IEEE International Conference on Embedded*

and Ubiquitous Computing, and International Symposium on Distributed Computing and Applications to Business, Engineering

and Science. IEEE Computer Society, pp.724-731.

Habash, N., R. Eskander and A. Hawwari. 2012. A Morphological Analyzer for Egyptian Arabic. *In: Proceedings of the Twelfth Meeting of the Special Interest Group on Computational Morphology and Phonology (SIGMORPHON2012)*, *7 June, Montreal, Canada*. Association for Computational Linguistics, pp.1–9.

Habash, N. Y. 2010. *Introduction to Arabic Natural Language Processing* [online]. Morgan & Claypool.

Hall, M., E. Frank, G. Holmes, B. Pfahringer, P. Reutemann and I. H. Witten. 2009. The WEKA Data Mining Software: An Update. *SIGKDD Explorations,* **11**(1), pp.10-18.

Harrat, S., K. Meftouh, M. Abbas and K. Smaili. 2014. Building Resources for Algerian Arabic Dialects. *In: 15th Annual Conference of the International Communication Association (Interspeech)*. pp.2123–2127.

Holes, C. 2004. *Modern Arabic: Structures, Functions, and Varieties* [online]. Washington, D.C.: Georgetown University Press.

Horesh, U. and W. M. Cotter. 2016. Current Research on Linguistic Variation in the Arabic-Speaking World. *Language and Linguistics Compass,* **10**(8), pp.370-381.

Huang, F. 2015. Improved Arabic Dialect Classification with Social Media Data. *In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, *17-21 September Lisbon, Portugal*. Association for Computational Linguistics, pp.2118–2126.

Huang, O. 2017. *Applying Multinomial Naive Bayes to NLP Problems: A Practical Explanation* [online]. [Accessed]. Available from: https://medium.com/syncedreview/applying-multinomial-naive-bayes-to-nlp-problems-a-practical-explanation-4f5271768ebf.

Ibrahim, H. S., S. M. Abdou and M. Gheith. 2015. Sentiment Analysis For Modern Standard Arabic And Colloquial. *International Journal on Natural Language Computing (IJNLC),* **4**(2), pp.95-109.

Ikonomakis, E. K., S. Kotsiantis and V. Tampakas. 2005. Text Classification Using Machine Learning Techniques *WSEAS TRANSACTIONS on COMPUTERS,* **4**(8), pp.966-974.

Itani, M., C. Roast and S. Al-Khayatt. 2017. Corpora For Sentiment Analysis Of Arabic Text In Social Media. *In: 8th International Conference on Information and Communication Systems (ICICS)*, *4-6 April Irbid, Jordan*. IEEE.

Joachims, T. 1997. A Probabilistic Analysis of the Rocchio Algorithm with TFIDF for Text Categorization. *In: Proceedings of the Fourteenth International Conference on Machine Learning*. Morgan Kaufmann Publishers Inc., pp.143-151.

Joachims, T. 1998. Text Categorization with Support Vector Machines: Learning with Many Relevant Features. *In: European conference on machine learning*. Springer Berlin Heidelberg, pp.137-142.

Jurafsky, D. 2011. *Text Classification and Naïve Bayes* [online]. Available from: https://web.stanford.edu/class/cs124/lec/naivebayes.pptx.

Kalach, N. 2015. Home Arabic: First Issues. *In: 11th International Conference of AIDA, Bucharest*. pp.337-344.

Karen, M. and M. Faiza. 2010. *Tunisian Arabic Corpus (TAC): 895,000 words.*

Kerl, C. 2010. *Woath it? Coase ah am, pet.* London: Coronet.

Khoshaba, M. P. 2006. *Iraqi Dialect Versus Standard Arabic* [online]. USA: Medius Corporation.

Khurana, S. and A. Ali. 2016. QCRI advanced transcription system (QATS) for the Arabic multi-dialect broadcast media recognition: MGB-2 challenge. *In:* IEEE, ed. *Spoken Language Technology Workshop (SLT)*. pp.292–298.

Kilgarriff, A., V. Baisa, J. Bušta, M. Jakubíček, V. Kovář, J. Michelfeit, P. Rychlý and V. Suchomel. 2014. The Sketch Engine: ten years on. *Lexicography,* **1**(1), pp.7-36.

Korde, V. and C. N. Mahender. 2012. Text Classification and Classifiers: a Survey. *International Journal of Artificial Intelligence & Applications (IJAIA), ,* **3**(2), pp.85-99.

Landis, J. R. and G. G. Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics, Wiley, International Biometric Society,* **33**(1), pp.159–174.

Lu, M. and M. Mohamed. 2011. *LAHGA: Arabic Dialect Classifier*. IR'11, Boulder, Colorado, USA.

Lulu, L. and A. Elnagar. 2018. Automatic Arabic Dialect Classification Using Deep Learning Models. *In: The 4th International Conference on Arabic Computational Linguistics (ACLing 2018)*, *November 17-19, Dubai, United Arab Emirates*. Procedia Computer Science, pp.262-269.

Ma, D. and M. Saunders. 2018 *SVM using SMO vs PDCO: Support Vector Machines using Sequential Minimal Optimization vs Primal-Dual interior method for Convex Objectives*. Unpublished.

Maamouri, M., A. Bies, F. Gaddeche, S. Krouna and D. T. Toub. 2009. Guidelines for Treebank Annotation of Speech Effects and Disfluency for the Penn Arabic Treebank V1.0 [online]. Available from: http://projects.ldc.upenn.edu/ArabicTreebank/.

Mahedero, J. P. G., A. M.´ınez, P. Cano, M. Koppenberger and F. Gouyon. 2005. Natural language processing of lyrics. *In: Proceedings of the 13th ACM International Conference on Multimedia*, *6-11 November, Singapore*.

Malmasi, S. and M. Dras. 2018. Native Language Identification With Classifier Stacking and Ensembles. *Computational Linguistics,* **44**(3), pp.403-446.

Malmasi, S., E. Refaee and M. Dras. 2015. Arabic Dialect Identification using a Parallel Multidialectal Corpus. *Pacific Association for Computational Linguistics*, pp.203-211.

Malmasi, S., M. Zampieri, N. Ljubešić, P. Nakov, A. Ali, J. Tiedemann and L. Tan. 2016. Discriminating between Similar Languages and Arabic Dialect Identification: A Report on the Third DSL Shared Task. *In: Proceedings of the 3rd Workshop on Language Technology for Closely Related Languages, Varieties and Dialects (VarDial), Osaka, Japan*.

Manning, C. D., P. Raghavan and H. Schütze. 2008. *Introduction to Information Retrieval* [online]. Cambridge University Press.

Mansour, M. A. 2013. The Absence of Arabic Corpus Linguistics: A Call for Creating an Arabic National Corpus. *International Journal of Humanities and Social Science,* **3**(12), pp.81-90.

Mdhaffar, S., F. Bougares, Y. Esteve and L. Hadrich-Belguith. 2017. Sentiment Analysis of Tunisian Dialect: Linguistic Resources and Experiments. *In: Proceedings of The Third Arabic Natural Language Processing Workshop (WANLP)*, *3 April, Valencia, Spain*. Association for Computational Linguistics, pp.55–61.

Meder, T., D. Nguyen and R. Gravel. 2016. The apocalypse on Twitter. *Digital Scholarship in the Humanities,* **31**(2), pp.398-410.

Mubarak, H. and K. Darwish. 2014. Using Twitter to Collect a Multi-Dialectal Corpus of Arabic. *In: Proceedings of the EMNLP Workshop on Arabic Natural Langauge Processing (ANLP)*, *25 October, Doha, Qatar*. pp.1–7.

Muntsa, P. and P. Llu´ıs. 2004. Comparing methods for language identification. *Procesamiento del Lenguaje Natural,* **33**(1), pp.155-162.

Nagoudi, B. E. M. and D. Schwab. 2017. Semantic Similarity of Arabic Sentences with Word Embeddings. *In: Proceedings of The Third Arabic Natural Language Processing Workshop (WANLP)*, *3 April, Valencia, Spain*. Association for Computational Linguistics, pp.18-24.

Oracle. 2010. *The Java EE 5 Tutorial* [online]. [Accessed March, 2017]. Available from: http://docs.oracle.com/javaee/5/tutorial/doc/javaeetutorial5.pdf.

Pak, A. and P. Paroubek. 2010. Twitter as a Corpus for Sentiment Analysis and Opinion Mining. *In: In Proceedings of the International Conference on Language Resources and Evaluation LREC, Valletta, Malta*. pp.1320-1325.

Pasha, A., M. Al-Badrashiny, M. Diab, A. E. Kholy, R. Eskander, N. Habash, M. Pooleery, O. Rambow and R. M. Roth. 2014. MADAMIRA: A Fast, Comprehensive Tool for Morphological Analysis and Disambiguation of Arabic. *In: In Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC)*. pp.1094-1101.

Patil, T. R. and S. S. Sherekar. 2013. Performance Analysis of Naive Bayes and J48 Classification Algorithm for Data Classification. *International Journal Of Computer Science And Applications,* **6**(2), pp.256-261.

Platt, J. C. 1998. Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines. [online].

Robertson, S. 2004. Understanding Inverse Document Frequency: On theoretical arguments for IDF. *Journal of Documentation,* **60**(5), pp.503-520.

Roul, R. K., O. R. Devanand and S. K. Sahay. 2014. Web Document Clustering and Ranking using Tf-Idf based Apriori Approach. *In: International Conference on Advances in Computer Engineering and Applications (IJCA)*. pp.74-78.

Sababa, H. 2018. *A Classifier to Distinguish Between Cypriot Greek and Standard Modern Greek*. thesis, University of Nicosia.

Sadat, F., F. Kazemi and A. 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)*, *24 August, Dublin, Ireland*. pp.22-27.

Saloot, M. A., N. Idris, A. Aw and D. Thorleuchter. 2016. Twitter corpus creation: The case of a Malay Chat-style-text Corpus (MCC). *Digital Scholarship in the Humanities, Vol. 31, No. 2,,* **31**(2), pp.227-243.

Shalev-Shwartz, S. and S. Ben-David. 2014. *Understanding Machine Learning: From Theory to Algorithms* [online]. New York, USA: Cambridge University Press.

Sharoff, S. 2006. Open-source Corpora: using the net to fish for linguistic data. *International Journal of Corpus Linguistics,* **11**(4), pp.435-462.

Shoufan, A. and S. Al-Ameri. 2015. Natural Language Processing for Dialectical Arabic: A Survey. *In: Proceedings of the Second Workshop on Arabic Natural Language Processing*, *26-31 July, Beijing, China*. Association for Computational Linguistics, pp.36–48.

Solorio, T., E. Blair, S. Maharjan, S. Bethard, M. Diab, M. Gohneim, A. Hawwari, F. AlGhamdi, J. Hirschberg, A. Chang and P. Fung. 2014. Overview for the First Shared Task on Language Identification in Code-Switched Data. *In: Proceedings of The First Workshop on Computational Approaches to Code Switching*, *25 October, Doha, Qatar*. Association for Computational Linguistics, pp.62–72.

Tutorialspoint. 2015. *Java servlets web application framework* [online]. Tutorials Point. Available from: https://www.tutorialspoint.com/servlets/servlets_tutorial.pdf.

Versteegh, K. 2014. *The Arabic Language.* Edinburgh, United Kingdom: Edinburgh University Press.

Wang, Y., Z. Zhou, S. Jin, D. Liu and M. Lu. 2018. Comparisons and Selections of Features and Classifiers for Short Text Classification. *In: IOP Conference Series: Materials Science and Engineering*.

Witten, I. H. and E. Frank. 2005. *Data Mining: Practical Machine Learning Tools and Techniques.* United States of America: Elsevier.

Xiang, G., B. Fan, LingWang, J. I. Hong and C. P. Rose. 2012. Detecting Offensive Tweets via Topical Feature Discovery over a Large Scale Twitter Corpus. *In: In Proceedings of the 21st ACM International*

Conference on Information and Knowledge Management. CIKM'12., *October 29–November 2, New York, USA*. ACM, pp.1980-1984.

Yun-tao, Z., G. Ling and W. Yong-cheng. 2005. An improved TF-IDF approach for text classification. *Journal of Zhejiang University SCIENCE* **6A**(1), pp.49-55.

Zaidan, O. F. and C. Callison-Burch. 2011. The Arabic Online Commentary Dataset: an Annotated Dataset of Informal Arabic with High Dialectal Content. *In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, *19-24 June, Portland, Oregon*. Association for Computational Linguistics, pp.37-41.

Zaidan, O. F. and C. Callison-Burch. 2014. Arabic Dialect Identification. *Computational Linguistics,* **40**(1), pp.171-202.

Zampieri, M., B. G. Gebre, H. Costa and J. v. Genabith. 2015. Comparing Approaches to the Identification of Similar Languages. *In: Proceedings of the Joint Workshop on Language Technology for Closely Related Languages, Varieties and Dialects*, *10 September, Hissar, Bulgaria*. Association for Computational Linguistics, pp.66-72.

Zerrouki, T. and M. Amara. 2009. *Arabic Stop words*.

Zwiefelhofer, D. B. 2008. *Find Latitude and Longitude* [online]. [Accessed 10/11/2016]. Available from: https://www.findlatitudeandlongitude.com/.

Appendix A
Lists of Seed Words in each Dialect

Table A.1 Seed words extracted from Twitter

Twitter							
English	**IPA**	**MSA**	**GLF**	**LEV**	**IRQ**	**EGY**	**NOR**
Sit	ʔʤls	اجلس- اقعد	اجلس- ايلس - اقعد			اوعد	قعمز
Man	rʤl	رجل	رجل - ريال - رجال	زلمه	رجال	راقل	راجل
Now	alħi:n	الان - الحين	الان - الحين	هسا-هلق- هلأ	هسا	دلوقتي	
Money	nqu:d	نقود	فلوس - دراهم	مصاري		فلوس	فلوس - دراهم
Look	ʔnðˤr	انظر	شوف	اطلع	باوع	بص	اشبح – راه-برق
Sorry	ʔsf	أسف				معلش	
I want	ʔri:d	أريد	ابغي – ابي	بدي	أريد	عاوز	
Yours	mlkh	ملكه	حقه-ماله	تبعو	ملكه-ماله	بتاعتو	دياله- بتاعه
What	ma:ða:	ماذا	ايش	شو		ايه	واش – شنوى
Well	ʤjd	جيد	زين	منيح		كويس	مليح – باهي
How are you	kjf ħa:lk	كيف حالك	كيف حالك – شلونك	كيفك		ازيك	كيفاش- كيف حالك
This	hða:	هذا	هذا	هيدا	هاذ	ده	هدا-هذيه

Go away	ʔbtʕd	ابتعد	روح – بعد	زيح		ابعد	حول-نحي
For	lʔʤl	لأجل	عشان	كرمال – منشان		عشان	علشان- علخاطر
Elderly	ʕʤu:z	عجوز	شايب	ختيار		عقوز	شايب
Hot	ћa:r	حار	حر	شوب		سخن	حامي
Nothing	lā ju:ʤd	لا يوجد	مافي			مفيش	مهناكش- مهناش
Next to you	bʤa:nbk	بجانبك	جنبك	حدك	يمك	جمبك	يالاك- بحدك
Much	kθi:r	كثير	كثير – واجد	كتير	هوايه	كتير	برشا – بزاف- هلبا-واجد
Two	ʔθnjn	اثنين	اثنين	تنين		اتنين	زوز
Hospital	mstʃfa:	مستشفى	مستشفى	مشفى- خستخانه	خستخانه	مستشفى – اسبتاليه	سبيطار- اسبيتار
Enter	tdxl	تدخل	تدخل- تدش			تخش	تخش
How	kjf	كيف	كيف			ازاي	شنوا-شنو
Become	jsˁbћ	يصبح	يصير		يظل	يبقى	يقعد
Send to me	ʔrsl li:	أرسل لي	طرشلي			ابتعتلي	بتعلي- دزلي
Married	mtzwʤ	متزوج	متزوج- متجوز	مجوز	متزوج	مقوز	متجوز
Ready	ʤa:hz	جاهز	جاهز	حاضر	تأهب	قاهز	واتي
Mouth	fm	فم	فم	تم		بوء	فم
Nose	ʔnf	أنف	خشم	منخار		مناخير	خشم
Calamity	da:hjh	داهيه - مصيبه	داهيه - مصيبه		داهيه- طركاعه	نيله	
What is the	ma:ða: bk	ماذا بك	ايشفيك	اشبيج	شوبك	مالك	خيرك- شنفيك

matter with you							
Wake up	ʔstjqðˤ	استيقظ	قام-قعد	قعد	فاق	صحى	ناض
You can	tsttˤjʕ	تستطيع	تقدر	تحسن	تكدر	تءدر	تقدر
Does not affect	lm jʕd jʔθr	لم يعد يأثر	ماعاد يأثر	مش عم بيأسر	ماديأثر	مبأش بيأسر	

Table A.2 Seed words extracted from Books

Books							
English	**IPA**	**MSA**	**GLF**	**LEV**	**IRQ**	**EGY**	**MAG**
Garrulous	fṣjħ	فصيح	فصيح	يلت	لغاوي- لغوة	لمض	
Every night	kl ljlh	كل ليلة	ليليا – كل ليلة			ليلاتي	كل ليلة
Mocks	jsxr	يسخر	يسخر – يتريق			يتمألس- يتريأ	
Rip	ʃq	شق		شق	شق	مرع	قلع
Sorry	ʔsf	أسف	معليش – أسف			معلهش	
Dislocate	nzʕ	نزع	نزع	خلع	خلع	ملخ	
Robe	ʤlba:b	جلباب	عبايه			ملايه	
Dress	θwb	ثوب	ثوب- دشداشه		دشداشه	جلابيه	سوريه
Pour	skb	سكب	كب		كب- انكب	دلق	بزاع- كب
Push	dfʕ	دفع	دف	دفع	درفع	زق – زء	دف

woman	sjdh	سيدة	حرمه	ست	حرمه	ست	
Get married	ʔtzwʤ	اتزوج	اتزوج			اتستت	
To leak	tsrb	تسرب	سرب	زرب	خر	سحسح	
Faint	fqd wʕjh	فقد وعيه	داخ	غط على قلبه	داخ	سخسخ	
Shrill	ħa:d	حاد	عالي	حاد	رفيع	سرسع	
thin	nħjl	نحيل	نحيف			مسلوع	
Hang up	tʕlq	تعلق	تعلق	علق	علق- أتشلبه	شعلق	اشعبط
Pleasure	sʕd	سعد	انبسط	اتمتع	اتلذذ	انشكح	
Fight	ʕra:k	عراك	مضاربة	مقاتله	عركة	عركة – خناقه	عركة
Guard	ħa:rs	حارس	حارس	حارس	ناطور	غفير	
Boat	qa:rb	قارب		شختوره	بلم	مركب	
Unemployed	ʕa:tˤl	عاطل	عاطل	بلا شغل	اجغ		
Careful	tħðr	تحذر		تحزر	تتأبي		
Toilet	ħma:m	حمام	حمام	توالت	ادبخانه		
Pharmacy	sˤjdljh	صيدلية	صيدلية	صيدلية	ازخانه	اقزخانه	صيدلية
Much	kθi:r	كثير		كتير	هوايه		واجد- هلبه
Disdain	anf	انف	اخجل	اخجل	انف	اكسف	تحشم
Measure	mqa:sa:ti:	مقاساتي	مقاساتي	قياساتي	أولجي		
Desert	hʤr	هجر		هجر	عاف		
Bribe	rʃwh	رشوه	رشوه	رشوه	برطل		
Hungry	ʤa:ʔʕ	جائع	جيعان	جوعان	جوعان	قيعان	جيعان
Eleven	ʔħd ʕʃr	أحد عشر	احدعش	ايدعش	دعش	حدعشر	احداش
To consult	tstʃi:r	تستشير	تشاور- تستشير	تستشير	تدانش		

Slipper	ħða:ʔ	حذاء	مداس	بابوج-صرمايه-شحاطه	مداس	شبشب	شبشب-سباط
Tour	nzhh	نزهه	تمشيه	كزدوره-جوله	خوره		تدهويره
Spoon	mlʕqh	ملعقه	ملعقه-قفشه	معلقه	خاشوقه		كشيك
Friendship	ʔsˤdqa:ʔ	اصدقاء	ربع-اصحاب-صحبه	رفقه	ربع	صحاب	اصحاب
Do you know	hl tʕlm	هل تعلم	تعرف	بتعرف	تعرف	تعرف	
Give	jʕtˤi:	يعطي	يعطي	يعطي	ينطي	يدي	
Balcony	ʃrfh	شرفه	بلكونه	بلكون	شرفه	برنده-بلكون	فيرانده
Orange	brtqa:l	برتقال	برتكان	بردءان	برتقال	برتقان	ليم
Window	na:fðh	نافذه	شباك-دريشه	شباك		شباك	روشن
Cup	kʔs	كأس	كاسه-قلاص	كأس	كلاص	كوبايه	طاسه-كبايه
Depend	jtkl	يتكل	اعتمد	يعتمد	عول		
Cream	qʃtˤh	قشطه	قشطه	اشته	قيمر	اشطه	
Shoe	ħða:ʔ	حذاء	جزمه	سباط	قندرة	قزمه	كندره-شلاكه
Blanket	ɣtˤa:ʔ	غطاء	لحاف	حرام	لحف	ملايه	ملايه-انصوله
Make dirty	wsx	وسخ	وسخ-بقع	لطخ	لوخ	وسخ	لبز
Table	tˤa:wlh	طاوله	طاوله	طاوله	ميز	طربيزه	طاوله-ميده
Slowly	tmhl	تمهل	شوي شوي	على مهل	يواش		بشويش

Appendix B
The Coordinate Points for each City

Dialect	Country	City	Longitude	Latitude
EGY	Egypt	Cairo	31.234131	30.031055
		Alexandria	29.915771	31.203405
		Port Said	32.299805	31.250378
		Asyut	31.190186	27.176469
		Sohag	31.695557	26.549223
		Tanta	31.014404	30.779598
		Luxor	32.640381	25.671236
NOR	Algeria	Algiers	3.076172	36.738884
		Oran	-0.637207	35.692995
		Annaba	7.756348	36.923548
		Ouargla	4.976807	32.166313
	Tunisia	Tunis	10.195313	36.81808
		Sfax	10.766602	34.75064
		Sousse	10.612793	35.826721
		Al-Qayrawan	10.096436	35.666222
	Morocco	Rabat	-6.844482	33.970698
		Casablanca	-7.580566	33.578015
		Marrakesh	-7.976074	31.625321
		Agadir	-9.602051	30.420256
	Libya	Tripoli	13.205566	32.879587
		Misrata	15.095215	32.342841
		Benghazi	20.170898	32.10119
		Sabha	14.458008	27.000408
GLF	Saudi Arabia	Riyadh	46.691895	24.686952
		Jeddah	39.221191	21.289374
		Makkah	39.858398	21.391705
		Medina	39.572754	24.507143
		Dammam	49.987793	26.372185
		Tabuk	36.5625	28.381735

		Abha	42.51709	18.271086
	Kuwait	Kuwait	47.988281	29.382175
	United Arab Emirates	Abu Dhabi	54.376831	24.45215
		Dubai	55.266724	25.204941
		Ras al-Khaimah	55.980835	25.799891
	Qatar	Doha	51.531372	25.279471
		Ar-Rayyan	51.410522	25.239727
		Al Khor	51.49292	25.676187
	Bahrain	Manama	50.597534	26.224447
IRQ	Iraq	Baghdad	44.362793	33.302986
		Ramadi	43.286133	33.422272
		Basrah	47.790527	30.514949
		Karbala	44.01123	32.593106
		Najaf	44.329834	32.026706
		Kirkuk	44.384766	35.46067
		Mosul	43.165283	36.350527
		Erbil	44.000244	36.199958
		Sulaymaniyah	45.450439	35.550105
		Falluujah	43.791504	33.339707
		Al-Nasiriyah	46.263428	31.043522
LEV	Jordan	Amman	35.930786	31.94284
		Irbid	35.851135	32.567648
		Az-Zarqa	36.095581	32.063956
		Jerash	35.908813	32.275522
	Palestine	Jerusalem	35.209808	31.76437
		Gaza	34.465485	31.500117
		Nablus	35.257874	32.219772
		Ramallah	35.200195	31.902044
		Haifa	34.992142	34.793624
	Lebanon	Beirut	35.499573	33.898917
		Tripoli	35.838776	34.442026
		Byblos	35.653381	34.127721
		Baalbek	36.205444	33.99575
	Syria	Damascus	36.274109	33.523079

		Aleppo	37.133789	36.206607
		Hama	36.757507	35.144617
		Homs	36.713562	34.741612
		Latakia	35.796204	35.543401
		Tartus	35.892334	34.890437

Appendix C
The Code of Frequent Terms Method

Begin

Define **EGYW** and **EGYF** for Egyptian dialect, **GLFW** and **GLFF** for Gulf dialect, **IRQW** and **IRQF** for Iraqi dialect, **LEVW** and **LEVF** for Levantine dialect, and **NORW** and **NORF** for North African dialect

Define **MSAF** a file of MSA words and stop words

```
FUNCTION sum(list, length):
   average=SUM(list)/ length
   RETURN average
ENDFUNCTION

FUNCTION multi(list):
   result=1
   for x in list:
      result *= x
   ENDFOR
   IF result ==1:
      result =0
   ENDIF
   RETURN result
ENDFUNCTION


INPUT document

// Check each word in the document if it is MSA word or not
FOR word IN document
     IF word IN MSAF
     THEN
         document <- document.replace(' '+word+' ', " ")
// Check if all words in the document are MSA words then enter a new
document
         IF Length(document)==0
         THEN
            INPUT document
         ENDIF
     ENDIF

// Check the rest of words in the document to decide each word belongs
to which dialect

category='Unclassified'

For word in document
```

```
    IF word in NORW
        Nweight= NORF(word)/ Length(NORW)
        NOR_V.append(Nweight)
    ELSE:
        NOR_V.append(0)
    ENDIF
    IF word in GLFW
        Gweight= GLFF(word)/ Length(GLFW)
        GLF_V.append(Gweight)
    ELSE:
        GLF_V.append(0)
    ENDIF
    IF word in IRQW
        Iweight= IRQF(word)/ Length(IRQW)
        IRQ_V.append(Iweight)
    ELSE:
        IRQ_V.append(0)
    ENDIF
    IF word in EGYW
        Eweight= EGYF(word)/ Length(EGYW)
        EGY_V.append(Eweight)
    ELSE:
        EGY_V.append(0)
    ENDIF
    IF word in LEVW
        Lweight= LEVF(word)/ Length(LEVW)
        LEV_V.append(Lweight)
    ELSE:
        LEV_V.append(0)
    ENDIF
ENDFOR
```

// Calculate average for each dialect vector

```
Avg_EGY=sum(EGY_V, Length(EGYW))
Avg_GLF=sum(GLF_V, Length(GLFW))
Avg_LEV=sum(LEV_V, Length(LEVW))
Avg_IRQ=sum(IRQ_V, Length(IRQW))
Avg_NOR=sum(NOR_V, Length(NORW))
```

// Check the average to compare which is the biggest average

```
IF Avg_EGY>Avg_GLF AND Avg_EGY>Avg_IRQ AND Avg_EGY>
     Avg_LEV AND Avg_EGY> Avg_NOR
THEN
    category='EGY'
ELSEIF Avg_NOR>Avg_EGY AND Avg_NOR>Avg_GLF AND Avg_NOR>
     Avg_IRQ AND Avg_NOR> Avg_LEV
THEN
    category='NOR'
```

```
ELSEIF  Avg_IRQ>Avg_EGY  AND  Avg_IRQ>Avg_GLF  AND  Avg_IRQ>
      Avg_LEV AND Avg_IRQ> Avg_NOR
THEN
     category='IRQ'
ELSEIF  Avg_LEV>Avg_EGY  AND  Avg_LEV>Avg_GLF  AND  Avg_LEV>
      Avg_IRQ AND Avg_LEV> Avg_NOR
THEN
     category='LEV'
ELSEIF  Avg_GLF>Avg_EGY  AND  Avg_GLF>Avg_IRQ  AND  Avg_GLF>
      Avg_LEV AND Avg_GLF> Avg_NOR
THEN
     category='GLF'
ENDIF

OUTPUT line
OUTPUT category
```

Appendix D
The Code of Voting Method

Begin

Define **EGYW** for Egyptian dialect, **GLFW** for Gulf dialect, **IRQW** for Iraqi dialect, **LEVW** for Levantine dialect, and **NORW** for North African dialect

Define **MSAF** a file of MSA words and stop words

```
FUNCTION sumColumn(matrix):
    total=SUM(matrixColumn)/
    RETURN total
ENDFUNCTION


INPUT document

// Check each word in the document if it is MSA word or not
FOR word IN document
        IF word IN MSAF
        THEN
            document <- document.replace(' '+word+' ', " ")
// Check if all words in the document are MSA words then enter a new document
            IF Length(document)==0
            THEN
                INPUT document
            ENDIF
        ENDIF

// Check the rest of words in the document to decide each word belongs to which dialect

category='Unclassified'
Create matrix[length(document)][5]
Row=0
M=5         // number of dialects

For word in document
        IF word in NORW
            Matrix[row][0]=1
        ELSE:
            Matrix[row][0]=0
        ENDIF
        IF word in EGYW
           Matrix[row][1]=1
    ELSE:
       Matrix[row][1]=0
    ENDIF
    IF word in IRQW
       Matrix[row][2]=1
    ELSE:
        Matrix[row][2]=0
    ENDIF
    IF word in LEVW
       Matrix[row][3]=1
    ELSE:
       Matrix[row][3]=0
    ENDIF
    IF word in GLFW
       Matrix[row][4]=1
    ELSE:
       Matrix[row][4]=0
    ENDIF
ENDFOR
```

//Using $\frac{1}{m}$ to represent the existence of a word in the dictionary instead of 1

```
nonZeros=numpy.count_nonzero(Matrix)
for i in range(Length(nonZeros)):
      if nonZeros[i]!=0:
        for j in range(m):
           if a[i][j]!=0:
              a[i][j]=1/nonZeros[i]
```

// Count number of words for each dialect column

vector=sumColumn(Matrix)

```
Sum_NOR=vector[0]
Sum_EGY=vector[1]
Sum_IRQ=vector[2]
Sum_LEV=vector[3]
Sum_GLF=vector[4]
```

// Check the average to compare which is the biggest average

```
ELSEIF  Sum_NOR>Sum_EGY  AND  Sum_NOR>Sum_GLF  AND
      Sum_NOR> Sum_IRQ AND Sum_NOR> Sum_LEV
THEN
    category='NOR'
```

```
IF Sum_EGY>Sum_GLF AND Sum_EGY>Sum_IRQ AND Sum_EGY> Sum_LEV AND Sum_EGY> Sum_NOR
THEN
    category='EGY'
ELSEIF Sum_IRQ>Sum_EGY AND Sum_IRQ>Sum_GLF AND Sum_IRQ> Sum_LEV AND Sum_IRQ> Sum_NOR
THEN
    category='IRQ'
ELSEIF Sum_LEV>Sum_EGY AND Sum_LEV>Sum_GLF AND Sum_LEV> Sum_IRQ AND Sum_LEV> Sum_NOR
THEN
    category='LEV'
ELSEIF Sum_GLF>Sum_EGY AND Sum_GLF>Sum_IRQ AND Sum_GLF> Sum_LEV AND Sum_GLF> Sum_NOR
THEN
    category='GLF'
ENDIF

OUTPUT line
OUTPUT category
```