

الجامعة  
البريطانية في  
دبي



The  
British University  
in Dubai

## **Investigating Cross-Lingual Hate Speech Detection on Social Media**

دراسه في الكشف عن خطاب الكراهية عبر اللغات على وسائل التواصل  
الاجتماعي

by

**HASSAN YOUSEF KHWILEH**

**Dissertation submitted in fulfilment  
of the requirements for the degree of  
MSc INFORMATICS**

at

**The British University in Dubai**

**January 2020**

## DECLARATION

I warrant that the content of this research is the direct result of my own work and that any use made in it of published or unpublished copyright material falls within the limits permitted by international copyright conventions.

I understand that a copy of my research will be deposited in the University Library for permanent retention.

I hereby agree that the material mentioned above for which I am author and copyright holder may be copied and distributed by The British University in Dubai for the purposes of research, private study or education and that The British University in Dubai may recover from purchasers the costs incurred in such copying and distribution, where appropriate.

I understand that The British University in Dubai may make a digital copy available in the institutional repository.

I understand that I may apply to the University to retain the right to withhold or to restrict access to my thesis for a period which shall not normally exceed four calendar years from the congregation at which the degree is conferred, the length of the period to be specified in the application, together with the precise reasons for making that application.

A handwritten signature consisting of several overlapping, slanted lines.

---

Signature of the student

## **COPYRIGHT AND INFORMATION TO USERS**

The author whose copyright is declared on the title page of the work has granted to the British University in Dubai the right to lend his/her research work to users of its library and to make partial or single copies for educational and research use.

The author has also granted permission to the University to keep or make a digital copy for similar use and for the purpose of preservation of the work digitally.

Multiple copying of this work for scholarly purposes may be granted by either the author, the Registrar or the Dean only.

Copying for financial gain shall only be allowed with the author's express permission.

Any use of this work in whole or in part shall respect the moral rights of the author to be acknowledged and to reflect in good faith and without detriment the meaning of the content, and the original authorship.

## Abstract

Social media platforms are becoming an integral part of our life. Massive amounts of content are being uploaded to social media platforms every second by online users. Social media sites are creating an exciting platform for online users to freely express their views, and share news or even thoughts and insights about any topic of their interest. Contrarily, social media platforms are becoming the ground for allowing toxic behaviour, online harassment, personal attacking and hate-speech content. This has resulted in many social media users closing their account to maintain their psychological and physical safety.

Major social media platforms such as Facebook, Twitter, YouTube are taking this problem very seriously, and making huge efforts and investment to maintain the trust, safety, integrity of the users in their platforms. However, recent research studies conducted in the United States on sample of online users, indicated that over 40% have personally experienced online harassment, and almost every online user is asking major online tech companies to act against it (Pew Research, USA, 2019).

With the availability of social media platforms in many languages and across different regions, Hate-speech and online harassment issues are becoming large-scale global problem that is affecting online users around the world. Therefore, there is an increasing demand to advance the current research and development in detecting online hate-speech not only for English but also for other languages. Previous research efforts have mainly focused on tackling hate-speech content for primary languages English, French and others, while very limited work has been done in other emerging languages such as Arabic where Internet penetration is exploding.

In this research, we investigate the task of building techniques for detecting online hate speech in Arabic language. Our contribution in this work can be summarised into two parts, the first part is to study the challenges of detecting hate speech for noisy, user-generated informal comments and tweets in Arabic, and the second part is to investigate novel approaches to build effective techniques for tackling this problem.

This work proposes a novel approach to handling Arabic hate-speech content that is based on Cross-lingual Arabic-to-English Text classification. The main hypothesis behind this approach is that by using effective and large-scale web translation resources such as Google Translate, we will be able to navigate both the social media noise, and the complexity of the dialectal and informal Arabic content. We test this approach on two very different datasets, the first one is specifically built by this work, that is a collection of Saudi-Arabic comments submitted on YouTube videos about specific Saudi controversial events in 2019, while the second one is collection of Egyptian tweets collected from Twitter platform about Egypt related political event. Both were annotated for hate-speech specific labels, and both are made available publicly by this research to encourage further research in this area.

Our extensive experimental investigation suggests that the proposed Cross-lingual Arabicto-English Text classification can indeed out-performs the traditional Arabic text classification for both datasets, YouTube comments and Twitter tweets.

Finally, our experiments present extensive comparative evaluation between different machine learning models and feature engineering approaches, and make some exciting suggestions to help upcoming research studies in the field of Arabic hate-speech detection.

## ملخص البحث

منصات وسائل الاعلام الاجتماعية أصبحت جزءا لا يتجزأ من حياتنا. يتم تحميل كميات هائلة من المحتوى إلى منصات وسائل الإعلام الاجتماعية كل ثانية من قبل المستخدمين على الإنترنت. مواقع وسائل الإعلام الاجتماعية تخلق منصة مثيرة للمستخدمين عبر الإنترنت للتعبير بحريه عن آرائهم ، وتبادل الأخبار أو حتى الأفكار والرؤى حول أي موضوع من اهتمامهم .

اضافة الى ذلك، منصات وسائل الإعلام الاجتماعية أصبحت الأرض الخصبة للسماح بالسلوك السام ، والتحرش عبر الإنترنت ، والهجوم الشخصي ، والمحتوى الذي يحض على الكراهية. وقد أدى ذلك إلى إغلاق العديد من مستخدمي وسائل الإعلام الاجتماعية حساباتهم للحفاظ على سلامتهم النفسية والجسدية .

منصات وسائل الإعلام الاجتماعية الرئيسية مثل الفيسبوك وتويتر ، يوتيوب تأخذ هذه المشكلة على محمل الجد ، وبذل جهود واستثمارات مالية ضخمة للحفاظ على الثقة وسلامة المستخدمين في منصاتها.

ومع ذلك ، أشارت الدراسات البحثية الأخيرة التي أجريت في الولايات المتحدة على عينة من المستخدمين على الإنترنت ، أن نسبة أكثر من 40% من المستخدمين واجهوا شخصيا التحرش عبر الإنترنت ، وتقريباً كل مستخدم على الإنترنت يطالب شركات التكنولوجيا على الإنترنت الرئيسية للعمل ضدها.

مع توافر منصات وسائل الإعلام الاجتماعية في العديد من اللغات وعبر مناطق مختلفة ، أصبحت قضايا الكراهية والكلام والتحرش عبر الإنترنت مشكلة عالمية واسعة النطاق تؤثر على المستخدمين عبر الإنترنت في جميع أنحاء العالم .

ولذلك ، هناك حاجة و طلب متزايد للمضي قدماً في البحث والتطوير الحالي في الكشف عن خطاب الكراهية عبر الإنترنت ليس فقط للإنجليزية ولكن أيضاً للغات أخرى.

ركزت الجهود البحثية السابقة بشكل رئيسي على معالجة محتوى خطاب الكراهية للغات الأولية الإنجليزية والفرنسية وغيرها ، في حين تم القيام بعمل محدود للغاية في اللغات الناشئة الأخرى مثل العربية حيث أن هناك كمية كبيرة من المحتوى العربي في منصات التواصل الاجتماعي.

في هذا البحث ، نبحث لبناء تقنيات ناجحة للكشف عن خطاب الكراهية على الإنترنت باللغة العربية.

ويمكن تلخيص مساهمتنا في هذا العمل في جزئين، الجزء الأول هو دراسة التحديات المتمثلة في كشف خطاب الكراهية للتعليقات العامية الصاخبة التي يولدها المستخدم في اليوتيوب وكذلك التغريدات في التويتر باللغة العربية ، والجزء الثاني هو بناء خوارزمية جديدة لبناء تقنيات فعالة لمعالجة هذه المشكلة .

ويقترح هذا العمل نهجاً جديداً للتعامل مع محتوى الكراهية العربية الذي يستند إلى تصنيف النصوص العربية إلى الإنجليزية عبر اللغات .

الفرضية الرئيسية وراء هذه الخوارزمية هي أنه باستخدام خدمات الترجمة الإلكترونية الفعالة واسعة النطاق مثل ترجمة Google ، سنتمكن من التنقل بين ضجيج وسائل الإعلام الإجتماعية ، وتعقيد المحتوى العربي غير الرسمي (اللغة العامية)

في هذه البحث سنعمل على إختبار هذه الخوارزمية على مجموعتين مختلفتين جداً من البيانات ، المجموعة الأولى تم تجميعها من ضمن هذا العمل ، وهي مجموعة من التعليقات العربية السعودية المقدمة على فيديوهات يوتيوب حول أحداث سعودية مثيرة للجدل في 2019 ، في حين أن الثانية هي مجموعة من تغريدات مصرية تم جمعها من منصة تويتر حول الحدث السياسي المتعلق بمصر .

تم تصنيف هذه البيانات (التعليقات في اليوتيوب والتغريدات في التويتر) الى خطاب الكراهية او غير ذلك بشكل يدوي عن طريق محكمين في اللغة العربية ، جميع البيانات المستخدمة في هذه البحث متاحة علنا بواسطة هذا البحث للتشجيع على إجراء مزيد من البحوث في هذا المجال .

ويشير تحقيقنا التجريبي المستفيض إلى أن التصنيف المقترح للنصوص العربية المترجمة الى اللغة الإنجليزية يمكن أن ينجح بالفعل في تصنيف النص العربي التقليدي لكل من مجموعات البيانات (تعليقات اليوتيوب وتغريدات تويتر).

وأخيراً ، تقدم تجاربنا المعدة في هذا البحث تقييماً مقارنة شاملاً بين نماذج التعلم الآلي المختلفة ونهج خوارزمية مميزة ، وتقدم بعض الإقتراحات الشاملة والمميزة للمساعدة في الدراسات البحثية القادمة في مجال الكشف عن خطاب الكراهية في العربية.

## Dedication

This dissertation study is dedicated to my father Yousef Khwileh and mother Huda Khwileh, my brother, I will always appreciate all they have done. I also dedicated this dissertation for my beloved wife and motivator Afnan and my beloved son Mohammad Khwileh.



## Acknowledgments

First of all, I would like to begin by thanking for Allah for supporting me with the perseverance and power to finish this dissertation study. I hope this work will add a useful knowledge for the science and research community. I would like also to appreciate and thank my supervisor Prof. Khaled Shaalan for the incessant provision of my MSc study and my research study during the past years, and for his motivation, patience, massive knowledge that he providing me during this study. Extra Special thanks to my dearest brother Dr. Ahmad Khwileh who support me in this dissertation by providing the academic advice, guidance to deliver this dissertation. Heartfelt thanks for Dr. Sherief Abdallah as his mentored modules and capacity to clarify the complicated knowledge of data mining and machine learning to made this work.

# Contents

Introduction	1
1.1 Online Hate Speech	3
1.2 Research Problem of Arabic Hate-Speech Detection	6
1.3 Scope of the Work	10
1.4 Research Questions	11
1.5 Thesis Structure	11
Background	12
2.1 Text Classification	12
2.2 Machine Learning	13
2.3 Text processing for Text Classification	14
2.3.1 Text Pre-processing	14
2.3.2 Stop Word Removal	15
2.3.3 Document Indexing and Term Weighting	15
2.4 Text Classification Methods	17
2.4.1 Rule-Based Systems	17
2.4.2 Machine learning Based Systems	18
2.5 Machine Learning Algorithms	18
2.5.1 Naive Bayes Methods	19
2.5.2 Support Vector Machine Methods	20
2.5.3 Random Forest	20

2.5.4	Deep Learning Methods	.....	21
2.6	Text Classification Evaluation	.....	25
2.7	Summary	.....	25
3	Related Work		27
3.1	Detecting Hate Speech for Social Media	.....	27
3.2	Cross-Lingual Text Classification	.....	29
3.3	Summary of previous work	.....	29
4	Experimental Setting and Evaluation		32
4.1	Saudi YouTube Comments Data Collection	.....	32
4.2	Egyptian Tweets Data Collection	.....	34
4.3	Arabic Data Pr-Processing	.....	35
4.3.1	Representation	.....	36
4.3.2	Tokenisation, Stemming, lemmatisation and stop words	.....	36
4.3.3	Text Cleaning and error correction	.....	36
4.4	Translated Data Pr-Processing	.....	37
4.5	Proposed Methodology for Text Classification	.....	38
4.5.1	Feature Engineering for Hate-Speech Detection	.....	39
4.5.2	Machine Learning Methods for Hate-Speech Detection	.....	40
4.6	Investigating Text Classification for Hate Speech Detection	.....	42
4.6.1	Cross-Lingual Arabic Text Classification Results	.....	43
4.6.2	Arabic Text Classification	.....	46
4.7	Summary and Conclusions	.....	47
4.7.1	Future Directions	.....	48

# List of Figures

- 1.1 Twitter growth in monthly active users for the past decade. . . . . 2
- 1.2 Twitter Trends demonstrating hateful hashtag attacking the Muslim protect group  
 . . . . . 7
- 1.3 Example of Reported hateful content in Twitter. . . . . 8
- 2.1 An Overview Text Classification process . . . . . 17
- 2.2 Deep Neural Networks . . . . . 21
- 2.3 An illustration of optimal separating hyperplane in SVM classification. Top  
 Figure is used for learning the the separating decision line, While bottom one is  
 showing how the model is applied using SVM. . . . . 24
- 4.1 Selected examples of annotated comments with labels from our Youtube com-  
 ments dataset. . . . . 34
- 4.2 Selected examples of annotated tweets with labels from Mubarak et al. (2017)  
 dataset. . . . . 35
- 4.3 Sample of Bad Google Translations . . . . . 38
- 4.4 Sample of Good Google Translation . . . . . 38
- 4.5 Summary of the proposed methodology for Investigating Hate speech Detection  
 for Arabic Content. . . . . 41
- 4.6 Labels distribution count for the tweets dataset. . . . . 42
- 4.7 Labels distribution count for the YouTube dataset. . . . . 43
- 4.8 Top ngram appearing in each label of the Youtube dataset ranked by the mean  
 TF-IDF score. . . . . 44
- 4.9 Top ngram appearing in each label of the tweets dataset ranked by the mean TF-IDF  
 score. . . . . 45

# List of Tables

3.1 An Overview of the related work in using Text Classification for Online Hate

	Speech Detection .....	31
4.1	Cross-lingual Text Classification for Youtube Comments DataSet .....	50
4.2	Cross-lingual Text Classification for Tweets DataSet .....	50
4.3	Arabic Text Classification for Youtube Comments DataSet .....	50
4.4	Arabic Text Classification for Twitter Dataset .....	51

# Chapter 1

## Introduction

Massive amounts of data are being generated by social media platforms that is currently available on the Web. The majority of this data comes in short (i.e. tweets, comments reviews) or semi-long textual format (i.e. forum posts, blogs articles). This exponential increase in the available social content can be attributed to the The availability of free and easy-to-use social media platforms such as YouTube, Facebook and Twitter.

With this massive amounts of content, there is an increasing demand to control the integrity and the quality of this content in social media platforms. Twitter, for example, is a major social media platform allowing users to express their own views in short text format. Twitter runs its business by monetizing (running ads) over the uploaded user-generated content. Twitter reported that the number Monthly Active Users (MAU) went up to over 330 million (Twitter inc, 2019) in 2018. These users generate between one to thousands tweets every month and they use these tweets to share a daily activity, a news item, or to freely express their opinion and views on a certain topic.

In the past decade twitter witnessed over 400% increase in terms of monthly active users as shown in Figure 1.1. However, out of the 330 million monthly active usage, it has been reported that around one third of this content is available for monetization (around 130 Million) (Twitter inc, 2019) . The report also indicated that over %70 of the active usage is coming from nonnative English speaking countries. The reason behind the gap in monetised usage and actual

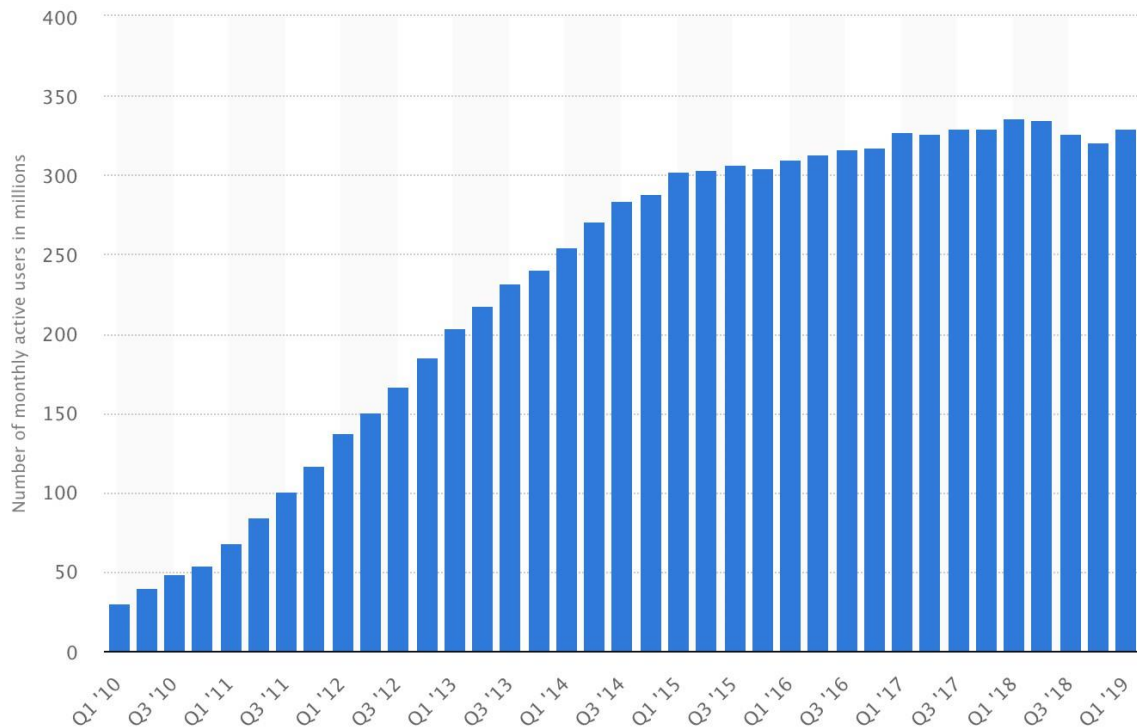


Figure 1.1: Twitter growth in monthly active users for the past decade.

active usage, is the content quality issues. Advertisers are usually concerned about their brand safety, and demand twitter to prevent their ads being run on violence or non-family safe content. Twitter, as well as most social media platforms, have policy and usage guidelines to remove this content.

Twitter <sup>1</sup>, as well as most social media platforms, has some content guidelines to prevent the following type of content:

- Adult Sexual Content: Twitter and many social platforms prohibits users from publishing non-family safe content.
- Copyright : Sharing stolen and copyright infringing content is not allowed on most social media platforms.

---

<sup>1</sup> <https://help.twitter.com/en/rules-and-policies/twitter-rules>

- Counterfeit Goods: The sale or promotion of counterfeit goods.
- Drug and Drugs Paraphernalia: Promoting or sale drug and drugs paraphernalia.
- Hateful Content or Racist : Twitter platform forbids the promotion of hateful speech or content.
- Illegal Products and Services: Twitter forbids the sale or promotion of illegal products, activities, or services.
- Inappropriate Content: Twitter platform prohibits the promotion of inappropriate content.
- Malware and Software Downloads: sharing malware software products.

Tech companies, such as Google and Facebook, have large teams of research scientists, engineers, Data analysts and policy experts dedicated to utilise state-of-the-art research methods to detect and control content quality. Out of these content issues, the detection and the classification of hateful content is arguably considered the most challenging tasks. The problem is even more challenging when it to comes to under-resourced less studied languages such as Arabic.

The aim of the this thesis is to investigate the task of identifying hateful content in Arabic user-generated content, mainly text using state-of-the-art text classification techniques. The details of our research problem are explained in the next section.

## 1.1 Online Hate Speech

Hate-speech can be identified as the content intended to demean and brutalise others, or the use of cruel and derogatory language on the basis of real or alleged membership in a social group (Karst and Mahoney, 2000).



The main challenge in the task of detecting hate-speech is the definition of it. The borderline discrimination between freedom of speech and hateful is arguably the most debated topics worldwide. Even when it comes to laws and regulations, countries have a widely varying law enforcement when it comes to regulating hate-speech.

There are over 30 hate-speech guidelines for law enforcement defined by different countries in the world (Wikipedia, 2019). This indicates that subjectivity hate speech is varied across regions, in which what might be considered to be hateful content in one country, might be considered freedom in another.

Across these different definitions, the scope of Hate-speech is defined as any content or material that attacks a person or a group of persons on the basis of protected attributes. The protected attributes can also vary but the most common, and least arguable ones, are the following

:

- Sex : Female or male or any sexual orientation or gender identity, any content directly attacking this attribute is widely considered hate-speech.
- Human Race and national origin : attacking grouping of humans based on shared physical or social qualities. Examples can be Asian, or African groups.
- Disability : group who suffers from a certain long-term disability are also protected and any content aims at making fun of or criticising this group is widely considered as hate speech content.

The type of hate-speech, we consider in this thesis is *online hate-speech*, in which the hateful content appears online in any of the user-generated social media platforms. It has been suggested that current social media sites bring both challenges and opportunities, and it needs complex balancing between principles and fundamental rights, with the defence of

human dignity and freedom of expression (Gagliardone et al., 2015). The reason why companies such as Twitter, YouTube (Youtube, 2017) and Facebook (Facebook video, 2019) have a certain guidelines and rules to control content being published by the users of their platform.

As explained before, Twitter has reported that over 79% of its monthly usage is being written in non-English languages(Twitter inc, 2019). Therefore, the task of identifying and removing hateful content is rather global one. This indicates that no hateful content is allowed to appear regardless of the language used, or the location where it is presented.

News sites and social media platforms are required to develop methods and systems to handle hate in their platforms. Such systems does not only require deeply technical knowledge on how to process and understand these languages but also a knowledge of the culture, religion and the current political situation and current conflicts in the region.

Taking Arabic online content as an example, with an estimated average of over 400 Million speakers in different nations <sup>2</sup>, Arabic language has the largest growth in terms of users in the previous decade with an estimated 2500% growth.(Internetworldstats.com, 2017). Arabic social media users, certainly in the middle east region which has been witnessing the biggest conflicts, have been utilising social media sites as freedom of speech platforms to freely express their views and insights about these geopolitical conflicts. However, this comes as responsibility of the social media platform to prevent any hateful discussion that can not only negatively impact the user experience but also the business growth of advertising this content as explained in the previous section. Figure 1.2 shows how users may indeed use hateful, and rather disturbing hashtags that can make it to the top trends of twitter in certain countries.

Social media platforms usually use two techniques to handle speech as follows.

---

<sup>2</sup> <https://en.wikipedia.org/wiki/Arabic>

- **Reactive Approach:** This is done by allowing users to report the content as showing in Figure 1.2. This report usually arrives to the inbox queue of human reviewer, who is knowledgeable of the twitter policy, and the language and most likely the region, will manually review the reported content and remove it if it is violating the policy.
- **Proactive Approach :** This approach is by utilising text classification systems that is designed to identify and take action against hateful content.

The Proactive approach is deemed to be more scalable and effective one. Since it relies on the machine to identify and take down the hateful content without any bias. However, building a system to automatically classify and take down hateful content, specially for non-English content can be very complex and non-trivial, calling for further research in this area.

## 1.2 Research Problem of Arabic Hate-Speech Detection


To support growth of social media users such as the Arabic speakers, social media sites are increasingly investing in building systems that classify hateful content and take action against it. Text classification has been long suggested to be the most effective technique to handle hate speech content. Hate speech detection systems combines techniques from natural language processing (NLP), to analyse and understand text, and Machine Learning (ML) to classify text as accurately as possible.

Building the NLP module is arguably the most complicated one, since it requires certain methods to effectively understand the language. This module has multiple core tasks such as parsing, morphological analysis, generation, tokenisation, part-of-speech (POS) tagging. For languages with little research focus, or limited technologies in these tasks, the NLP module is even more complicated.

Arabic language for example, has very rich and complex morphology, is considered to be a highly inflected and derived language. Arabic language has a varying format and dialects.

Modern Standard Arabic (MSA) is the official language of the Arab countries (Middle East) and is the main language of the education, media, and culture. In the past decade, NLP for MSA has been heavily studied in the literature making huge progress on how to semantically understand and analyse Arabic MSA text for several tasks (Farghaly and Shaalan, 2009; Habash, 2010) .

Trends · Worldwide · change

#TheBestSound  Promoted

#blamethemuslims

#babymakingsong

#jlsquestionandansaaa

RestartNaEliana

ALEX LOVES JACK

Teen Titans

White Chicks

Luciana Gimenez

Millicent

Figure 1.2: Twitter Trends demonstrating hateful hashtag attacking the Muslim protect group



### An update on your report

Thanks again for letting us know. Our investigation found this account violated the [Twitter Rules](#):



**Amy Mek**  
@AmyMek

- Violating our rules against [hateful conduct](#).

We appreciate your help in improving everyone's experience on Twitter. You can learn more about reporting abusive behavior [here](#).

Figure 1.3: Example of Reported hateful content in Twitter.

However, Arabic MSA is not the native language of any country. In twitter and other social media platforms, users tend to use their own formal language known as Arabic dialect. Arabic dialect varies based on the geographical location of the Arabic speakers (Habash, 2010; Diab and Habash, 2012). The following are few examples of the most common ones in social media.

- Egyptian Arabic which is the dialects of the Nile valley: Sudan and Egypt.
- Levantine Arabic which is the dialects of Jordan, Palestine, Lebanon, Syria.

- Gulf Arabic which is the dialects of Kuwait, Oman, United Arab Emirates(UAE), Bahrain, Saudi Arabia and Qatar.
- Maghrebi arabic which is the dialects of Algeria , Morocco, Libya, Tunisia, Mauritania.

For the task of identifying hate speech for Arabic content in social media, the proposed system must be able to effectively process and understand formal Arabic and differentiate between the dialects. However, the current research and technologies in processing Arabic dialect is very limited. This indicates that problem of identifying hate speech for certain dialect remains *unsolved* for many social platforms. Many news outlets and entities or even video channels in Youtube, disable comments completely as result to prevent hateful comments from appearing in their websites or channels. While such solution can effectively work, it can negatively impact the freedom of speech, and hinder the user engagement rates.

In this thesis, we consider the problem of detecting Arabic hate-speech for one of the most dominant dialects in social media which are the Egyptian and Gulf saudi dialects. The challenges for this task can be summarised with the help of the following points :

- The current little knowledge and limited research on how to effectively and semantically process dialectal Arabic.
- The noise and inconsistency associated with informal language being used in social media. Quality and the format of the written text can vary from user to another.
- The complexity of the hate speech definition in the Arabic region with a high-level of subjectivity in what considered to be hateful.

### 1.3 Scope of the Work

Our research in this thesis seek to build a novel framework to detect hate-speech for Arabic language in Twitter and YouTube. We deal with an informal real-world comments collection

harvested from social media platform with specific focus in identifying approaches to handle hate-speech.

We investigate the utilisation of dialect machine translation to navigate the problem of the language complexity of the Arabic dialects. A typical text classification system for hate-speech detection may include modules of processing dialects and having three modules as follows :

- Text Analysis : This is simply checking if the tweet or the comment contains words from a predefined list of unsuitable or explicit hateful words.
- Natural Language Processing (NLP): This is where the system is required to perform more sophisticated analysis of the language morphology to extract certain semantic features to identify hate speech.
- Machine Learning processing (ML) : This is where the system is required to utilise the engineered features and predict whether the content is hateful and worth removing.

In this thesis, our focus on the ML module and text analysis module. The NLP module is replaced with machine translation and an evaluation of such replacement is extensively studied in this thesis. In other words, we do not intend to implement new approach or method for processing Arabic dialect. We leave studying this part of the system for future research and directions in this task.

## 1.4 Research Questions

Our proposed research questions in this thesis are indicated as follows.

1. Research Question 1 : Can Cross-lingual English-to-Arabic text classification be beneficial in detecting online hate speech? how does this approach compare the existing Arabic text classification for Hate speech?



2. Research Question 2 : Which feature engineering is most effective Arabic text classification in Detecting online Hate speech for Arabic.
3. Research Question 3 : Which feature engineering is most effective for Cross-Lingual Textclassification in Detecting Arabic online Hate speech

## 1.5 Thesis Structure

This next chapters of this thesis is going to cover the following.

- Chapter 2 provides a brief background on fundamental Text Classification concepts, processes, techniques and evaluation. We provide an overview of some of the widely used Machine learning models in text classification such as Naive Bayes, Support Vector Machines and others that are used in this thesis.
- Chapter 3 is used to review literature work that is relevant to this research. The first part reviews existing work in hate-speech detecting in social media, while the second part presents a survey in of previous cross-lingual text classification methods.
- Chapter 4 of this thesis is used to explain the experimental settings, the data collections, and the type of text processing and cleaning we used in our work. Chapter 4 also present the experiments and the result obtained from running the cross-lingual Arabic text classification on our test collections. The last section of this chapter, Chapter 4, concludes the work and provide directions for future work in the field of Arabic hate-speech detection.

# Chapter 2

## Background

This chapter is dedicated to describe general background about the contribution of this thesis. The chapter starts with a summary of the basic machine learning methods that are relevant to this work, and then introduces more related methods and applications such as Text classification and Cross Language text classification (CLTC).

### 2.1 Text Classification

Text classification is a fundamental task in natural language processing that used to automatically assign assign tags or categories (i.e classes, labels) to textual content. The main goal of text classification is to enable users to obtain valuable information and insights from textual assets. In particular, Text classification employs both Machine Learning (ML) and Natural Language Processing (NLP) techniques to discover patterns and classify from the different types of the available text Sebastiani (2005).

With the current growth in social media content, text classification is assumed to have a high business potential value due to its application in making use of this data Korde and Mahender (2012). Therefore, research in text classification is gaining more value recently with demand not only to scale across the web but also to understand multiple variations of formal and informal text on social media.

## 2.2 Machine Learning

Artificial Intelligence is the science of making smart machines that can act and think like human. Machine learning (ML) is an approach utilised to teach machine how to be smart. ML approaches usually divided into four categories as follows.

- *Supervised Machine Learning* : An algorithm is developed to uses labelled data to learn the relationships between input and output, and apply this learnings to predict future input.
- *Unsupervised Machine Learning*: The algorithm is given only input data and the goal is to find a hidden structure from unlabelled data. The training data are divided into quantitative or qualitative properties, which are used as features during the learning process.
- *Semi-Supervised Machine Learning*: This approach comes in the middle between unsupervised learning and supervised learning, labelled and unlabelled data are used for training, usually huge amount of labelled data and a small amount of labeled data, Semisupoervised learning used when the labelled data need relevant resources to learn from it and train over it. (Bishop, 2006)
- *Reinforcement Learning (RL)*: is another emerging ML technique, where the learning is conducted by interacting with the environment. The algorithm learns errors, rewards by utilising a pre-defined reward function and take actions that are expected to maximize rewards. The Reinforcement learner should discover which actions will return more rewards by trying them all. Trial and error are the most important distinguishing features of RL algorithms.

For the problem we are tackling in this thesis, we are using a supervised learning problem. The goal is to classify new tweets or comments to which certain classes of hate-speech.

## 2.3 Text processing for Text Classification

As explained before, Text classification uses machine learning to predict labels and extract information. The machine learning process utilises textual features to learn text classification model that is able to classify text into labels.

Text processing is an essential process in text classification and it is used to translates the natural textual document into a machine-learning-ready structure. Text processing enables ML models to extract highly important textual-features to make correct predictions.

As shown in Figure 2.1, Feature extraction is critical to feed the machine with the right features for accurate predictions. These features can be words that are often associated with certain Text classes. For example, assume that there is an app review A written as ( I hate the User Interface of this app) and another review B submitted as ( I love the User Interface of this app. Words such as hate and love can be considered informative features to help us classify the reviews into negative and positive review. Now, there are also other nouns the machine learning model needs to consider such as ( User, Interface, app) and other help words such as ( the, of, this). The processing of both groups during the text pre-processing stage explained in the next section.

### 2.3.1 Text Pre-processing

The objective of preprocessing is to represent all document as a feature vector, to split the text into different words as shown in Figure 2.1. the key preprocessing phase essential for the indexing the documents is Selecting the keyword and this is the feature selection process. This phase is vital in determining the quality of the next phase, that is, the classification task. In text preprocessing phase, its important to highlight the significant keywords that can help in classifying the document, and ignore the words that do not contribute to distinguishing

among the documents (Srividhya and Anitha, 2010). In order to perform this, the approaches explained in the next sections are considered.

### 2.3.2 Stop Word Removal

Stop words are a part of natural language. The reason for removing stop words is important to make the text lighter and because it is not important for the classifier analysis and removing them improves the accuracy of the classifier. Stop words are mostly useless and unlikely to convey any valuable or unique information about the text. They include articles, pro-nouns, prepositions, and preposition and other that do not represent meaning of the documents. These words are often referred to as stop words (Vijayarani et al., 2015). There are around 400- 500 Stop words in English language. Examples of such words include 'of', 'it', 'she', he, 'the' (Srividhya and Anitha, 2010).

Stemming Stemming the text are used to fetch the stem or root of a word in the text. Stemming change the words back to their original stems. This is done based language-dependent linguistic knowledge of the words that appears in the text. The hypothesis of the stemming argues that words within the same word root or stem mostly define same or quite near concepts in text data. For instant, the words, users, user, using, used, all can be stemmed to the word 'USE'. The goal of stemming is to eliminate several suffixes, to decrease the number of words in the text, also to improve the accurately matching stems and to save memory space and time in order to obtain a better classification result. (Srividhya and Anitha, 2010)

### 2.3.3 Document Indexing and Term Weighting

Document indexing aims to transform the textual content of a document into bag of words that represented as vector. Document indexing contains of selecting the suitable group of keywords ( bag of words) built on the full corpus of documents and giving weights to those keywords for specific document, thus representing each document by a vector of keyword

weights. The weight usually is connected to the term frequency of occurrence in the document and the sum of documents that contains that term. (Srividhya and Anitha, 2010)

Term weighting is another essential step in text processing that is concerned about ranking the words according to their importance. The important of word is modelled using some heuristics such as the term frequency (TF), which is how often the word appears across in the document and across the corpus.

Vector space model (vsm), is the most well-established model for text ranking and representation (Salton et al., 1975). In VSM, documents are represented as vectors, and different terms have varying weights according to their level of topical importance in a text. The term weight is assigned to each term in the collection or corpus to represent how important this term across the documents.

In traditional text classification, there are two main features that influence the weight of a term in a document explained as follows.

- Term Frequency  $tf$  is a weight that represents the availability of any word in documents. The frequency is usually calculated by how often the term appears in the document and in the collection. TF is usually computed by measuring  $1+\log(tf)$  (Buckley et al., 1993; Singhal et al., 1996), rather than taking the raw frequency, to decrease the weight of high frequent terms.
- Inverse Document Frequency ( $idf$ ) While  $tf$  captures how frequency of the term is in the collection, the  $idf$  represents the *uniqueness* of the term in the collection.  $idf$ , is defined by the inverse of the  $df$  (document frequency) so that it gives a higher weight to these terms which are uniquely identify the document topic (i.e have a lower  $df$ ) (Salton and Buckley, 1988). The  $idf$  metric is calculated using  $idf(t)=\log(\frac{N}{df(t)})$ , where  $N$  is the number of documents in the data and  $df(t)$  is the number of documents in which  $t$  occurs (Sparck-Jones, 1973).

Both TF and IDF measures are considered one of the most important features in text classification modelling, often combined multiplied together using one of the TF-IDF formulas (Allan et al., 1995; He and Ounis, 2006). In Text classification, The TF-IDF score of a term is utilised to represent the probability of text belonging to certain class. In the next section we explain the different types of Text classification systems.

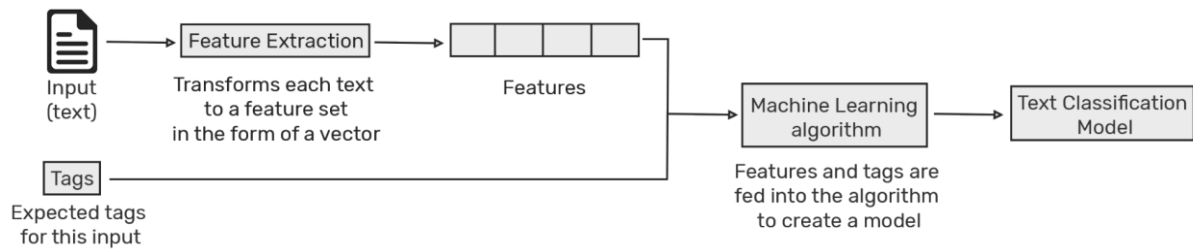


Figure 2.1: An Overview Text Classification process

## 2.4 Text Classification Methods

In this section we will provide an overview of most popular methods to Text classification.

### 2.4.1 Rule-Based Systems

In rule based approaches, the classification is conducted using a group of linguistic rules, each rule has a pattern or antecedent, and a predicted group that takes the formula of *if then* rules.

The basic idea of rule based is when the premises of a rule (i.e. *the if statement*) is satisfied by the data, the system asserts the conclusions of the rule (*the then statement*) as true to extract knowledge or information from the textual data.

Generally speaking, these rules are planned and constructed directly from domain experts in certain language (i.e. linguists, translators). Due to the rapid increase in the amount of textual online content available become increasingly popular towards the design of rule-based systems. Unsupervised Machine learning techniques can used to induce linguistic rules directly from text.

These linguistic rules, are often very helpful and can be used for different tasks such as association rules, classification rules and regression rules (Jain and Srivastava, 2013). Other benefit of the Rule based technique is human readable, and can be easily upgraded or updated over time for small systems.

Rule based systems have also disadvantages. For examples, these techniques often require a knowledge of the language and text domain. Rule based systems also be very time consuming, especially if we are constructing linguistics rules for a large complex classification system that needs extensive analysis and testing. Rule based systems are hard to maintain and if you dont scale will result as decrease accuracy of the existing ones (Monkey Learn, 2019).

#### 2.4.2 Machine learning Based Systems

Machine Learning (ML) classification systems automatically build a classifier model by training on a group of pre-classified (labelled) documents. During the training process, an ML model is trained to learn the diverse associations between textual content (words, sentences, paragraphs) and that specific document class (Khan et al., 2010). To learn such a model, textual feature extraction is employed to convert each textual unit into a numerical representation (i.e. a vector of terms weights).

One of the most commonly used feature extraction approaches to text classification, is the bag-of-words (POW) approach, where the frequency of a term in a particular index of terms extracted from the whole training corpus represented by a vector. For instance, if we have declare our word index to have the arguments [This, was, awful, and, bad, experience], and we need to vectorize this piece of text [This was bad], will be represented as a vector representation of that text as[1, 1, 0, 0, 0, 1, 0]. Then the ML system will train the classifier with data that comprises of pairs of feature groups and each associated class or label [e.g. politics, comedy] to create a classification model as shown in 2.1.



## 2.5 Machine Learning Algorithms

As explained before, ML have been heavily utilised in the Text classification field. In this section, we will give an brief of machine learning main methods used in this thesis, for more extensive review, we refer the reader to these excellent surveys (Berry and Castellanos, 2004; Aggarwal and Zhai, 2012)

### 2.5.1 Naive Bayes Methods

Naive Bayes classifiers are popular and simple probabilistic models for text classification. Naive Bayes is built on implementing Bayes theorem with robust independence assumptions across the features. These independence assumptions mark the features order irrelevant, therefore that the present of single feature will not affect present of other features in classification process (Khan et al., 2010). These assumptions can contribute to the efficient of computation of Bayesian classification process but limits its applicability (Brucher et al., 2002).”

The process of applying Naive Bayes algorithm to text classification can be fairly simple, summarised as below.

- Documents are represented using bag-of-word (POW) method, where each document  $X$  is transformed into a set of (word, a frequency of word) pairs.
- During the training process, a probabilistic model  $P(X|Y = A)$  is constructed for documents in class  $A$ .
- Once the model is learned, in production, to classify a document, the model selects the To classify, select class  $Y$  which is most likely to generate  $X$  as shown in equation 2.1. The naive assumptions are that order of the words in document  $X$  makes no difference but repetitions of words do.

$$\hat{y} = \underset{y}{\operatorname{argmax}} P(X|y) * P(y) \quad (2.1)$$

Nave Bayes text classifier has been extensively used in research because of its simplicity in both the training and classifying process. While it can be considered less accurate over other approaches, many researchers suggested that it is effective enough to for text classification in several domains, since it requires a relatively small amount of training data to estimate the parameters necessary for classification (Ting et al., 2011).

## 2.5.2 Support Vector Machine Methods

Support vector machines are based on the Structural Risk Minimisation principle constructed from the computational learning theory. The awareness of this principle is mainly to discover a hypothesis to guarantee a minimised true error during the classification process.

SVM text classification typically require both negative and positive training samples to learn a decision line that can optimally separate the negative from the positive data in a dimensional space. This decision line is often referred to as is *hyperplane* (Khan et al., 2010; Brucher et al., 2002). An illustration of optimal separating hyperplane is demonstrated in Figure 2.3

SVM is considered to be highly effective method for text classification. The key advantage of SVM is that it is able to classify documents with high dimensional input space, and ignore most of the irrelevant features. This is very relevant for text classification where sparse textual features is a major efficiency issue . While the main disadvantage of the SVM is their comparatively complex training process as it requires memory high consumption to learn the decision lines, specially for large scale classification tasks, where multiple classes are available Khan et al. (2010); Aggarwal and Zhai (2012).

### 2.5.3 Random Forest

Random forest or Random forrest or Randon forest is classification method with other task for making decision using machine learning, it contains of a huge number of different decision trees that run as an ensemble which is divide-and-conquer method used to make the performance better. the output of the run multiple decision trees is individual decision tree(Segal, 2004). The first algorithm credited and created by Tin Kam Ho (Wikipedia contributors, 2019) via the random subspace method, and the trademark is owned and developed by Leo Breiman (Wikipedia contributors, 2019)

Random forest increase and maintain the accuracy for small and large dataset, the process of averaging the results in random forest of different decision trees supports the model to overcome the known problem of overfitting, which is one of the advantage of random forest, it can be used without preparing the input data (prepossessing) .The disadvantage of random forest is time-consuming, and the complexity of the model (Statnikov et al., 2008).

### 2.5.4 Deep Learning Methods

Deep Learning (deep neural networks) methods are new emigrating machine learning techniques constructed based on artificial neural networks. The word "deep" refers to the number of learning layers that are implemented to transform the data. Deep Neural Networks (DNN) is composed of multiple processing layers, each layer has several connected neurons that used to extract high-level features from the raw input.

For example, in Text processing and classification, lower layers may identify nouns or verbs of the sentence, while higher layers may identify the concepts or topics that are relevant and beneficial for human consumption.

Over the last decade, DNN methods achieved a revolutionary Machine learning performance in multiple applications of in visual object recognition, Text Classification, speech recognition and many other domains. It is also currently considered the state-of-the-art

Machine learning method for many research areas including health, oil and gas and other industries.

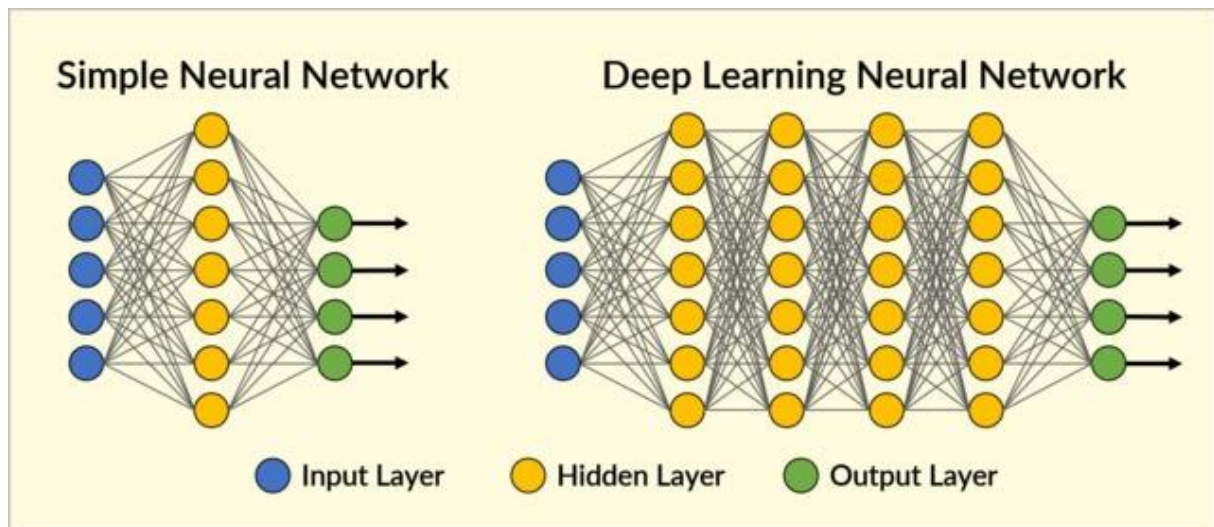


Figure 2.2: Deep Neural Networks

Deep neural systems are basically feed-forward systems in which information is moved from the underlying info layer to the last yield layer as appeared in Figure 2.2. At the first stage, the neurons are created to represent some features, and random wights are assigned to each one, the hidden layer (Shown in Figure 2.2) is used to as middle mathematical function to combine the features neurons and their weights and return an output. The hidden layer can be a linear or non-linear function. During the training and validation process, *the back propagation algorithm* is utilised to go backward to through the network and adjust the model parameters according to the training data for better fitting Yu et al. (2008). That way DNN can find a more suitable training parameters to fully process the data. LeCun et al. (2015). During the back propagation process, DNN also utilise techniques Empirical Risk Minimisation (ERM) and dropout methods to balance between over-fitting and under fitting. The optimisation techniques of DNN is an active research area, interested readers can refer to Labach et al. (2019), for more detailed review on this topic.

The key advantage using of DNN in text classification tasks is the ability to handle multidimensional textual features, where massive amount of semantic features can be

extracted from textual documents. In particular, DNN utilise multi-layer approach, explained before, to effectively deal with noise and the sparsity of text and extract semantic features. DNN enables the use of a distributed representation of the words such as *Word embedding* where terms or expressions from the vocabulary are mapped to vectors of genuine numbers utilizing multi layer DNN representation. Two very popular word-based DNN architectures were proposed by Mikolov et al. (2013) as follows.

- The Continuous Bag of Words (CBOW) architecture : This DNN-based representation predicts the current word based on the surrounding words.
- The Skip-gram model architecture : This DNN-based representation predicts surrounding words given the current word.

The use of these word embedding approaches in text classification allows the incorporation of different linguistic annotations and external knowledge much better than the previously machine learning-based classification models (Mikolov et al., 2013; LeCun et al., 2015; Khan et al., 2010)..

The disadvantage of using DNN is computing cost is very high and consumes high physical memory usage and CPU usage, sometimes require GPU or TPU size processing units which may not be always available for small or low-budget tasks. Another drawback is that DNN decisions are extremely difficult to trace or debug due the the model complexity. This may negatively affect the acceptance of these methods in many real-life applications.

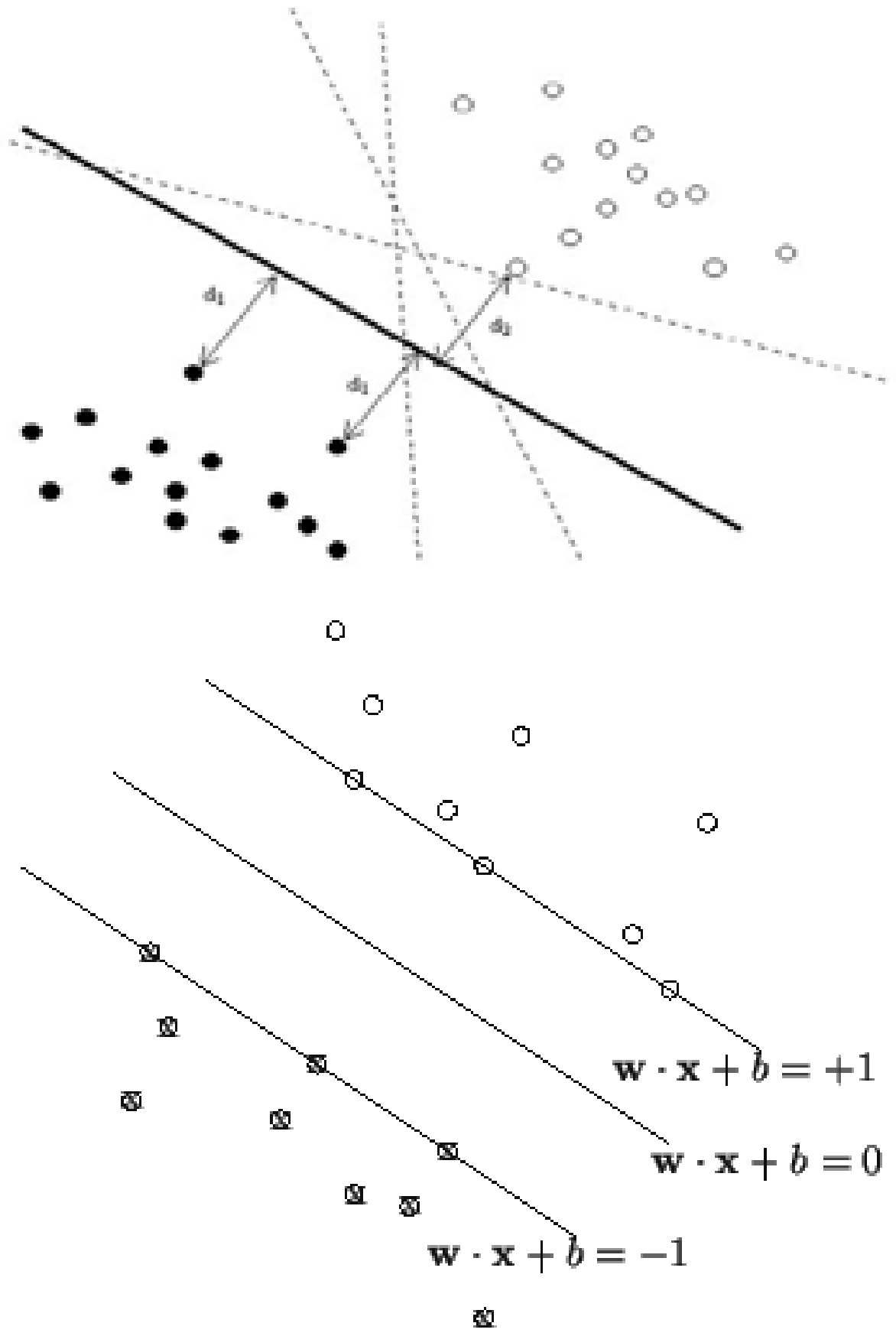


Figure 2.3: An illustration of optimal separating hyperplane in SVM classification. Top Figure

24 is used for learning the the separating decision line, While bottom one is showing how the model is applied using SVM.

## 2.6 Text Classification Evaluation

Text classification systems are built using human-labelled corpus, that is a set of documents and labels or classes) prepared for specific task. This the training corpus is often randomly divided using Cross-validation methods into equal-length sets of examples (e.g. 4 sets with 25% of the data). For each set, a text classifier is trained using the remaining samples (e.g. 75% of the samples). Next, the classifiers make predictions on their respective sets, and the final results are compared against the human-selected labels. This evaluation allow the prediction to be judged by counting the following measures.

- Correct prediction : True positive (TP) and True negative (TN).
- Wrong prediction : false positives, false negatives (FP and FN).

After getting these metrics, Text classifiers are evaluated using the popular machine learning metrics to test how well a classifier works, these metrics are explained as follows.

- Accuracy: the percentage of texts that were correctly classified.
- Precision: the percentage of correct predictions out of the total number of examples that it predicted for a given class.
- Recall: the percentage of correct predictions for a given class out of the total number of examples it should have predicted for that given class.
- F1 score : the harmonic mean of precision and recall metrics.

## 2.7 Summary

This chapter presented an overview of Text classification methods and techniques. It also introduced the main machine learning models in Text classification such as the Naive Bayes and Support Vector Machines, as well as the latest methods in utilising Deep learning for text classification.

In the next chapter we present a literature review of the existing work in online hate-speech which are also related to this research work.



# Chapter 3

## Related Work

This chapter reviews existing work that related to our research thesis. In the next sections, we provide brief review of the related work in online hate speech and cross-lingual text classification. In the last section, We explain our proposed research methodology and experiments in this thesis.

### 3.1 Detecting Hate Speech for Social Media

The Following chapter intends to outline past work on applying content classification systems in identifying the hate speech available online.

As clarified in chapter one, hate speech can be characterised as any hate speech that objectifies a particular bunch characteristics, for example, ethnic inception,sex , religion, or sexual direction. In the following section we review couple of papers that are closely related to this work. For more extensive , we refer readers to this excellent survey conducted by Schmidt and Wiegand (2017)

The research published by Greevy and Smeaton (2004) implemented a text classification system that is able to classify racism web-content on the World Wide Web. Greevy and Smeaton (2004) proposed a methodology using bigram and Bag-of-word technique feature selection to train a Support Vector Machines (SVM) models to classify these web-content which may identified as hate speech or racist for specific people or sweat. They concluded that the polynomial kernel is the best function for the bag of word representation and reached the best precision and

recall.

The research presented in (Burnap and Williams, 2014) implemented a supervised text classification system that can identify hate speech in tweets. They proposed a meta-classifier that combine many classifiers which are: probabilistic, spatial-based and rule-based. Tweets were collected during time window around a particular controversial event like the 9/11 event. The dataset has around 450,000 tweets. The authors showed that using meta-classification techniques to detect hate-speech on twitter can reach a very high accuracy of 0.95.

Another relevant research work conducted by Warner and Hirschberg (2012) to implemented a supervised text classification system that is able to find anti-semitic speech user posts in social online clusters. They discussed diverse type of hate speech and the current problems that they make for online businesses. Their proposed technique is to use template-based plan to generate features from the corpus for SVM classification . (Warner and Hirschberg, 2012) reported that using a template-based plan is much more effective (they measure the performance In terms of f-measure reported to be 0.63), than bigram, trigram templates for classifying hate speech in online comments and news sections.

Another relevant research done by Kwok and Wang (2013) evaluated a supervised text classification system that can detect racist tweets in contrast to blacks. They demonstrated how racist tweets in contrast to black race can be detected using Naive base classifier, achieving an accuracy of 76% average. One key finding in their study is that they show how tough and subjective this mission is in terms of annotation agreement which as as low as 33% only between the three annotators they hired.

## 3.2 Cross-Lingual Text Classification

Cross-lingual text classification (CLTC) is the mission of classifying text of source language using systems that is originally developed for another target language. The target language (i.e English) usually has more resources and developed systems than a source language (i.e Arabic) to accurately classify text. CLTC is usually conducted by crossing the language barrier between the source language and the target language.

The main challenge in CLTC is to maintain the same accuracy when classifying documents in different languages where labelled training data are not available.

Xu and Yang (2017) presents a innovative method to CLTC that builds on model distillation, which adjusts and covers a framework originally projected for model compression. Using easy probabilistic predictions for the online documents in a label-rich language as the supervisory labels in a similar corpus of documents, Xu and Yang (2017) trains a classifiers successfully for different languages in which labeled training data are not existing. An confrontational feature adaptation technique is also applied in Xu and Yang (2017) through the model training to decrease distribution disparity. Xu and Yang (2017) shown experiments on two benchmark CLTC datasets, giving English as the source language and French, German, Chinese and Japan as the unlabelled target documents languages. The proposed approach in Xu and Yang (2017) had the beneficial or equivalent performance of the other state-of-art approaches.

### 3.3 Summary of previous work

Online hate speech detection has recently gained lots of attention from the research community in machine learning and Natural language processing. This is because the present upsurge of social media content available online which is calling for more controlling over the quality of this content. The task of detecting hateful content has been implemented through many type of available online content such starting with webpages (Greevy and Smeaton, 2004), news stories and online comments (Warner and Hirschberg, 2012), then moving into microblogging and tweets (Kwok and Wang, 2013; Burnap and Williams, 2015; Pak and Paroubek, 2010) Table 3.1 shows summary of the techniques proposed by the previously explained papers.

After providing this review, in the next chapter, provide the experimental settings and evaluation of our work.

Paper	Dataset	Technique	Performance
<i>(Greevy and Smeaton, 2004)</i>	8 datasets of web-pages from PRINCIP Project.	-SVM kernels functions.- Polynomial, radial basis function, sigmoid and Linear. BOW representations.	bigram is the best for optimal precision BOW is greatest representation for maintaining Recall.
<i>(Djuric et al., 2015)</i>	56,280 comments classified as hate speech and 895,456 clean comments generated by 209,776 users	paragraph2vec technique- CBOW (continuous Bag of word) model as an element of paragraph2vec	BOW using TF-IDF and BOW. using Area Under theCurve (AUC) metric.
<i>(Burnap and Williams, 2014)</i>	450,000 tweets.	They merge multiple classifiers ( rule-based, probabilistic, spatial based) into one classifier-	Achieved accuracy of 0.95
<i>(Warner and Hirschberg, 2012)</i>	9,000 text paragraphs	-SVM with a linear kernel function. a 10-fold cross-validation on the labelled dataset.	f-measure reported to be 0.63
<i>(Kwok and Wang, 2013)</i>	24582 tweets	Naive Bayes classifier unigram feature extraction 10-foldcross-validation technique	accuracy of 76% average.
<i>(Pak and Paroubek, 2010)</i>	300000 text posts collected from twitter messages.	-unigrams, bigrams, and trigrams. Naive Bayes technique	81% of accuracy

Table 3.1: An Overview of the related work in using Text Classification for Online Hate Speech Detection

# Chapter 4

## Experimental Setting and Evaluation

In this Chapter, we explain our experiments towards answering our proposed research questions as follows.

- Research Question 1 : Can Cross-lingual English to Arabic text classification be beneficial in detecting online hate speech?
- Research Question 2 : Which feature Engineering is most effective Arabic text classification in Detecting online Hate speech for Arabic.
- Research Question 3 : Which feature processing is most effective for Cross-Lingual Text classification in Detecting Arabic online Hate speech .

To answer these questions, we target two controversial social events, one in Saudi, where we collect YouTube comments on videos, the other one is in Egypt, where we collect tweets and comments. Both collects are explained in the next sections.

### 4.1 Saudi YouTube Comments Data Collection

In an attempt to recreate the infrastructure for tourism as part of the Saudi Vision 2030, the Kingdom of Saudi Arabia, for the first time ever initiated few parties and entertainment fun events in the capital, Riyadh. Although, it was well reserved by many, some local reviewers, however criticised the organisation and some details of these events saying that it crosses many boundaries. Most of these events were either streamed live online or filmed and uploaded to YouTube. We build a python based tool that aggregate comments on videos that

are about Riyadh Seasons in 2019. The tool uses Google Cloud <sup>3</sup>, and colab <sup>4</sup> notebook to run the data aggregation process.

For accessing YouTube comments, we utilise the YouTube Search API <sup>5</sup> to search for collect the most commented videos about the Riyadh events. The python-based tool runs as follows.

1. Activate Google Cloud and Google Colab in python 3 environment for access YouTubeAPI.
2. Call the YouTube Search API to aggregate videos about the Arabic keyword

رأى كركم في السعودية

3. Result is restricted to Arabic language, and Location is Saudi Arabia.
4. The videos are ranked by views, we took the top ranked 1000 results.
5. The script will then take the comments of each video and store in separate row of file.
6. each row has the video ID, and the associated comments. We restricted the number of comments for each videos for 100 comments.
7. We had around 1000 comments, after filtering outs the non-word based comments (emoji,marks etc..) we had around 750 comments that are ready for analysis in our hate speech detection task.

After having this dataset of comments, we use two human annotators, we went through each comment and evaluated select the suitable label out of the following three :

- Natural Label : for any comments that has no offensive or hate words. This can be any comments that are expressing opinion or saying things without hating or offending. Users have the right to express any opinions without censorship or restraint.

---

<sup>3</sup> <https://cloud.google.com>

<sup>4</sup> <https://colab.research.google.com/>

<sup>5</sup> [developers.google.com/youtube/v3/docs/search/list](https://developers.google.com/youtube/v3/docs/search/list)

- Hate Label : for any comment that has abusive or threatening speech or writing that expresses prejudice against anyone. This could be against the organisers of the event or the ones who are appearing in the videos.
- Offensive Label : Any comment that has any offensive swearing communications, i.e (You, son of etc..)

Examples of each label selected from our dataset is showing in Figure 4.1

Comments	TextLabel
! : نطالب بإلغاء موسم الرياض ! انشروها معي !	Neutral
مثل هانول البنات .. عار . على السعوديه واليمن ,مالهن حل..الا تودوهن.على الحد الجنوبي الجبهه	Hate
(لباس المرأة العاري دليل غضب الله عليها لان آدم وحواء عندما غضب الله عليهما نزع عنهما لباسهما و أراهما سوءاتهما )	Hate
((هذه الداشره لاتمثل حتى اهلها انما تمثل نفسها ))	Hate
(إن لك ألا تجوع فيها ولا تعري )	Neutral
(اقترب للناس حسابهم وهم في غفلة معرضون) والله اصبحنا نرى أشياء لا تكاد تصدق انها تحصل في هذه الامه يا أسفي عليك وعلى قلة الحياء تبا	Neutral
(بأخْت هَارُونَ مَا كَانَ أَبُوكِ امْرَأَ سَوْءٍ وَمَا كَانَتْ أُمُّكَ بَيْعِيًا).	Neutral
@حجازية بني الحارث خلاص علا راوسكم.. معاد حد بيخش علا مواقع اباحيه الرياض جمبك 😂😂 استحو وصلت للرقص ونساكم محجبات ه	Neutral
*تركت التتور ويدات بالشوارع العامه تتبرج وترقص وتدور*	Hate
*هذي وسخه ومتأكد ماهي سعوديه*	Offensive
*ودي اشفقها بسلكك الشاحن*	Hate

Figure 4.1: Selected examples of annotated comments with labels from our Youtube comments dataset.

## 4.2 Egyptian Tweets Data Collection

Our second dataset is collected from previous study done by Mubarak et al. (2017), who made available a large corpus of classified user comments that were deleted from a popular Arabic news site due to violations the sites rules and guidelines. The dataset contains 1100 Arabic tweets and comments. These comments and tweets are written in Modern Standard Arabic (MSA) and mostly egyptian dialects. The authors collected the 100 tweets by identifying 10 controversial twitter influence rs (tweeps) according to twitter statistics site called SocialBakers.com. The author then randomly picked 10 tweets that have 10 or more comments/replies. In total, Mubarak et al. (2017) had presented 100 original tweets plus

1,000 comment/reply tweets aggregated from each tweets thread <sup>6</sup>. The author then used CrowdFlower.com (a data annotation platform for crowdsourcing) to judge each tweet by 3 different annotators into Offensive, hate and natural (clean) labels. The authors claimed that the annotators reached a very high average inter-annotator agreement was 85%. Example of these tweets together with their labels is shown in Figure 4.2.

Tweets	Label
مبروك و سامحونا لعجزنا التام. عقبال اللي جوه. اللي بره يا عاجز يا بيزايد على العاجز	Neutral
كلنا بره ومش هنبطل نزايد على العجايز الي جابونا وري	Hate
بدل ما انت قاعد بره كده تعالي ازرع الصحرا	Hate
قذر اتفووو ماتيجى مصر وتورينا نفسك كدا يا جبان	Hate
وهكذا رجال الشو اللي محرومين من عمل برنامج الغريبة انهم بيقلوا المطبليطة وهم كلهم مطبليطة ايضا	Hate
أنت أزاى لبؤة كدة ؟	Offensive

Figure 4.2: Selected examples of annotated tweets with labels from Mubarak et al. (2017) dataset.

## 4.3 Arabic Data Pr-Processing

The aim of this work is provide approaches for detecting hate speech on Arabic social content. To achieve this, we need to deal with the noise associated with social content. The following steps are following in order to prepare our dataset for processing :

### 4.3.1 Representation

Both datasets are represented in CSV format, and using python and panda apis <sup>7</sup>, we were able to transform the data into well structured tables (dataframes).

<sup>6</sup> 4The 1,100 annotated tweets can be downloaded from [http:// alt.qcri.org/hmubarak/offensive/TweetClassification-Summary.xlsx](http://alt.qcri.org/hmubarak/offensive/TweetClassification-Summary.xlsx).

<sup>7</sup> ://pandas.pydata.org/



### 4.3.2 Tokenisation, Stemming, lemmatisation and stop words

Both Youtube comments, and tweets, were tokenised into words using the standard pyarabic library version of tokenization (Zerrouki). After few initial experiments with farsa<sup>8</sup> and other Arabic stemming, we found that, for these social noisy content, it is better not to perform any stemming as this is very informal content and is dialect-based one, currently there is no effective tool for stemming Saudi and Egyptian social language. We also chose to leave the stop words in Arabic for the same reason.

### 4.3.3 Text Cleaning and error correction

To perform text cleaning in our data set, we took the following steps :

- We removed all special characters including emoji from all comments and tweets.
- All numbers were transformed into text using the pyarabic library (Zerrouki).
- To simplify the text for classification, We also used the same library to strip Harakat, Shadda, and tatweel from the text<sup>9</sup>.
- We performed error correction as this is informal social text and contain many spelling errors, we used Ghalatawi, an Arabic AutoCorrect library that is known to be very effective for spelling errors (Zerrouki et al., 2014). Ghalatawi utilises a rule-based method for error correction that work based on two manually implemented methods, a list of words and regular expressions.

## 4.4 Translated Data Pr-Processing

To enable cross lingual classification, we translated each comment and tweet into English. The motivation behind this is that once we have the text in English, its much easier to utilise the advanced techniques for hate speech detection. We used the Google Translate API<sup>10</sup>, that is

---

<sup>8</sup> <http://qatsdemo.cloudapp.net/farasa/>

<sup>9</sup> <https://pypi.org/project/PyArabic/>

<sup>10</sup> <https://cloud.google.com/translate/docs/apis>

based on Google Cloud to translate the Arabic text into English for both datasets. Google Translate is a nonprofit multi-lingual statistical machine translation system, that is implemented by Google. It is available for use as website interface, iOS Android mobile devices. Google Translate service translate more than 100 languages, google claimed over 500 million total users used google translation, with over than 100 billion texts translated every day. The main reason why chose this translation tool over others, is that it has shown to be the most effective tool for translating social content in Arabic (Khwileh et al., 2017). The reason behind this effectiveness is that Google Translation is trained based on parallel dataset that is collected from the Arabic web, which in fact, includes many formal and social content for Arabic.

After conducting the translation, we performed error analysis on the content translated by Google Translate to check the quality before processing. Note that its known to be very difficult to translate, previous research such as (Khwileh et al., 2016; Khwileh and Jones, 2016) showed that Arabic translation quality particularly is more likely to be hindered by translation errors comparing to other languages such as English, French and Italian. In our task, we found that the quality of translation highly varied, some were of excellent quality and has almost 100% accurate translation, some were really bad and changed the whole meaning. Examples of poor and good translations from both data sets are shown in Figure 4.3 and Figure 4.4.

Some examples such as `???? ¼PAJ. KĐ@Qmì'@ YÊK. úÍ@` were completely missed and rather transliterated into (Aly.bld Aharam.nbark) .

While others were 100% accurate such as `???? jJ. ' ÊAK . é®ç JÓ ø @ áÓ`

`ø X IJ . Ê@` was translated into (The girl ever from any region exactly ???).

*This huge variation in quality can be attributed the fact that these comments were written*

Bad Translation	
Arabic	English Translation
*ودي اشنقها بسلكك الشاحن*	* Woody Ashengaha charger Bslkk *
إلى بلد الحرام نبارك	Aly.bld Aharam.nbark
امريكا رح تبليكم في يوم	America Rah Etbiekm on

Figure 4.3: Sample of Bad Google Translations

Good Translation	
Arabic	English Translation
إن عذاب الله شديد. ❤️	The severe punishment of God ❤️
البنات ذي من اي منطقته بالمحيط؟؟؟	The girl ever from any region exactly ???
الحجب من اهل السعودية انهم يوالون سلمان مع كل هذا الفساد ويقولون هم ولاء الامر. ماذا تنتظرون من هذا الزنديق انه سوف يرجع لكم الجاهلية الاولى	The wonder of the people of Saudi Arabia they are loyal to Salman with all this corruption They say the rulers of it. What do you expect from this heretic because he will you ignorance first.

Figure 4.4: Sample of Good Google Translation

*by online users who have varying background, interest and their writing style and quality can be very different.*

## 4.5 Proposed Methodology for Text Classification

To perform these four tasks, we use the text processing and cleaning explained as follows. In this section, we perform our main experiments which can be summarised in these four tasks.

- Task 1 : Using Text Classification to detect online hate-speech from Saudi Arabic YouTube Comments.
- Task 2 : Using Text Classification to detect online hate-speech from Egyptian Arabic

tweets.

- Task 3 : Using Cross-Lingual Text Classification to detect online hate-speech from Saudi Arabic YouTube Comments.
- Task 4 :Using Cross-Lingual Text Classification to detect online hate-speech from Egyptian Arabic tweets
- For Arabic text on Tasks 1 and 2, we use the methods explained in Section 4.3 for data pr-processing including Tokenisation, text cleaning and error correction.
- For English translated text on Tasks 3 and 4, we use The NLTK toolkit for English tokenisation, and lemmatization, and stopwords. NLTK <sup>11</sup> is very well-known leading toolkit for processing human language data, and it has been well-regarded by the NLP community to be the most effective library.

#### 4.5.1 Feature Engineering for Hate-Speech Detection

After processing the data, we utilise multiple feature engineering approaches, these utilises matrix notation to represent documents in bag of words to extract features that are helpful for classification.

- Count Vectors : represents the frequency count (TF) of each term in each document.
- WordLevel TF-IDF Vectors : represents the TF-IDF scores of every single term in different documents of the datasets.
- N-Gram TF-IDF Vectors (phrase-based) : represents the TF-IDF scores of n terms in different documents of the datasets. <sup>12</sup>.

---

<sup>11</sup> <https://www.nltk.org/>

<sup>12</sup> We set the ngram up to have phrases of 2 and 3 terms

- CharLevel TF-IDF Vectors : calculate the Tf-idf scores of character level n-grams in the corpus.

All featuring processing methods are based on calculating both calculating term frequency and TF-IDF representation explained in Chapter 2, Section 2.

#### 4.5.2 Machine Learning Methods for Hate-Speech Detection

After the feature extraction process, to perform the actual classification, since we have labelled datasets, we use three different supervised machine learning methods as follows.

- Naive Bayes Classification (Li et al., 2018).
- Logistic Linear Regression (Abadeh et al., 2015).
- Support Vector Machine (SVM) (Huang et al., 2018).
- Random Forest Classification (Segal, 2004).

These are standard and well-known methods which were explained details previously in Chapter 2. These methods were implemented in our four tasks with help of scikit-learn library<sup>13</sup>, which is standard library for using machine learning algorithms in Python.

Finally for our evaluation, we use three main metrics Accuracy, Recall and F1, which were also explained in Chapter 2, Section 2.6.

Figure 4.5 summarise the flow we followed to design our proposed experiments.

---

<sup>13</sup> <https://scikit-learn.org/stable/>

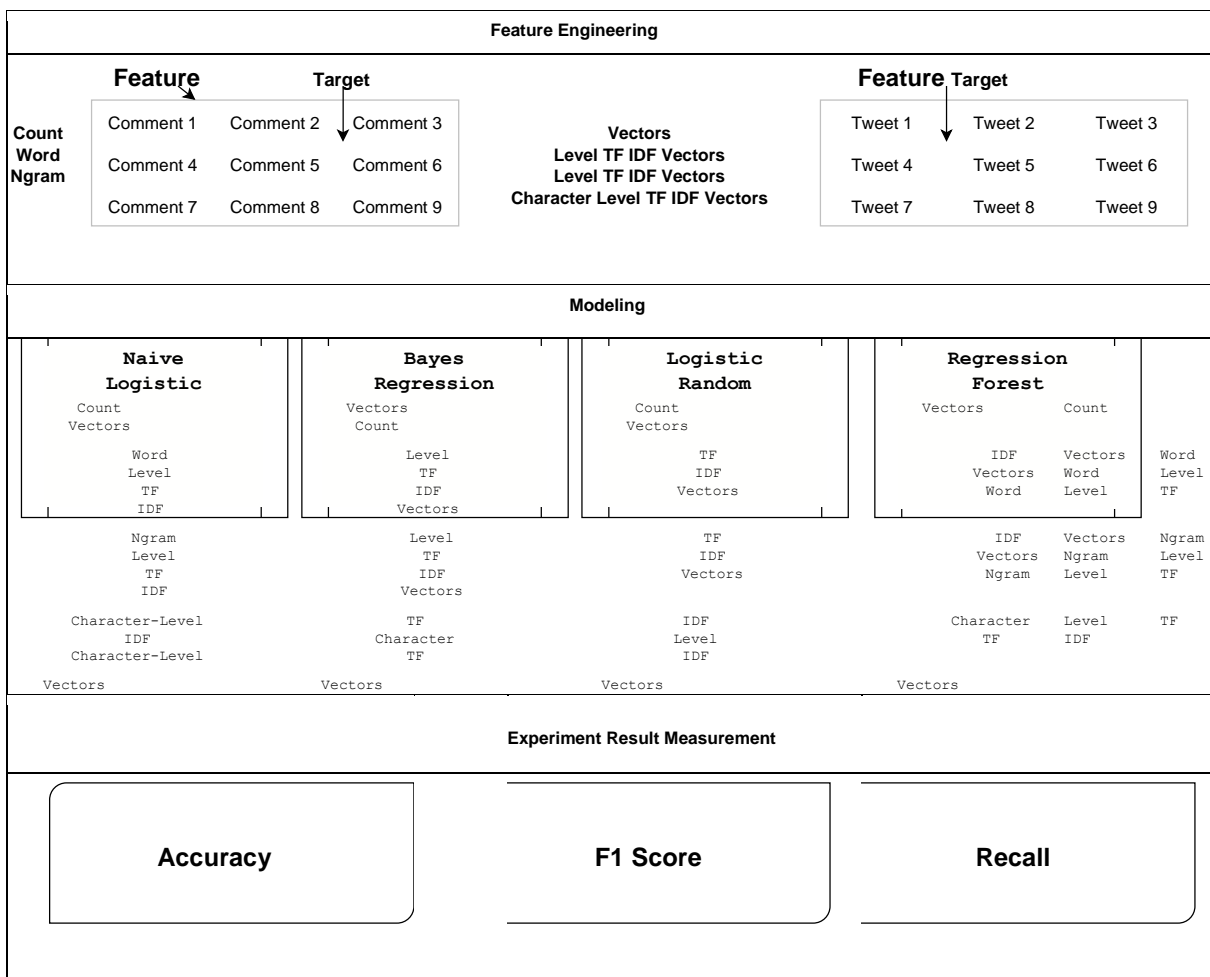
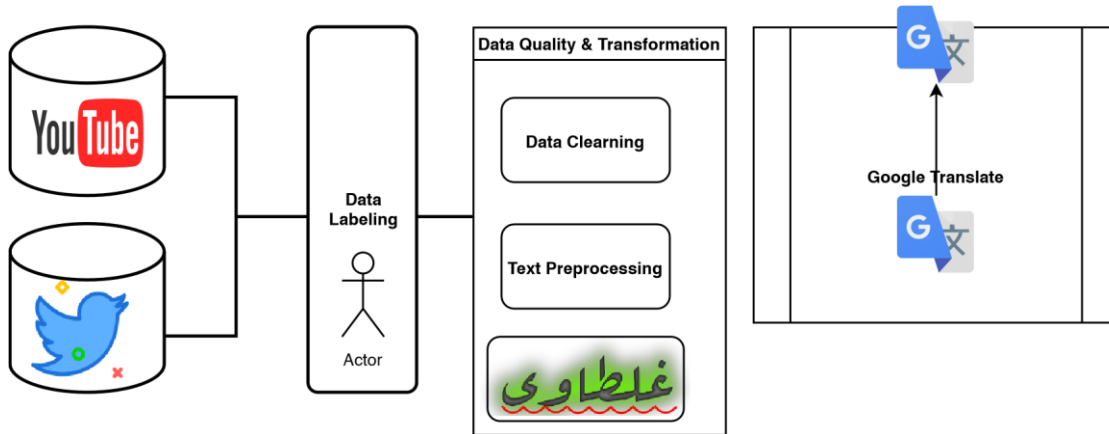


Figure 4.5: Summary of the proposed methodology for Investigating Hate speech Detection for Arabic Content.

## 4.6 Investigating Text Classification for Hate Speech Detection

We perform the proposed methodology explained in the previous section on both datasets. Before running the classification, we investigate both datasets to understand the distribution of labels and the top phrases appearing each dataset for each label.

Figure 4.6 shows the labels distribution we want to predict for the Egyptain tweets dataset, while Figure 4.7 shows the labels for the Saudi Youtube comments. To perform our experiment, we split the datasets using 10 fold cross-validation methods into training and testing datasets, each iteration has split of 80% for training and 20% for testing. We also use the standard implementation of cross validation by scikit-learn library.

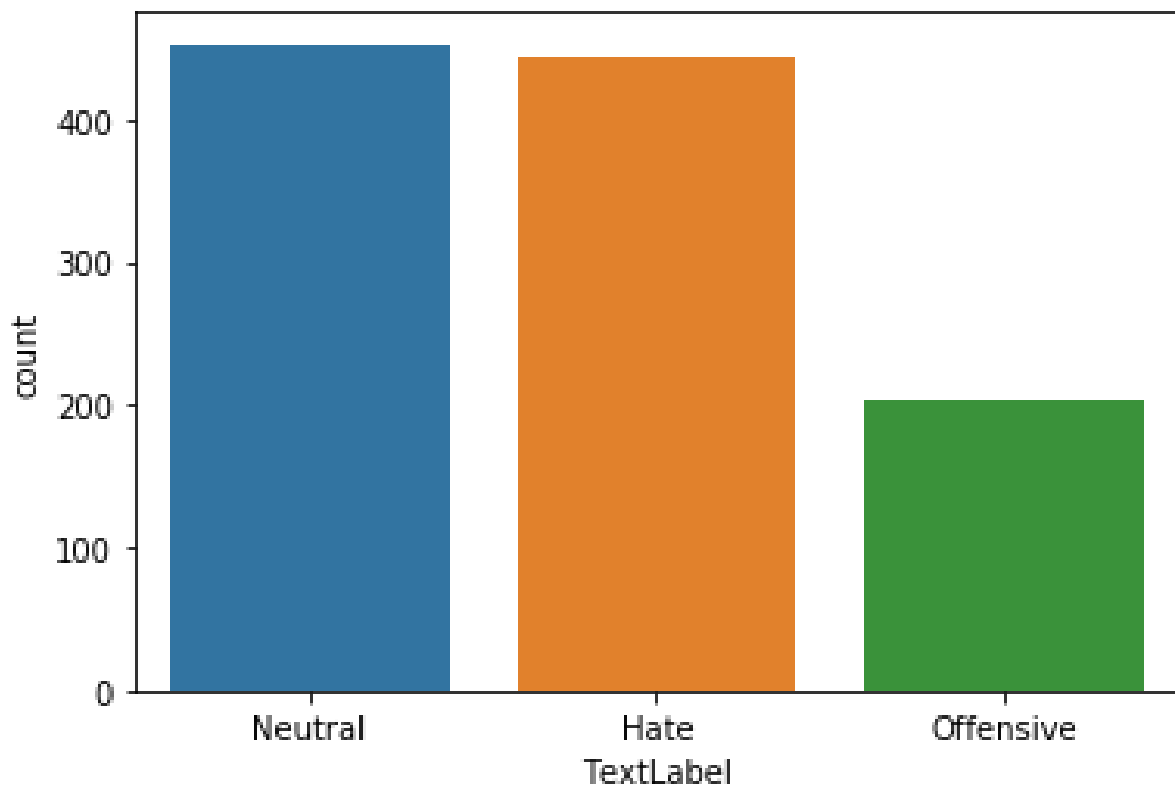


Figure 4.6: Labels distribution count for the tweets dataset.

To show the most common terms in both datasets grouped by label, we use the calculate the mean TF-IDF scores for each phrase in the datasets. We choose the ngram to be 2 and 3 to

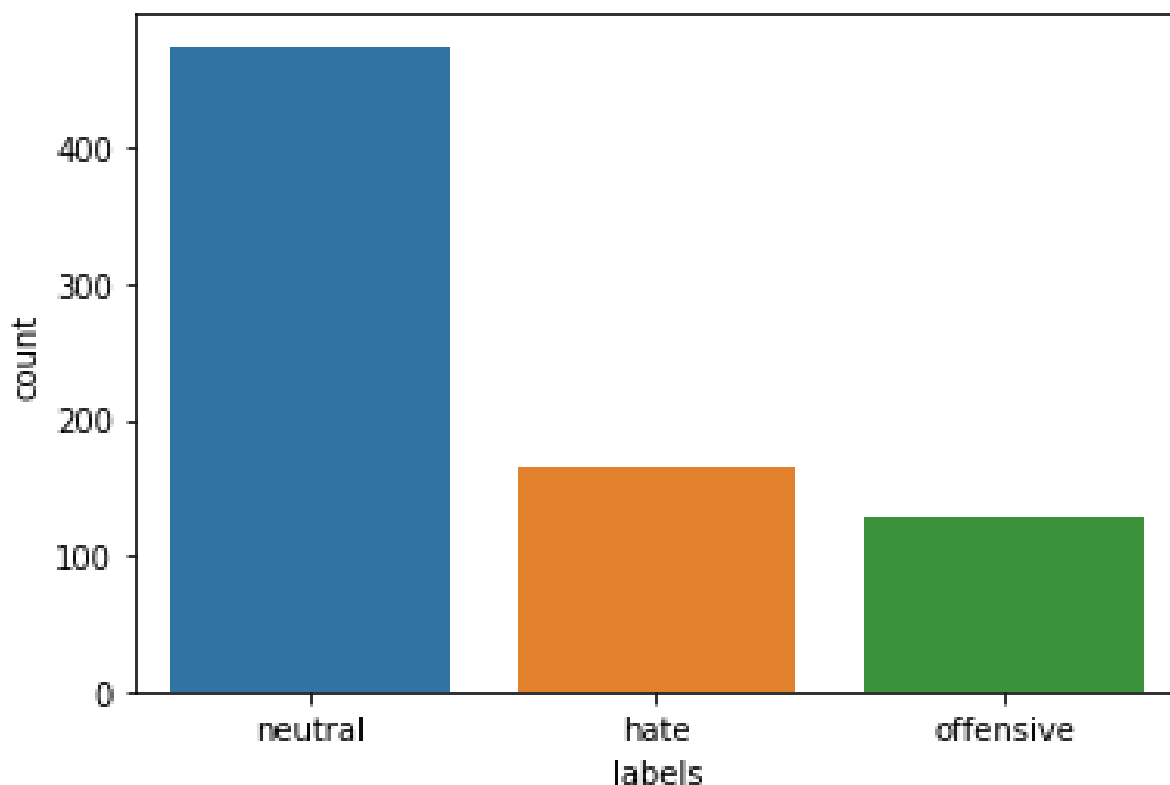


Figure 4.7: Labels distribution count for the YouTube dataset.

show meaningful words to understand the trends in our datasets. We used the TF-IDF vectoriser implemented by scikit learn to learn the TF-IDF scores of the ngrams.

Figure 4.8 and Figure 4.9 shows the top phrases appearing in both datasets. Both figures are showing some interesting insights. We can see that the natural comments are mostly religious comments reciting verses from the holy Quran or trying to express their view without hurting anyone. While Hate are composed of mostly named entities, public figures and races which were mostly attacking the women appearing in the videos. Offensive labels can have a mix of insult and hate-related words from local dialect of Saudi and Egyptian.

#### 4.6.1 Cross-Lingual Arabic Text Classification Results

In this section we show the result obtained from running our experiments as previously explained to compare different machine learning methods combined with each of the feature extraction methods.



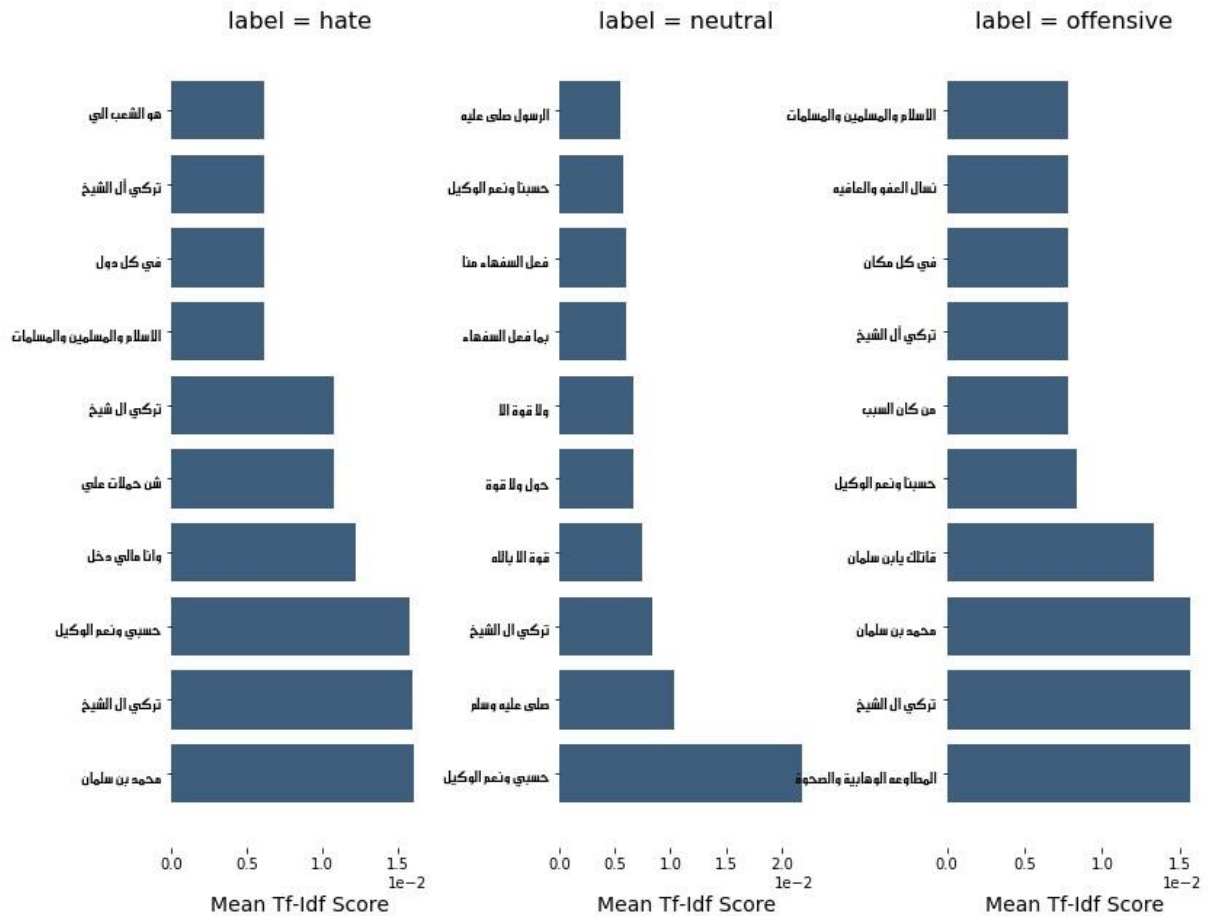


Figure 4.8: Top ngram appearing in each label of the Youtube dataset ranked by the mean TF-IDF score.

Table 4.1 shows each the performance result for each model in terms of accuracy, recall and f1 for utilising the Cross-lingual classification for the Youtube comments dataset. While Table 4.2 shows the cross-Lingual performance results for the Tweets datasets. The best performance is highlighted in bold in both tables.

Both Tables are showing the following insights which will help us address our research questions.

- In terms of accuracy for both data sets, Naive Bayes method gets the best performance. Naive Bayes achieved a relatively good performance giving that we are dealing with the translation noise of the comments.
- In terms of F1, and Recall, the Random forest and Naive bayes achieved slightly better

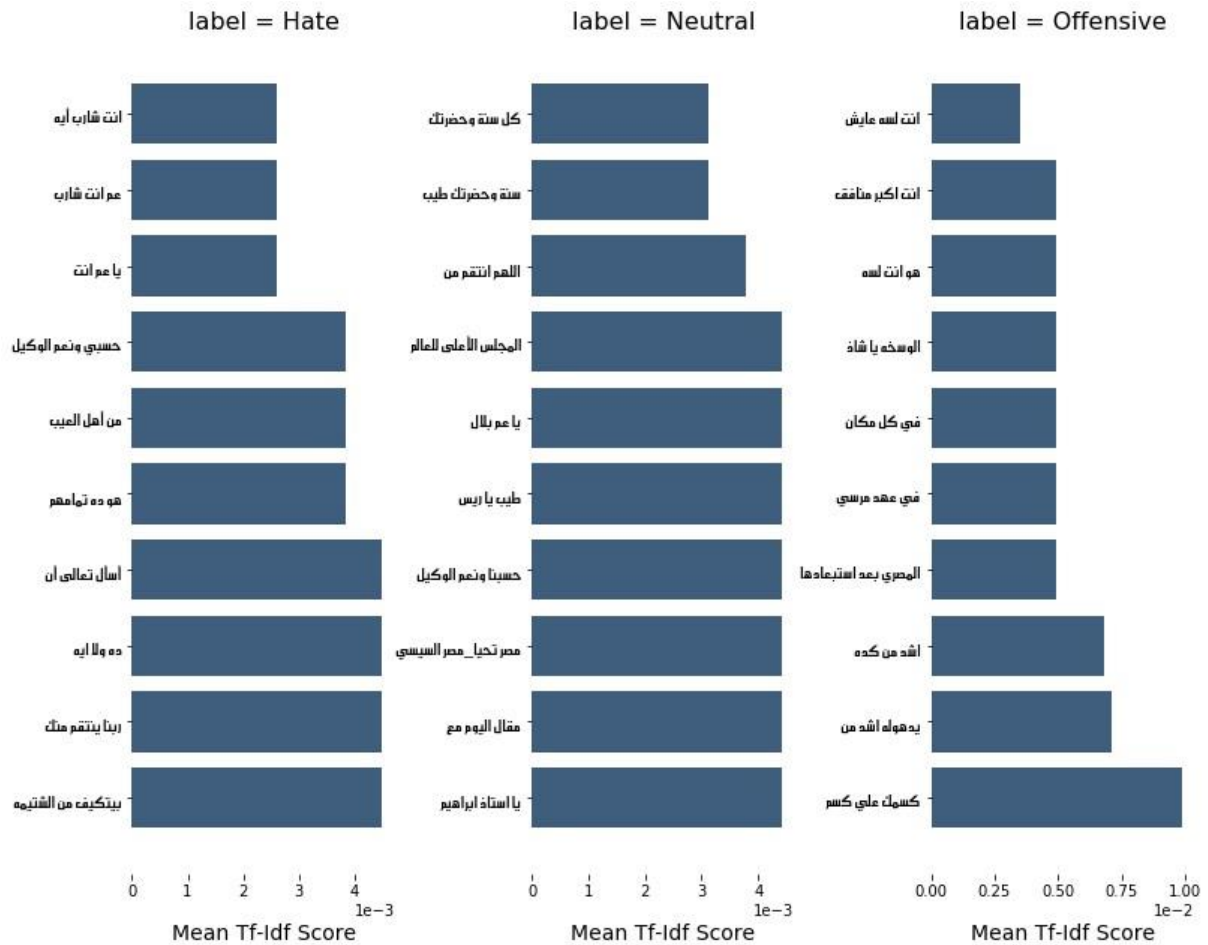


Figure 4.9: Top ngram appearing in each label of the tweets dataset ranked by the mean TF-IDF score.

results than the rest for both and tweets and Youtube comments dataset.

- Comparing the feature engineering methods, across both datasets, the Count vector representation consistently achieved better representation for all models.
- Overall, results from 4.2 and Table 4.1, shows that both Naive Bayes methods, and Random Forest are more suitable for this task over SVM and Logistic Regression, this can be attributed to that that these models could be more suitable for larger experiments and larger datasets with more data and a large-scale parameter tuning<sup>14</sup>.

<sup>14</sup> We leave this part of parameter tuning for SVM and Logistic regression for future work

## 4.6.2 Arabic Text Classification

Table 4.3 shows each the performance result for each model in terms of accuracy, recall and f1 for utilising the Cross-lingual classification for the Youtube comments dataset. While Table 4.4 shows the cross-Lingual performance results for the Tweets datasets. The best performance is highlighted in bold in both tables.

Looking at both tables, we can suggest the following insights :

- We can see that again both Naive Bayes and Random Forest performed slightly better than SVM and logistic for Arabic classification. As previously explained, this can be attributed to the size of our dataset, parameter tuning and dealing with noise, both Naive Bayes and Random Forest can be better choice.
- In terms of feature representation, results from Table 4.3 and Table 4.4 shows Character level representation is more effective for Arabic text classification in this task. This can be attributed to the Arabic text tokenisation used in this task, some local and dialect-based words are very hard to and process with additional noise coming from slang, spelling errors and compound phrases and words.

Finally, comparing the cross-lingual classification results showing in the previous section in tables 4.1 and 4.2, to the Arabic classification results showing in tables 4.3, 4.4, we can see that across all tasks, using all methods, cross-lingual classification achieve better or mostly similar performance to that in Arabic classification.

Although it has some translation errors, translating the comments and tweets, allowed us to navigate many complexity of the Arabic dialectal languages in Saudi and Egyptian. Google translation, since it is originally trained on web content and have good quality in terms of dealing with the noisy social text, was an effective method to fix many issues in the comments by replacing them into the formal language of English.

Another reason behind the effectiveness of cross-lingual classification is the text processing methods in English which were more effective and suitable for detecting hate speech in our explored datasets.

## 4.7 Summary and Conclusions

The work presented in this thesis investigated the task of detecting online hate-speech in Arabic social media. In particular, we studied building text classifiers for detecting hateful content in Arabic social media. We discussed the challenges and issues of hate-speech detection in Arabic online content and proposed novel approach for addressing these issues. We investigated the use of cross-lingual approaches to Arabic hate-speech detection on two datasets, the first one we built specifically for this study that is collected from YouTube online user comments in SaudiArabic, and the other one we used Egyptian tweets dataset that is collected from twitter about certain political views.

For our experiments, we designed two different hate-speech tasks for each collection, Crosslingual English-to-Arabic text classification and Arabic text classification. The goal for each task is to be able to automatically classify comments and tweets into the correct labels from Natural, Hate and Offensive label.

For each task we evaluated four different widely-used Machine Learning models, namely, Logistic Regression, Support Vector Machines, Naive Bayes, and Random Forest. For each model, we evaluated four different feature extraction methods, namely, Count Vectors, WordLevel TF-IDF, ngram-level TF-IDF and character-level TF-IDF vectorization of the text.

These extensive experimentation allowed us to perform comparative evaluation between Cross-lingual and monolingual techniques to Arabic text classification for hate-speech detection in informal social media content.

*Research Question 1 : Can Cross-lingual English-to-Arabic text classification be beneficial in detecting online hate speech? how does this approach compare the existing Arabic text.* Our experimental evaluation across different methods studied in this thesis, indicated that, although it may have some translation noise and add some errors, cross-lingual approaches can be more effective for handling Arabic social text for the task of hate-speech detection. This effectiveness can be attributed to different aspects as follows.

- Hate-speech detection on social media, in general, is very difficult task due to the noise associated with informal comments and tweets. Adding translation is going only to help due to the fact that it replaces dialectal and informal words into more simplified version.
- The effectiveness of Google translate is a key component in this task. Since it is trained on the noisy web content, it can be highly effective in terms of navigating noise and textual complexity of Arabic Saudi and Egyptian dialects.
- One the other hand, Arabic text classification relies mostly on available Arabic text processing tools that were rather designed to work for Modern Standard Arabic rather than specific dialects. To the best of our knowledge there is currently, no available tools for processing noisy social Egyptian and Saudi comments.

*Research Question 2 : Which feature engineering is most effective Arabic text classification in Detecting online Hate speech for Arabic.* Our experiments indicated that for Arabic social text, the use of character-level TF-IDF scores of terms is the most effective. This can be attributed to the nature of the textual content in both Saudi and Egyptian datasets. Comments and tweets can have words of a varying formats and style, and using character-level would be the most effective to process this text and extract the most important words features for detecting hate-speech,

Research Question 3 : Which feature engineering is most effective for Cross-Lingual Text classification in Detecting Arabic online Hate speech. For English, we found that count vector is the most effective. Although it is considered to be the most simply feature extraction approach, count vector is very robust when dealing the translated text with multiple translation edits and errors.

#### 4.7.1 Future Directions

Overall, our work presented a novel and totally different direction to detect hate-speech from dialectal Arabic content. We hope this study will open up new direction for further work in

investigating Cross-lingual approaches not only in hate-speech detection but for many other Arabic text classification tasks for informal text content such as sentiment analysis and many others.

In terms of the experiments, further investigation could be carried in larger datasets that contain hundred thousands of tweets or comments. We believe cross-lingual text classification would highly benefit from scalable machine learning approaches such as Logistic regression, Support Vector Machines and Deep Neural Networks.

Finally, we suggest further work on this area to test newer emerging effective feature extraction approaches such as Word Embedding which captures not only the frequency of words in terms of count and TF-IDF scores but also the semantic relationship between words which can be effective for hate-speech detection.

We hope these points can drive more and much needed work in the area of Arabic social media text processing.

Cross-lingual Text Classification for Youtube Comments DataSet							
Naive Bayes				Linear Classifier (SVM)			
	Accuracy	Recall	F1		Accuracy	Recall	F1
Count Vectors	0.6354	0.535	0.5019	Count Vectors	0.6197	0.2065	0.255
WordLevel	0.6197	0.2065	0.255	WordLevel	0.6197	0.2065	0.255
N-Gram	0.6197	0.2065	0.255	N-Gram	0.6197	0.2065	0.255
CharLevel	0.6197	0.2065	0.255	CharLevel	0.6197	0.2065	0.255
Logistic Regression				Random Forest			
	Accuracy	Recall	F1		Accuracy	Recall	F1
Count Vectors	0.625	0.4757	0.3558	Count Vectors	0.6302	0.4817	0.5273
WordLevel	0.6197	0.2065	0.255	WordLevel	0.6145	0.4389	0.4678
N-Gram	0.6197	0.2065	0.255	N-Gram	0.6197	0.3196	0.7098
CharLevel	0.6302	0.4103	0.2996	CharLevel	0.5989	0.395	0.4083

Table 4.1: Cross-lingual Text Classification for Youtube Comments DataSet

Cross-lingual Text Classification for Tweets DataSet							
Naive Bayes				Support Vector Machine			
	Accuracy	Recall	F1		Accuracy	Recall	F1
Count Vectors	0.5418	0.5522	0.5535	Count Vectors	0.3963	0.1321	0.1892
WordLevel	0.5272	0.6781	0.448	WordLevel	0.3963	0.1321	0.1892
N-Gram	0.429	0.4882	0.2865	N-Gram	0.3963	0.1321	0.1892
CharLevel	0.5309	0.6797	0.4155	CharLevel	0.3963	0.1321	0.1892
Logistic Regression				Random Forest			
	Accuracy	Recall	F1		Accuracy	Recall	F1
Count Vectors	0.5345	0.5769	0.5506	Count Vectors	0.5418	0.541	0.5664
WordLevel	0.4945	0.5646	0.4378	WordLevel	0.5309	0.5589	0.5174
N-Gram	0.4254	0.4824	0.2845	N-Gram	0.4218	0.4416	0.3883
CharLevel	0.56	0.6436	0.5088	CharLevel	0.4872	0.5271	0.4627

Table 4.2: Cross-lingual Text Classification for Tweets DataSet

Arabic Text Classification for Youtube Comments DataSet							
Naive Bayes				Support Vector Machine			

	Accuracy	Recall	F1		Accuracy	Recall	F1
Count Vectors	0.6041	0.4572	0.4448	Count Vectors	0.6354	0.2118	0.2590
WordLevel	0.6354	0.2118	0.2590	WordLevel	0.6354	0.2118	0.2590
N-Gram	0.6354	0.2118	0.2590	N-Gram	0.6354	0.2118	0.2590
CharLevel	0.7406	0.6462	0.5794	CharLevel	0.6458	0.5473	0.2987
Logistic Regression				Random Forest			
	Accuracy	Recall	F1		Accuracy	Recall	F1
Count Vectors	0.6302	0.3290	0.2919	Count Vectors	0.6406	0.6405	0.5476
WordLevel	0.6354	0.2118	0.2590	WordLevel	0.6354	0.2118	0.2118
N-Gram	0.6354	0.2118	0.2590	N-Gram	0.6302	0.2122	0.2122
CharLevel	0.6406	0.5467	0.2974	CharLevel	0.6614	0.8853	0.7192

Table 4.3: Arabic Text Classification for Youtube Comments DataSet

Arabic Text Classification for Twitter Dataset							
Naive Bayes				Support Vector Machine			
	Accuracy	Recall	F1		Accuracy	Recall	F1
Count Vectors	0.56	0.5502	0.5467	Count Vectors	0.5018	0.5723	0.4011
WordLevel	0.5309	0.6812	0.4387	WordLevel	0.5018	0.6673	0.3932
N-Gram	0.4109	0.6037	0.3224	N-Gram	0.4363	0.4680	0.2908
CharLevel	0.6972	0.7102	0.6588	CharLevel	0.4781	0.5823	0.4396
Logistic Regression				Random Forest			
	Accuracy	Recall	F1		Accuracy	Recall	F1
Count Vectors	0.52	0.4998	0.4979	Count Vectors	0.5927	0.6592	0.6602
WordLevel	0.5309	0.5789	0.3702	WordLevel	0.52	0.6052	0.6314
N-Gram	0.4436	0.5224	0.3236	N-Gram	0.4254	0.5510	0.4828
CharLevel	0.6109	0.4927	0.3927	CharLevel	0.6818	0.7083	0.7440

Table 4.4: Arabic Text Classification for Twitter Dataset



# Bibliography

- Abadeh, S. S., Esfahani, P. M. M., and Kuhn, D. (2015). Distributionally robust logistic regression. In *Advances in Neural Information Processing Systems*, pages 1576–1584.
- Aggarwal, C. C. and Zhai, C. (2012). A survey of text classification algorithms. In *Mining text data*, pages 163–222. Springer.
- Allan, J., Ballesteros, L., Callan, J. P., Croft, W. B., and Lu, Z. (1995). Recent experiments with INQUERY. In *Proceedings of The Fourth Text REtrieval Conference, TREC 1995, Gaithersburg, Maryland, USA, November 1-3, 1995*.
- Berry, M. W. and Castellanos, M. (2004). Survey of text mining. *Computing Reviews*, 45(9):548.
- Bishop, C. M. (2006). *Pattern recognition and machine learning*. springer.
- Brucher, H., Knolmayer, G., and Mittermayer, M.-A. (2002). Document classification methods for organizing explicit knowledge.
- Buckley, C., Allan, J., and Salton, G. (1993). Automatic routing and ad-hoc retrieval using SMART: TREC 2. In *TREC*, pages 45–56.
- Burnap, P. and Williams, M. L. (2014). Hate speech, machine classification and statistical modelling of information flows on twitter: Interpretation and communication for policy decision making.
- Burnap, P. and Williams, M. L. (2015). Cyber hate speech on twitter: An application of machine classification and statistical modeling for policy and decision making. *Policy & Internet*, 7(2):223–242.
- Diab, M. and Habash, N. (2012). Arabic dialect processing tutorial. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Tutorials*, page 3. Association for Computational Linguistics.

- Djuric, N., Zhou, J., Morris, R., Grbovic, M., Radosavljevic, V., and Bhamidipati, N. (2015). Hate speech detection with comment embeddings. In *Proceedings of the 24th International Conference on World Wide Web Companion*, pages 29–30. International World Wide Web Conferences Steering Committee.
- Facebook video (2019). Facebook. <https://www.facebook.com/facebook/videos>. Retrieved: 2019-01-30.
- Farghaly, A. and Shaalan, K. (2009). Arabic natural language processing: Challenges and solutions. *ACM Transactions on Asian Language Information Processing (TALIP)*, 8(4):14.
- Gagliardone, I., Gal, D., Alves, T., and Martinez, G. (2015). *Countering online hate speech*. Unesco Publishing.
- Greevy, E. and Smeaton, A. F. (2004). Classifying racist texts using a support vector machine. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 468–469. ACM.
- Habash, N. Y. (2010). Introduction to arabic natural language processing. *Synthesis Lectures on Human Language Technologies*, 3(1):1–187.
- He, B. and Ounis, I. (2006). Query performance prediction. *Information Systems*, 31(7):585–594.
- Huang, S., Cai, N., Pacheco, P. P., Narrandes, S., Wang, Y., and Xu, W. (2018). Applications of support vector machine (svm) learning in cancer genomics. *Cancer Genomics-Proteomics*, 15(1):41–51.
- Internetworldstats.com (2017). Internet world users by language top 10 languages. <http://www.internetworldstats.com/stats7.htm>. Retrieved: 2017-01-04.
- Jain, N. and Srivastava, V. (2013). Data mining techniques: a survey paper. *IJRET: International Journal of Research in Engineering and Technology*, 2(11):2319–1163.
- Karst, L. W. L. K. L. and Mahoney, D. J. (2000). *Encyclopedia of the american constitution*.

- Khan, A., Baharudin, B., Lee, L. H., and Khan, K. (2010). A review of machine learning algorithms for text-documents classification. *Journal of advances in information technology*, 1(1):4–20.
- Khwileh, A., Afli, H., Jones, G. J., and Way, A. (2017). Identifying effective translations for cross-lingual arabic-to-english user-generated speech search. *WANLP 2017 (co-located with EACL 2017)*, page 100.
- Khwileh, A., Ganguly, D., and Jones, G. J. (2016). Utilisation of metadata fields and query expansion in cross-lingual search of user-generated internet video. *Journal of Artificial Intelligence Research*, 55:249–281.
- Khwileh, A. and Jones, G. J. (2016). Investigating segment-based query expansion for user-generated spoken content retrieval. In *Content-Based Multimedia Indexing (CBMI), 2016 14th International Workshop on*, pages 1–6. IEEE.
- Korde, V. and Mahender, C. N. (2012). Text classification and classifiers: A survey. *International Journal of Artificial Intelligence & Applications*, 3(2):85.
- Kwok, I. and Wang, Y. (2013). Locate the hate: Detecting tweets against blacks. In *Twenty-Seventh AAAI Conference on Artificial Intelligence*.
- Labach, A., Salehinejad, H., and Valaee, S. (2019). Survey of dropout methods for deep neural networks. *arXiv preprint arXiv:1904.13310*.
- LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *nature*, 521(7553):436–444.
- Li, T., Li, J., Liu, Z., Li, P., and Jia, C. (2018). Differentially private naive bayes learning over multiple data sources. *Information Sciences*, 444:89–104.
- Mikolov, T., Yih, W.-t., and Zweig, G. (2013). Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 746–751.

- Monkey Learn (2019). Text classification. <https://monkeylearn.com/text-classification/?fbclid=IwAR1R0Yk1PzPaU68dNdHrf0xrIN3yX4nnDjiM3k7qQ6MttaNkmnFGT> Retrieved: 2019-11-30.
- Mubarak, H., Darwish, K., and Magdy, W. (2017). Abusive language detection on Arabic social media. In *Proceedings of the First Workshop on Abusive Language Online*, pages 52–56, Vancouver, BC, Canada. Association for Computational Linguistics.
- Pak, A. and Paroubek, P. (2010). Twitter as a corpus for sentiment analysis and opinion mining. In *LREC*, volume 10, pages 1320–1326.
- Pew Research, USA (2019). Online harassment 2017. . Retrieved: 2019-11-30.
- Salton, G. and Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Information Processing and Management*, 24(5):513–523.
- Salton, G., Wong, A., and Yang, C. S. (1975). A vector space model for automatic indexing. *Commun. ACM*, 18(11):613–620.
- Schmidt, A. and Wiegand, M. (2017). A survey on hate speech detection using natural language processing. In *Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media*, pages 1–10.
- Sebastiani, F. (2005). Text categorization. In *Encyclopedia of Database Technologies and Applications*, pages 683–687. IGI Global.
- Segal, M. R. (2004). Machine learning benchmarks and random forest regression.
- Singhal, A., Salton, G., Mitra, M., and Buckley, C. (1996). Document length normalization. *Information Processing and Management*, 32(5):619–633.
- Sparck-Jones, K. (1973). Index term weighting. *Journal of Documentation*, 9(11):619–633.
- Srividhya, V. and Anitha, R. (2010). Evaluating preprocessing techniques in text categorization. *International journal of computer science and application*, 47(11):49–51.

Statnikov, A., Wang, L., and Aliferis, C. F. (2008). A comprehensive comparison of random forests and support vector machines for microarray-based cancer classification. *BMC bioinformatics*, 9(1):319.

Ting, S., Ip, W., and Tsang, A. H. (2011). Is naive bayes a good classifier for document classification. *International Journal of Software Engineering and Its Applications*, 5(3):37–46.

Twitter inc (2019). Quarterly results- twitter. <https://investor.twitterinc.com/financial-information/quarterly-results/default.aspx>. Retrieved: 2019-01-30.

Vijayarani, S., Ilamathi, M. J., and Nithya, M. (2015). Preprocessing techniques for text mining- an overview. *International Journal of Computer Science & Communication Networks*, 5(1):7–16.

Warner, W. and Hirschberg, J. (2012). Detecting hate speech on the world wide web. In *Proceedings of the Second Workshop on Language in Social Media*, pages 19–26. Association for Computational Linguistics.

Wekipedia (2019). Hatespeech law by country. [https://en.wikipedia.org/wiki/Hate\\_speech#Hate\\_speech\\_laws\\_by\\_country](https://en.wikipedia.org/wiki/Hate_speech#Hate_speech_laws_by_country). Retrieved: 2019-01-30.

Wikipedia contributors (2019). Random forest — Wikipedia, the free encyclopedia. [https://en.wikipedia.org/w/index.php?title=Random\\_forest&oldid=925199071](https://en.wikipedia.org/w/index.php?title=Random_forest&oldid=925199071). [Online; accessed 10-January-2020].

Xu, R. and Yang, Y. (2017). Cross-lingual distillation for text classification. *arXiv preprint arXiv:1705.02073*.

Youtube (2017). Youtube. <http://www.youtube.com/>. Retrieved: 2019-01-30.

Yu, B., Xu, Z.-b., and Li, C.-h. (2008). Latent semantic analysis for text categorization using neural network. *Knowledge-Based Systems*, 21(8):900–904.

Zerrouki, T. pyarabic, an arabic language library for python.

Zerrouki, T., Alhawiti, K., and Balla, A. (2014). Autocorrection of arabic common errors for large text corpus. In *Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP)*, pages 127–131.