People's Democratic Republic of Algeria

Ministry of Higher Education and Scientific Research

University of DJILALI BOUNAAMA Khemis-Miliana

Faculty of Science and Technology

Department ofMathematics & Informatics

Thesis Presented

*For obtaining the **Master** diploma in computer science*

***Speciality**Software Engineering and distributed systems*

Theme :

Arabic Sentiment Analysis: Corpus-Based Approach

Presented by:Infront of the jury:

GHAZI ZahraExaminer 1:

KEDDAOUI ImaneExaminer 2 :

Supervisor :GOUDJIL Mohammed

University Year: 2019/2020

Dedication

I dedicate this modest work to the light of my life:

My dear parents, my father AHMED, and my mother KHEIRA.

For their greatest love, support, encouragement of patience

And continuous support during my years of study.

To all my family, brothers (ABD ERRAHMANE, AYMEN, WALID) and

Sisters (CHAHRAZED *and her husband* ***MOHAMMED, RABIAA,***

FATIHA, WASSILA) for their moral support.

To my dear partner IMANE.

To all my friends, and to everyone I love and to all people

Who gave me encouragement

And bother to support me during this project.

Zahra

Dedication

First and foremost, I would like to thank ***ALLAH*** *Almighty*

for giving me the strength, knowledge, ability and opportunity to undertake this research study and to persevere and complete it satisfactorily. Without his blessings, this achievement would not have been possible

I dedicate my dissertation work

to my family, my family in law and many friends*.*

A special feeling of gratitude ***to my loving parents, MOHAMED and FATIHA*** *whose words of encouragement and push for tenacity ring in my ears.*

My two and only brothers: YAAKOB and SIDAHMED,to my husband MOURAD and my grandmother "MANI"

have never left my side and are very special.

A special dedicate to my best friends: ***FATIMA, NOURELHOUDA, MOUNA, IMENE, SOUMIA AND FELLA***

IMANE.

Acknowledgements

In the name of **ALLAH**, the Most Gracious and the Most Merciful.

First of all, we thank **ALLAH** Almighty for giving us the will and courage to carry out this work.

Our sincere thanks go to **Dr. MOHAMED GOUDJIL,** the supervisor of this thesis, for his guidance, encouragement, wisdom and patience and for pushing us further than we thought we could go.

We would like to thank all of the jury members who honored us reviewing this work.

Last but not the least, we thank all our family and friends who helped us, encouraged us and anyone who contributed to the development of this work, with advice, or even with a smile.

Statement of Original Authorship

The work contained in this thesis has not been previously submitted to meet requirements for an award at this or any other higher education institution. To the best of our knowledge and belief, the thesis contains no material previously published or written by another person except where due reference is made.

Signature: GHAZI Zahra, KEDDAOUI Imane

Date: 17/11/2020

الخلاصة :

في السنوات الأخيرة، أصبح هناك عدد هائل من المستخدمين الذين يشاركون محتوياتهم عبر مجموعة واسعة من الشبكات الاجتماعية.وبالتالي، يتوفر حجم هائل من البيانات الإلكترونية على الإنترنت و الذي يحتوي على أفكار المستخدمين ومواقفهم وآرائهم تجاه بعض المنتجات أو الأحداث أو الأخبار أو أي مواضيع مثيرة للاهتمام. لذلك، أصبح تحليل المشاعر موضوعًا مرغوبًا فيه لأتمتة عملية استخلاص آراء المستخدم. اللغة العربية هي إحدى لغات مشاركة المحتوى على نطاق واسع عبر الشبكة الاجتماعية. لكن اللغة العربية لديها العديد من العقبات التي تجعل تحليل المشاعر مشكلة صعبة. يشارك معظم المستخدمين محتوياتهم بلغة عربية غير رسمية. بالإضافة إلى ذلك ، هناك الكثير من اللهجات العربية المختلفة. ومن ثم ، فإن أبحاث تحليل المشاعر في مجال اللغة العربية يتم تطويرها ببطء مقارنة باللغات الأخرى مثل الإنجليزية. تقترح هذه المذكرة نهجًا قائمًا على مجموعة لتحليل المشاعر في مجال اللغة العربية ، وقد تم تنزيل هذه المجموعة من الإنترنت ومن أجل تحسين أداء تحليل المشاعر في المجال العربي ، قمنا بإضافة بعض التعديلات المهمة عليها ، بحيث يمكن للمجموعة مقارنة مصنفين (Decision Tree, Naïve Bayes) مع مواقف مختلفة (قبل المعالجة المسبقة ، بعد إزالة كلمات التوقف ، بعد الاشتقاق)

الكلمات المفتاحية :

تحليل المشاعر ، الآراء، الشبكات الاجتماعية، اللغة العربية، المنهج القائم على الجسم، المصنفات

Abstract:

In recent years, there became a massive number of users who share their contents over a wide range of social networks. Thus, a huge volume of electronic data is available on the Internet containing the users' thoughts, attitudes, views and opinions towards certain products, events, news or any interesting topics. Therefore, sentiment analysis becomes a desirable topic in order to automate the process of extracting the user's opinions. One of the widely content sharing languages over the social network is Arabic Language. However Arabic language has several obstacles that make the sentiment analysis a challenging problem. Most users share their contents in informal Arabic. Additionally, there are lots of different Arabic dialects. Hence, Arabic sentiment analysis research is developed slowly compared to other languages such as English. This thesis pa proposes a corpus-based approach for Arabic sentiment analysis, this corpus was downloaded from internet and in order to improve the performance of sentiment analysis in the Arabic domain we have added some important modification on it, so the corpus can compare two classifiers (Decision Tree, Naïve Bayes) with different situations (before pre-processing, after removing stopwords, after stemming).

Key-words:

Sentiment analysis, opinions, social networks, Arabic language, corpus-based approach, classifiers.

Table of content

General Introduction

Chapter I : Sentiment Analysis (Background)

Chapter II : Related works in Arabic sentiment analysis

Chapter III : Implementation

General Conclusion

Bibliographical references

Annexes

List of figures

List of tables

List of Abbreviations

API	Application Program Interface
CA	Classical Arabic
DA	Dialectal Arabic
DT	Decision Tree
IOT	Internet of Things
KNN	K-Nearest Neighbors
ME	Maximum Entropy
ML	Machine Learning
MSA	Modern Standard Arabic
NB	Naïve Bayes
NLP	Natural Language Processing
POS	Part Of Speech
SA	Sentiment Analysis
SVM	Support Vector Machine
TF	Term Frequency
TF-IDF	Term Frequency Inverse Document Frequency

General Introduction

General introduction

Over the last decade, Internet users started to contribute more to the Internet's contents by adding comments or opinions to the webpages. This is due to the Web 2.0 technology which permits Internet surfers to share their thoughts and views via social networks (such as Twitter and Facebook), personal Blogs, forums, etc. This new technology results in a massive amount of raw data for which creative data mining techniques are needed in order to extract valuable knowledge.

Sentiment Analysis (SA) is one of the most vital research fields of Natural Language Processing (NLP) nowadays. It has emerged as an active research field with the proliferation of textual data on the Web especially in social media websites. It can be accomplished through both supervised and unsupervised learning techniques. In both techniques, labeled data is required for training and testing in supervised learning and for testing in unsupervised learning. Most NLP tasks need an annotated corpus for training machine learning classifiers, the corpus has to be in machine-readable form i.e. it has to be annotated for the machine to understand it.SA of English has been thoroughly researched; however, research on SA of Arabic has just flourished.

Arabic is ranked fourth among languages on the web although it is the fastest growing language on the web among other languages. Arabic is a morphologically rich language where one lemma could have hundreds of surface forms; this complicates the tasks of SA. Moreover, Arabic language is in a state of diglossia where the formal language used in written form differs radically from the one used in every-day spoken language. The formal language is called Modern Standard Arabic (MSA) and the spoken language differs in different Arabic countries producing numerous Arabic dialects sometimes called informal Arabic or colloquial Arabic. The language used in social media is known to be highly dialectal. Previous research on SA of Arabic was merely for MSA, but recently researchers started addressing Dialectal Arabic (DA). Dialects differ from MSA phonologically, morphologically and syntactically .Moreover, dialects do not have standard orthographies. Most Arabic NLP solutions are designed for MSA and perform poorly on DA. It is very difficult and almost impossible for one NLP solution to process all the variants of Arabic. As such, an Arabic NLP solution has to specify the Arabic variant it can process beforehand.

The remainder of this thesis is organized as follow.chapter 1 exhibits sentiment analysis in background and the definition of our chosen approach “corpus based approach” and its characteristics. Chapter 2 presents the related works in Arabic sentiment analysis with a corpus based approach, and then we talked about the most common challenges and gaps in Arabic sentiment analysis. Finally In chapter 3 we presented the results of our work.

Chapter 1:

Sentiment Analysis (Background)

1. Introduction

In this chapter we will get to know about sentiment analysis, the first part shows the definition of Sentiment analysis as well as the different applications and challenges that has faced it, the main approaches and levels of sentiment analysis are illustrated in the second part of this chapter, finally, in the third part we will see opinion classification techniques and the methodologies of sentiment analysis.

2. Sentiment Analysis:

Sentiment analysis (SA) which is commonly known as opinion mining is a process of studying public opinion about an entity, it refers to the use of natural language processing, text analysis, computational linguistics, and biometrics, and it concerned with the analysis of human opinion, sentiment, and emotion expressed in positive or negative comments, tweets, questions and requests, by analyzing a large numbers of documents regarding a specific topic or subject, such as reviews and survey responses, online and social media. [1]

Sentiment analysis plays an important role in decision making and the recommender system. Decision making includes purchasing a product or making an investment, for example: we Suppose that a customer wants to buy any item online, so before buy that item the customer generally reads reviews about that item or product on social media, but the richness of social media with huge amounts of reviews and comments would make it very challenging, and impossible to read by an investor or a buyer. [2]

Sentiment analysis makes this task easy because it describes the polarity of review so that a buyer can directly know whether a given review is positive or negative without reading the whole sentence which helps to take the right decision about that item, product or investment.

3. Challenges of sentiment analysis:

Though sentiment analysis is very powerful, it faces many challenges that are still not addressed and resolved efficiently, these challenges could be related to performance, data, methods and techniques used etc. The Sentiment Analysis process becomes successful when the challenges or issues are identified correctly and sorted out properly.

Based on holistic perspective views of sentiment analysis challenges we highlight the most important challenges bellow:

- **Sarcasm :**

Sarcasm is a form of expression where the literal meaning is opposite to the intended although there is technically no negative term in the language, it is intended to convey a negative sentiment. For example: when I do something wrong, and my sister says: “oh yeah! Good job”.

The speaker’s vocal or physical cues are usually necessary to signal sarcasm, such as accompanying a statement by rolling one’s eyes or speaking with a specific tone, this tone is dependent on the language of the speaker. so to figure it out, it requires a good understanding of the context, the culture of the situation, the topic, the people and also the language involved in the sarcastic statement. Some people thus have difficulty understanding sarcasm if they have a hard time reading vocal and physical cues, and it is also very hard for formachines to determine the polarity of a speaker's opinion. [3]

- **context dependency:**

Some words may be objective out of context but could assume subjectivity in a specific context or domain, e.g., the adjective ‘long’ is neither positive nor negative, but it could be positive in some domains, for example: ‘long battery’, or negative in other contexts, eg: ‘long queue’. Often the context of a word is dependent on words far away in the sentence and, hence, outside the window of adjacent words. Parse tree models in conjunction with word vectors are able to identify large sub-structures effectively. [4]

- **Emojis:**

Emojis are pictorial representations of human expression using graphical icons used to express moods of human feelings that are hard to interpret [5]. It is usually used on social media especially in Facebook posts, comments and also in tweets. Emojis is divided to two categories, ones encoded in only one or two characters such as 😁 happy coded as :) ☹ and sad coded as :(emojis, and ones with a longer combination of characters and a smaller nature, such as shrug emojis which is coded as ¯ \ _ (ツ) _ / ¯.

Emojis play an important role in the sentiment of texts, particularly in tweets. When performing sentiment analysis a lot of preprocessing might be needed., For example, is case we perform sentiment analysis on tweets, we'll need to pay special attention to character-level, as well as word-level, we might also want to preprocess social media content and transform both Western and Eastern emojis into tokens and whitelist them (i.e. always take them as a feature for classification purposes) in order to help improve sentiment analysis performance.[6]

- **Comparative opinions:**

Comparative opinions can be expressed as a comparison of two or more entities based on some of their shared features, It is usually conveyed using the comparative or superlative form of an adjective or adverb, they are different from regular opinions in that they have different semantic meanings and different syntactic forms, To determine the polarity for comparative sentences can be a challenge. For example Battery life of phone X is better than phone Y. This review has the positive word 'better' but the author's preferred object is not easy to determine which is the key piece of information in a comparative review [7], so treating comparisons in sentiment analysis became another challenge worth tackling.

- **Negation:**

Negation is a theoretical challenge, where Most traditional text processing relies on the fact that small differences between two pieces of text don't change the meaning very much [5]. But in sentiment analysis if negation didn't handled properly, it can give completely wrong results.

Negation words or phrases, such as never, not, no, none, nothing, neither, and nor, can reverse the polarities of the opinion words. For example in Sentiment analysis however, “There is a good chance that this phone will break easily" is very different from "There is a good chance that this phone will not break easily", this review shows positive polarity but presence of negation changes the effect completely [7].

- **Big Data:**

The contents on social web platforms including blogs, social networks and forums offer enormous amounts of heterogeneous data generated by the users of these web communities.

In general the sentiment analysis of big data is a challenging task due to volume, variety, velocity, variability and veracity of data, which are the main characterize the big data, especially in:

1) **Data Collection**

Data collection is a preliminary step for any sentiment analysis task but is one of the main challenges for researchers, not all sort of data sets are not available free for the interested researchers in sentiment analysis field, some of social networks sites provide APIs for enabling data collection from their sites. Although, due to the volume, variety, velocity of big data the collection of data set through using APIs is still a challenging task, since the APIs like Twitter API enables users to retrieve only one hundred tweets each time, compared to the volume of data available online.

2) **Data Preprocessing**

Preprocessing is another essential task for sentiment analysis and one of the major challenges in the big data world. Data volume restricts the filtering of relevant data from non-relevant data which may compromise the sentiment analysis results, big data variety and velocity limiting the feature extractions which are one of critical tasks in preprocessing of sentiment analysis data set.

3) **Data Storage and Analytics**

Data storage is one of the technical issues that is addressed by some or advanced storage techniques, because of the abundant size of the processed data.

Another challenge is velocity of big data since sentiment analysis on dynamic and real time events in the big data world is a challenging task that needs to be addressed efficiently taking into consideration the people's opinions are changes over time [8].

4. Application of sentiment analysis:

Since people are expressing their sentiments and opinions through social media, blogs, etc. and about different topics, sentiment analysis has many applications in various Fields, and some of them are mentioned below:

- **Applications in industrial and commercial sectors:**

Nowadays people tend to look upon reviews of services or products which are available online before they buy them based on the product reputation derived from the opinion of others, and for many businesses, the online opinion decides the success or failure of their product. Thus, Sentiment Analysis plays an important role in businesses. Businesses also wish to extract sentiment from the online reviews in order to improve their products and in turn improve their reputation and help in customer satisfaction. [9]

- **Applications in Medical and Health sectors:**

Applying medical data to sentiment analysis can help to identify, predict and track patient satisfaction levels by tweet rates, depression rates, patient comments and complaints, and it can also detect mental health issues such as help in the prediction of suicide rate. [10]

- **Applications in social sector:**

Sentiment analytics has been widely used across various domains and social applications, such as sociology, psychology, policy and sports, which allow showing trends in human emotions especially on social media, by detecting users opinion polarity, prediction of electoral results and public awareness and monitoring the views of students in education, and also determination of satisfaction about different social issues. [10]

- **Applications In Smart Homes:**

Smart homes are supposed to be the technology of the future. In future entire homes would be networked and people would be able to control any part of the home using a tablet device. Recently there has been a lot of research going on the Internet of Things (IoT). Sentiment Analysis would also find its way in IoT. Like for example, based on the current sentiment or emotion of the user, the home could alter its ambiance to create a soothing and peaceful environment [9].

5. Sentiment analysis approaches:

The Machine Learning approach uses a set of training records to train a model that is used later to predict new records without a label, each record is labeled to a class. When a new unlabeled record is given, the model is used to predict its label class. These classes are positive, negative and rarely neutral. In the Machine Learning Approach we can distinguish two different sub-approaches based on the learning method: the corpus-based approach (supervised learning approach) and the lexicon-based approach (unsupervised learning approach).

- **Corpus-based approach:**

The corpus-based approach uses machine learning classifiers, this approach requires labeled data to train classifiers, which learns from labeled training documents, to make predictions to the label class of the new unlabeled record, and the model would be corrected when those predictions are wrong. The training process continues until the model achieves a desired level of accuracy on the training data. And therefore detect the polarity of opinions (positive, negative and neutral).

This approach relies on methods such as Decision Tree, Support Vector Machine, Cross-validation, and Naïve Bayes….

- **Lexicon-based approach :**

Lexicon-based approach also called a dictionary approach is one of the two main approaches to sentiment analysis and it involves calculating the sentiment from the semantic orientation of words or phrases that occur in a text. With this approach a dictionary of positive and negative words is required, with a positive or negative sentiment value assigned to each of the words. Different approaches to creating dictionaries have been proposed, including manual and automatic approaches.

Generally speaking, in lexicon-based approaches a piece of text message is represented as a bag of words. Following this representation of the message, sentiment values from the dictionary are assigned to all positive and negative words or phrases within the message, a combining function, such as sum or average, is applied in order to make the final prediction regarding the overall sentiment for the message. Apart from a sentiment value, the aspect of the local context of a word is usually taken into consideration, such as negation or intensification.

6. Levels of Sentiment analysis:

Sentiment analysis has been mainly investigated at four different levels: document level, sentence level, aspect level and concept level.

1) Document level :

document level aims to classify a personal document as expressing an overall positive or negative opinion, this level considers the whole document as the primary information unit, and it works best when the document is subjective and written by a single person and expresses an opinion/sentiment on a single entity so it can't be applied to documents that evaluate or compare multiple entities.

2) Sentence level :

Sentiment analysis at this level classifies individual sentences to two classifications:

1) Classify sentences to objective sentences which present some factual information or subjective sentences that express personal feelings, views, emotions, or beliefs.
2) Classifying the resulting subjective sentences as expressing positive, or negative opinion , it also helps to determine the none sentences with no opinions, so the classification would be to positive, negative or neutral sentences.

This level deals with sarcasm and comparative sentences, and it's more challenging because the sentiment orientation of words is highly context-dependent [11].

3) Aspect level :

Aspect-based sentiment analysis is a fine-grained task, it aims to identify entities and extract its aspects, then summarize the opinions concerned with the aspect, and classify it to positive, negative or neutral. Positive opinion on an object can be positive on just an attribute of the object, but not on the object as a whole.

4) Concept level :

While all the levels above relay on the syntactic form of the text, this level classifies text based on the semantics by using semantic networks, which allow the aggregation of conceptual and affective information associated with natural language opinions.

It relies on the implicit features associated with natural language concepts. It is able to detect also sentiments that are expressed in a subtle manner, e.g., through the analysis of concepts that do not explicitly convey any emotion, but which are implicitly linked to other concepts that do so. The analysis at concept-level is intended to infer the semantic and affective information associated with natural language opinions and, hence, to enable a comparative fine-grained feature-based sentiment analysis. Rather than gathering isolated opinions about a whole item, users are generally more interested in comparing different products according to their specific features, or even sub-features[12].

7. Opinion classification techniques:

In natural languages, the sentiment is normally included in the subjectivity concept as the latter represents the language's aspects of opinions and impressions. Therefore, sentiment analysis involves performing a subjectivity classification task first so that a unit of text (term, phrase, sentence or document) is classified as either objective or subjective. Then, the subjective text is classified into the polarity it implies which might be positive, negative, neutral or even mixed. Sentiments can be annotated at various levels of granularity: word or phrase, aspect, sentence and document. Regardless of the level at which sentiment is captured, the sentiment classification process is conducted using one of three main approaches classifiers: [13]

- **Corpus-based approach**: in corpus-based approach a different classifiers can be used in classification steps we mentioned the most common of them: support vector machine(SVM), Naïve Bayes (NB), Decision tree (DT), maximum entropy (ME), K-Nearest Neighbors,
- **Lexicon-based approach**: for the lexicon-corpus N-grams, Baseline Approach, Conjunction Rules…classifiers are used for sentiment analysis of the datasets.
- **Hybrid approach**: in order to improve the performance of sentiment classification, both classifiers of corpus-based approach and lexicon-based approach can be used because this approach combines the both of the lexicon and corpus approaches.

8. Sentiment analysis methodologie:

The general process of sentiment analysis consists of5 steps:

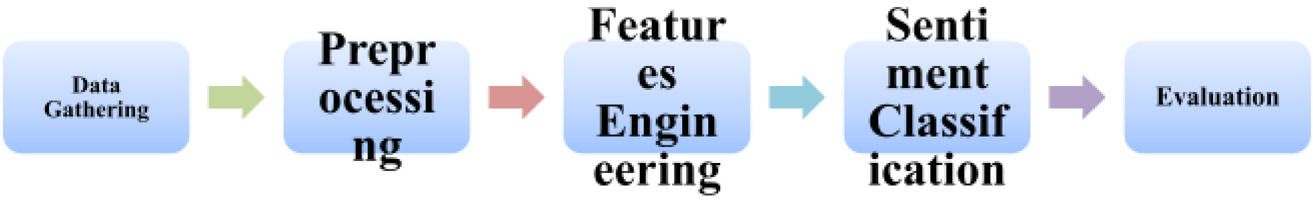


Figure 1.1: The general process of sentiment analysis.

1) Data gathering :

The key part for mastering sentiment analysis is working on different datasets and experimenting with different approaches. First, you'll need to get your hands on data and procure a dataset which you will use to carry out your experiments.

In sentiment analysis data is gathered from social media (Facebook and twitter....etc.), product analytics, customer service, market research and online commercial websites where people usually express their opinions and feelings and give their comments and reviews (Product reviews, restaurant reviews and movie reviews ...)

The corpora are either built using crawlers, Application Programming Interface (API) or collected manually, although crawlers have the advantage of collecting large numbers of posts, they do require to remove unwanted data, so any data collection from online sources is subject to the ethical and legal constraints governing re-use of the data for research. [14]

After the data has been prepared, it is converted into documents, which have been additionally modified for further processing.

2) Preprocessing :

Preprocessing is the process of cleaning the data readying the text for classification. Online texts usually contain lots of noise and unnecessary parts such as tags, scripts. Pre-processing the data reduces the noise which helps to improve the performance of the classifier, it also speeds up the classification process, thus helping in real time Sentiment Analysis.

Appropriate text pre-processing including data transformations and filtering can significantly improve the performance so before we could classify a given document, it is essential to first apply the preprocessing process which consists of cleaning, normalizing and preparing the text document to the classification step. We describe below the most common methods of preprocessing phase:

- **Tokenization:**

This process aims to transform a text document to a sequence of tokens separated by white spaces. The output of this process is a text free from punctuation marks and special characters. This is the basic definition of tokenization. However, this process takes a deeper dimension in Arabic language, it is called Segmentation. The segmentation process consists of separating a word from its clitics (including proclitics and enclitics) and the determiner Al. As a definition, clitics are a kind of affixes that can be associated to Arabic words; they can be prepositions, conjunctions, future markers, etc. As examples, the segmentation of the word«وبالعمل » (and with the work) gives « عمل» (work), and the segmentation of the word « يعتبرهم » (he considers them) gives «يعتبر» (he considers). The first example illustrates proclitic removal while the second one is a case of enclitic removal [15].

- **Normalization:**

Normalization means replacing specific letters within the word with other letters according to a predefined set of rules; i.e., the unification of characters. Some writing forms (Hamza and Alif) need normalization, which consists for instance in converting “ إ “ ,”أ ” and “ آ ” into “ا ” because most of the Arabic texts neglect the addition of Hamza on Alif. Another kind of impurity encountered is the elongation where users repeat letters for exaggeration. We shorten the elongated words by replacing the repeated letters with a simple occurrence instead [14].

There are many tasks performed simultaneously to achieve normalization. It includes the conversion of all text to either upper or lower case, eliminating punctuations and conversion of numbers to their equivalent words. This increases the uniformity of preprocessing on each text [16].

- **Stemming:**

It is the process of removing affixes from words, and reducing these words to their roots. It can significantly improve the efficiency of the classification by reducing the number of terms being input to the classification. Many stemming methods have been developed for Arabic language. The two most widely used stemming methods are:

- **The heavy stemming:** Allows transforming each surface Arabic word in the document into its proper root
- **The light stemming:** Allows removing prefixes and suffixes from words, as the removal of infixes can change the word meaning completely and consequently the sentiment polarity [14].

Stemming would reduce the words « المدرسة » (the school), « المدرس » (the teacher) and « الدراسة » (the study) to one stem « درس » (to study).While light-stemming would reduce the words « الدراسات » (the studies) and « مدرسان » (two teachers) to respectively « دراسة » (a study) and « مدرس » (a teacher). The main idea for using light stemming is that many word variants do not have similar meanings or semantics although these word variants are generated from the same root. For example, the stemming of the two words « رائع » (wonderful) and « مروع » (horrible) gives the word « روع » (horrify). We can see that the polarity of « رائع » (wonderful) is inversed by stemming. Hence, light-stemming allows retaining words' meanings [15].

- **Lemmatization:**

Lemmatization is the process of merging two or more words into a single word. This analyzes the word morphology and eliminates the ending of the word like shocked to shock, caught to catch etc [16].

- **Stop Words removal:**

Stop Words refer to function words (such as articles, prepositions, conjunctions, names, and pronouns) which provide structure in language rather than content, they are extremely frequent words and considered as valueless for taking them as features. We remove stop words that do not affect the classification task. Negation words should not be removed; they reverse the sentiment from positive to negative and vice versa. Note that the list of stop words (called stoplist) is typically established manually, it is domain and language-specific [14].

- **Filtering:**

The purpose of filtering is to remove character sequences that may be noisy and thus affect the quality of data. After converting text corpus into UTF”8 encoding, it is necessary to clean up the texts by removing punctuation marks, special characters, non-Arabic characters, dates,
time, numbers, single letters, links, and diacritics, etc. None of these impurities represents any polarity. Therefore They should be removed [14].

- **Term Frequency Thresholding:**

This process used to eliminate words whose frequencies are either above a pre-specified upper threshold or below a pre-specified lower threshold. This process helps to enhance classification performance since terms that rarely appear in a document collection will have little discriminative power and can be eliminated. Likewise, high frequency terms are assumed to be common and thus not to have discriminative power either[15].

3) **Features engineering :**

Feature engineering is the process of using domain knowledge to extract features from raw data via data mining techniques. These features can be used to improve the performance of machine learning algorithms. Feature engineering can be considered as applied machine learning itself. [17]

In the concept of feature Extraction and Selection, the preprocessed data forwarded to this phase facilitates syntactic features extraction since some preprocessing tasks like POS tagging, stemming and lemmatization, negation and emotion tagging can be considered key indicatives of the sentiment . In addition, through tokenization, the common bag-of-words and n-grams feature schemes are produced. Feature vectors can then be formulated via binary weighting due to the presence/absence of a word or n-gram in a specific input text. Furthermore, the relative importance of a term or an n-gram which is usually decided by its frequency of occurrence in the dataset, reduces the features' dimensionality by keeping terms of specific frequency values. On the other hand, sentiment lexicons provide another set of features where a term's sentiment score or intensity value define the text features. All these features are called "hand-crafted"; they have been used in most of the presented SA works . More recently, a novel type of features has emerged, the so-called text embeddings where words, phrases and sentences are mapped into real-valued, low-dimensional feature vectors to be used within deep learning systems[13]. Many techniques can be used for feature extraction, and some of them are mentioned below:

- **TF-IDF:**

TF-IDF is the short form for term frequency-inverse document frequency.TF-IDF is one of the largely used methods in information retrieval and text mining.TF-IDF is a weight metric which determines the importance of words for that document.

TF-IDF is considered as a better way to convert the textual representation of information into a Vector Space Model (VSM).

- **TF**

Term Frequency measures number of times a particular term t occured in a document d. Frequency increases when the term has occurred multiple times.TF is calculated by taking ratio of frequency of term t in document d to number of terms in that particular document d.

$$TF(t,d) = \frac{\text{Number of times term } t \text{ appears in a document } d}{\text{Total number terms in a document } d}$$

- **IDF:**

TF measures only the frequency of a term t.Some terms like stop words occur multiple times but may not be useful.Hence Inverse Document Frequency(IDF) is used to measure term's importance.IDF gives more importance to the rarely occurring terms in the document d. IDF is calculated as:

$$IDF(t) = log_e \frac{\text{Total number of documents}}{\text{Total number of documents with term } t \text{ in it}}$$

The final weight for a term t in a document d is calculated as:

TF-IDF(t/d) = **TF**(t/d) * **IDF**(t)

For example : Suppose there is a document which contains 200 words and out of these 200 words mouse appears 10 times than term frequency will be 10/250=0.04 and suppose there are 50000 documents and out of these only 500 documents contains mouse. Than IDF (mouse) = 50000/500=100, and TF-IDF (mouse) will be 0.04*100= 4. [18]

- **Bag-of-Words Model**

A common way to represent text documents in a simplified manner is by using a bag-of-words model. The technique lists term occurrence and optionally the frequency of term occurrence, disregarding grammar and term order. Machine learning classifiers can use the resulting model directly as feature vectors [19].

- **Doc2Vec:**

An extended version to word2vec, named doc2vec model was put to improve the learning of embeddings from word to word sequences. doc2vec can be applied for the word n-gram, sentence, paragraph or document.

Doc2vec is a set of approaches to represent documents as fixed length low dimensional vectors. Doc2Vec is a three layer neural network with an input ,one hidden layer and an output layer.Doc2vec was proposed in two forms: dbow and dmpv.In Word2vec two algorithms continuous bag of words (CBOW) and skip-gram(SG) algorithms are implemented using deep Learning, in Doc2vec these algorithms correspond to distributed memory(DM) and distributed bag of words(DBoW)[18].

- **Part-of-Speech Tagging:**

Part-of-speech (POS) tagging is the process of categorizing the tokens of a sentence into the different parts of speech (such as nouns, verbs, adjectives and adverbs) based on their definitions as well as the contexts. This way, POS tagging attempts to solve the problem of word ambiguity. There are many POS taggers for regular languages trained on treebanks particularly for the newswire domain such as the Penn Treebank. However, the conversational language of Twitter causes an out-of-domain problem for these traditional POS taggers, degrading their performance [19].

4) Sentiment classification :

In this step of pre-processing, classifiers play a major role in the classification, and we choose them depending on the approach type used.

- **Corpus-Based methods :**In corpus-based approach there are several methods and techniques used for classification, we mentioned some of them bellow:
- **Decision Tree(DT) :**decision Trees performs classification by using yes or no conditions. A decision tree consists of edges and nodes. The root node doesn't contain incoming edges, nodes which contain outgoing edges are called test nodes, nodes with no outgoing nodes are called decision nodes. Edges represent the conditions. Each decision node holds a class label. When the unlabeled data samples are to be classified, they pass through series of test nodes finally leading to the decision node with a class label and the class label is assigned to that unlabeled data sample.[18]
- **Naive Bayes(NB):**

Bayes theorem forms the basis for Naive Bayes classifier. It assumes that features are strongly independent and calculates the probability. Multivariate Bernoulli event model is one of the naive bayes classifiers used in sentiment analysis. If Xi is a boolean expressing the occurrence or absence of the ith term from the vocabulary , then the likelihood of a document given a class C_k is given by:[18]

$$P(x|C_k) = \prod_{i=1}^{n} p_{k_i}^{x_i} (1 - p_{k_i})^{1-x_i}$$

- **K-Nearest Neighbors :**

The k-nearest Neighbors algorithm (KNN) is a non-parametric technique used for classification. This method is used for text classification applications in many re- search domains. The initial application of k-Nearest Neighbors (k NN) to text categorization was reported by Masand and colleagues. The basic idea is to determine the category of a given query based not only on the document that is nearest to it in the document space, but on the categories of the k documents that are nearest to it. Having this in mind, the Vector method can be viewed as an instance on the k-NN method, where k = 1.

The best choice of k depends on the data, generally, larger values of k reduce the effect of noise on the classification but make boundaries between classes less distinct. Figure 1.2 shows that the test sample (circle) should be classified either to the first class of squares or to the second class of triangles. If k = 3 it is classified to the second class because there are 2 triangles and only 1 square inside the inner circle. If k = 5 it is classified to first class (3 squares vs 2 triangles inside the outer circle). A good k can be selected by various heuristic techniques, for example, cross-validation. The special case where the class is predicted to be the class of the closest training sample (i.e. when k = 1) is called the nearest neighbor algorithm [20].

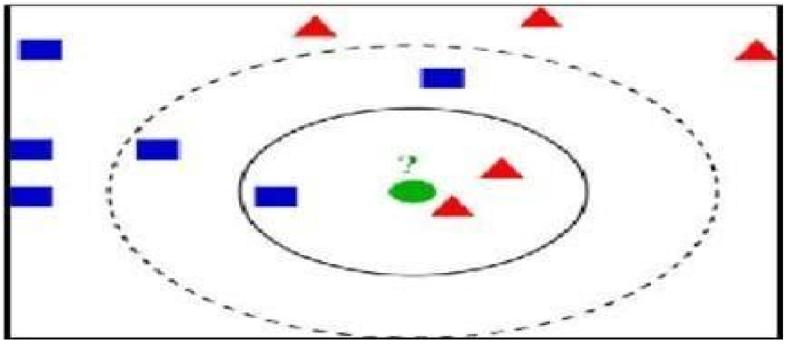


Figure 1.2: Example of KNN classification

- **Support Vector Machine SVM:**

Support vector machines (SVM) are a popular class of supervised learning algo- rithms. SVM was developed by Vapnik and Chervonenkis in 1963. B.E.Boser et al. adapted this version into a nonlinear formulation in the early 1990. SVM was originally designed for binary classification tasks. However, many researchers work on multi-class problems using this dominate technique. The Figure 1.3 indicates the linear and non-linear classifier which is used for 2 SVM is particularly applicable to large and high-dimensional classification problems. It is based on the structural risk minimization principle where input points in N-dimensional space are mapped into a higher dimensional space and then a maximal separating hyper plane is found Consider a training set of labeled instances xt∈Rn i = 1. . . L, belong to a set of

categories Figure 1.4 is an example of an optimal hyperplane for separating two classes, From Figure 1.4, SVM builds the classification model on the training data using a linear separating function to classify unseen instances .For linearly separable vectors, the kernel function is simple. It takes the form: F (x) = W * X + b.

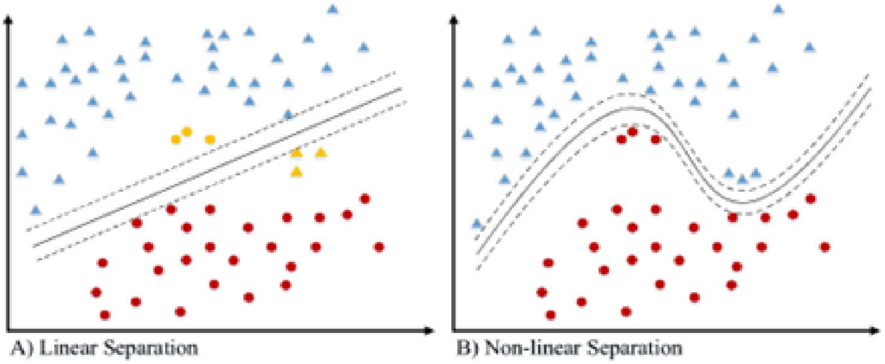
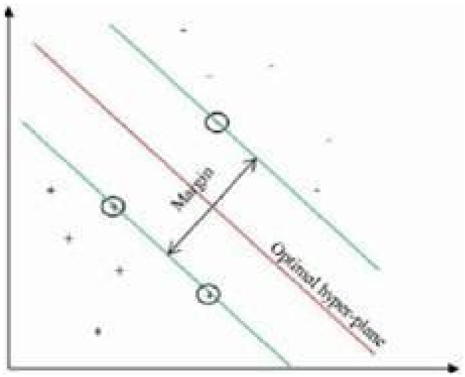


Figure 1.3: The linear and non-linear Support Vector Machine.

W is called a weight vector for optimal hyper-plane and b is known as the bias. The class of X (test instance) can be found using the following linear decision function: y = sign(f (x))   .

Support Vector Machines have been applied successfully in many text classification tasks since most text classification problems are linearly separable and *SVMs* are robust in high dimensional space and robust



with sparse data [20].

Figure 1.4: the linear separation between two classes (points marked with circles are support vectors)

- **Maximum Entropy**:

In the Maximum Entropy Classifier, no assumptions are taken regarding the relationship between the features extracted from the dataset. This classifier always tries to maximize the entropy of the system by estimating the conditional distribution of the class label. Maximum entropy even handles overlap features and is the same as logistic regression method which finds the distribution over classes. The conditional distribution is defined as MaxEnt makes no independence assumptions for its features, unlike Naive Bayes. The model is represented by the following:

$$P_{ME}(c \mid d, \lambda) = \frac{\exp[\sum_{i} \lambda_i f_i(c,d)]}{\sum_{c} \exp[\sum_{i} \lambda_i f_i(c,d)]}$$

Where c is the class, d is the tweet and λ i is the weight vector. The weight vectors decide the importance of a feature in classification [9].

- **Lexicon-Based methods :**

The lexical approach begins with the preprocessing of the text to be analyzed. The total score is initialized to zero. Then it is checked if the lexicon is present in the dictionary of words and if present whether it is positive or negative and the score is updated accordingly. The final score will thus classify the text as positive or negative. Some variants of the lexical approach are:

- **Baseline Approach:**

Here the dictionary is limited to a number of positively and negatively tagged words.

- **Stemming:**

It basically means getting rid of prefixes and suffixes, algorithm strips off the prefixes and suffixes like ‘watching’ and ‘watched’ becomes ‘watch’. So every word is stemmed before working with it.

Example: “I really liked this movie” will be changed to “I really like this movie”

- **Part of Speech Tagging:**

In the corpus linguistics, part-of-speech tagging, also called grammatical tagging is the process of marking a word in a text as corresponding to a particular part of speech, based on both its definition, as well as its context.(i.e relationship with adjacent and related words in a phrase, sentence or a paragraph).

It has been found that certain parts of speech such as adjectives and adverbs express polarity more often.

- **N-grams:**

Instead of considering each word by itself consecutive words in order can be considered to find out common phrases. These normal sequences of words are called n-grams. If you keep a track of longer phrases as well then it will give better results. In the example given below ‘knock your socks off’ or only ‘socks off’ is indicative of a good review.

Example: “This movie will knock your socks off”

- **Conjunction Rules:**

The main purpose of applying conjunction rules is to extract the precise meaning or expression from a given sentence using grammar rules. Generally, a sentence only expresses one opinion orientation unless there are some conjunctions such as ‘but’, ‘although’, ‘however’, ‘while’ that changes the direction of the sentence. Conjunction rules explanations are shown as below:

a. Although (Phrase A), (Phrase B).

E.g: Although this camera is nice, sadly it has short battery life.

In this case, phrase A can be cut off, and phrase B can be used for sentiment identification.

b. (Phrase A), but (Phrase B).

E.g. The camera appearance is not beautiful, but very durable.

In this case, phrase A can be cut off, and phrase B can be used for sentiment identification.

c. Although (Phrase A), (Phrase B), but (Phrase C).

E.g. Although this camera is nice, too bad it has short battery life, but I still like it.

In this case, phrase A and B can be cut off, and phrase C can be used for sentiment identification.[9]

o **StopWords:**

Stop words are usually the non-semantic words like articles, prepositions, conjunctions and pronouns. Articles such as ‘a’ and ‘the’ and pronouns such as ‘he,’ ‘they,’ and ‘I’ provide little or no information about sentiment. In computing, stop words are words, which are filtered out prior to, or after, processing of natural language data (text). We can get a list of such words from the Internet and use it as a pre-processing step to get rid of such words because we know that they do not hold any information. They basically just confuse the classifier and introduce problems.

o **Negation method:**

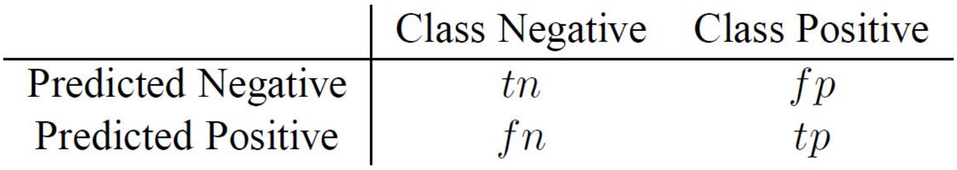
A negation word, such as ‘not’ inverts the evaluative value of an affective word. For example, ‘not good’ is similar to saying ‘bad’. By using a technique proposed by Das and Chen[2], a tag ‘NOT_’ can be added to every adjective between the negation word and the first punctuation mark following the negation word. Ex: ‘I do not like this movie’ becomes ‘I do NOT_like this movie’. In unigrams, the value of ‘like’ is positive, but there is a negation word ‘not’, therefore a ‘NOT’ is replaced and joined with the consequent word. As a result, ‘NOT_like’ can affect the value of the word and hence the polarity of the entire sentence.[21]

 Hybrid methods :

The hybrid method combines both of the corpus-based method and the lexicon-based method. The output of lexicon based method is used as input for machine learning algorithms, the true classified reviews used as a training set for the machine learning algorithm. After building the model using specific features. The unclassified reviews from corpus-based technique are used to test the model in the lexicon method technique.

4) Evaluation :

In text classification, the classifier evaluation is usually concerned with the classifier effectiveness rather than its efficiency. That means it evaluates the correctness of the classifier predictability rather than its computational complexity. The common statistics that are used in text mining are precision and recall in addition to accuracy and F-measure, they are usually computed from a confusion table that presents the correctly and wrongly classified cases for all categories. The confusion table is presented in Table 2.5.



	Class Negative	Class Positive
Predicted Negative	tn	fp
Predicted Positive	fn	tp

Table 2.1: The results of the classification presented in a confusion table

Tn: are true negative cases which is the number of correctly predicted negative documents.

Tp: is true positive. That is the number of correctly predicted positive classes.

Fp: is false positive which is negative classes predicted as positive,

Fn: is false negative, that is positive classes predicted as negative.

The evaluation metrics are computed based on the values in the confusion table as follows: Accuracy is the number of correctly predicted documents out of all documents. It is computed by:

Accuracy $= \frac{Tp+Tn}{tp+tn+fp+fn}$

Precision is the number of true positive out of all positively assigned documents, and it is given by:

Precision $= \frac{Tp}{Tp+Fp}$

Recall is the number of true positive out of the actual positive documents, and it is given by:

Recall $= \frac{Tp}{Tp+Fn}$

Finally F-measure is the harmonic average of the precision and recall as computed in Equation (3). It is a trade-off between precision and recall values. The f-measure for all three similarities used in this research (cosine, Jaccard and dice) is around their recall and precision values,and it is computed as:

F-measure $= \frac{2 * precision * recall}{precision + recall}$

Where its value ranges from 0 to 1 and indicates better results the closer it is to 1. [22]

8. Conclusion

In this chapter we have seen the definition of sentiment analysis, then we've discussed the most important challenges of sentiment analysis and its applications, next we've illustrated the main approaches and levels of sentiment analysis

Finally, we had seen opinion classification techniques of the different approaches, and the methodologies of sentiment analysis.

CHAPTER 2:

RELATED WORKS IN ARABIC SENTIMENT ANALYSIS

1. Introduction:

In this chapter we will give an overview about Arabic language, we will also talk about the root and the pattern of it, in the next step we will see the challenges that faces the natural language processing, then we will present the most interesting works that have been performed in Arabic Sentiment Analysis., finally we will see the negation in Arabic language and the challenges and gaps in Arabic sentiment analysis.

2. Arabic language overview:

Arabic is the official language of 27 countries, it is spoken by more than 400 million speakers, it is also recognized as the 4th most used language on the Internet.

There are three different forms of Arabic language: Classical Arabic (CA), Modern Standard Arabic (MSA) and Dialectal Arabic (DA). Classical Arabic is the language of Qur’an, the holy book of Islam, one of the world’s major religions. Modern Standard Arabic (MSA) is the dialect used in education, books, television, newspapers, and in conversation among educated Arabs who have different local dialects. Local dialects (also known as colloquial Arabic)

There are different dialects of Arabic, such as Egyptian, Maghreb, Hassaniya, Maltese, Sudanese, Levantine, Iraqi, Gulf, Hijazi, Najdi, and Yemeni which exists based on geographical location and country - even within the same country the dialect may vary in different areas.[22]

The Arabic language is a Semitic language that consists of 28 letters, as follows: أ, ب, ت, ث , ج , ح, خ , د, ذ, ر, ز, س , ش, ص , ض, ط, ظ, ع, غ , ف , ق , ك , ل , م , ن , ه , و and ي. There are three vowels, the letters (ا, و , and ي), and the rest are consonants.

Arabic text has right to left alignment. Unlike English, the shape of Arabic letters are not fixed, they change according to their location in the word. Moreover, short vowels, or diacritics, exist for Arabic language. These diacritics change the meaning of the word and its pronunciation, for example: كَتَبَ(katabah) means “wrote” whereas كُتُبٌ(kutub) means “books”. Although the same letters are used in both words, the diacritics significantly change the meaning, grammar and pronunciation.

In addition to the complexity of diacritics and the different ways a letter can be written in Arabic, the Arabic language includes is morphologically complex [23].

3. Root and pattern in Arabic language:

The key concepts in Arabic morphology are the concepts of root and pattern, which interlock to form the final shape of the word.

Roots: that are mainly three and four (and rarely five) radicals, i.e., consonants, comprise the smallest meaningful language unit.

Pattern: is the group of letters that have been used to derive the words. These two features possess the lexical and grammatical meanings, respectively early studies of Arabic morphology show that Arabic has almost 5000 roots while another has estimated that the roots of nouns and verbs together comprise 10,000. There are also around 400 different patterns in Arabic that may be added to the root.

These two features show how the root of Arabic words can form a variety of word forms that are the derivation and inflection. The derivation is the process of word/lemma formation from its root. This process occurs by combining a specific consonantal root with a desired pattern. The inflection of the Arabic word is caused by the contextual position of a word . Each different type of words (noun, verb or pronoun) has different inflectional categories that may be applied to them. In the case of nouns and adjectives, four inflectional categories are applied: gender, number, case and definiteness. Verbs have a larger number of features: aspect, person, voice, mood, gender and number. Finally, pronouns tend to possess four different features: person, gender, number, and case.

The Arabic language is highly inflectional and derivable. Arabic has a small number of roots, but this increases its complexity. The agglutinative feature of the word structure adds considerable difficulty to the language morphology. Arabic words may work with three types of affixes: prefixes, infixes, and suffixes. Affixes may be one letter long or combination of multiple letters. In addition to their complex nature, the level of ambiguity of Arabic, morphemes is notable. Determining whether a letter is an affix or part of the stem is not an easy task, especially when there is an absence of short vowels. These characteristics affect the NLP tools that deal with Arabic, such as the part-of-the speech tagger, morphology analyzer, named entity recognition and syntactic parsing. [24]

Meaning	Transcription	Arabic
a book	Kitaab	كِتاب
Awriter	Kitaab	كاتِب
an office	Maktab	مَكْتَب
Alibrary	Maktaba	مَكْتَبة
Aletter	Maktuub	مَكْتوب
Hewrites	Yaktab	يَكْتَب
Hewrote	Kataba	كَتَبَ

Table 2.1: Derivation of three letters root [25]

- Most roots are trilateral meaning they consist of 3 letters. There are some four letter roots and a few roots with five or six letters, but they all are few and far between.
- **Patterns** are the set molds of words that roots can be inserted into. Together, the root letters placed inside the patterns are words. Patterns also carry meanings,

similar to how suffixes and prefixes do so. The following table shows the relationship between roots and patterns[24]:

Meaning	Transcription	Root Inserted intoPattern	PatternMeaning	Pattern	BaseMeaning	Root
writer	kaatib	كاتِب	the "do"- er	فاعِل	write	ك-ت-ب
sleeper	naa'im	نائِم	the "do"- er	فاعِل	sleep	ن-ا-م
walker	maashii	ماشِي	the "do"- er	فاعِل	walk	م-ش-ى

Table 2.2: Placing the roots for the patterns [25]

4. Challenges of Arabic Natural language processing:

The Arabic language is both challenging and interesting. It is interesting due to its history, the strategic importance of its people and the region they occupy, and its cultural and literary heritage. It is also challenging because of its complex linguistic structure.

The challenges of Arabic language are into its characteristics and their related computational problems at orthographic, morphological, and syntactic levels. In automating the process of analyzing Arabic sentences, there is an overlap between these levels, as they all help in making sense and meaning of words, and in disambiguating the sentence.

- Arabic orthography:

- Within the orthographic patterns of the written words, the shape of a letter can be changed depending on whether it is connected with a former and subsequent letter, or just connected with a formal letter,

- the word behaves as homograph that is semantically and phonologically ambiguous For instance, the unvoweled word “كتب “)ktb), supports several alternatives such as " كَتَبَ")he wrote((kataba كَتَبَ) it was written,)kutibaكُتِبَ).
- the most critical use of Hamza letter (“ء“) ”الهمزة “brings in more challenges. With the very significance of Hamza being an additional letter seen at the top or bottom of the letters following the sounds of “و“, ”ا ,“or “ى ,“i.e. “ؤ“, ”أ ,“or “ئ “,respectively.
- Defective (weak) verb (الفعل المعتل) (“is any verb that its root has a long vowel as one of its three radicals. These long vowels will go through a change when the verb is conjugated. For example, consider the case of a negated present tense verb that is preceded by the apocopative particle Lam—“حرف الجزم لم ”.
- Most Arabic proper nouns (NEs) are indistinguishable from forms that are common nouns and adjectives (non-NEs) which might cause ambiguity. For example, the noun)“الجزيرة ‘’ “Aljazeera”) can be recognized as an organization name or a noun corresponding to island. In written Arabic, there are two types of vowels: diacritical symbols and long vowels.
- Arabic text is dominantly written without diacritics which leads to major linguistic ambiguities in the high level of ambiguity of the Arabic script poses special challenges to developers of NLP areas such as Morphological Analysis, Named Entity Extraction and Machine Translation. These difficulties are exacerbated by the lack of comprehensive lexical resources, such as proper noun databases, and the multiplicity of ambiguous transcription schemes.

-Arabic morphology:

Arabic is a Semitic language that has a powerful morphology and a flexible word order. It is difficult to put a border between a word and sentence, yielding morpho-syntactic structure combinations for a word along the dimensions of parts of speech, inflection, declension, clitics, among other features. Arabic morphology and sentence structure give the ability to incorporate a broad number of adds to each word which makes the combinatorial expansion of possible words, another morphologic challenge in Arabic language is that we can compose a word to another by a conjunction of two words. This conjunction can be with nouns, verbs, or particles.

- **Syntax is intricate:**

  - Historically, as Islam spread, the Arab grammarians wanted to lay down the basis of grammar rules that prevents the incorrect reading of the Holy Qur'an. Arabic syntax is intricate. Automating the process that makes the computer analyze the Arabic sentences is truly a challenging problem from the computer perspective.
  - Anaphora Resolution is classically recognized as a very difficult problem in NLP. It is one of the challenging tasks that is very time consuming and requires a significant effort from the human annotator and the NLP system in order to understand and resolve references to earlier or later items in the discourse.
  - Agreement is a major syntactic principle that affects the analysis and generation of an Arabic sentence which is very significant to difficult NLP applications such as Machine Translation and Question Answering. Agreement in Arabic is full or partial and is sensitive to word order effects [26].

5. Related works:

Several researches have proposed solutions to sentiment analysis. In this section, we focus on those works that handle Arabic sentiment analysis. We divide those works to two sections:

1) Arabic sentiment corpus:

Three tendencies were observed in the creation of sentiment corpus in Arabic: (1) Manual construction, (2) Automatic construction, and (3) Semi-automatic construction.
In most cases, the annotation was made by native annotators.

- Abdul-Mageed and Diab[27] introduced AWATIF, a multi-genre corpus that comprises 10,723 Arabic sentences retrieved from three sources namely: the Penn Arabic Treebank (ATB), a selection of web forums, and a list of Wikipedia talk pages. The sentences were manually annotated as objective or subjective, and the subjective sentences were annotated as positive or negative.

- Medhaffar et al [28] presented TSAC (Tunisian Sentiment Analysis Corpus). TSAC contains 17,060 Tunisian Facebook posts. These posts were manually annotated, including 8215 positive statements and 8845 negative statements. TSAC was collected from official pages of Tunisian radios and TV channels. .
- A multi-corpus was presented by Abdul-Mageed et al. [29] including DARDASHA, a collection of 2798 chat messages retrieved from Maktoob, TAGREED, a group of 3015 Arabic tweets, TAHRIR, a dataset composed by3008 sentences from Wikipedia talk pages, and MONTADA, a dataset produced by 3097 sentences retrieved from web forums. This corpus was manually annotated by native speakers as positive or negative. .
- Alharbi and Khan [30] proposed the manual construction of a corpus dedicated to distinguishing between comparative and non-comparative opinion. Three native Arabic speakers performed the annotation process. Two labelers annotated the sentences, and the third labeler made decisions about sentences that raised a conflict between the first and second labelers. The corpus contains 43% of comparative text and approximately 57%of non-comparative.
- In the context of annotations based on rating reviews, Aly and Atiya [31] presented LABR (Large-scale Arabic Book Reviews), a collection of book reviews that contain 63,257 book reviews, each of them on a scale that ranges from 1 to 5 stars. Reviews with 4 or 5 stars were annotated as positive while reviews with 1 or 2 stars were annotated as negative. Reviews with three stars were annotated as neutral. ElSahar and El-Beltagy [32] followed the same procedure proposed by [31] to annotate 7 collections of reviews named ATT, HTL, MOV, PROD, RES1, RES2, and RES.ATT and HTL are two datasets of tourist place reviews and hotel reviews extracted from TripAdvisor.com. Both datasets comprise 2154 reviews and 15,572 reviews, respectively. MOV is a dataset of movie reviews extracted from elcinema.com that contains 1524 reviews. PROD is a dataset of product reviews extracted from souq.com that contains 4,272 reviews. RES1 is a dataset of restaurant reviews extracted from qaym.com that contains 8364reviews. RES2 is a dataset of restaurant

reviews extracted from tripadvisor.com that contains 2642 reviews. Finally, RES is a combination of RES1 and RES2 that contains 10,970 reviews.

- Guellil et al [33] based on their created sentiment lexicon [34], in Algerian to write down a large collection of messages. The corpus comprises messages written in Arabic and Arabizi, and it contains 800 messages of which 4000 were written in Arabic and 4000 were written in Arabizi.
- Gamal et al [35] constructed a large sentiment corpus (containing 151,548 tweets where both positives and negative classes contain 75,774 tweets) dedicated to MSA and Egyptian dialect as well. For building their corpus, the authors first relied on the work of [31] for extracting and manually annotated 4404 phrases commonly used for expressing sentiment.
- AraSenTi-Tweet, Al-Twairesh et al.[36] is a corpus created using a semi-automatic approach. This corpus contains 17,573 Saudi tweets that were manually annotated into four classes as positive, negative, neutral, or mixed tweets. To create this corpus, the authors recovered a collection of tweets using a list of sentiment words. After cleaning the recovered tweets, three native Arabic annotators reviewed the constructed corpus to validate the post tags inferred from the list of sentiment words.

2) Corpus-Based classification methods:

Several researches have proposed solutions to sentiment analysis. In this section, we focus on those works that handle Arabic sentiment analysis using a corpus-based approach. We classify those works according to the approaches that handle sentiment analysis. Several authors used supervised machine learning approaches to analyze sentiments.

- Rushdi - Salah et al [37] have proposed an Opinion Corpus for Arabic (OCA) collected from different web pages with comments about movies, they used cross-validation to compare the performance of two of the most widely used learning algorithms: SVM and NB. In their experiments, the tenfold cross-validation (k = 10) has been used to evaluate the classifiers. The authors used the Rapid Miner software with its text mining plug-in, which contains different tools designed to assist in the preparation of text documents for mining tasks (tokenization, stop word removal, and stemming, among others), better accuracy was achieved with SVM classifier, and it is up to 91%.

- Abdul-Mageed et al [38] developed a machine learning system called “SAMAR” for subjectivity and sentiment analysis for Arabic social media genres, this system is based on MSA and its dialects (mainly Egyptian dialects).

  In their work, they applied an automatic tokenization tool in which they extracted the stem of words by isolating the prefixes and suffixes. They take into consideration several features (e.g. word form, Part of speech tagging). Finally, they applied SVM light as the classifier in their experiments and they achieved accuracy up to 71.3%.
- Gamal et al [35] validate their corpus by applying different machine learning algorithms such as SVM, NB, Ridge Regression (RR), etc. For extracting features, they used TF-IDF. They used tenfold cross-validation and obtained their better results by using RR (which are up to 99.9% for either *F*1-score and accuracy metrics). However, the authors carried out only intrinsic experiments then this result concerns a part of the constructed corpus.
- For evaluating the constructed corpus,Rahab et al [39] carried out many experiments for showing the impact of word weighting approach, classification method, and of the light stemming. They used many word weighting algorithms such as Term frequency, Term occurrence, Term frequency, and Inverse document frequency (TF–IDF), and binary term occurrence BTO. The authors also relied on many classifiers such as SVM, NB, and k-nearest neighbors (KNN), and they opted for tenfold cross-validation. The best results were obtained using TF with NB classifier, and by relying on light stemming they are up to 75% (accuracy).
- For validating their lexicon, Abdulla [40] proposed a SA algorithm for calculating SO by considering negation and intensification. For verifying this lexicon, they relied on the ArTwitter corpus that they also annotated manually. For showing the importance of the size of the glossary on the results, they carried out different experiments where a more critical part of a lexicon is used in each one. The best results were obtained with the totality of glossary, and they are up to 59.6% (accuracy). However, the authors also showed that the results obtained using a corpus-based approach

(relying on SVM and *K*-fold validation) outperforms the lexicon-based method with an accuracy of up to 87.2%.

- For evaluating their corpus, Alahmary et al. [41] employed the Word2vec model for learning vector representations of the words in an unsupervised way. Afterward, they applied two deep learning models which are long short-term memory (LSTM) and bidirectional long short-term memory (Bi-LSTM). They also compared the deep learning results to the results obtained from SVM, which is a well-known machine learning algorithm. LSTM classifier outperforms the other classifiers with an accuracy of up to 92%.
- For validating their corpus (ASTD), Nabil et al. [42] applied many classifiers such as SVM, Binomial NB (BNB), Logistic Regression (LR)… etc. on both balanced and unbalanced dataset. They used TF–IDF and *n*-gram for extracting features. The best accuracy that they obtained was up to 69.1%, and it was on the unbalanced corpus, using TF–IDF and SVM classifiers.

6. Negation in Arabic sentiment analysis:

Negation in the Arabic language is used to negate the idea of the sentence. There are two styles of negation: The first style uses negation terms, called explicit negation, the second style is implicit negation that does not use negation terms or words. Instead, some of the words or forms in a sentence carry a negation meaning, the scope of this work will be focused solely on one type of negation, explicit negation.

Negation in sentiment analysis is considered a real challenge, and researchers try hard to find solutions to this challenge in order to improve the performance of sentiment analysis. Little work has been undertaken in Arabic in order to address the issue of negation, either in the negation detection problem itself or the effect of negation in sentiment analysis.

We highlight bellow some related works to negation in Arabic sentiment analysis:

- Elhawary and Elfeky [43] considered the negation concept in their work. They relied on the Arabic lexicon to calculate the sentiment orientation score of each word or phrase. While the counting process is running, the negated word of the phrase is flipped. There are two main issues here in this work. Firstly, the authors did not mention the Arabic negation words used, stating only that they used around twenty words as negation words. Secondly, there is the issue of how they determined the negated words or phrases that come with the negation word in the sentence. This might affect the process of sentiment analysis, since it has the possibility of changing the polarity (i.e. its polarity type and strength). A further limitation of this work is that the sentiment orientation was calculated depending on the Arabic lexicon, rather than using machine learning to classify the sentiment.

- Ferret, al (2010) [44] also considered negation while attempting to capture the sentiment of Arabic text. The negation issue is considered in this work by only counting the frequency of the negation words in the sentence while attempting to build a semantic feature of the sentence depending on Arabic sentiment lexicon. The used features were the frequency of each positive, negative, neutral word, special character and the frequency of the negation words. The authors do not consider the ways in which words might be affected by the negation words. This resulted in a lower accuracy when compared to other methods used by the authors. As in the previous work, the authors here did not mention the list of negation words used. In addition, relying on a simple representation (i.e., frequency counts of negation words or polarity words) would not capture all the semantics and syntax of the sentence that might be useful in sentiment classification.
- Hamouda and El-Taher [45] attempted to build a sentiment analyzer for comments on Arabic Facebook news pages. They compared different machine learning algorithms with different features. One of these was dealing with negation in Arabic. They counted only five different negation words, whereas there are many more than these, even without counting negation words in the dialects. They only added the percentage of negation words in either the post or the comment as the feature, without considering the effect of negation on the word or phrase. They claimed that adding negation word features besides the features of all words in the posts and comments gives the best performance. The general issue here is that their proposed method may work only for the domain that they have chosen, which is the posts and

the comments in Arabic Facebook news pages. This might, or might not, work with regular Arabic sentiment analysis.[24]

7. Challenges and gaps in Arabic sentiment analysis:

Here, we highlight the major challenging issues encountered while conducting Arabic Sentiment Analysis:

- **Complex morphology:**

being a Semitic language, Arabic adopts the root-and pattern representation where a single set of consonants called the “root” is used to derive a variety of words by adding vowels (a,o,i) (أ, و , ي) or short vowels (diacritics) in addition to other consonants Habash [46]. The inflectional morphology, however, is observed through the ability of Arabic language to express a word in several grammatical categories while keeping the same meaning. The world's inflected forms can be obtained for several categories such as person, tense, voice (active/passive), number, gender, etc. Consequently, with such high derivational and inflectional morphology, handling Arabic texts through customizing current English SA systems and tools might be limited [46]. Thus, special preprocessing tasks supported by Arabic-oriented morphological analyzers should be combined in Arabic Sentiment Analysis systems.

- **Lack of resources:**

Lack of resources: despite the abundant online Arabic content, there is a lack of Arabic sentiment datasets and sentiment lexicons. During the last decade, some datasets have been constructed either for MSA or DA, nevertheless, the number of sentiment datasets which are publicly available remains little. Besides, most of these datasets do not have enough amounts of data which affects the evaluation of ASA systems when compared to English SA models since the sentiment analysis accuracy depends on the size of the manipulated data. On the other hand, the difficulties that accompany the construction and annotation process of sentiment lexicons have hindered the provision of large-scale and highly-coverage Arabic lexicons, especially with the existence of different Arabic dialects and domains.

- **Negation and sarcasm:**

Negation in Arabic is expressed using specific negation words which indicate the meaning “not”; some of them are: **“لا“ ,”ما”** and **“لا”**. Negation should be accurately detected and handled as it can convert the meaning of a sentence yielding a quite opposite polarity. This task becomes more difficult and challenging when dealing with DA where negation words are so different from formal MSA ones and have several meanings such as “مو” meaning “not” in the Levantine dialect that can be used for negation (e.g. **“السلطة مو تازة”**) or interrogative (e.g. **“تجي بكرة, مو ”**) which might mislead the sentiment classifier. Another ambiguity faced by ASA models is the sarcasm issue in which the explicit polarity totally opposites the meant sentiment as in e.g**”بعد الانتظار لساعتين،نفذت كل التذاكر، كم أنا محظوظ“**, where the word''محظوظ'', which means “lucky”, indicates a positive sentiment while in the example it actually refers to the opposite;

- **Arabizi usage**:

Arabizi is considered a newly-emerged Arabic variant written using the Arabic numeral system and Roman script characters. It is commonly used while expressing DA across social media and poses a challenge to sentiment analysis when it is mentioned along with Arabic (e.g. “**3an jad7 كتير الفلم elou”**.). This requires proper tools to interpret Arabizi into either MSA or DA before conducting the sentiment classification task;

Dialects variances: DA forms the majority of the online opinionated Arabic content as it is commonly used across social media platforms. DA combines various dialects which differ according to the geographical location. Each dialect has its own vocabulary, syntactic and grammatical rules in addition to special idioms. On the other hand, despite that all dialects are derived from MSA and hence do share some vocabulary, common words or expressions among two dialects might have drastically different sentiments. For example, “**يعطيك العافية**'' is a complement of a positive sentiment that means “May God grant your health” in the Levantine dialect, while this very same phrase has an aggressive meaning of “Burn in hell” in the Tunisian dialect. Considering these variances, an ASA system that targets one dialect might not be efficient for another as it is developed with a dialect-dependent tools such as the morphological analyzer, stopwords / negation words and sentiment lexicons.[13]

- **Agglutination of Arabic words:**

The agglutination of Arabic words could lead to errors when classifying sentiments. For example, the word”مَا كلَا”(meaning “food”) is translated to English as “Hedid not eat” but in the sentence “مَا كلَا مَا شرَب”the meaning is “he did not eat, and he did not drink”. When”مَا كلَا''is used as a word, it could not be separated and thetis equivalent to exactly one word and one meaning. However, in the sentence, the word” مَا كلَا”could be separated as ” مَا ”+”كلَا” (where ” مَا” represents a negative pronouns). Hence both use cases of the same word have different polarities depending on the context where the word is used.

8. Conclusion:

Through this chapter, we have seen an overview of Arabic language, we also seen the root and pattern in Arabic language, then we had talked about challenges of Arabic in natural language processing, after that we had seen the common works related to Arabic sentiment analysis, and the negation in Arabic language, finally we have discussed the main challenges and gaps in Arabic sentiment analysis.

CHAPTER 3:

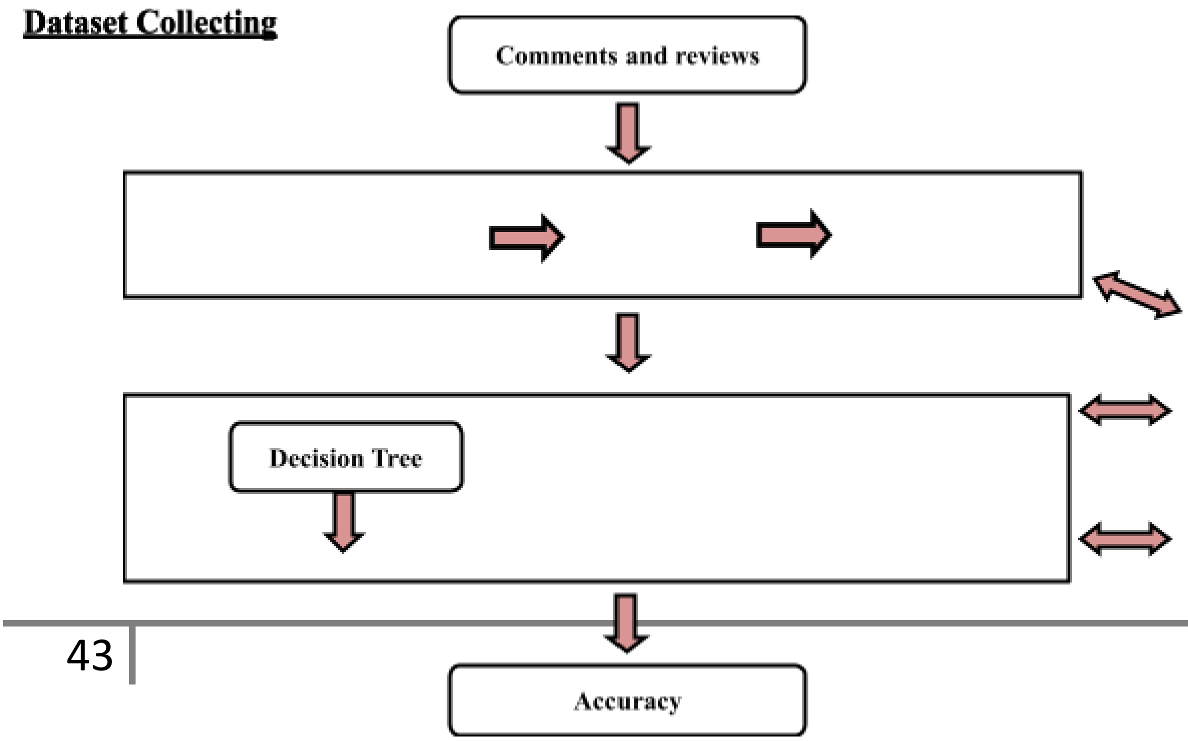
IMPLEMENTATION

1. Introduction :

In this chapter we will present the architecture of the adopted sentiment analysis model. This includes several key points mainly: the dataset collection, preprocessing operations, scoring and sentiment calculation, and performance evaluation. The sentiment calculation involves comparison of two classifiers (Decision Tree and Naïve Bayes) with different situations (before pre-processing, after removing stopwords and after stemming).Finally, we will present the implementation work while discussing the obtained results.

2. Architecture of the adopted sentiment analysis model :

This work presents the architecture of the adopted sentiment analysis model. The architecture contains four main phases namely: dataset collection, preprocessing operations, scoring and sentiment classification, and performance evaluation. In our work, the sentiment calculation involves comparison of two classifiers (Decision Tree and Naïve



Bayes) with different situations (before pre-processing, after removing stopwords and after stemming).

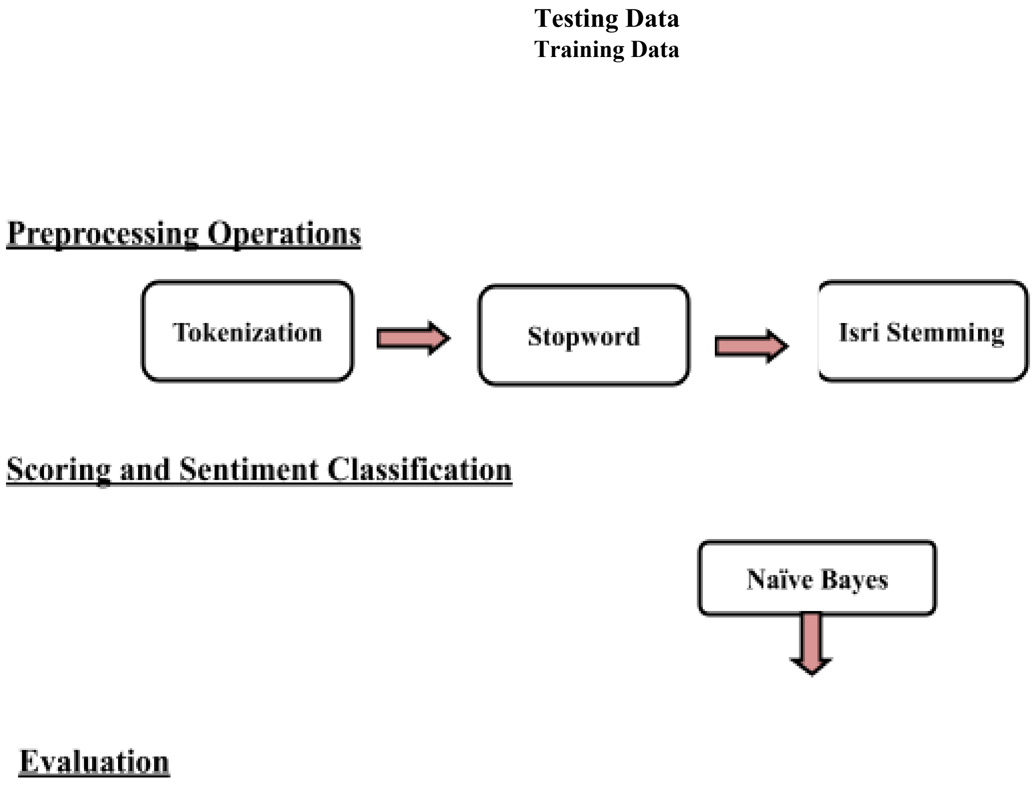


Figure 3.1: Architecture of the Adopted Sentiment Analysis Model

2.1 Dataset Collection

To operate and test a dataset for sentiment analysis, it is normal to adopt three steps mainly: perform web crawling, filter the reviews, and extract some attributes from each reviewer. The collection of data is important to compile the dataset from users` reviews. The dataset is considered as input to the preprocessing phase. In our case, we have created the dataset manually, which is reviews about translation program, this data is either positive or negative, the data is written in standard Arabic type, we have divided it to two categories : the training data used to train the classifiers and the testing data used for the test and obtain the results.

2.2. The Preprocessing Operations

This phase is focused on text preprocessing which is important for sentiment analysis. This phase presents several operations mainly: tokenization, removing stopwords and light stemming. The function of each operation is briefly mentioned as follows:

2.2.1 Tokenization of Arabic Text

Tokenization is an important step and is performed for each comment in the adopted dataset.

Tokenization divides and/or splits any comment into multiple tokens or words based on whitespace character or any delimiter. The tokens may be number, article, preposition, modifier, punctuation, or any word of different types (nouns, verbs, adjectives, adverbs,…etc.).

2.2.2 Removing stopwords

Stopwords are those words that are not important in calculating a term weighting. The stopwords are rejected or removed as they do not have any effect on identifying the sentiment classification. The stopwords are sometimes called noise words and they may be prepositions, pronouns, modifiers, specifying tools, and other tools. The stopwords mentioned in Table 4.1 are considered a part of stopwords that exist in the Arabic Language (MSA and Colloquial). The original list of Arabic stop words contains about 607 stopwords.

عن	أمام	إلى	تلك	التي	هذا
كانوا	خلف	على	ذلكما	اللذان	هذه
كن	كان	في	تلكما	اللتان	هذان
هو	كانت	حتى	ذلكم	اللاتي	هاتان
هي	كانا	فوق	تلكم	اللائي	هؤلاء
هم	كانتا	تحت	من	ذلك	الذى
...	...	...	...	...	...

Table 3.1*:* Examples of Some Arabic Stopwords

2.2.3 Normalization

Normalization is important for a good accurate sentiment analysis model. Normalization is concerned with removing a set of letters. This Includes removing punctuations, weak vowels, removing elongations (or repeated characters), and removing non-letters like numbers. Normalization also is concerned with replacing some letters by others. Letters like (alef maddah), (alefHamza ,(أ and (alefkasraإ) can be replaced with (alefا). The Same thing also can take place by replacing final (taamarbutaة) by (haamarbutaه) and so on. [14], [13],[2].

2.2.4 The Arabic Light Stemming

Light stemming is simple and proves to be effective in a lot of computational linguistic applications. A light stemmer is adapted to suit the task of sentiment analysis. The light stemmer removes both the Arabic prefixes and the ArabicAffixes.[14], [13], [2]. Table 2 shows examples of some Arabic prefixes, suffixes, and affixes.

Prefix	Prefix	Suffix	Suffix	Affix	Affix
الدنيا	خلف	على	ذلكما	اللذان	هذه
يعمل	ناكل	صالحين	شربوا	ليقولون	يتكلمون
فاليعلم	يستسلم	كتابك	معطفك	فادخلوا	يستغفرون
كالصخر	استغفر	لعبتكم	سبيلكم	بايديكم	نشكركم
...	...	...	...	...	...

Table 3.2: Examples of Some Arabic Prefixes, Suffixes, and Affixes

2.3. Scoring and Sentiment Calculation

This step involves several important themes for the adopted sentiment analysis approach.This Includes Comparison of two classifiers (Decision Tree and Naïve Bayes) with different situations (before pre-processing, after removing stopwords and after stemming).

3. Implementation work

To evaluate the performance of the adopted sentiment analysis model, it should be operated and tested. The model was implemented using Python and some natural language utilities that were used for some preprocessing operations such as tokenization, removing stopwords and light stemming.

The model was run on a laptop with 4-GB memory size and 2.50 GHZ processing speed. The model also was run and tested using datasets divided to two categories: training data and testing data, as mentioned before. A brief descriptive statistic of the adopted datasets is mentioned as shown in Table 3.3.

The project" Arabic slang language"[47] was created by IBRAHIMAlsharif,the Quality AssuranceManager Projects Manager from Palestine, This example compares three classifiers (Decision Tree, Naïve Bayes and Maximum Entropy) with different situations (before pre-processing, after removing stopwords and after stemming) .

DATA	Positive training data	Positive testing data	Negative training data	Negative testing data
Number of comments	43	10	45	10

Table 3.3: Descriptive Statistics of the Adopted Datasets

- **Python:**

Python is an interpreter, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typinganddynamicbinding;makeitveryattractiveforRapidApplicationDevelopment, as well as for use as a scripting or glue language to connect existing components together.

Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed [20]

- **Python libraries:**

Python library is a collection of functions and methods that allows you to perform many actions without writing your code. Each library in Python contains a huge number of useful modules that you can import for your everyday programming. For example Scikit-learn, NLTK ,NumPy. . .etc. but in our work we have used NLTK and Textblob.

- **Natural Language Toolkit (NLTK):**

NLTK is a leading platform for building python programs to work with human language data ,it provides easy to use interfaces to over 50 corpora and lexical resources such as wordnet, along with a suite of text processing libraries for classification ,tokenization ,stemming, tagging, parsing, and semantic reasoning ,wrappers for industrial-strength NLP libraries ,and an active discussion forum.

NLTK with python provides a practical introduction to programming for language processing .written by creators of NLTK, it guides the reader through the fundamentals of writing python programs, working with corpora, categorizing text, analyzing linguistic structure, and more.[48]

- **TextBlob:**

TextBlobis a Python library for processing textual data. It provides a simple API for diving into common natural language processing (NLP) tasks such as part-of-speech tagging, noun phrase extraction, sentiment analysis, classification, translation, and more.

The good thing about textblob is that it is just like python strings;you can transform and play with it.

- **Anaconda:**

Anaconda is a conditional free and open source distribution of python and R programming languages for scientific computing (data science, machine learning applications ...etc), that aims to simplify package management and deployment. The distribution includes data science packages suitable for windows, linux, and macOS . it is developed and maintained by anaconda, Inc., which was founded by Peter Wang and Travis Oliphant in 2012 as an Anaconda, Inc. product ,it is also known as Anaconda Distribution or Anaconda Individual Edition ,while other product for the company are Anaconda Team Edition and Anaconda Enterprise Edition, both of which are not free.[wiki]

- **Jupyter Notebook:**

The Jupyter Notebook is an open source web application that allows you to create and share documents that contain live code, equations, visualization and narrative text .uses include data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning, and much more.

Jupyter supports over 40 programming languages, including python, R, Julia, and scala

Notebooks can be shared with others using email, Dropbox, Github and the Jupyter notebook Viewer.

The code can produce rich, interactive output: HTML, images, videos,LaTex, and custom MIME types.

4.Discussion of results

Results :

1) **Decision Tree classifier:**

Test Data	**Accuracy** (before pre-processing of training data)	**Accuracy** (Stop words removed from training data)	**Accuracy** (After stemming of training data)
Training array it self	1.0	1.0	1.0
Training array after removing stop words	1.0	1.0	1.0
Training array after stemming	1.0	1.0	1.0
Test array	0.54	0.54	0.54
Test array after removing stop words	0.54	0.54	0.54
Test array after stemming	0.54	0.54	0.54

Table 3.4 :results using decision tree classifier

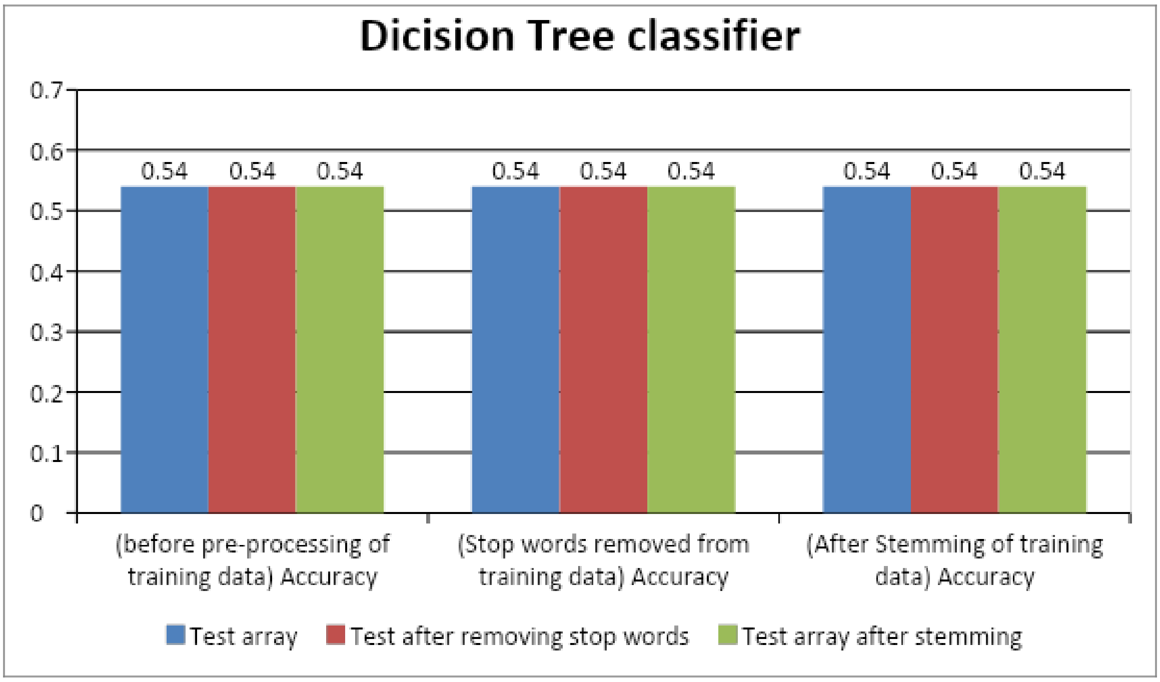


Figure 3.2: Histogram of results using Decision tree classifier

2) Naïve Bayes classifier:

Test Data	**Accuracy** (before pre-processing of training data)	**Accuracy** (Stop words removed from training data)	**Accuracy** (After stemming of training data)
Training array it self	0.95	0.97	0.71
Training array after removing stop words	0.91	0.97	0.70
Training array after stemming	0.66	0.69	0.97
Test array	0.54	0.5	0.5
Test array after removing stop words	0.45	0.5	0.5
Test array after stemming	0.45	0.45	0.58

Table 3.5: results using naïve Bayes classifier

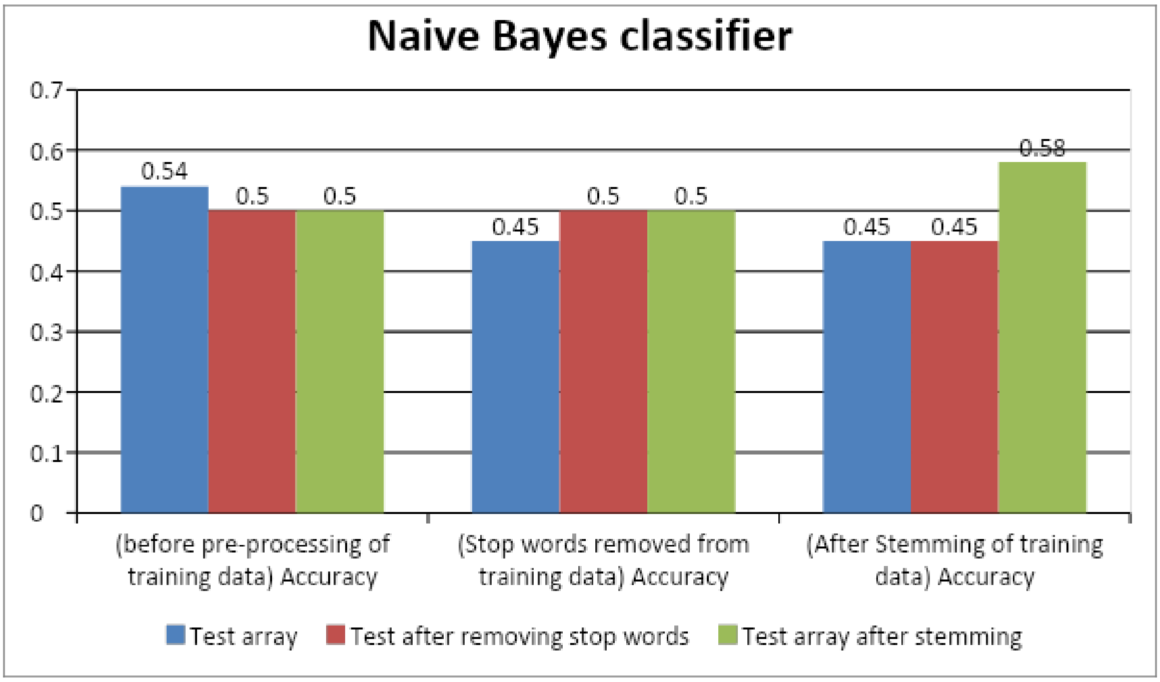


Figure 3.3: Histogram of results using naïve Bayes classifier

Discussion:

From the experimental work, it was noticed that: the sentiment model behaves better using The decision tree classifier,this is clear from the obtained results for the measuring criteria: accuracy.

The behavior of the sentiment model was high using training data with decision tree classifiers in all cases (before pre-processing, after removing stopwords and after stemming) with accuracy up to 100%. And 50% accuracy obtained using testing data in all cases (before pre-processing, after removing stopwords and after stemming).

The accuracy using naïve Bayes classifier was low compared with the accuracy using decision tree classifier, it was between (97 % - 66%) using training and in all cases (before pre-processing, after removing stopwords and after stemming) , and it was between (45% - 58%) using testing data in all cases (before pre-processing, after removing stopwords and after stemming).

The behavior of the sentiment model was improved after stemming using naïve Bayes classifier where the accuracy was up to 58%. But in other cases the accuracy was less than the accuracy obtained using the decision tree classifier.

4.Conclusion

In this chapter we had presented the architecture of the adopted sentiment analysis model..we also had seen the comparison of two classifiers (Decision Tree and Naïve Bayes) with different situations (before pre-processing, after removing stopwords and after stemming).Finally, we had presented the implementation work while discusses the obtained results.

GENERAL CONCLUSION

General conclusion

This dissertation discussed the methods for sentiment classification and especially the corpus-based approach and comparison of two classifiers (Decision tree and naive Bayes) in different situations(before preprocessing, after removing stopwords and after stemming).

It is concluded that the Decision tree classifier obtained the highest accuracy compared to naive bayes classifier.

These two classifiers are commonly used by many researches for comparing their proposed work. After finishing this work, it is very clear that the expansions in sentiment classification and feature selection algorithms are still required and hence an open area of research.

For sentiment analysis data is taken from blogs, social media websites like Facebook, Twitter, Amazon, flip kart etc. People freely express their view on these media about certain topics, products, and politics. By analyzing these reviews one can extract the information about their area and can do improvement. Since so much research has been done in the field of sentiment analysis, still it faces many challenges. Sometimes people express their views in a sarcastic way that is hard to detect. Due to these challenges, sentiment analysis still remains an area of research. In order to improve the classification result deep data analysis is required based on context.

in the case of the Arabic language. Arabic is considered a rich morphology language that may need new approaches to achieve the sentiment classification tasks. These limitations, characteristics and challenges of Arabic lead to the need for new Arabic sentiment resources, proposing and studies involving new features in a suitable manner, comparisons of Machine Learning classifiers, and dealing with the central sentiment influential factor, negation.

BIBLIOGRAPHICAL REFERENCES

Bibliographical References

[1] Wikipedia, sentiment analysis, September 2020

https://en.wikipedia.org/wiki/Sentiment_analysis

[2] Dipti Sharma, Dr. MunishSabharwal, Dr. VinayGoyal, and Dr. MohitVij, Sentiment Analysis Techniques for Social Media Data: A Review, September 2019

[3]Mohsen Farhadloo, Erik Rolland, Fundamentals of Sentiment Analysis and Its Applications, March 2016.

[4]ItiChaturvedi, Erik Cambria, Roy E.Welsch, Francisco Herrera, Distinguishing between facts and opinions for sentiment analysis: Survey and challenges, ELSEVIER Volume 44, November 2018, Pages 65-77.

[5]Ms. Puja M. Dadhe and Dr. R.N. Jugele, Inspection of Retrospection - Challenges in Sentiment Analysis, Compliance Engineering Journal, Volume 11, Issue 1, 2020 Page No: 123-128.

[6]monkeylearn , sentiment analysis, 2020

https://monkeylearn.com/sentiment-analysis/

[7] SeemaKolkur, GayatriDantal, ReenaMahe, Study of Different Levels for Sentiment Analysis, International Journal of Current Engineering and Technology ,2015, Volume 5, Issue 2 , Page No:768 – 770.

[8] Osamaha.mghaleb, annasarovijendran, the challenges of sentiment analysis on social web communities, International Journal of Advance Research in science and engineering, Volume 06, Issue 12, 2017, page N°: 117- 125.

[9] Vishal A. Kharde, S.S. Sonawane, Sentiment Analysis of Twitter Data: A Survey of Techniques, International Journal of Computer Applications, Volume 139 – No.11, April 2016.

[10] Madhav Singh Solanki , Sentiment Analysis of Text using Rule Based and Natural language Toolkit, International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-8 Issue-12S, October 2019

[11] AshishKatrekar, An Introduction to Sentiment Analysis ,global logic https://www.globallogic.com/wp-content/uploads/2019/12/Introduction-to-Sentiment-Analysis.pdf

[12] Erik Cambria, An Introduction to Concept-Level Sentiment Analysis

[13] HalaMulki, Hatem Haddad, Ismail Babaoglu, Modern Trends in Arabic Sentiment Analysis: A Survey, TAL. Volume 58 N°3, 2017, page N° 15 – 39.

[14] SadikBessou, Rania Aberkane, Subjective Sentiment Analysis for Arabic Newswire Comments, Journal of Digital Information Management, Volume 17 N° 5, 2019, PP 289 – 295.

[15] AsmaaMountassir, HoudaBenbrahim, and IlhamBerrada, Sentiment Classification on Arabic corpora: A preliminary cross-study, Document numérique, Volume 16 – n° 1/2013, PP 73 – 96 DOI: 10.3166/dn.16.1.73-96

[16] RavinderAhuja, Conference on Pervasi, shrutikuhli, shauryagupta, pratyushahuja, The Impact of Features Extraction on the Sentiment Analysis, Procedia Computer Science152, 2019.

[17] Wikipedia, feature engineering, https://en.wikipedia.org/wiki/Feature_engineering

[18] Avinash M and Sivasankar, A Study of Feature Extraction techniques for Sentiment Analysis, 2019 arXiv:1906.01573v1

[19] JørgenFaret, Johan Reitan, Twitter Sentiment Analysis: Exploring the Effects of Linguistic Negation, 2015.

[20] SahnouneChaouche Farouk, HouachemiAmira, Automatic Text Classification: a comparative study of three algorithms, 2019

[21]ChetashriBhadane,HardiDalal, HeenalDoshi, Sentiment analysis: Measuring opinions,Procedia Computer Science 45 (2015) 808 – 814.

[22] Lama SalehAlsudias, Arabic Blogging Sentiment Analysis, 2014.

[23] ITANI Maher, ROAST Chris, ALKHAYATT Samir, Corpora for sentiment analysis of Arabic text in social media, In: 8th International Conference on Information and Communication Systems (ICICS), 2017. IEEE, 64-69.

[24] Saud SalehAlotaibi, Sentiment Analysis in the Arabic Language Using Machine Learning, 2015

[25] softschool, roots and pattern , 2020
https://www.softschools.com/languages/arabic/roots_and_patterns/

[26] KhaledShaalan, SanjeeraSiddiqui, ManarAlkhatib ,Azza Abdel Monem, Challenges in Arabic Natural Language Processing, 2018

[27] Abdul-Mageed M, Diab MT (2012b) Awatif: a multi-genre corpus for modern standard Arabic subjectivity and sentiment analysis. In: LREC. Citeseer, pp 3907–3914

[28] Medhaffar S, Bougares F, Esteve Y, Hadrich-Belguith L (2017) Sentiment analysis of Tunisian dialects: Linguistic resources and experiments. In: Proceedings of the third Arabic natural language processing workshop, pp 55–61

[29] Abdul-Mageed M, Diab M, Kübler S (2014) Samar: subjectivity and sentiment analysis for Arabic social media. Comput Speech Lang 28(1):20–37

[30] Alharbi FR, Khan MB (2019), Identifying comparative opinions in Arabic text in social media using machine learning techniques. SN ApplSci 1(3):213

[31] Aly M, Atiya A (2013) Labr: a large scale Arabic book reviews dataset. In: Proceedings of the 51st annual meeting of the association for computational linguistics (volume 2: short papers), vol 2, pp 494–498.

[32] ElSahar H, El-Beltagy SR (2015) Building large Arabic multi-domain resources for sentiment analysis. In: International conference on intelligent text processing and computational linguistics. Springer, pp 23–34

[33] Guellil I, Adeel A, Azouaou F, Hussain A (2018b) Sentialg: Automated corpus annotation for Algerian sentiment analysis. arXiv preprint arXiv :1808.05079

[34] Guellil I, Azouaou F, Saâdane H, Semmar N (2017c) Une approche fondée sur les lexiques d'analyse de sentiments du dialecte algérien.

[35] Gamal D, Alfonse M, El-Horbaty ESM, Salem ABM (2019) Twitter benchmark dataset for Arabic sentiment analysis. Int J Mod EducComputSci 11(1):33

[36] Al-Twairesh N, Al-Khalifa H, Al-Salman A, Al-Ohali Y (2017) Arasenti- tweet: a corpus for Arabic sentiment analysis of Saudi tweets. ProcediaComputSci 117:63–72

[37] Rushdi-Saleh M, Martيn-Valdivia MT, Ureٌa-Lَpez LA, Perea-Ortega JM (2011) Oca: opinion corpus for Arabic. J AssocInfSciTechnol 62(10):2045–2054

[38] Abdul-Mageed M, Diab M, Kübler S (2014) Samar: subjectivity and sentiment analysis for Arabic social media. Comput Speech Lang 28(1):20–37

[39]Rahab H, Zitouni A, Djoudi M (2019) Sana: sentiment analysis on newspapers comments in Algeria. J King Saud UnivComputInfSci.https ://doi.org/10.1016/j.jksuc i.2019.04.012

[40]Abdulla NA, Ahmed NA, Shehab MA, Al-Ayyoub M (2013) Arabic sentiment analysis: lexicon-based and corpus-based. In: 2013 IEEE Jordan conference on applied electrical engineering and computing technologies (AEECT). IEEE, pp. 1–6

[41] Alahmary RM, Al-Dossari HZ, Emam AZ (2019) Sentiment analysis of Saudi dialect using deep learning techniques. In: 2019 international conference on electronics, information, and communication (ICEIC). IEEE, pp 1–6

[42] Nabil M, Aly M, Atiya A (2015) Astd: Arabic sentiment tweets dataset. In: Proceedings of the 2015 conference on empirical methods in natural language processing, pp 2515–2519

[43] M. Elhawary and M. Elfeky. Mining arabic business reviews. In Data Mining Workshops (ICDMW), 2010 IEEE International Conference on, pages 1108 {1113, dec. 2010.

[44]N. Farra, E. Challita, R. A. Assi, and H. Hajj. Sentence-level and document-level sentiment mining for arabic texts. In Data Mining Workshops (ICDMW), 2010 IEEE International Conference on, pages 1114 {1119, dec. 2010.

[45] Alaa El-Dine Ali Hamouda, Fatma El-zahraa El-taher, Sentiment Analyzer for Arabic Comments System, International Journal of Advanced Computer Science and Applications, Vol. 4, No.3, 2013

[46] Habash N. Y., “Introduction to Arabic natural language processing”, Synthesis Lectures on Human Language Technologies, vol. 3, no 1, p. 1-187, 2010.

[47] ImaneGuellil, FaicalAzouaou, Marcelo Mendoza, Arabic sentiment analysis: studies, resources, and tools, Social Network Analysis and Mining, 9(1)-56, Springer, https://doi.org/10.1007/s13278-019-0602-x

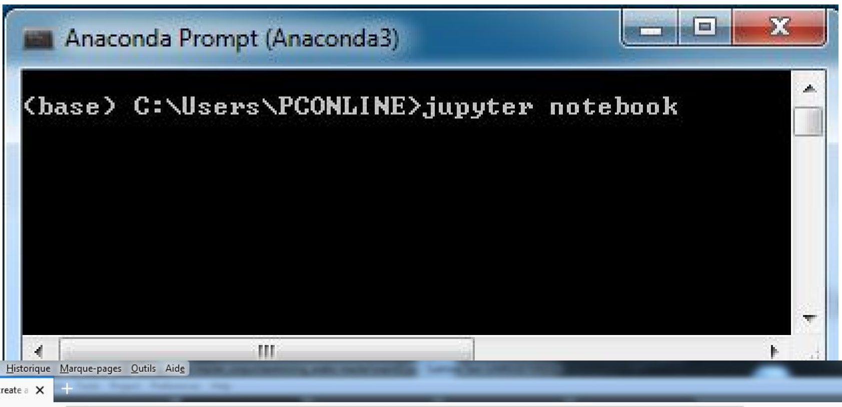
[47]https://github.com/IbrahimAlsharif/textmining_arabic

ANNEXES

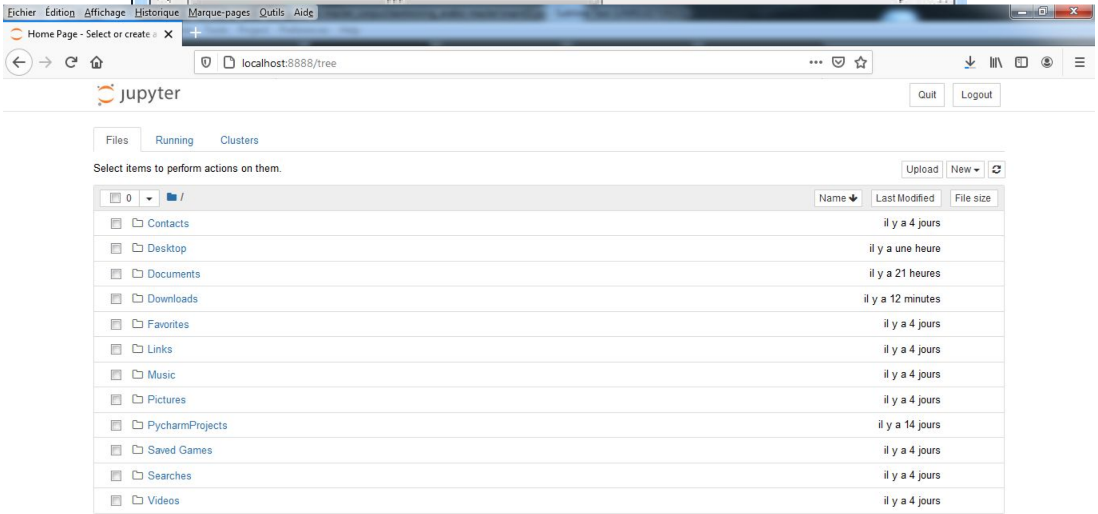
Annexes

Run the project :

- In order to Start Jupyter Notebook type "`Jupyter notebook`"command in the anaconda prompt.



```
(base) C:\Users\PCONLINE>jupyter notebook
```

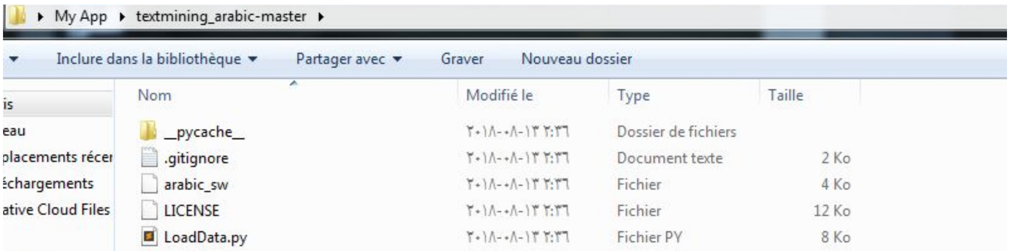


Select items to perform actions on them.

Name	Last Modified
Contacts	il y a 4 jours
Desktop	il y a une heure
Documents	il y a 21 heures
Downloads	il y a 12 minutes
Favorites	il y a 4 jours
Links	il y a 4 jours
Music	il y a 4 jours
Pictures	il y a 4 jours
PycharmProjects	il y a 14 jours
Saved Games	il y a 4 jours
Searches	il y a 4 jours
Videos	il y a 4 jours

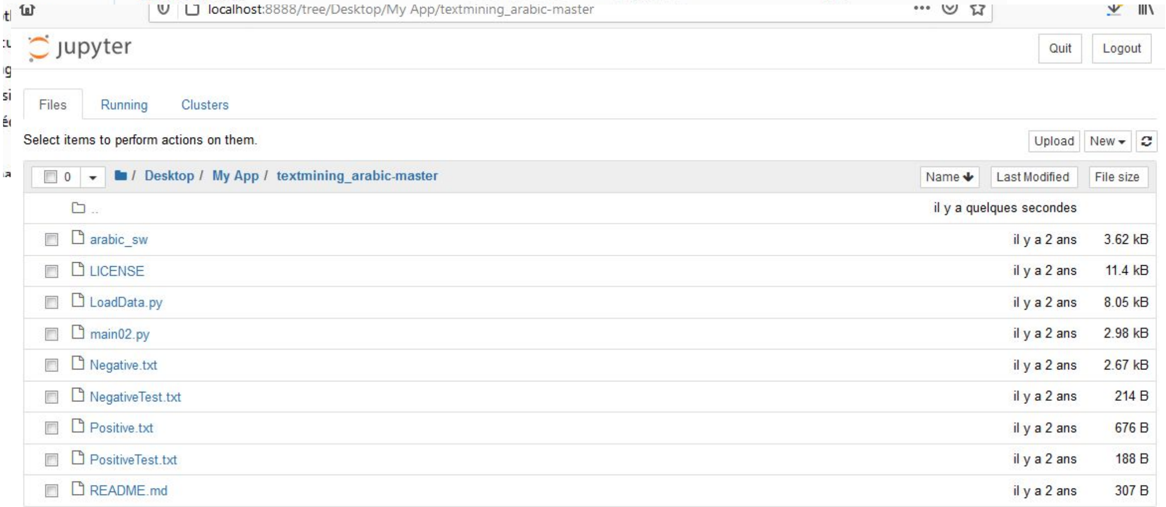
- The anaconda prompt will direct you to your browser and open a window with this link http://localhost:8888/tree and with this content.

- For me I created a new folder on my desktop with “My App” name, in this folder I extracted textmining_arabic-master.zip and this is the content of it:



My App ▸ textmining_arabic-master ▸

Nom	Modifié le	Type	Taille
__pycache__	٢٠١٨-٠٨-١٣ ٢:٣٦	Dossier de fichiers	
.gitignore	٢٠١٨-٠٨-١٣ ٢:٣٦	Document texte	2 Ko
arabic_sw	٢٠١٨-٠٨-١٣ ٢:٣٦	Fichier	4 Ko
LICENSE	٢٠١٨-٠٨-١٣ ٢:٣٦	Fichier	12 Ko
LoadData.py	٢٠١٨-٠٨-١٣ ٢:٣٦	Fichier PY	8 Ko



localhost:8888/tree/Desktop/My App/textmining_arabic-master

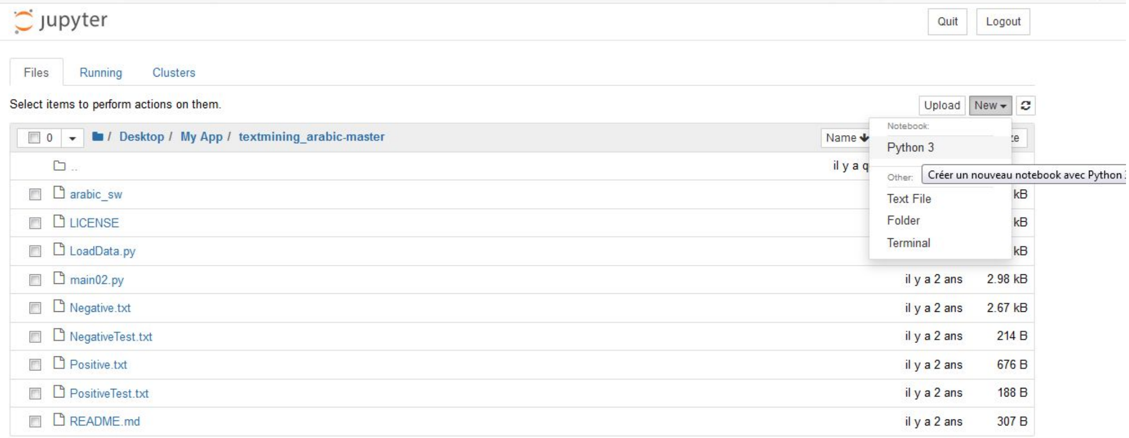
Select items to perform actions on them.

□ 0 ▾ / Desktop / My App / textmining_arabic-master

Name ↓	Last Modified	File size
..	il y a quelques secondes	
□ arabic_sw	il y a 2 ans	3.62 kB
□ LICENSE	il y a 2 ans	11.4 kB
□ LoadData.py	il y a 2 ans	8.05 kB
□ main02.py	il y a 2 ans	2.98 kB
□ Negative.txt	il y a 2 ans	2.67 kB
□ NegativeTest.txt	il y a 2 ans	214 B
□ Positive.txt	il y a 2 ans	676 B
□ PositiveTest.txt	il y a 2 ans	188 B
□ README.md	il y a 2 ans	307 B

- In my jupyter notebook page which is opened on my browser I select Desktop, then My App, after that I select textmining_arabic-master folder where I can easily find my extracted files.

- Now from the “New” list I chose python3 which allowed me to create a new Jupyter notebook.

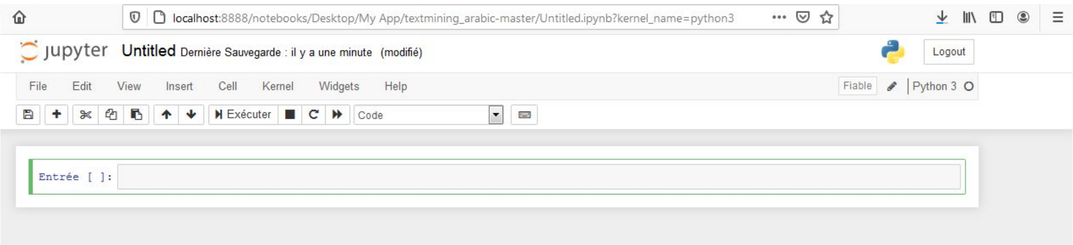


Select items to perform actions on them.

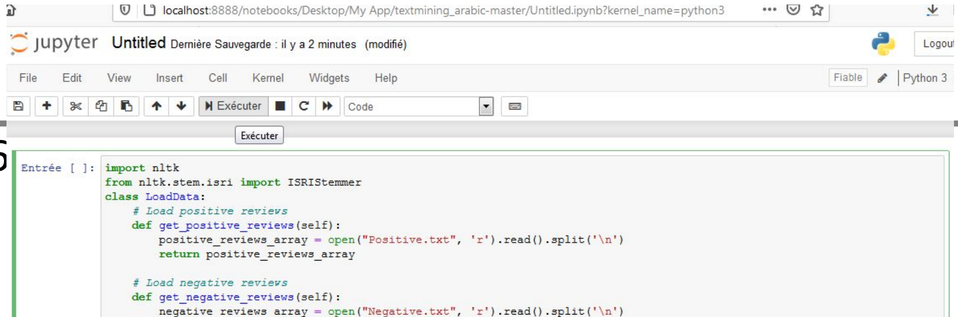
/ Desktop / My App / textmining_arabic-master

..	il y a q	
arabic_sw		kB
LICENSE		kB
LoadData.py		kB
main02.py	il y a 2 ans	2.98 kB
Negative.txt	il y a 2 ans	2.67 kB
NegativeTest.txt	il y a 2 ans	214 B
Positive.txt	il y a 2 ans	676 B
PositiveTest.txt	il y a 2 ans	188 B
README.md	il y a 2 ans	307 B

It will direct me to a new window, in this window I will execute the code


localhost:8888/notebooks/Desktop/My App/textmining_arabic-master/Untitled.ipynb?kernel_name=python3
jupyter
Untitled Dernière Sauvegarde : il y a une minute (modifié)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Fiable
Python 3
Exécuter
Code
Entrée [ ]:

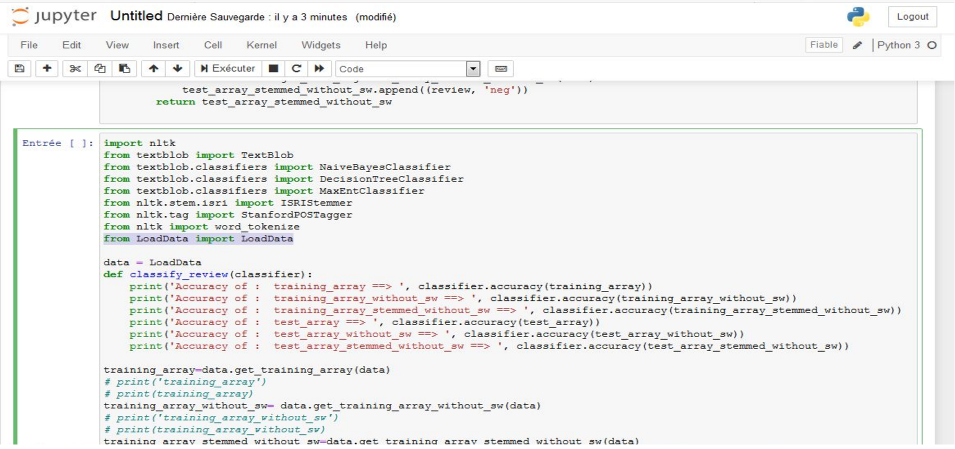
- So in the first stage I put my LoadData code (copy and paste from my LoadData.py), then hit the executor.


localhost:8888/notebooks/Desktop/My App/textmining_arabic-master/Untitled.ipynb?kernel_name=python3
jupyter
Untitled Dernière Sauvegarde : il y a 2 minutes (modifié)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Fiable
Python 3
Exécuter
Code
Exécuter
Entrée [ ]:

```
import nltk
from nltk.stem.isri import ISRIStemmer
class LoadData:
    # Load positive reviews
    def get_positive_reviews(self):
        positive_reviews_array = open("Positive.txt", 'r').read().split('\n')
        return positive_reviews_array

    # Load negative reviews
    def get_negative_reviews(self):
        negative_reviews_array = open("Negative.txt", 'r').read().split('\n')
```

- In the second stage I put my main code (copy and paste from my main.py), then hit the executor.
- Note: in this part I shall delete this line: `from LoadData import LoadData` from my main code because we're working in the same project so we don't have to import the code executed In the first stage.



jupyter Untitled Dernière Sauvegarde : il y a 3 minutes (modifié)
Logout
File Edit View Insert Cell Kernel Widgets Help
Fiable Python 3
Exécuter Code
Entrée [ ]:

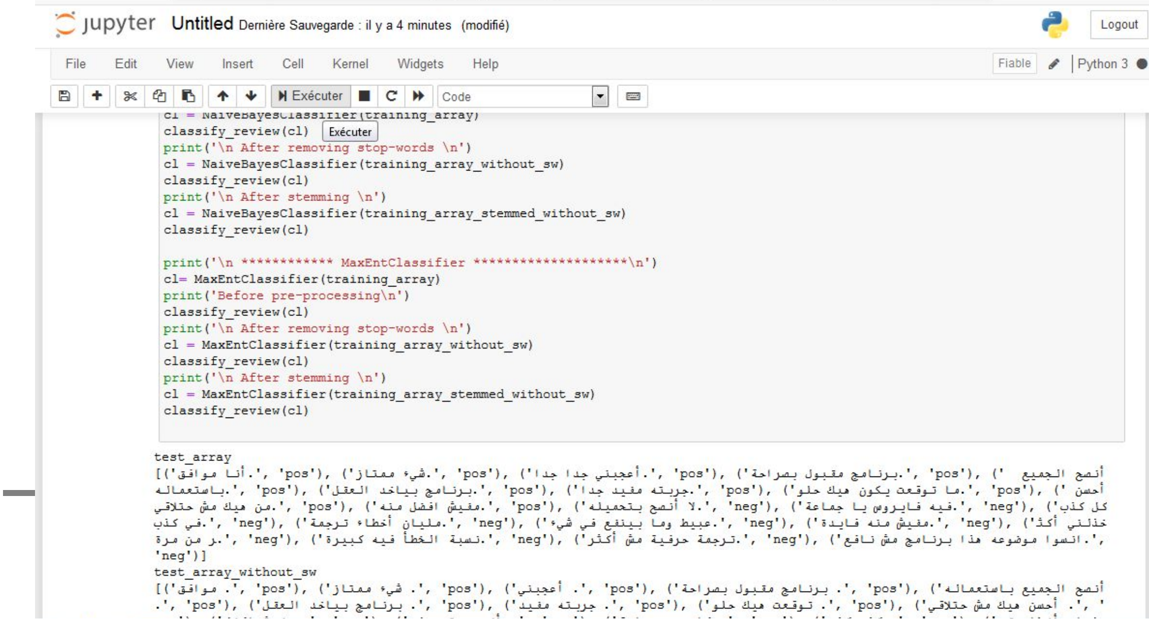
```
        test_array_stemmed_without_sw.append((review, 'neg'))
    return test_array_stemmed_without_sw
```

```
import nltk
from textblob import TextBlob
from textblob.classifiers import NaiveBayesClassifier
from textblob.classifiers import DecisionTreeClassifier
from textblob.classifiers import MaxEntClassifier
from nltk.stem.isri import ISRIStemmer
from nltk.tag import StanfordPOSTagger
from nltk import word_tokenize
from LoadData import LoadData

data = LoadData
def classify_review(classifier):
    print('Accuracy of :  training_array ==> ', classifier.accuracy(training_array))
    print('Accuracy of :  training_array_without_sw ==> ', classifier.accuracy(training_array_without_sw))
    print('Accuracy of :  training_array_stemmed_without_sw ==> ', classifier.accuracy(training_array_stemmed_without_sw))
    print('Accuracy of :  test_array ==> ', classifier.accuracy(test_array))
    print('Accuracy of :  test_array_without_sw ==> ', classifier.accuracy(test_array_without_sw))
    print('Accuracy of :  test_array_stemmed_without_sw ==> ', classifier.accuracy(test_array_stemmed_without_sw))

training_array=data.get_training_array(data)
# print('training_array')
# print(training_array)
training_array_without_sw= data.get_training_array_without_sw(data)
# print('training_array_without_sw')
# print(training_array_without_sw)
training_array_stemmed_without_sw=data.get_training_array_stemmed_without_sw(data)
```

- After a few seconds the results would appear.

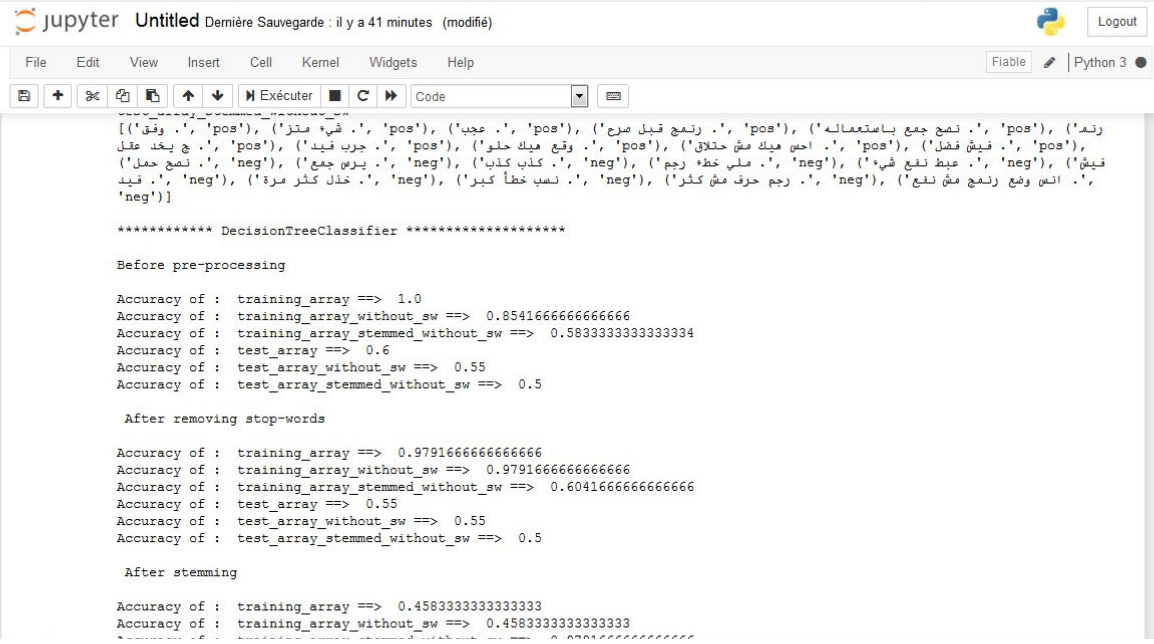


jupyter Untitled Dernière Sauvegarde : il y a 4 minutes (modifié)
Logout
File Edit View Insert Cell Kernel Widgets Help
Fiable Python 3
Exécuter Code
Exécuter

```
cl = NaiveBayesClassifier(training_array)
classify_review(cl)
print('\n After removing stop-words \n')
cl = NaiveBayesClassifier(training_array_without_sw)
classify_review(cl)
print('\n After stemming \n')
cl = NaiveBayesClassifier(training_array_stemmed_without_sw)
classify_review(cl)

print('\n ************ MaxEntClassifier ********************\n')
cl= MaxEntClassifier(training_array)
print('Before pre-processing\n')
classify_review(cl)
print('\n After removing stop-words \n')
cl = MaxEntClassifier(training_array_without_sw)
classify_review(cl)
print('\n After stemming \n')
cl = MaxEntClassifier(training_array_stemmed_without_sw)
classify_review(cl)
```

```
test_array
[('أنا موافق.', 'pos'), ('شيء ممتاز.', 'pos'), ('أعجبني جدا جدا.', 'pos'), ('برنامج مقبول بصراحة.', 'pos'), (' أنصح الجميع
باستعماله.', 'pos'), ('برنامج بياخد العقل.', 'pos'), ('جربته مفيد جدا.', 'pos'), ('ما توقعت يكون هيك حلو.', 'pos'), ('أحسن
من هيك مش حتلاقي.', 'pos'), ('مفيش افضل منه.', 'pos'), ('لا أنصح بتحميله.', 'neg'), ('فيه فايروس يا جماعة.', 'neg'), ('كل كذب
في كذب.', 'neg'), ('مليان أخطاء ترجمة.', 'neg'), ('عبيط وما بينفع في شيء.', 'neg'), ('مفيش منه فايدة.', 'neg'), ('خذلني أكث
ر من مرة.', 'neg'), ('نسبة الخطأ فيه كبيرة.', 'neg'), ('ترجمة حرفية مش أكثر.', 'neg'), ('انسوا موضوعه هذا برنامج مش نافع.',
'neg')]
test_array_without_sw
[('موافق .', 'pos'), ('شيء ممتاز .', 'pos'), ('أعجبني .', 'pos'), ('برنامج مقبول بصراحة .', 'pos'), ('أنصح الجميع باستعماله
.', 'pos'), ('برنامج بياخد العقل .', 'pos'), ('جربته مفيد .', 'pos'), ('توقعت هيك حلو .', 'pos'), ('أحسن هيك مش حتلاقي .', '
```



```
[('وقق .', 'pos'), ('شيء متز .', 'pos'), ('عجب .', 'pos'), ('رنمج قبل صرح .', 'pos'), ('نصح جمع باستعماله .', 'pos'), ('رنم
عقل ج يخد .', 'pos'), ('جرب فيد .', 'pos'), ('وقع هيك حلو .', 'pos'), ('احس هيك مش حتلاق .', 'pos'), ('فيش فضل .', 'pos'),
('نصح حمل .', 'neg'), ('يرس جمع .', 'neg'), ('كذب كذب .', 'neg'), ('ملي خطء رجم .', 'neg'), ('عبط نفع شيء .', 'neg'), ('فيش
فيد .', 'neg'), ('خذل كثر مرة .', 'neg'), ('نسب خطأ كبر .', 'neg'), ('رجم حرف مش كثر .', 'neg'), ('انس وضع رنمج مش نفع .',
'neg')]

************ DecisionTreeClassifier ********************

Before pre-processing

Accuracy of :  training_array ==>  1.0
Accuracy of :  training_array_without_sw ==>  0.8541666666666666
Accuracy of :  training_array_stemmed_without_sw ==>  0.5833333333333334
Accuracy of :  test_array ==>  0.6
Accuracy of :  test_array_without_sw ==>  0.55
Accuracy of :  test_array_stemmed_without_sw ==>  0.5

 After removing stop-words

Accuracy of :  training_array ==>  0.9791666666666666
Accuracy of :  training_array_without_sw ==>  0.9791666666666666
Accuracy of :  training_array_stemmed_without_sw ==>  0.6041666666666666
Accuracy of :  test_array ==>  0.55
Accuracy of :  test_array_without_sw ==>  0.55
Accuracy of :  test_array_stemmed_without_sw ==>  0.5

 After stemming

Accuracy of :  training_array ==>  0.4583333333333333
Accuracy of :  training_array_without_sw ==>  0.4583333333333333
Accuracy of :  training_array_stemmed_without_sw ==>  0.9791666666666
```

- To get a clear view of the results,I have summarized it in these tables with its graphics.

Results of the original application :

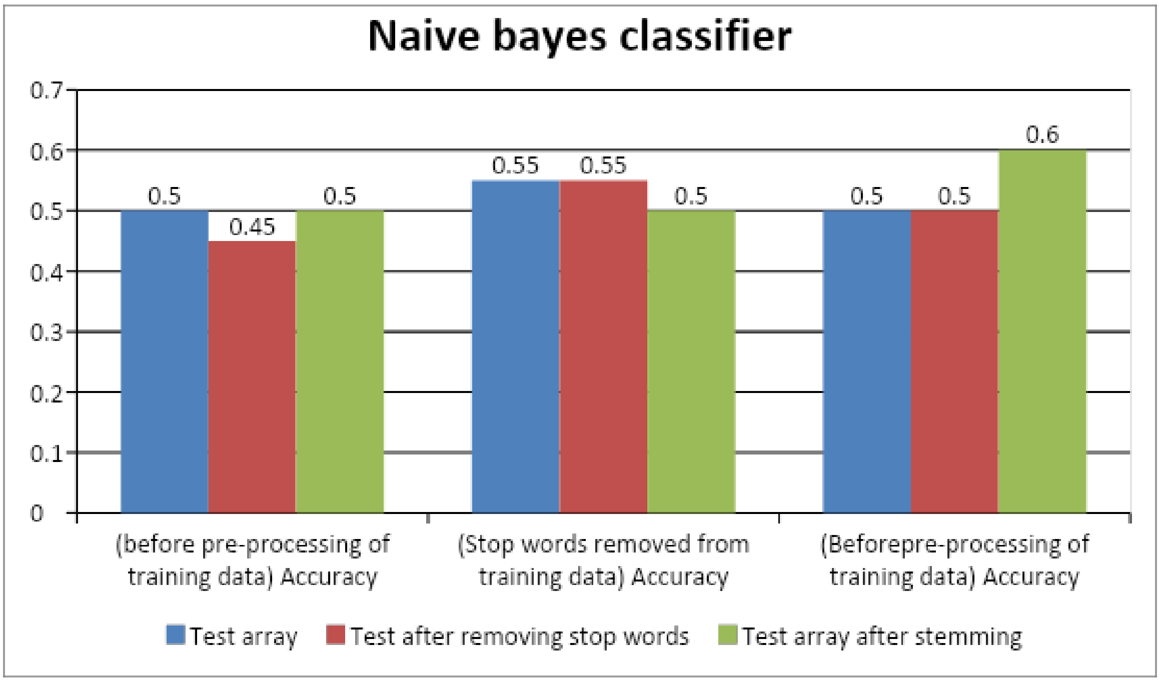
1) Decision Tree classifier:

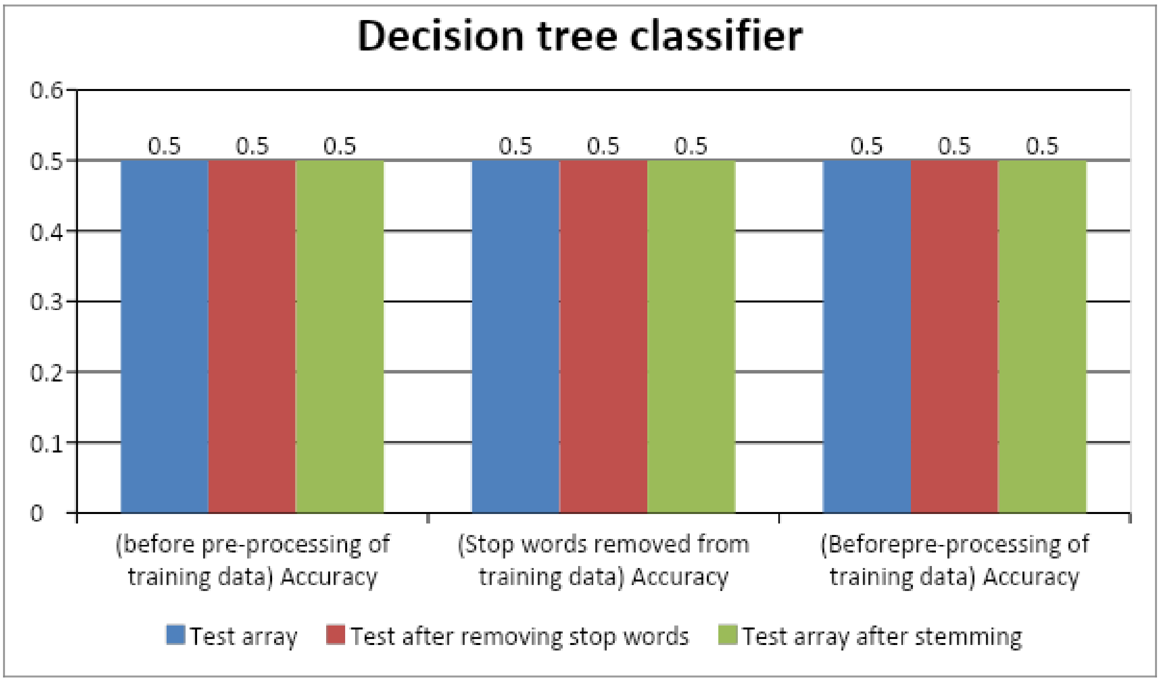
Test Data	Accuracy	Accuracy	Accuracy (After stemming of training data)

	(before pre-processing of training data)	(Stop words removed from training data)	
Training array it self	1.0	0.9791	0.4345
Training array after removing stop words	0.916	0.9791	0.4345
Training array after stemming	0.625	0.6041	0.9791
Test array	0.5	0.55	0.5
Test array after removing stop words	0.45	0.55	0.5
Test array after stemming	0.50	0.50	0.6

2) Naïve Bayes classifier:

Test Data	**Accuracy** (before pre-processing of training data)	**Accuracy** (Stop words removed from training data)	**Accuracy** (After stemming of training data)
Training array it self	0.8125	0.8333	0.45833
Training array after removing stop words	0.75	0.8333	0.4375
Training array after stemming	0.4166	0.4166	0.8333
Test array	0.50	0.50	0.50
Test array after removing stop words	0.50	0.50	0.50
Test array after stemming	0.5	0.50	0.50


Naive bayes classifier
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
0.5
0.45
0.5
0.55
0.55
0.5
0.5
0.5
0.6
(before pre-processing of training data) Accuracy
(Stop words removed from training data) Accuracy
(Beforepre-processing of training data) Accuracy
Test array
Test after removing stop words
Test array after stemming


Decision tree classifier
0.6
0.5
0.4
0.3
0.2
0.1
0
0.5
0.5
0.5
0.5
0.5
0.5
0.5
0.5
0.5
(before pre-processing of training data) Accuracy
(Stop words removed from training data) Accuracy
(Beforepre-processing of training data) Accuracy
Test array
Test after removing stop words
Test array after stemming