

MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH

University of Kasdi Merbah Ourgla

Faculty of Modern Information and Communication
Technology

Computer science and information technology department



Academic Master Thesis

Field: Computer science and information technology

Branch: Computer science

Specialty: Fundamental computing

Theme

Algerian Dialect text clustering based on Emotion
detection

Presented by:
REHAIEM ELMOUNTASSIR
DIDA MAROUANE

Supervised by:
Dr.M. MEZATI

2021/2022

Acknowledgement

In the name of Allah, the most gracious, the most merciful. First and foremost, we are thankful to Almighty Allah for giving us the strength, knowledge, ability, and opportunity to realize this study and complete it satisfactorily.

Secondly, we would like to express our deepest gratitude to our Advisor Dr.M. MEZATI for his invaluable patience and guidance through the process to accomplish this thesis.

A special thanks to Mrs. W. SAADI for her orientation and advices during this work.

I would like to extend my sincere thanks to the members of the jury for their precious time devoted to study our work.

We are also grateful to our teachers through the years, whom we couldn't undertake this journey without. Lastly, we would like to thank our families for their support.

Abstract

Currently, social media is considered as a big space to express opinions and share thoughts, Facebook and twitter are a rich source of information that plays an important role in the Algerian society. Despite the existence of many studies that have focused on supervised text classification for the Arabic language. the lack of labelled datasets interest in the Algerian dialect poses a challenge. The purpose of our study is to build a model for text clustering in the context of emotion detection in Arabic text . For the objective of our approach tweets that are used as text data were extracted from twitter via twitter API for the Algeria region. Using an unsupervised Machine Learning (ML) technique for natural language processing (NLP), this work is divided into two main phases, the first is the preprocessing in which the raw data text is cleaned to feed the second phase which is treatment, in this phase, different clustering algorithms are being applied on the cleaned text. After this work, the obtained result is a dataset classified according to the Ekman emotional model into six (06) categories (Happiness, Anger, Fear, Surprise, Sadness, Disgust). This dataset can be helpful to make trained models for Emotion Detection on dialectical Algerian Tweets.

Key words: NLP, Emotion Detection, Machine Learning, Algerian Dialect, Text clustering, Twitter.

Résumé

Actuellement, les médias sociaux sont considérés comme un grand espace pour exprimer des opinions et partager des pensées, Facebook et Twitter sont une riche source d'information qui joue un rôle important dans la société algérienne. Malgré l'existence de nombreuses études qui se sont concentrées sur la classification supervisée de textes pour la langue arabe. le manque d'intérêt des ensembles de données étiquetés pour le dialecte algérien pose un défi. Le but de notre étude est de construire un modèle de clustering de textes dans le contexte de la détection d'émotions dans un texte arabe. Pour l'objectif de notre approche, les tweets utilisés comme données textuelles ont été extraits de Twitter via l'API Twitter pour la région Algérie. Utilisant une technique de Machine Learning (ML) non supervisé pour le traitement du langage naturel(NLP), ce travail est divisé en deux phases principales, la première est le prétraitement dans lequel le texte brut des données est nettoyé pour alimenter la deuxième phase qui est le traitement, dans ce phase, différents algorithmes de clustering sont appliqués sur le texte nettoyé. A l'issue de ce travail, le résultat obtenu est une dataset classé selon le modèle émotionnel d'Ekman en six (06) catégories (Bonheur, Colère, Peur, Surprise, Tristesse, Dégoût). Cet ensemble de données peut être utile pour créer des modèles entraînés pour la détection d'émotions sur les tweets algériens dialectiques.

Most clés: NLP, Détection d'émotions, Machine Learning, Dialecte Algérien, clustring de textes, Twitter.

Contents

Introduction	1
1 Natural Language Processing (NLP)	3
1.1 Introduction	3
1.2 Natural Language processing	3
1.2.1 Definition	3
1.2.2 NLP branches	4
1.2.3 NLP Applications	5
1.3 Arabic language	7
1.3.1 Classic Arabic (CA)	7
1.3.2 Modern Standard Arabic (MSA)	7
1.3.3 Dialectal Arabic	7
1.3.4 Algerian dialect (AD)	7
1.4 Difficulties of Arabic language processing	8
1.5 Conclusion	9
2 State of the art of text categorization	10
2.1 Introduction	10
2.2 Text classification	10
2.3 Text clustering	11
2.3.1 Typical use cases for text clustering	12
2.3.2 Clustering methods	13
2.3.2.1 Hierarchical clustering	13
2.3.2.2 Agglomerative method	13
2.3.2.3 Decisive	13
2.3.2.4 Non-hierarchical clustering	14
2.3.2.5 k-means	14
2.3.3 Text clustering categories	15
2.3.3.1 Word-Based Clustering	15
2.3.3.2 Knowledge-Based Clustering	15
2.3.3.3 Information-Based Clustering	15
2.3.4 Challenges of text clustering	15
2.3.4.1 Sparse feature vector	16
2.3.4.2 Polysemy	16
2.3.4.3 Synonymy	16
2.4 Text clustering evaluation techniques	16
2.5 Prior works on text categorization	16
2.5.1 Supervised approaches	17
2.5.2 Unsupervised approaches	18
2.6 Conclusion	19
3 Emotion detection from social media	20
3.1 Introduction	20
3.2 What is an emotion	20
3.3 Emotion detection	20
3.4 Emotion models	21
3.4.1 Discrete emotion models	21
3.4.2 Dimensional emotion models	22
3.5 emotion detection from text	23

3.6	Emotion detection from social media	24
3.6.1	Twitter	24
3.6.2	Why twitter	24
3.7	Challenges of emotion detection in social media	25
3.8	Approach for emotion detection for text	25
3.8.1	Machine Learning-based method	25
3.8.2	Lexicon-based method	26
3.8.2.1	keyword-based methods	26
3.8.2.2	Corpus-based methods	26
3.8.3	Hybrid based methods	27
3.9	Related Works	27
3.9.1	Sentiment Analysis studies	27
3.9.2	Emotion Detection studies	28
3.9.3	Algerian dialect Studies	29
3.10	Conclusion	30
4	Conception	31
4.1	Introduction	31
4.2	Twitter API	32
4.3	Data collection	32
4.4	Text Preprocessing	32
4.4.1	Deleting the Arabic diacritics	33
4.4.2	Stop words removal	33
4.4.3	Punctuation removal	34
4.4.4	Numbers removal	34
4.4.5	Latin letters removal	34
4.4.6	Special characters removal	34
4.4.7	Emojis handling	34
4.4.8	Tokenization of words	36
4.5	Text Vectorization	37
4.6	Clustering	37
4.6.1	Cluster analysis with NMF	38
4.6.2	Emotion extraction	39
4.6.3	Clusters	39
4.7	conclusion	39
5	implementation	41
5.1	Introduction	41
5.2	Presentation of technologies and languages	41
5.2.1	VS Code	41
5.2.2	jupyter	41
5.2.3	Python language	42
5.2.4	Scikit-learn	42
5.2.5	NumPy	43
5.2.6	Pandas	43
5.2.7	Matplotlib	43
5.2.8	NLTK	44
5.2.9	Googletrans	44
5.3	Twitter api	45
5.4	Data collection	45
5.5	Tweets	46
5.6	Text Vectorization	50
5.7	Clustering	51
5.8	Results	53
5.9	Conclusion	55
	conclusion	55

List of Figures

1.1	Position of NLP in AI	4
1.2	NLP applications	6
3.1	Russel’s emotion model	23
3.2	Plutchik’s emotion model	23
4.1	Algerian Dialect Emotion Analysis Process	31
4.2	Text Preprocessing	33
4.3	Stop words removal	33
4.4	Emojis	35
4.5	Handling emojis	35
4.6	Classification of emoji	36
4.7	Tokenization example.	36
4.8	tf-idf Vectorization	37
4.9	Cluster analysis with NMF	38
5.1	VS Code logo	42
5.2	Jupyter	42
5.3	Python	43
5.4	Script of data collection from the Twitter API.	45
5.5	Capture of data	46
5.6	Script of word tokenization and stop words removal.	46
5.7	Script to delete the Arabic language diacritics.	47
5.8	code Removing Special Characters	47
5.9	Script to get emojis from tweets.	48
5.10	Emojis	48
5.11	Emoji translation by Googletrans.	49
5.12	Handling emojis	49
5.13	Preprocessing	50
5.14	TF-IDF Vectorizer script.	51
5.15	Text Vectorization	51
5.16	Clustering by NMF	52
5.17	H Matrix output NMF	52
5.18	Emotion extraction from matrix H.	52
5.19	Emoji lexicon	53
5.20	Emoji google Translator	54
5.21	Emoji Subjective translation	54

List of Tables

- 1.1 NLP applications examples 6
- 3.1 Related works with Emotion detection. 30

Introduction

Social media contains a huge quantity of data text that provides a massive amount of information. This data is mainly created by people that use these sites to freely express their opinions and thoughts as publications, comments, or messages. Twitter is the most popular online microblogging, social networking, and news site, where people can communicate through short messages called "tweets". The text in the Twitter platform is short and have a limit of 280 characters. Twitter, unlike other social media platforms, has no geographical limitations, no conversational boundaries, and no time constraints. In addition, it consists of real-life discussions, a broad variety of content, and a real-time stream. That's what makes it a favorable site for opinion expression, communication, business and entertainment. Thousands of the tweets generated daily on Twitter are in the form of natural language text data. This text can be in a formal or dialectal language. This text data is considered a gold mine for precious information used for political, economic, and medical matters. The information in these short messages is usually hidden and unstructured, which makes the manual extraction process painstaking for humans.

Natural language processing (NLP) is a technique that allows machines to understand and generate and interact with natural language the same way humans do. NLP has the ability to organize the scattered data in cases like affective computing (e.g., sentiment and emotion detection) and text classification. This latter is an interesting application for NLP, in which the objective is to classify unstructured text into organized groups.

Text classification is divided into two approaches based on machine learning techniques, which is the foundation stone of this task. The first is the supervised learning approach, which automatically categorizes a set of text into one or more preset classes based on their subjects. The second is the unsupervised approach or text clustering, in which, unlike the first approach, there are no predefined classes or categories. Clustering is the process of grouping a set of texts so that the texts placed in one group (cluster) have the same properties.

The Arabic language is spoken by over 420 million people as a native language, and it's the official language in 22 countries. Arabic is a highly structured and derivational language in which morphology is extremely significant. The Arabic language, specifically the Algerian dialect, in this case, poses significant challenges to researchers and developers of natural language processing (NLP) systems for Arabic text, as we shall cover later in chapter 1.

Although there is a lot of prior research that focused on supervised text classification in the context of sentiment analysis or emotion detection, there is a critical lack of labeled datasets for emotion detection in Algerian dialect text. This problem pushes practitioners to the clustering method to create labelled datasets.

The objective of our work is to create an Algerian dialectal Arabic text clustering model in the context of emotion detection. This model can categorize tweets into six emotion classes (happiness, sadness, disgust, fear, anger and surprise). The result of this process is a labelled dataset ready to feed supervised learning approaches

for the Algerian dialect text in the context of emotion detection. This work is organized as follows:

- chapter1: we covered natural language processing from different aspects, beginning with its definition, branches and application domains. then we introduced an overview about Arabic language and its challenges for NLP.
- chapter2: we introduced a state of the art for text clustering.
- chapter3: we presented different aspects related to emotion detection from social media.
- chapter4: we described our followed method for applying NLP techniques and clustering algorithms.
- chapter5: we defined the used tools in the course of this works in addition to results and discussions.

Chapter 1

Natural Language Processing (NLP)

1.1 Introduction

Since machines can not handle human natural language, appears the necessity for Natural language processing techniques. In this chapter we present the basic theoretical principles of Natural language processing. We try to clarify and cover the subject from different sides by defining every related aspect to the topic from the overall to the detailed. Beginning with Natural language processing definition, branches and Applications followed by a an overview about Arabic languages and it's categories and the difficulties of applying NLP techniques on the Arabic text.

1.2 Natural Language processing

1.2.1 Definition

Natural language processing (NLP) is a subfield of Artificial intelligence (AI) related to computational linguistics which is concerned with modeling language. NLP helps computers or machines to understand, manipulate, and explain human language as it is spoken and written. In this process, natural language is translated into structured data. Machines can do tasks such as speech recognition, sentiment detection or generating questions and responses. The natural language processing system is considered as pipeline because it contains several rational stages of data processing.

Natural language is the language in which humans communicate with each other and which has evolved through use and repetition like Arabic and English. It is different from programming language that it can take the form of speech or writing. Nowadays, machines can understand, interact and do tasks based on the Natural language, like Amazon's Alexa, Apple's Siri and

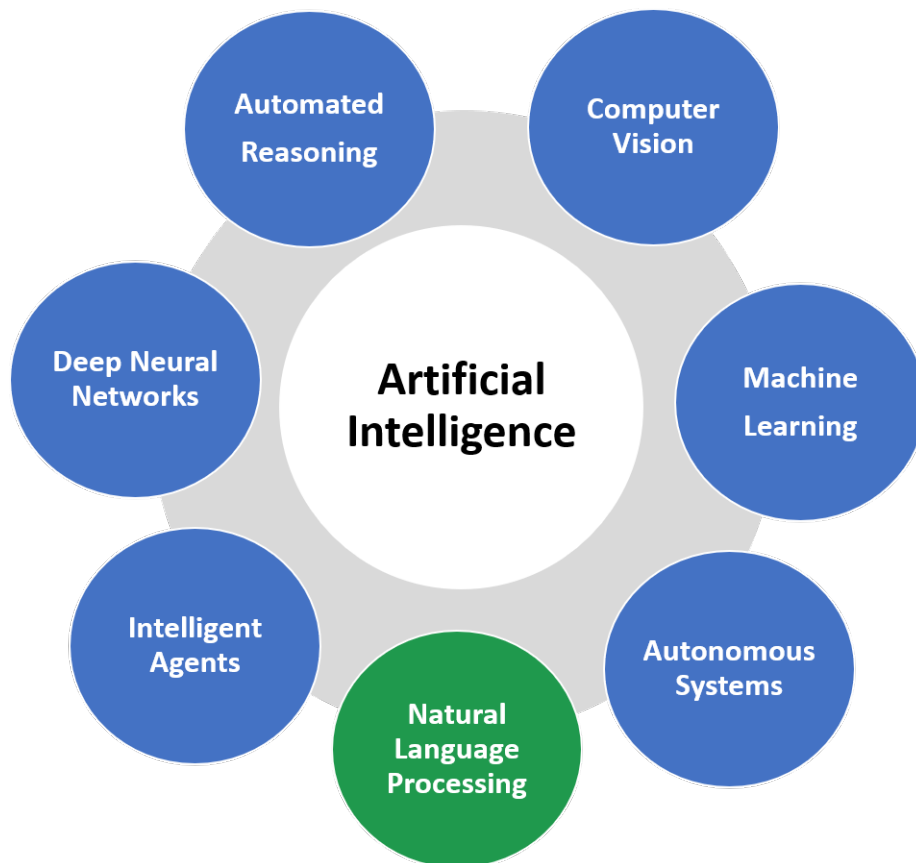


Figure 1.1: Position of NLP in AI

Google.

Natural language processing exists since 1940s, at that time NLP systems were implemented manually with code as a set of rules. Since 1990s most of the NLP researches are relied on machine learning. This latter provides automatic learning by using a huge amount of examples to build systems can deal with NLP, yet the biggest advantage of Machine learning for NLP is the accuracy. [1]

1.2.2 NLP branches

We can define two aspects for NLP:

- Natural Language Understanding (NLU) is a branch of natural language processing that breaks down the elements of speech to assist computers comprehend and interpret human language. Machine learning models in natural language understanding (NLU) improve over time as they learn to detect syntax, context, linguistic patterns, unique meanings, sentiment and intent. Human-computer interaction is enabled through NLU.
- Natural language generation (NLG) is the use of artificial intelligence techniques to generate written or spoken narratives from a data set. This allows the chatbot to query data

repositories, such as integrated back-end systems and third-party databases, and use the results to generate a response.

1.2.3 NLP Applications

Humans pick up their natural language through the years with practice and repetition. This language can be difficult to decipher because it contains expressions and sentiments that go beyond literal meanings. That's why we often don't appreciate the complexity of our language in the eyes of the machine and the difficulty of NLP application. As illustrated in figure 1.2 NLP has numerous applications in diverse domains in both understanding or generating natural language, like sentiment analysis, emotion detection, plagiarism detection, behavioral detection, and fact checking.

Natural language processing helps search engines provide useful and meaningful results by considering the natural language of web pages. NLP is used by search engines to surface relevant results based on comparable search habits or user intent, allowing the average person to find what they're looking for without having to be an expert in search. Functions like autocorrect, autocomplete, and predictive text are everywhere on our smartphones and computers.

Autocomplete and predictive text work in a similar way to search engines in that they forecast what we might say depending on what you enter, either by finishing the word or proposing a similar one. Text editors and browsers use NLP to correct spelling and grammar mistakes.

Email filtering is one of the most fundamental and early online applications of NLP. It began with spam filters, which identify specific words or phrases that indicate a spam message. But, like early NLP adaptations, filtering has improved. Gmail's email classification is one of the more common, newer implementations of NLP. Based on the contents of emails, the algorithm determines if they belong in one of three categories (main, social, or promotions).

With the use of Natural Language Understanding (NLU), the machine can recognize many moods that may be expressed through the user's instruction. An NLP tool will often analyze customer interactions, such as social media comments or reviews, or even brand name mentions, to see what's being said. The sentiment analysis of these interactions can assist brands in determining the success of a marketing campaign or in monitoring trending customer complaints before deciding how to respond or improve service for a better customer experience.

NLP can generate short text in the case of smart assistants' replies like Amazon's Alexa or Apple's Siri; voice recognition allows these assistants to recognize speech and interact with it; and long text in the form of creative writing like poetry and movie scripts; text mining like summarization. Text summary is an extremely hard task to do because we need to summarize large volumes of data with the extraction of every single bit of information.

One last common employment for NLP is translation software or machine translation. This software ends linguistic barriers in social media, education, and scientific research. As a result of the availability of huge amounts of data and powerful equipment, as well as breakthroughs in the fields of machine learning and neural networking, machine translation has improved substantially. [2]

NLP systems can act like humans and do processes that are considered a human capability. These systems are in the form of chatbots that can schedule meetings, handle emails and even write some sentences in books. NLP applications and their use cases are described in Table(1.1).

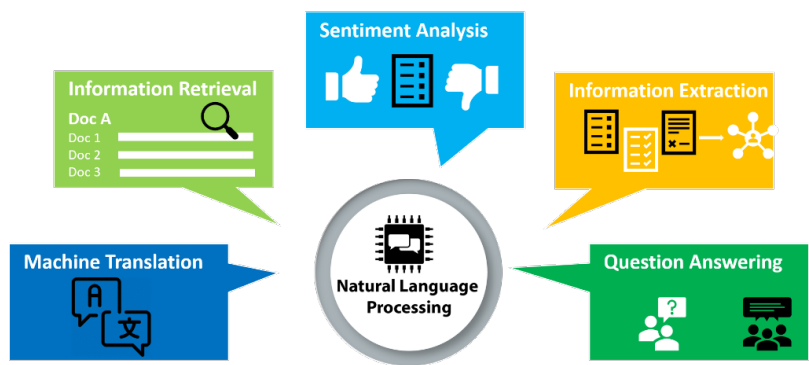


Figure 1.2: NLP applications

Application	Examples
Search	Web Documents Autocomplete
Editing	Spelling Grammar Style
Dialog	Chatbots Virtual assistants Scheduling
Writing	Index Concordance Table of contents
Email	Spam filter Classification Prioritization
Text mining	Summarization Knowledge extraction Medical diagnoses
Law	Legal interference Precedent search Subpoena classification
News	Event detection Fact checking HEAdline composition
Attribution	Plagiarism detector Literary forensics Style coaching
Sentiments analysis	Community morale monitoring Product review triangle Costumer care
Behavior detection	Finance Election forecasting Marketing
Creative writing	Movie scripts Poetry Song lyrics

Table 1.1: NLP applications examples

1.3 Arabic language

Arabic language [3] is a Semitic language spoken by approximately 420 million people. It is the official language in approximately 22 countries. Arabic is a broad term that encompasses three distinct groups: classic Arabic, modern Standard Arabic and dialectal Arabic.

1.3.1 Classic Arabic (CA)

is primarily defined as the Arabic used in the Qur'an and the earliest literature from the Arabian peninsula, but it also serves as the foundation for much literature up to the present day.

1.3.2 Modern Standard Arabic (MSA)

(Alfus'ha in Arabic) is the variety of Arabic that has been retained as the official language in all Arab countries and as a common language. It is basically a modernized version of classical Arabic. Standard Arabic is not the first language most Algerians pick up. MSA is learned at school and through exposure to formal broadcast programs (such as the daily news), religious practice, and newspapers.

1.3.3 Dialectal Arabic

Also known as colloquial Arabic or vernaculars are spoken dialects of Arabic. They are not written, unlike classical Arabic and MSA. These dialects have a hybrid form with numerous variations. They are influenced by both ancient indigenous languages and European languages such as French, Spanish, English, and Italian. The differences in these dialects of spoken Arabic throughout the Arab world can be significant enough to render them incomprehensible to one another. As a result of the significant differences between dialects, we can regard them as distinct languages based on the geographical location in which they are practiced. As a result, the majority of the literature describes Arabic dialects from an east-west perspective.

1.3.4 Algerian dialect (AD)

In Algeria, as in other places, spoken Arabic differs from written Arabic. Algerian dialect has a vocabulary inspired by Arabic, but the original words have been phonologically altered, with many new words and phrases. loanwords from French, Turkish, and Spanish. Like all Arabic dialects, Algerian Arabic has dropped the case endings of the written language.

Algerian dialect is not used in schools, on television, or in newspapers, which typically use standard Arabic, English, or French, but is more likely to be heard in songs, if not only in

songs. Algerian houses and streets Algerian Arabic is widely spoken. The vast majority of Algerians use it every day. [4]

1.4 Difficulties of Arabic language processing

NLP plays a key role in the preparation stage for every Arabic language matter. When applying NLP techniques to the Arabic language, several challenges appear beyond what is expected for English. The Arabic language consists of 28 distinct characters that are written from right to left. There are Arabic language features that are inherently challenging for Arabic NLP researchers and developers. These features include the nonconcatenative nature of Arabic morphology, the absence of the orthographic representation of Arabic short vowels from contemporary Arabic texts, and the need for an explicit grammar of MSA that defines linguistic constituency in the absence of case marking.

Another big issue with the Arabic script is that the letters change shape based on their position in the text. The originality of the words is a final challenge; 85 percent of Arabic words were formed from their roots [5]. Important aspects such as anaphoric relations, subjectless sentences, and discourse analysis must be described.

1.5 Conclusion

Humans use natural language to communicate with each other. To achieve the goal of understanding natural language by machines, natural language processing (NLP) techniques were invented and developed.

Since NLP is the pillar domain to any text handling technique, in this chapter, we focused on natural language processing by presenting a brief NLP definition and its branches. In addition, we displayed a set of NLP applications. Then we defined the Arabic language and its groups and also spotted the light on the Algerian dialect. Finally, we mentioned some difficulties with Arabic language processing.

The next chapter contains the state of the art for text clustering which is one of NLP tasks on text analysis.

Chapter 2

State of the art of text categorization

2.1 Introduction

The necessity of communication has pushed people to use several ways, such as social media. As a consequence of this use, a huge amount of data was generated. This data contains mostly unstructured informations. Text data is a good example of unstructured information, which is one of the most basic types of data that may be created in the majority of cases. Humans can quickly comprehend and interpret unstructured language, but machines have a far more difficult time understanding it. Therefore, there is a great need to develop methods and algorithms to properly analyze this flood of text in a wide range of applications. Machine learning techniques have proved their effectiveness and efficiency in text analysis. In this chapter we will present the most used machine learning techniques in the context of text analysis, specially text clustering.

2.2 Text classification

Automatic text classification is a supervised machine learning task that automatically categorizes a set of text into one or more predetermined classes based on their topic. Thus, the primary goal of text classification is to develop methods for natural language processing text categorization.

This method is used to assign topics to one or more preset class tags. These subjects were classified as different fields using various methodologies. Text classification methods are used in a variety of activities such as searching for linked documents, categorizing subject by document from appropriate documents, establishing documents in different subjects, Sentiment analysis, Emotion detection ...etc.

As a result, the goal of classification is to automatically attach the appropriate categorization

to any text that wishes to be categorized. Furthermore, text classification may be utilized in a variety of NLP applications. Traditional text classification approaches rely on many human-designed structures such as dictionaries, knowledge rules, and separate tree kernels.

Novel approaches of Text classification use supervised learning techniques to improve the effectiveness and efficiency. Supervised learning is considered as a difficult and complex task due to the lack of objectivity in human's participation in the process of classes labeling, which is not practicable with enormous datasets. Despite the fact that the work flow matches the techniques used in AI operations, it is time demanding. In Machine Learning. When distinct data distributions, different outputs, and different feature spaces arise, as in heterogeneous text corpora, supervised learning becomes costly. [6]

In supervised learning, an algorithm learns from example data and associated target responses or trained on labeled data, which can be numeric values or string labels such as classes or tags, in order to predict the correct response when new examples are presented. The training data is a smaller part of the larger data set and serves to give the algorithm a basic idea of the problem, the solution, and the data points to deal with. The training data set is also similar to the final data set in its properties and provides the algorithm with the labeled parameters required for the problem. [7, 8]

2.3 Text clustering

Text mining is the process of analysing unstructured information, typically found in natural language text, to discover new patterns. One of the most common text mining tasks is text clustering. Text clustering [9] is an unsupervised learning in which there is no predefined categories or classes. It is the process of grouping a set of texts so that the texts in one group (cluster) have the same properties as the texts in other groups or clusters. Its goal is to classify and group data with similar attributes together. Text clustering is primarily used in knowledge discovery and data mining. It includes Keyword Extraction and Named Entity Recognition because keywords in the text help in making text clusters, as well as word forms recognized as people, places, and organizations, which are also included in grouping the data points. Text can be clustered at various levels of granularity by considering cluster objects as documents, paragraphs, sentences, or phrases. Clustering algorithms uses basically supervised learning techniques.

In unsupervised learning, an algorithm learns from simple examples with no associated response, allowing the algorithm to determine data patterns on its own. This type of algorithm typically restructures data into something else, such as new features that may represent a new

class or series of feature values that are unrelated. They are extremely beneficial in providing humans with insights into the interpretation of data and the generation of new useful inputs for supervised machine learning algorithms. It is a type of learning that is similar to the methods humans use to figure out that certain objects or events are from the same class, such as by observing the degree to which two objects resemble each other. [8]

2.3.1 Typical use cases for text clustering

- Client segmentation is a common application for text clustering, which is based on either a narrow measure (the type of products they prefer to buy) or broader criteria (the sum of demographic characteristics). After analyzing the data, it is easier to develop different strategies for each client segment and tailor the offering accordingly.
- Text clustering can also be used to explore and visualize data. Instead of going through all of the samples, we could examine only a few representative samples from each cluster group. since samples within the same cluster are assumed to be similar.
- Another important application is detecting anomalies in data without prior labeling. Data samples that fall outside of the main clusters are typically candidates for data anomalies. Finding anomalous transactions in customer transaction data is one example of this approach.
- Taxonomy generation is the process of generating topics or concepts, as well as their relationships, by clustering texts from a given corpus. The task of taxonomy generation entails compiling a list of topics or categories from the corpus and linking each to relevant texts.
- Since clustering is an unsupervised learning method, it may be used in document classification [10] to improve the quality of results in its supervised counterpart. In particular, word-clusters and co-training approaches can be utilized to increase the classification accuracy of supervised applications using clustering techniques.
- Clustering algorithms give a coherent summary of the collection in the form of cluster-digests or word-clusters, which may be utilized to provide summary insights into the underlying corpus's general content. Such methods, particularly sentence clustering, can also be employed for document summarization.
- Text clustering can be applied in the field of affective computing [11], especially emotion detection, to improve the performance of emotion detection for short text. By representing short texts with word cluster features.

2.3.2 Clustering methods

Text clustering is an unsupervised machine learning problem that takes two forms of methods, the Hierarchical and the non-Hierarchical methods.

2.3.2.1 Hierarchical clustering

The hierarchical clustering [9] method, which combines relevant similarity measures, has become the standard technology for text clustering. Hierarchical clustering is a text clustering approach that may produce hierarchical nested classes. The hierarchical clustering approach considers the category to be hierarchical; in other words, when the category is changed, the item also changes.

A variety of application domains are interested in hierarchical clustering solutions, which are in the form of trees called dendrograms. Hierarchical trees display data at various levels of abstraction. Because clustering solutions are consistent at different levels of granularity, flat partitions of varying granularity can be extracted during data analysis, making them ideal for interactive exploration and visualization. Furthermore, clusters frequently have subclusters, and hierarchical structures naturally represent the underlying application domain.

Hierarchical clustering methods are generally divided into Agglomerative (bottom-up) hierarchical clustering method and decisive (top-down) hierarchical clustering method.

2.3.2.2 Agglomerative method

The main idea behind agglomerative clustering is to integrate text into clusters based on its similarity to one another. Bottom-up hierarchical clustering begins with a single cluster, first takes an object as a separate category, and then combines two or more appropriate categories repeatedly. Hierarchical clustering does not loop until a stop condition is reached (the number of parameters is generally K). Bottom-up hierarchical clustering may be thought of as a method of generating a tree that contains information on the class hierarchy and the similarity of all classes.

2.3.2.3 Decisive

The top-down hierarchical clustering approach begins with the whole work of the item and subsequently divides it into additional groups. The standard approach is to build a minimal spanning tree on comparable graphs, then remove the side that has the least resemblance to the spanning tree (or is the furthest away from the spanning tree). It can generate a new category if it deletes one side. When the lowest similarity reaches a certain level, the cluster may come

to a halt. In general, the top-down technique requires more computing than the bottom-up method, and its applications are inferior to the latter.

Hierarchical clustering has the following advantages: it can be applied to any shape; it can be used for any similarity or distance form; and it has the inherent flexibility of clustering granularity. The disadvantage is that the termination condition is inaccurate, and it must be determined by human experience; once the clustering result forms, it can typically no longer be rebuilt to enhance the result, and the erroneous categories formed by error do not have the ability to be repaired.

2.3.2.4 Non-hierarchical clustering

This method, also called partitional clustering [12], begins with each point as part of a random or guessed cluster and moves points between clusters iteratively until a local minimum is found with respect to some distance metric between each point and the center of the cluster to which it belongs. In other words, this method chooses a given number "n" of objects in such a way that the sum of other objects' distances to the closest of the "n" objects is minimized. At the same time, each object is assigned to one of the "n" selected objects, resulting in a clustering into "n" subsets.

Partitional clustering is not ideal for many applications because the approximate number of clusters must be known in advance and each data point must be placed in some cluster.

The most used partitioning algorithm is K-means.

2.3.2.5 k-means

The purpose of the k-means algorithm is to partition the data set into k groups based on the input parameters k. The algorithm employs an iterative updating method, with each round based on k points of reference points being grouped around k clusters, with each cluster centroid being utilized as a reference point in the following round of iteration. Iteration brings the selected reference point closer to the genuine cluster centroid, improving the clustering effect.

The k-means algorithm has the following advantages: it has good geometry and statistical significance in the numerical attribute; it is less susceptible to order; it can run in parallel; and it can carry out the cluster under the random norm. However, its disadvantages include: requiring the user to provide the number of clusters in advance; being unable to analyze categorical attribute data; and being sensitive to isolated points. Cannot detect non-sphere clusters or clusters with large size differences; usually falls into local optimum solution but is unable to reach globally optimal solution; is subject to aberrant point interference; lacks scalability;

clustering results are occasionally uneven.

2.3.3 Text clustering categories

In general, text clustering methods are classified into three categories based on different text representations: Word-based clustering, Knowledge-based clustering, and Information-based clustering

2.3.3.1 Word-Based Clustering

In general, the concept is the fundamental unit of automatic text processing. Words constitute concepts. Word is the atomic carrier of information that cannot be broken down. The key words can be used to represent the document. It is necessary to select an efficient word segment method for preprocessing the documents, and extracting representative key words is critical for the word-based clustering method. The Vector Space Model (VSM) is a popular method of document representation in which each document is represented as a vector (a series of words). Because of the high dimensionality and sparseness, the performance of the word-based clustering method will deteriorate. [13]

2.3.3.2 Knowledge-Based Clustering

An explicit knowledge base is essential for knowledge-based clustering. The knowledge representations include semantic webs, predicates, objects, rough set based constructs, neural networks etc. A knowledge base with a strong specialty must be manually constructed for the knowledge-based clustering method, which is unlikely to be portable. Knowledge-based clustering methods can perform text clustering quickly and accurately for specific applications. [13]

2.3.3.3 Information-Based Clustering

Information-based clustering is context-dependent. Only useful information is extracted during the text clustering process. Information-based clustering analyzes phrases, text segments surrounding them, and latent semantic information. This method, in particular, can be used to handle text that lacks key words or key phrases, overcoming the limitations of information-based clustering methods such as ambiguous words, thesaurus, phrases ...etc. [13]

2.3.4 Challenges of text clustering

There is several difficulties facing researchers in the field of text clustering. Particularly, short text clustering presents a unique challenge when compared to general text clustering. In the next section we will present most common challenges for short text clustering.

2.3.4.1 Sparse feature vector

Each document is represented by a feature vector in document or large text clustering methods. This vector contains numerical values for features that correspond to document terms. Because the number of words in short texts is very low, the feature vector generated from short texts is generally sparse in nature. The sparsity of the feature vector is a major issue in clustering short text data, and resolving it is a difficult task. [14]

2.3.4.2 Polysemy

The second issue is the existence of various interpretations of a single word. It may be easier to assign a meaning to a word in a document because we can easily identify the meaning from the context of the text. However, in a short text, it is definitely a difficult task; the reason is that there are few words and thus the context of a specific word cannot be understood. [14]

2.3.4.3 Synonymy

This is the case when there are two or more words with the same meaning. For example, the words "happy," "pleased," and "contented" all mean the same thing. As a result, deciding where to place such words is a difficult task, especially when such words are found in short texts. [14]

2.4 Text clustering evaluation techniques

A clustering evaluation requires the use of an independent and reliable method for assessing and comparing clustering experiments and results. In theory, the clustering researcher has developed an intuition for clustering assessment, but in practice, the big volume of data on one hand, and the complex subtleties of data representation and clustering algorithms on the other, make intuitive judgment difficult.

An intuitive, introspective evaluation is thus only feasible for small groups of items, but large-scale trials necessitate an objective procedure. There is no absolute system for measuring clustering, but a range of assessment metrics from other fields, such as theoretical statistics, machine vision ..etc. There is three factors by which clustering can be evaluated: Clustering tendency, number of clusters and clustering quality.

2.5 Prior works on text categorization

When talking about text classification in general we can define two approaches for this task: Supervised and Unsupervised. this partitioning is based on the used machine learning technique.

In this section we display some of prior works for each technique.

2.5.1 Supervised approaches

Geli Fei and Bing Liu (2015) [15] proposed a new technique for solving the classification problem. The technique's main innovation is the transformation of document representation from the traditional ngram feature space to a center-based similarity (CBS) space to build much better classifiers. Burdisso [16] introduced the SS3 framework, which is based on machine learning techniques. SS3 was created as a general framework for dealing with early risk detection issues on social media. KeYuan Wu [17] proposed a fuzzy logic-based method for text classification in social media. The aim of the study is to determine the relevance between Hurricane Sandy 2012 related Twitter text data and Twitter's messages. Julia Ive [18] applied a hierarchical recurrent neural network (RNN) architecture with an attention mechanism on mental health-related social media data. This architecture showed an improvement in overall classification results when compared to previously reported results on the same data. Manzhu Yu [19]'s research examined the capability of a convolutional neural network (CNN) in text classification for disaster-related tweets in order to facilitate the rapid identification of disaster response and relief content. CNN showed better accuracy compared to support vector machine (SVM) and logistic regression (LR). Al-Garadi [20] described the development and evaluation of supervised automated text classification models for identifying self-reports of prescription medication usage on Twitter. and proposed a classifier based on transformer context-preserving bidirectional encoder representations (BERT). Hajibabae [21] suggested a text classification model that included a modular cleaning phase and tokenizer, three embedding techniques, and eight classifiers. This experiment provides a promising result for detecting offensive language on a dataset obtained from Twitter. LIU and CHEN [22] suggested a medical social media text classification algorithm that integrates consumer health terminology utilizing datasets including patient descriptions from social media. Akshi Kumar [23] primarily analyzed the text classification algorithms used in the process of mining unstructured data in order to provide a definitive appraisal of their utilization in terms of their distinct strengths, weaknesses, opportunities, and threats (SWOT). Martins [24] aimed on establishing a lexical baseline for hate speech in social media, by applying classification methods on an annotated dataset for the same purpose. Hissah ALSaif [25] built a text classification model for detecting violence in Arabic dialects on Twitter using different feature-reduction approaches, like SVM and information gain (IG). Singh [26] suggested a new SentiVerb and Spell Checker framework for extracting opinions from social media text (SMT). and classify them as positive or negative opinions using a dictionary-based technique and a binary classifier based on SMT sentiment score.

2.5.2 Unsupervised approaches

The unsupervised approach is based mainly on clustering. Jing [27] presented a new clustering technique based on ontologies-based distance measure, and the results showed that ontologies-based distance measure improves the performance of text clustering approaches. Zhong's paper [28] integrates an efficient online spherical k-means (OSKM) method with an existing scalable clustering technique. To accomplish quick and adaptable clustering of text streams. LIU [29] discussed typical text clustering algorithms, analyzed and compared various aspects of clustering algorithms such as the applicable scope, initial parameters, termination conditions, and noise sensitivity. Xu [30] proposed a Short Text Clustering using Convolutional Neural Networks (STCC), in which they assume that CNN is more beneficial for clustering by considering one constraint on learned features through a self-taught learning framework without using any external tags/labels. Zhang [31] provided an overview of available approaches for document clustering based on frequent patterns. and presented a novel method for document clustering termed Maximum Capturing, as well as compared it to other methods. Aggarwal [9] provided a detailed study of the text clustering problem. includes challenges, application domains, important methods employed, and a number of achievements in the context of social networks and linked data. Rangrej [32] examined the performance of several document clustering algorithms, including K-means, an SVD-based method, and a graph-based approach, using short text data gathered from Twitter.

2.6 Conclusion

Text classification tasks, in both supervised and unsupervised forms, find use in a variety of applications such as sentiment analysis and emotion detection. In our search for approaches related to this task, we noticed that the percentage of approaches that employed clustering was way lower than its supervised variant.

In this chapter we presented a state of the art for text categorization . Beginning by defining the supervised text classification. Then we define text clustering, spotting the light on some of its typical use cases and clustering methods, as well as the categories and challenges of text clustering. In the fourth section, we discussed clustering evaluation techniques. Finally, we presented a collection of previous works on text clustering.

When talking about text clustering for the Arabic language in the context of Emotion detection, specifically the Algerian dialect, there are no studies on this matter. The following chapter provides an overview of emotion detection in social media.

Chapter 3

Emotion detection from social media

3.1 Introduction

Affective computing is a branch of AI that allows practitioners to handle concepts like sentiments and emotions in several data types. In our case we focus on text data. In this chapter we will present different aspects related to emotion detection from social media. At first we define emotion and the emotion detection concepts, in addition we detail emotion detection models, then emotion detection in social specially in Twitter. The challenges that appears in the course of the process are also presented. Moreover, we arrive to present different approaches for emotion detection from text. finally, we put a set of related works with this context including works dealt with the Algerian dialect.

3.2 What is an emotion

On our way to looking for a definition of emotion, we found no exact definition. Emotion can be defined as a subjectively experienced affective state of consciousness expressed as strong feelings usually directed toward a specific object.

The term "emotion" should be used to refer to a collection of responses triggered by parts of the brain to the body and by parts of the brain to other parts of the brain. employing both neural and humoral mechanisms. The end result of gathering such responses is an emotional state characterized by physiological and behavioral changes. [33]

3.3 Emotion detection

Human beings use Emotions as a way of expression. Emotion Detection [34] and Recognition is an important NLP task. Emotion detection (ED) from social media is a recent field of research

that is closely related to Sentiment Analysis. Sentiment Analysis aims to detect positive, neutral, or negative feelings, whereas Emotion Analysis aims to detect and recognize types of feelings through the expression of social media posts, such as anger, disgust, fear, happiness, sadness, and surprise.

3.4 Emotion models

When we arrive to ED systems we need to know what and how many classes we are presenting our emotion, that what refers to emotion models. There is a lot of proposed approaches for this matter that suggest and explain how to represent emotions. Two important types of approaches for emotion models are Discrete emotion models and dimensional emotion models.

3.4.1 Discrete emotion models

The discrete model of emotions is based on distinct emotion classes or categories, among these models:

- The Paul Ekman model [35] categorizes emotions into six (06) fundamental categories. According to the theory, there are six (06) essential emotions that arise from various brain systems as a result of how an experiencer sees a circumstance, making emotions independent. These basic emotions are joy, sadness, anger, disgust, surprise, and fear. However, the combination of these feelings can result in the production of more complex emotions such as guilt, shame, pride, lust, greed, and so on.
- The Robert Plutchik model [36], which, like Ekman, presumes that there are a few core emotions that occur in opposite pairs and combine to generate complex emotions, In addition to the six (6) essential emotions proposed by Ekman, he identified eight such fundamental emotions, including acceptance, trust, and anticipation. Joy vs. sadness, trust vs. disgust, anger vs. fear, and surprise vs. anticipation are the eight opposite emotions. According to Plutchik, there are variable degrees of intensity for each emotion as a function of how an experiencer interprets events.
- The Orthony, Clore, and Collins model [37] disagreed with Ekman and Plutchik's comparison of "fundamental emotions." They did agree, however, that emotions emerged as a function of how individuals viewed events and that emotions varied in strength. They divided emotions into 22 categories, adding 16 to the basic emotions proposed by Ekman, resulting in a much broader representation of emotions, with additional classes of relief,

envy, reproach, self-reproach, appreciation, shame, pity, disappointment, admiration, hope, fears-confirmed, grief, gratification, gloating, like, and dislike.

3.4.2 Dimensional emotion models

The dimensional model assumes that emotions are not independent and that they are related, thus the necessity to arrange them in a spatial space. Thus, dimensional models place emotions on a dimensional space (unidimensional, i.e., 1-D, and multidimensional, i.e., 2-D and 3-D) to show how connected emotions are and, in general, to reflect the two primary fundamental behavioral states of good and bad. The relative degrees (low to high) of occurrence of unidimensional and multidimensional are both influenced. Although unidimensional models are rarely employed, their basic concept pervades most multidimensional models.

- Russell introduces the circumplex of affect, a circular two-dimensional model [38] significant in dimensional emotion representation. The model distinguishes emotions in the Arousal and Valence domains, with Arousal distinguishing emotions by Activations and Deactivations and Valence distinguishing emotions by Pleasantness and Unpleasantness. The Circumplex model of Affect indicates that emotions are connected rather than autonomous. Russell's model is represented in figure(2.1)
- Plutchik presents a two-dimensional emotional wheel [36] with Valence on the vertical axis and Arousal on the horizontal axis. The emotions are depicted on the wheel in concentric circles, with the innermost emotions being derivatives of the eight fundamental emotions, followed by the eight fundamental emotions, and ultimately combinations of the primary emotions on the wheel's outermost regions. The wheel depicts how related emotions are based on their position on the wheel. Plutchik's emotional wheel is seen in Figure (2.2).
- Russell and Mehrabian also provide a three-dimensional emotion model [39] comprised of Valence/Pleasure, Arousal, and Dominance. According to the 2-D model, arousal and valence describe how pleasant/unpleasant or active/inactive an emotion is. The third dimension of Dominance describes how much control experiencers had over their emotions.

Discrete emotion models are more employed in emotion classification, due to its simplicity compared to the dimensional ones. On the other hand, Dimensional emotion models are highly suggested in projects that express emotional similarities.

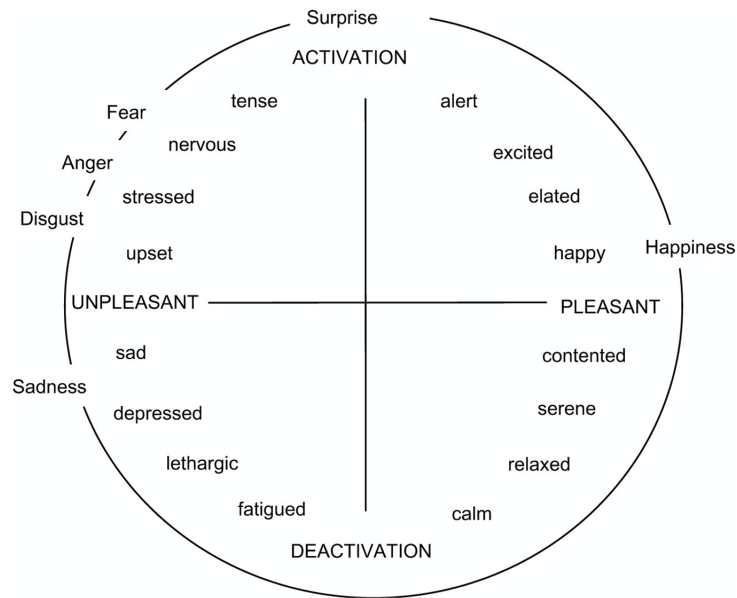


Figure 3.1: Russel's emotion model

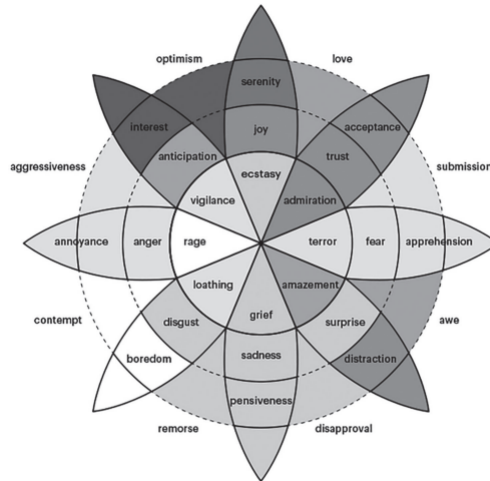


Figure 3.2: Plutchik's emotion model

3.5 emotion detection from text

Nowadays, people prefer to share their thoughts, opinions, sentiments and emotions in public. Social media(e.g.,Twitter,Facebook) provides an open space for opinion expression in a form of short text(like tweets and or posts). This text may contain some emotions indicators. this is what makes social media a rich source of emotional text data and an appropriate data source for behavioral studies like emotions for individuals and populations.

Emotion detection systems in social media will be effective in several applications, ED technologies will have a huge impact on individuals and public matters, like mental health counseling through social media and polling organizations that use emotions to estimate public felling towards any subject to take fast actions about it.

3.6 Emotion detection from social media

Social media is used by More than 4.5 billion of people around the world to share information and make connections. On a more personal level, social media allows people to communicate with friends and family, learn new things, explore new interests, and be entertained. Professionally, social media can be used to expand knowledge in a specific field and to build a professional network by connecting with professionals in any domain.

We can divide the term "social media" into two simpler terms. The first is "social," which means interaction with other people by sharing and receiving information from them. The second term is "media," which refers to a communication tool such as TV, radio and newspapers.

Social media is an internet-based technology that allows people to share ideas, thoughts, and information via virtual networks and communities. Social media allows users to share content such as personal information, documents, videos, and photos in real time. Users interact with social media through web-based software or applications on a computer, tablet, or smartphone.

3.6.1 Twitter

Twitter is a social media platform that allows users to communicate via short messages known as tweets. Twitter is also a news dissemination platform, a tool for public opinion formation, and a source of humour in our daily lives. A tweet is a publicly posted message on Twitter that cannot exceed 280 characters and can include text, an image, a video, or a link. Every day, over 500 million tweets get posted, and Twitter has over 450 million active users.

3.6.2 Why twitter

Because of the massive amounts of real-time data produced on a consistent basis witter data has recently become a popular dataset among Natural Language Processing (NLP) researchers. Aside from its massive size, Twitter data has several distinguishing characteristics, including real-life conversations, uniform length (280 characters), rich variety, and a real-time data stream. Twitter is a goldmine for a diverse set of applications that include opinion mining, and sentiment analysis towards matters of social or political concern.

Twitter allows users to express their ideas, opinions, feedback, and comments in the form of "tweets." Users can also retweet, follow, share, and like content. It has a number of distinguishing features that make it a good option for natural language processing for text content. The content is largely unbiased, not geographically restricted, and is indexed. Twitter even suggests online applications and provides instructions for using or extracting tweets for scientific experiments.

3.7 Challenges of emotion detection in social media

In the way of emotion detection in social media we have to spot the light on some challenges that appears during the course of the process. All these challenges are related to the nature of text data. It would be easy if there were words that expresses the emotion clearly, but most of the time the emotion is expressed implicitly and ambiguity. Most of the social media posts and messages text is written in informal way. Some text data contain more than an emotion, others have spelling errors, some have grammatical mistakes, some use dialect, some contain more than a language in a single statement. Numerous topics and emotional states make it hard to manually create a corpus of labelled data that covers all ED cases. Let's suppose that we want to perform a supervised approach for ED by training an automatic classifier with data that we label manually; this process will be a painstaking due to the Numerosity of topics and emotional states. One last challenge is the consistency of annotation for emotions, human annotators are not reliable because of the subjectivity in judgment. Hence, different annotators will classify emotions into different emotion classes. [40]

3.8 Approach for emotion detection for text

There are several ways of detecting textual emotions. Actually, emotion recognition is a subset of affective computing, and the computational methods utilized in this work are part of it.

Several researchers have classified the area into several groups. These researchers' methodologies have been defined and may be classified into four types: keyword-based methods, lexicon-based methods, machine learning methods, and the hybrid method. Existing works make use of unigrams (one word), n-grams (multiple words), emoticons, hashtag words, punctuations and negations as features for emotion detection task [41].

3.8.1 Machine Learning-based method

For textual emotion identification, both supervised and unsupervised machine learning methods are utilized, with a model designed to train a classifier using a portion of the dataset and then test the classifier with the remainder of the dataset. An annotated emotion dataset is utilized for training and testing the supervised classifier in the supervised technique. The most often used classifiers are Naive Bayes, Support Vector Machine, and Decision Tree. The data used in the unsupervised classification is not labeled with the classes, according to the definition. The classifier begins with multiple seed words for each emotion, which are subsequently linked to phrases. Sentences are assigned to emotions in this manner. This trains the classifier model,

which is then used to label the testing data. Unsupervised method is a more generalized one but in most cases, supervised classification achieves better accuracy.

3.8.2 Lexicon-based method

The lexicon-based method is one of the semantic analysis approaches . The emotion orientations of the entire document or collection of sentence(s) are calculated using semantic orientation of lexicons. The dictionary of lexicons can be manually or automatically generated. Many researchers utilize the WorldNet dictionary. To begin, lexicons are discovered throughout the document, and then WorldNet or another type of online thesaurus can be utilized to locate synonyms and antonyms to expand that dictionary. Among these approaches, we can find:

3.8.2.1 keyword-based methods

The most intuitive and straightforward approach is keyword-based emotion detection. The goal is to find and match patterns that are comparable to emotion keywords. The first goal is to identify the word in a sentence that represents the feeling. This is generally accomplished by using a Parts-Of-Speech tagger to tag the words of a phrase and then extracting the Noun, Verb, Adjective, and Adverb (NAVA) terms. Most linguistic and emotion-based studies concluded that these are the most likely emotion-carrying words. These words are then matched against a set of terms indicating emotions based on a specific emotion model. The emotion of the specific sentence is whichever emotion matches the keyword.

When a word matches numerous emotions in the list, different approaches might be used. Each word in certain keyword dictionaries has a probability score for each emotion, and the emotion with the greatest probability score is chosen as the word's emotion. In some other works, the first emotion associated with the term is chosen as the dominant emotion. The keyword reference list or keyword dictionary varies based on the researcher. Researchers typically build keyword dictionaries based on emotions and the words with which they are associated.

3.8.2.2 Corpus-based methods

This finds sentiment orientation of context-specific words. The two methods of this approach are:

- **Statistical approach:** The words which show erratic behavior in positive behavior are considered to have positive polarity. If they show negative recurrence in negative text they have negative polarity. If the frequency is equal in both positive and negative text then the word has neutral polarity.

- The semantic approach assigns sentiment values to words and words that are semantically near to those words; this can be accomplished by discovering synonyms and antonyms for that term.

3.8.3 Hybrid based methods

This method combines keyword-based implementation with Machine learning-based implementation. The key advantage of this method is that it can produce higher accuracy results by training a combination of classifiers and integrating knowledge-rich linguistic information from dictionaries and thesauri. This has the advantage of offsetting the high expense of using human indexers for information retrieval activities and minimizing the complexity faced when combining multiple lexical resources.

3.9 Related Works

Recently, there has been a lot of interest in Arabic text, and many works have been created in this context. Because NLP techniques for emotion detection and sentiment analysis are so close and both deal with text data, the next section presents related works based on these two aspects.

3.9.1 Sentiment Analysis studies

Most of the studies focused on the sentiments analysis in the first place such as M.O. Hegazi [42] who presents an approach to extract information from social media Arabic text and provides an integrated solution for the challenges in preprocessing Arabic text in social media in four stages data collection, cleaning, enrichment, and availability, they claim that their study is the first that aims to provide a framework (an integrated approach) for dealing with social media Arabic text and merge the concepts of data manipulation and database. Al-Khatib and El-Beltagy [43], in which they present an Arabic tweet dataset that they have built to serve this task. SEDAT [44] is a system to detect sentiments and emotions in Arabic tweets. The authors of this paper used word and document embeddings and a set of semantic features with the application of CNN-LSTM and fully connected neural network architectures in their study. Khaled Mohammad Alomari's [45] study that investigate different supervised machine learning sentiment analysis approaches when applied to social media Arabic text in either Modern Standard Arabic (MSA) or Jordanian dialect, and introduces an Arabic Jordanian twitter corpus for Sentiments Analysis. Maghfour [46] presents a sentiments analysis for (SA) and Moroccan dialect study based on comparing two approaches. The first one is the classical

approach that considers all Arabic text as homogeneous. The second one, that we propose, require a text classification beforehand sentiment classification, based on language categories: the standard and the dialectal Arabic. Matrane [47] this paper discused several attempts of the literature at solving the challenge of Sentiment analysis of regional dialects, this approach based on AraBERT(Arabic BERT) word embedding for Moroccan dialect (MD) sentiment analysis.The method goes through a pipeline of steps starting with preprocessing, lexicon-based translation and feature extraction. Afterwards a comparative study, in 2-way classification, of machine learning algorithms as SVM, DT, LR, RF, NB and deep learning algorithms such as LSTM, BiLSTM and LSTM-CNN from state of art. On the other hand,the authors of this paper managed to train thier model with four different outputs in 4 way classification. As a result, BiLSTM proved to be the best in both 2-way classification scoring 83% accuracy, and in 4-way classification achieving scores ranging between 62% and 92% of accuracy for each of the 4 classes. Mdhaffar [48] and her team worked on the SA for Tunisian dialect,they used machine learning techniques to define the polarity of comments written in Tunisian dialect.this study is based on comparing the performance of SA systems with an MAS and Multi-dialectal data sets, and with an annotated Tunisian dialect corpus that they collect from Facebook.

3.9.2 Emotion Detection studies

As for Emotion Detection and classification which is derived from SA, Alqahtani [49] focused on developing an Arabic language model for emotion classification of Arabic tweets. Their article provides a practical overview and detailed description of material that can server their objective. Rabie [50] contribution in Emotion detection for Arabic text consists in adding preprocessing steps that have improved the classification results by 4.4% compared to the original khoja stemmer. In addition, they have extracted a sample word-emotion lexicon from that corpus. this experiment demonstrates that this sample word-emotion lexicon enhances the emotion detection results by 22.27% compared to the Sequential Minimal Optimization (SMO) classification using the train/test option. Amira.f [51] The purpose of this paper was the automatic recognition of emotions in Arabic text.the base of this study is a sized Arabic lexicon, which used to annotate Arabic children’s stories for the six fundamental emotions:Joy, Fear, Sadness, Anger, Disgust, and Surprise. As a result this method achieved a 65% accuracy rate.Samar Al-saqqa [52] ,This survey investigated the most recent state-of-the-art approaches for emotion detection in text and discussed their classification based on the methodologies utilized, the used emotional model, and the various datasets used. The results of the paper tend to emphasize the limitations and gaps of these recent efforts and drive potential future research to solve these gaps in emotion detection field.(Sawsan N. Cassab,2018) built an Arabic emo-

tion Ontology (ArEmontology) for conceptualizing emotions in Arabic using standard language based on emotion theories. In addition to a proposition of a mechanism of emotion detection for Arabic text using classification and ArEmontology semantics, this came with a result of 65% percent accuracy in detecting one of six Paul Ekman’s models. N. Alswaidan [53], with a Hybrid-based approach for emotion detection proposed three models, a human-engineered feature-based (HEF) model, a deep feature-based (DF) model, and a hybrid of both models (HEF+DF) for emotion recognition in Arabic text. After evaluating and comparing the performance of those models on the SemEval-2018, IAEDS, and AETD datasets. Results showed that HEF+DF outperformed the HEF and DF on all datasets.

3.9.3 Algerian dialect Studies

Considering the Algerian dialect as a natural language there is diverse studies that aim on Algerian dialect text on social media such as, Assia Soumeur [54], were interested in the SA of Algerian users’ comments on various Facebook pages. they did a pre-processing of a corpus of such comments, and trained two neural network models, MLP and CNN to classify comments as negative, neutral or positive. they were able to obtain a 81.6% accuracy with the MLP network and 89.5% accuracy with the CNN.(Meflah , 2017) in which they present a state of art about the sentiment analysis for the Algerian dialect.

Table(1.2) represent related works based on the applied approach, sentiment analysis(SA) or emotion detection(ED), and on the language/ dialect used:

study	year	type	language/dialect
MO hegazi	2021	SA	Standard Arabic
El-beltagi	2017	SA and ED	Standard Arabic
SEDAT	2018	SA	Standard Arabic
Khaled M.O	2017	SA	Standard Arabic and Jordanian dialect
Maghfour	2018	SA	Standard Arabic and Moroccan Dialect
Matrane	2022	SA	Moroccan Dialect
Lamia Belghit	2017	SA	Tunisian Dialect
Alqahtani	2022	ED	Standard Arabic
Rabie	2014	ED	Standard Arabic
Amria.f	2013	ED	Standard Arabic
Samar Al-saqqa	2018	ED	Standard Arabic
Sawsan N. Cassab	2018	ED	Standard Arabic and Syrian dialect
Al-swaidan	2020	ED	Standard Arabic
Assia soumeur	2018	SA	Algerian Dialect
Meflah	2017	SA	Algerian Dialect

Table 3.1: Related works with Emotion detection.

3.10 Conclusion

In this chapter we presented emotion detection (ED) from social media, especially text data. We started by defining emotion and ED concepts and their models. Moreover, we talked about emotion detection from social text and from social media, specifically Twitter. as well as some challenges for ED on social media. We also explored approaches for text ED. Finally, a collection of works related to the ED context was presented.

In the next chapter, we will propose our approach for text clustering in the context of ED.

Chapter 4

Conception

4.1 Introduction

The main approach to this work is based on emotion detection through text. It was relied on Twitter tweets because it becomes a huge repository of a lot of important data, as it is a major destination for data analysts and researchers, especially those interested in emotion analysis. In order to do our analysis, we must go through a number of important steps of collecting, processing and categorizing data. In this section, we will explain each step in detail.

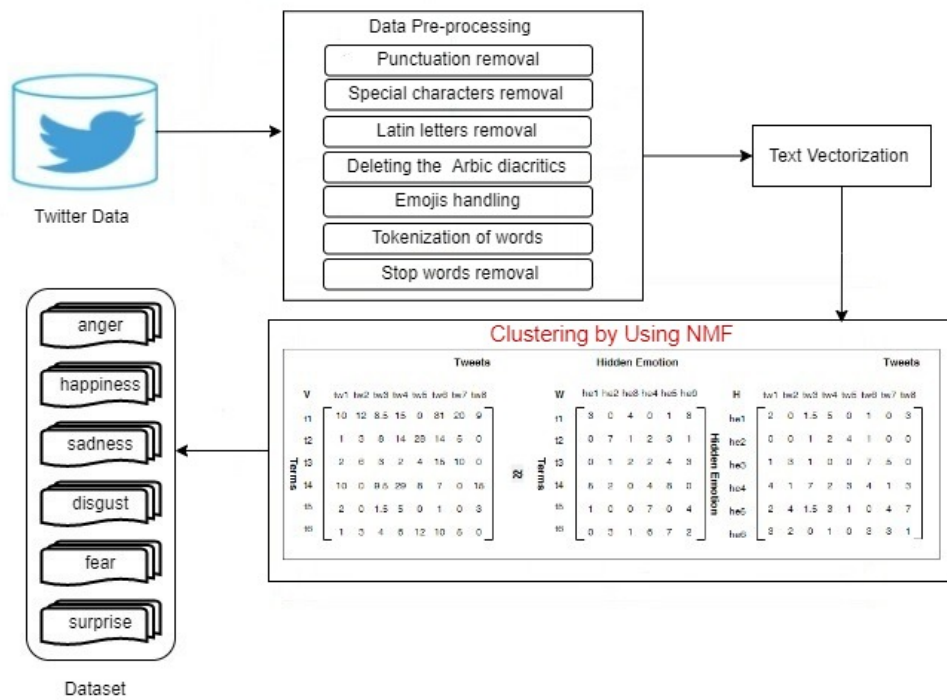


Figure 4.1: Algerian Dialect Emotion Analysis Process

4.2 Twitter API

We can access Twitter via the web or on a mobile device. To share information on Twitter as widely as possible, companies, developers, and users can be provided with programmatic access to Twitter data through APIs (application programming interfaces).

At a high level, APIs are the way computer programs "talk" to each other so that they can request and deliver information. This is accomplished by allowing a software program to call what's known as an endpoint; an address that corresponds with a specific type of information (endpoints are generally unique like phone numbers). Twitter provides API access to aspects of its service so that people may create software that interacts with Twitter, such as a solution that enables a firm to respond to client feedback on Twitter. Twitter data differs from most other social platform data in that it represents information that users choose to post publicly. The API platform gives customers comprehensive access to public Twitter data that they have decided to share with the rest of the world. The APIs also allow users to manage their own non-public Twitter information (e.g., Direct Messages) and share it with developers whom they have approved.

4.3 Data collection

A tweet crawler that collects a set of related tweets by querying the Twitter web service. Another way to use the software interface is the Twitter Application (API). This is an API provided by Twitter that gives developers the ability to use Twitter functionality such as retrieving from tweets using the chosen keyword and language.

First of all, our focus will be on Twitter data. It can be fetched from the official API. We can access the API and search for any query. It returns some data, including the following:

- user: String. Username.
- text: String. Tweet text (max. 280 chars).
- created at: Date Time. Timestamp with the post creation time.

4.4 Text Preprocessing

Natural Language Processing (NLP) is a branch of data science that deals with textual data. Apart from numerical data, text data is also widely used in analyzing and solving business problems. It is important to process the data before it is used for analysis or forecasting. We perform text preprocessing to prepare the text data for model building and this is the first step

in NLP projects. All pretreatment steps are shown in the figure:



Figure 4.2: Text Preprocessing

4.4.1 Deleting the Arabic diacritics

The Arabic language is vast and morphologically complex. A collection of morphological properties that results in a large number of rich word forms, resulting in more diversity (several versions of the same phrase).

We delete the diacritics to collect the weight of words with the same writing.

4.4.2 Stop words removal

Stop words are commonly removed from text prior to training machine learning models since they appear frequently and provide little to no unique information that may be used for clustering.

All these words have been deleted from all tweets, as an example of some stop words in the Arabic language: [أنا، نحن، أنتن، أنت، أو، إلى، في] :



Figure 4.3: Stop words removal

4.4.3 Punctuation removal

A frequent text preprocessing method is the removal of punctuation from textual data. The removal of the punctuation method will assist in treating each text equally. When the punctuation is deleted, the words "data?" and "data" are treated the same.

Tweets contain a lot of punctuation, in this step we remove all punctuation from tweets as the %'() * +, -. /::? @ ,|

4.4.4 Numbers removal

Because we're dealing with text, the number may not be very useful in text processing. As a consequence, numbers in text can be removed.

4.4.5 Latin letters removal

We eliminate all of the Latin characters in this step since the tweets collected are written in the Arabic language and Algerian dialect.

4.4.6 Special characters removal

Special characters are considered non-alphanumeric characters. These characters are frequently seen in remarks, allusions, monetary numbers, and so on. These characters contribute nothing to text comprehension and add noise to algorithms. We can get rid of these letters and numbers.

In tweets we always find links tagging someone is a way to share strong feelings. We remove all tags on people because they are not useful for sentiment analysis for example @sami, @marouane,

4.4.7 Emojis handling

Emoji are small digital images or icons that are used to represent a thought or feeling. These are tiny enough to be included in the text. In Japanese, "e" stands for image and "moji" stands for character.

Removing emojis from text to parse text may not be a good decision. Sometimes they can provide strong information in text, especially in emotion analysis and removing them may not be the right solution.

In emotion analysis we try to identify a feeling from a text. In this case, removing the emoji is not the right decision because we may lose valuable information.

Emojis convey an emotional expression in a text message to analyze the text that we may need to handle carefully.










:smiling face with tear
:loudly crying face
:weary face
:slightly smiling face
:rolling on the floor laughing
:face with tears of joy
:smiling face with heart-eyes
:face blowing a kiss
:crying face

Figure 4.4: Emojis

we want text that represents those emojis. We compile a list of emojis with corresponding words and this dictionary was obtained with the help of Unicode emoji. We will use this dictionary to convert emojis into corresponding words and we will translate these corresponding words into Arabic in different ways.

- Subjective translation

In this part we want text that represents those emojis. We compile a list of emoji with the corresponding words and this dictionary has been obtained with the help of Unicode emoji. We will use this dictionary to convert emojis into corresponding words and we translate these corresponding words into Arabic in two ways.

(2) At this stage, you translate words literally using Google Translator

(1) At this stage, we translate the emotion of the absence of some words and focus on the importance of emotion only

(1)	(2)
 :مبتسم	 :وجه مبتسم بالدموع
 :يبكي	 :وجه يبكي بصوت عالٍ
 :مرهق	 :وجه مرهق
 :مبتسم	 :وجه مبتسم قليلا
 :يضحك	 :يتدحرج على الارض من الضحك
 :فرحان	 :وجه بدموع الفرح
 :مبتسم	 :وجه مبتسم بعيون قلب
 :يقبل	 :وجه يرسل قبلة
 :يبكي	 :وجه يبكي

Figure 4.5: Handling emojis

- Emoji lexicon

We created an emoji dictionary based on the top 100 most often used emojis and catego-

rized them into six categories based on their official annotations and emotions expressed: happiness, anger, sadness, fear, disgust, and surprise.

- **Happiness:** { 😊 😄 😁 😂 😃 😅 😆 😇 😈 😉 😊 😋 😌 😍 😎 😏 😐 😑 😒 😓 😔 😕 😖 😗 😘 😙 😚 😛 😜 😝 😞 😟 😠 😡 😢 😣 😤 😥 😦 😧 😨 😩 😪 😫 😬 😭 😮 😯 😰 😱 😲 😳 😴 😵 😶 😷 😸 😹 😺 😻 😼 😽 😾 😿 😺 😻 😼 😽 😾 😿 }
- **Sadness:** { 😭 😞 😟 😠 😡 😢 😣 😤 😥 😦 😧 😨 😩 😪 😫 😬 😭 😮 😯 😰 😱 😲 😳 😴 😵 😶 😷 😸 😹 😺 😻 😼 😽 😾 😿 }
- **Anger:** { 😠 😡 😢 😣 😤 😥 😦 😧 😨 😩 😪 😫 😬 😭 😮 😯 😰 😱 😲 😳 😴 😵 😶 😷 😸 😹 😺 😻 😼 😽 😾 😿 }
- **Surprise:** { 😲 😳 😴 😵 😶 😷 }
- **Disgust:** { 🤢 🤮 🤨 🤩 }
- **Fear:** { 😱 😲 😳 }

Figure 4.6: Classification of emoji

4.4.8 Tokenization of words

We use the method (Tokenization of words) to split a sentence into words. The output of word tokenization can be converted to Data Frame for better text understanding in machine learning applications. Machine learning models need numeric for clustering. Word tokenization becomes a crucial part of the text (string) to numeric data conversion.

Tokenization in the Arabic language is a challenging task because of its rich morphology. Tokenization is done as shown in Figure 4.6.



Figure 4.7: Tokenization example.

4.5 Text Vectorization

This section describes a typical approach for transforming text documents into vector representations. The suggested tf-idf (Term Frequency-Inverse Document Frequency) methodology, vectorizations, and tests for grouping tweets using Non-Negative Matrix Factorization are detailed.

The tf-idf score rises proportionately to the number of times a certain word appears in a given tweet (term frequency) and falls correspondingly to the total number of tweets in the corpus (inverse-document frequency).

The tf-idf matrix converts all tweets into rows, with all words kept as column vectors. The tf-idf score is calculated by taking the product of tf and idf.

$$TFIDF(t, d, D) = TF(t, d) * IDF(t, D)$$

Term Frequency (TF):

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

Inverse Document Frequency (IDF)::

$$IDF(t, D) = \log\left(\frac{\text{Total number of documents}}{\text{Number of documents containing } t}\right)$$

	text	tf	idf
0	Eddard Stark is a king in the north.	1/8	log(4/3)
1	A king but one king : kings are everywhere.	2/8	log(4/3)
2	Hodor was different : he was not a king .	1/8	log(4/3)
3	But the North could not change without him.	0	log(4/3)

	king	was	the	not	a	he	one	north	kings	is	in	him	everywhere	A	different	could	change	but	are	Stark	North	Hodor	Eddard	
0	0.015617	0.0	0.5	0.0	0.5	0.0	0.0	1.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	1.0	
1	0.031234	0.0	0.0	0.0	0.0	0.0	1.0	0.0	1.0	0.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	0.0
2	0.015617	2.0	0.0	0.5	0.5	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0
3	0.000000	0.0	0.5	0.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	1.0	1.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0

Figure 4.8: tf-idf Vectorization

4.6 Clustering

Clustering, or cluster analysis, is an unsupervised learning problem. It is often used as a technique of data analysis to discover interesting patterns in the data, such as groups of customers based on behavior, opinions, emotions, etc.

There are many clustering algorithms to choose from and there is no single best clustering algorithm for all cases. In this work, Non-negative matrix factorization (NMF) was selected

and, according to our review, we found that it may give acceptable results in text classification. Before applying this algorithm, we must first convert text values into numeric, because all classification algorithms accept numeric input, we convert text values into numeric values and it is called Text Vectorization and this works with the so-called TF-IDF and we will discuss what was mentioned in more detail

4.6.1 Cluster analysis with NMF

The Data Analysis phase is dedicated to revealing the hidden topics from tweets. This can be done by exploiting the factors W and H obtained at the end of the NMF Decomposition phase. Particularly, the approach adopted for the extraction of the hidden topics is illustrated in Figure 4.9 After topics have been extracted and interpreted, tweets can be clustered accordingly. Furthermore, this phase allows to effectively visualize each cluster and to easily display its semantics using appropriate tools

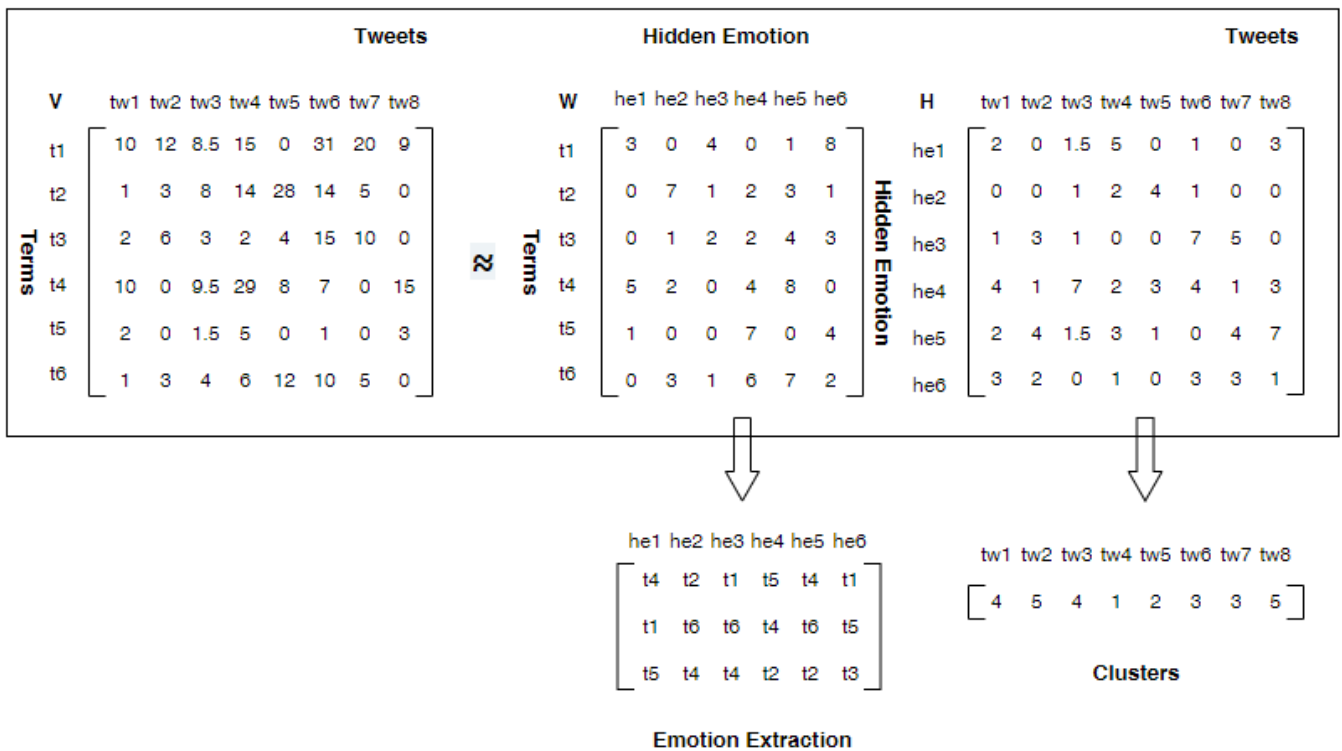


Figure 4.9: Cluster analysis with NMF

In the case of emotion analysis, for example, the expression data matrix V can be represented as a tweets matrix, where m is the number of tweets and n the number of terms. The k columns of W , therefore, will have the dimension of a single array (m tweets) and are known as basis terms or factors. Similarly, the n columns of H are known as encoding vectors and are in one-to-one correspondence with a single term of the emotion data matrix. Consequently, each row

of H has the dimension of a single tweet (n terms) and it is denoted as basis emotion.

4.6.2 Emotion extraction

As previously highlighted, the columns of the matrix factor W stand for the hidden emotions embedded into the vector space describing the tweets (after the pre-processing phase). In particular, a column W_k in W associates a weight W_k to each term in V : the higher this weight, the more important the term in defining the hidden emotion. To allow an easier interpretation of hidden emotions, for each column of W the topmost r terms (in order of their weight) can be selected. The example reported in Figure 4.8 illustrates how to extract the hidden emotions using the matrices W and H , by electing the topmost three terms from each column of W . It is important to highlight that the emotions are automatically discovered by analyzing the original tweets; in fact they usually are not known in advance but are learned from data.

4.6.3 Clusters

Each tweet exhibits multiple topics with different relevance. Through NMF it is possible to suggest the importance of each emotion in each tweet. The encoding matrix H maps the hidden emotions (rows of H) with the tweets (columns of H), and elements h_{ij} indicate the importance (weight) that the i -th emotion has in the j -th tweet. In the example in Figure 4.8, the first tweet $tw1$ is about the hidden emotions $he1$ and $he3$ with weights 2, and 1, respectively, while it does not refer to the emotion $he2$. Each tweet is represented as a vector in the new space spanned by the column vectors W_j . Thus NMF can be used for hard document clustering by assigning the tweets to the nearest basis in the space. This is equivalent to assigning each tweet to the emotion with the highest weight in the column of H . Since NMF and spherical k -means have been proved to be equivalent, each hidden emotion in W can be seen as the centroid of a cluster. However, due to the non negativity of NMF (and differently from the k -means) each centroid is also interpretable. In the example (reported in Figure), the tweets $tw1$, $tw3$, $tw4$, $tw8$ are assigned to the first cluster (that is about the terms $t4$, $t1$ and $t5$), the tweet $tw5$ to the second cluster and the tweets $tw6$ $tw5$ to the third cluster.

4.7 conclusion

Arabic text clustering has its own characteristics compared to other languages, and needs special treatment. In this chapter, we proposed our method for Arabic text clustering in the

concept of emotion detection. Beginning with the data collection to the preprocessing phase, finishing with the algorithms used in the processing phase.

The next chapter will discuss the implementation of our work with the results.

Chapter 5

implementation

5.1 Introduction

The main approach is based on Emotion Detection through text. To fetch data, we have relied on Twitter because it is a huge repository of a lot of important data, as it is a major destination for data analysts and researchers, especially those interested in sentiment analysis. We have used Python 3.0.1 and the available python libraries. we chose Python due to its ease of development and the available range of powerful libraries (eg scipy, numpy, sklearn).

In this chapter we will explain the different steps of implementing our project, starting with the required software and input methods to the final results, all after passing the necessary tests to determine some statistics according to the output results. It will be presented in three main parts: The first part reveals the programming language and the appropriate development environment. The second part describes the work proposal required in detail, and in the last part we present the results and their accuracy.

5.2 Presentation of technologies and languages

5.2.1 VS Code

Visual Studio Code (famously known as VS Code) is a free open source text editor by Microsoft. VS Code is available for Windows, Linux, and macOS. Although the editor is relatively lightweight, it includes some powerful features that have made VS Code one of the most popular development environment tools in recent times.

5.2.2 jupyter

JupyterLab is the latest web-based interactive development environment for notebooks, code, and data. Its flexible interface allows users to configure and arrange workflows in data science,

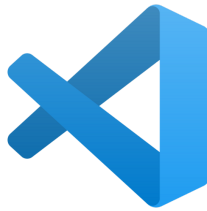


Figure 5.1: VS Code logo

scientific computing, computational journalism, and machine learning. A modular design invites extensions to expand and enrich functionality.



Figure 5.2: Jupyter

5.2.3 Python language

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, 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.

5.2.4 Scikit-learn

Scikit-learn is a popular and robust machine learning library that has a vast assortment of algorithms, as well as tools for ML visualizations, preprocessing, model fitting, selection, and evaluation.

Building on NumPy, SciPy, and matplotlib, Scikit-learn features a number of efficient algorithms for classification, regression, and clustering. These include support vector machines, rain forests, gradient boosting, k-means, and DBSCAN.



Figure 5.3: Python

Scikit-learn boasts relative ease-of-development owing to its consistent and efficiently designed APIs, extensive documentation for most algorithms, and numerous online tutorials.

Current releases are available for popular platforms including Linux, MacOS, and Windows.

5.2.5 NumPy

NumPy is the fundamental package for scientific computing in Python. It is a Python library that provides a multidimensional array object, various derived objects (such as masked arrays and matrices), and an assortment of routines for fast operations on arrays, including mathematical, logical, shape manipulation, sorting, selecting, I/O, discrete Fourier transforms, basic linear algebra, basic statistical operations, random simulation and much more.

5.2.6 Pandas

Pandas is an open-source library that is made mainly for working with relational or labeled data both easily and intuitively. It provides various data structures and operations for manipulating numerical data and time series. This library is built on top of the NumPy library. Pandas is fast and it has high performance productivity for users.

5.2.7 Matplotlib

Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python. Matplotlib makes easy things easy and hard things possible.

Create publication quality plots. Make interactive figures that can zoom, pan, update. Customize visual style and layout. Export to many file formats. Embed in JupyterLab and Graphical User Interfaces. Use a rich array of third-party packages built on Matplotlib.

5.2.8 NLTK

Natural Language Toolkit 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.

Thanks to a hands-on guide introducing programming fundamentals alongside topics in computational linguistics, plus comprehensive API documentation, NLTK is suitable for linguists, engineers, students, educators, researchers, and industry users alike. NLTK is available for Windows, Mac OS X, and Linux. Best of all, NLTK is a free, open source, community-driven project.

NLTK has been called “a wonderful tool for teaching, and working in, computational linguistics using Python,” and “an amazing library to play with natural language.”

5.2.9 Googletrans

Googletrans is a free and unlimited python library that implemented Google Translate API. This uses the Google Translate Ajax API to make calls to such methods as detect and translate.

5.3 Twitter api

Twitter is a social networking site and news site, people pass through it through short messages called tweets, and each tweet is limited to a certain number of characters, which is 280, and Twitter is a goldmine of data. Unlike other social platforms, almost every user's tweets are completely public and can be pulled. This is a big plus if you are trying to get a large amount of data to run analytics on. Twitter data is also very specific. The Twitter API allows you to perform complex queries such as pulling every tweet about a specific topic for the past 20 minutes, or pulling a specific user's tweets that haven't been retweeted.

```
import tweepy

# == OAuth Authentication ==
# The consumer keys can be found on your application's Details
# page located at https://dev.twitter.com/apps (under "OAuth settings")
consumer_key="M2p08LERCJ1Hijzh4jDAfH6b1"
consumer_secret="u1AMtSZUneVKpdj42KRzNtzjiWesIJ6kpZVc4wtEDd7HfCTAWt"

# After the step above, you will be redirected to your app's page.
# Create an access token under the the "Your access token" section
access_token="3352155622-sekw2jMcLsH9ugHk4GS0a2ecrelZErIBEb9fEyn"
access_token_secret="toXAH1mV2cR7mstdF18h4AExs0jpCuzZ0VYyR9CkhBhtD"

# OAuth process, using the keys and tokens
auth = tweepy.OAuthHandler(consumer_key, consumer_secret)
auth.set_access_token(access_token, access_token_secret)

# Creation of the actual interface, using authentication
api = tweepy.API(auth,wait_on_rate_limit=True)

results = api.search_tweets(q="place:d73762e172ac7c37",count=100)
```

Figure 5.4: Script of data collection from the Twitter API.

5.4 Data collection

After creating a free Twitter developer account, which gives us access, we were able to fetch a large number of tweets specific to the Algeria region from the Twitter API, and this was done during a relatively large period for reasons of Twitter's limitations on the free service, as we were uploading about 6000 tweets were obtained with repetition, This was done using the Python code called Tweepy library.

We deleted the duplicate tweets after obtaining the required data and collecting it in one CSV file and this was done using code, to finally reach 3500 tweets consisting of three columns,

as shown in the following figure.

	user	text	created_at
0	user 1	@Ad64141455 😊 ما هو لونك المفضل 😊	2022-04-25 09:20:36+00:00
1	user 2	@tagmebshbdrhm بسك ما قستيهاش	2022-04-25 09:20:35+00:00
2	user 3	@gladiato2 @rafikh459 @jouuumii و فهمتش و...بصح ما	2022-04-25 09:18:40+00:00
3	user 4	@Dolgador محرز	2022-04-25 09:15:35+00:00
4	user 5	@Benayadachraf ميهمناش	2022-04-25 09:14:32+00:00

Figure 5.5: Capture of data

5.5 Tweets

Natural language processing, text preprocessing is the practice of cleaning and preparing textual data. NLTK and re are popular Python libraries for handling many text preprocessing tasks. We did this by following a number of steps :

- Stop words removal

The NLTK library is one of the oldest and most widely used Python natural language processing packages. Stop word removal is supported by NLTK, and the list of stop words can be found in the NLTK library. To remove stop words from a phrase, break your text into words and then remove the word if it appears in NLTK's list of stop words.

- Word tokenization

Word encoding is the process of breaking down a large sample of text into words. This is a requirement in natural language processing tasks where each word must be captured and subjected to further analysis such as categorizing, calculating particular emotions, etc. The Natural Language Toolkit (NLTK) is a library used to achieve this. We installed NLTK to encode the word

```
# Stop words removal and word Tokenization
stopwords = nltk.corpus.stopwords.words('arabic')
data.text = data.text = data.text.apply(
    lambda x: ' '.join([word for word in word_tokenize(str(x)) if word not in stopwords])
)
```

Figure 5.6: Script of word tokenization and stop words removal.

- Deleting the Arabic diacritics

Regular-expressions (regex) can be used to get rid of these diacritics.


```

import demoji
demoji.download_codes()

dall = {}
|
|
for i in df.text:
|     dall.update(demoji.findall(i))

p = pd.DataFrame(dall,index=['text']).T.reset_index()
p.to_csv('emoji.csv',encoding="utf-8-sig",index=False)

```

Figure 5.9: Script to get emojis from tweets.

After saving the file, we have the emoji with emotional expressions, as shown in the image, to be ready after the translation process for Arabic to process the text. As shown in the following figure,

	emoji	text
0	😭	smiling face with tear
1	😱	loudly crying face
2	😓	weary face
3	😊	slightly smiling face
4	🤪	rolling on the floor laughing
5	😂	face with tears of joy
6	😭	crying face
7	😍	smiling face with heart-eyes
8	😘	face blowing a kiss
9	😉	winking face

Figure 5.10: Emojis

1. Emoji lexicon

At this stage, the emoji was divided into 6 feelings and this was divided manually to be ready in the process of replacing the emoji with the appropriate text in the text processing

2. google Translator

In this step, unlike in others, we employ Python code to import the googletans package, which converts the emoji expressions collected from the demoji library from English to Arabic.


```

from googletrans import Translator, constants

translator = Translator()
emoji = pd.read_csv('emoji.csv', encoding="utf-8")
emoji.text = emoji.text.apply(
    lambda x: ' '.join([translator.translate(x, dest="ar").text])
)

emoji.head(n=10)

```

Figure 5.11: Emoji translation by Googletrans.

3. Subjective translation In this step, we translate emojis subjectively. We assign the global emotion to each emoji based on the definition of emojis.

(1)			(2)			(3)		
emoji	text		emoji	text		emoji	text	
0	حزين	🙄	0	وجه مبتسم بالدموع	😓	0	مبتسم	😊
1	حزين	❤️	1	وجه يبكي بصوت عالٍ	😭	1	يبكي	😭
2	حزين	😓	2	وجه مرهق	😓	2	مرهق	😓
3	سعيد	😊	3	وجه مبتسم قليلا	😊	3	مبتسم	😊
4	سعيد	😄	4	يتدحرج على الارض من الضحك	🤪	4	يضحك	🤪
...
96	سعيد	🍷	295	هدية ملفوفة	📺	295	هدية ملفوفة	📺
97	سعيد	🍷	296	حوري البحر	🌿	296	حوري البحر	🌿
98	حزين	👤	297	شمس خلف سحابة صغيرة	☁️	297	شمس خلف سحابة صغيرة	☁️
99	سعيد	👤	298	شمس خلف سحابة	☁️	298	شمس خلف سحابة	☁️
100	سعيد	👤	299	يد النصر: لون بشرة فاتح	👋	299	يد النصر: لون بشرة فاتح	👋

Figure 5.12: Handling emojis

After taking previous methods in dealing with emoji, we can obtain a dictionary of words for emoji, and we will deal with it at the stage of the text processor

After completing the processing of the text, we get a clean text that is ready to enter the next stage to work on it in analyzing feelings, as shown in the figure 5.13

	user	text	created_at
0	user 0	@Ad64141455 😊 ما هو لونك المفضل 😊	2022-04-25 09:20:36+00:00
1	user 1	@tagmebshbdrhm بسك ما قستيهاش	2022-04-25 09:20:35+00:00
2	user 2	@gladiato2 @rafikh459 @jouuumii ...بصح ما فهمتش و	2022-04-25 09:18:40+00:00
3	user 3	@Dolgador محرز	2022-04-25 09:15:35+00:00
4	user 4	@Benayadachraf ميهمناش	2022-04-25 09:14:32+00:00
5	user 5	@Ad64141455 🤔❤️🤔🤔🤔🤔	2022-04-25 09:14:00+00:00
6	user 6	😊😊😊😊😊 https://t.co/OKZE7HYx5O	2022-04-25 09:13:16+00:00
7	usre 7	واصفح عني إلهي . https://t.co/qm7zdhuZLG	2022-04-25 09:06:01+00:00
8	user 8	09:49 Temp. 19.3°C, Hum. 51%, Dewp. 8.2°C, Bar...	2022-04-25 09:00:10+00:00
9	user 9	...لو تأملت في حالك لوجدت ان الله أعطاك أشياء د	2022-04-25 08:58:52+00:00



	user	text	created_at
0	user 0	وجه مبتسم بالدموع لونك المفضل وجه مبتسم بالدموع	2022-04-25 09:20:36+00:00
1	user 1	بسك قستيهاش	2022-04-25 09:20:35+00:00
2	user 2	... بصح فهمتش وش دخل البدر فالناس كارهة زعما كفاه	2022-04-25 09:18:40+00:00
3	user 3	محرز	2022-04-25 09:15:35+00:00
4	user 4	ميهمناش	2022-04-25 09:14:32+00:00
5	user 5	... وجه مرهق قلب مجروح وجه يبكي بصوت عال وجه يبكي	2022-04-25 09:14:00+00:00
6	user 6	...وجه مبتسم قليلا وجه مبتسم قليلا وجه مبتسم قليل	2022-04-25 09:13:16+00:00
7	user 7	واصفح عني إلهي	2022-04-25 09:06:01+00:00
9	user 8	...تأملت حالك لوجدت ان الله أعطاك أشياء ان تطلبها	2022-04-25 08:58:52+00:00
12	user 9	...تحصل أيد شيء كامل ستحصل أشياء ناقصة تكتمل برضا	2022-04-25 08:55:24+00:00

Figure 5.13: Preprocessing

5.6 Text Vectorization

Scikit-Learn provides a transformer called the `TfidfVectorizer` in the module called `feature-extraction.text` for vectorizing documents with TF-IDF scores. Under the hood, the `TfidfVectorizer` uses the `CountVectorizer` estimator we used to produce the bag-of-words encoding to count occurrences of tokens, followed by a `TfidfTransformer`, which normalizes these occurrence counts by the inverse document frequency.

The input for a `TfidfVectorizer`, strings that contain a collection of raw tweets, similar to that of the `CountVectorizer`. As a result, a default tokenization and preprocessing method is applied unless other functions are specified.

```

from sklearn.feature_extraction.text import TfidfVectorizer

tfidf_vectorizer = TfidfVectorizer(
    min_df = 2,
    max_df = 0.95,
    max_features = 500
)
tfidf = tfidf_vectorizer.fit_transform(df.text.values.astype('U'))

```

Figure 5.14: TF-IDF Vectorizer script.

The vectorizer returns a sparse matrix representation in the form of ((tweets, term), tfidf) where each key is a document and term pair and the value is the TF-IDF score.

	آخر	الله	ألهم	آله	آمين	أبطال	أبي	أبيض	أتذكر	أجمعين	...	يقول	يكون
0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0
1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0
2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0
3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0
...
2853	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0
2854	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0
2855	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0
2856	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0
2857	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0

2858 rows × 1000 columns

Figure 5.15: Text Vectorization

5.7 Clustering

After import the sklearn package and properly preparing the data, we are now ready to apply the NMF algorithm. As illustrated in the graphic below, split the data into six cluster.

```

from sklearn.decomposition import NMF

nmf_model = NMF(n_components=6,random_state=42).fit(tfidf.toarray())
emotion_results_nmf_model = nmf_model.transform(tfidf.toarray())
df['nmf_model'] = emotion_results_nmf_model.argmax(axis=1)

```

Figure 5.16: Clustering by NMF

After the clusters process, we get the output of an NMF algorithm with data categorized from six group, and to extract the hidden emotion we need the W matrix resulting from the arithmetic operations of this algorithm whose dimensions are shown in Figure 5.16

	آخر	آله	آلههم	آله	آمين	أبطال	أبي	أبيض	أتذكر	أجمعين	...	يقول
0	0.000416	0.002400	0.004309	0.006582	1.309889	0.000000	0.001973	0.000000	0.000000	0.028889	...	0.000340
1	0.010406	0.011214	0.000784	0.014487	0.000000	0.000000	0.017713	0.000000	0.003141	0.024336	...	0.008158
2	0.003444	0.000062	0.000000	0.000000	0.000000	0.000000	0.000000	0.005212	0.000000	0.000000	...	0.000000
3	0.000000	0.000000	0.000059	0.000000	0.000000	0.000432	0.000000	0.000000	0.000000	0.000000	...	0.018690
4	0.012430	0.000000	0.000149	0.000000	0.000000	0.000117	0.000000	0.000000	0.000000	0.000000	...	0.000000
5	0.003221	0.000000	0.001213	0.005205	0.000000	0.008017	0.000000	0.121926	0.000160	0.005155	...	0.000019

6 rows × 1000 columns

Figure 5.17: H Matrix output NMF

It is difficult in the Arabic language to define the hidden sentiments in each category since several words might convey the same feeling, and for this reason, emoji transformed into text can assist us a lot in clarifying this task. We shall notice that the word sad has a higher weight with the number 2 in this category, as seen in the Figure 5.18 .

1	حاجة	حال	حالك	حاول	حب	حربها	حزر	حزين	حسبنا	حسبي
2	0.001346	0	0.001396	0	0.004007	0	0.003761	0	0	0
3	0	0	0	0	6.49E-05	0	0	0	0	0
4	7.02E-05	0.007356	0	0	0	0	0	3.202232	0	0
5	0.006773	0.009752	0.004423	0.004692	0.003143	0.009272	3.33E-05	0	0.014237	0.022578
6	7.42E-05	4.91E-05	0	0	0	0	0	0	6.51E-05	0
7	0.000371	0.013845	0.00036	0.000286	0	0	0	0	0	0

Figure 5.18: Emotion extraction from matrix H.

5.8 Results

After clustering We obtained three outcomes for each result, based on the prior treatment of the emoji, after collecting and extracting the hidden emotion and labelling the block as it is. When identifying the most correct emotion, it might be tough to compare these results with those in Arabic.

We were able to determine the emotion of each category in the first example, Figure 5.19 where emojis were gathered, and a higher percentage of fear and happiness were discovered compared to other emotions.

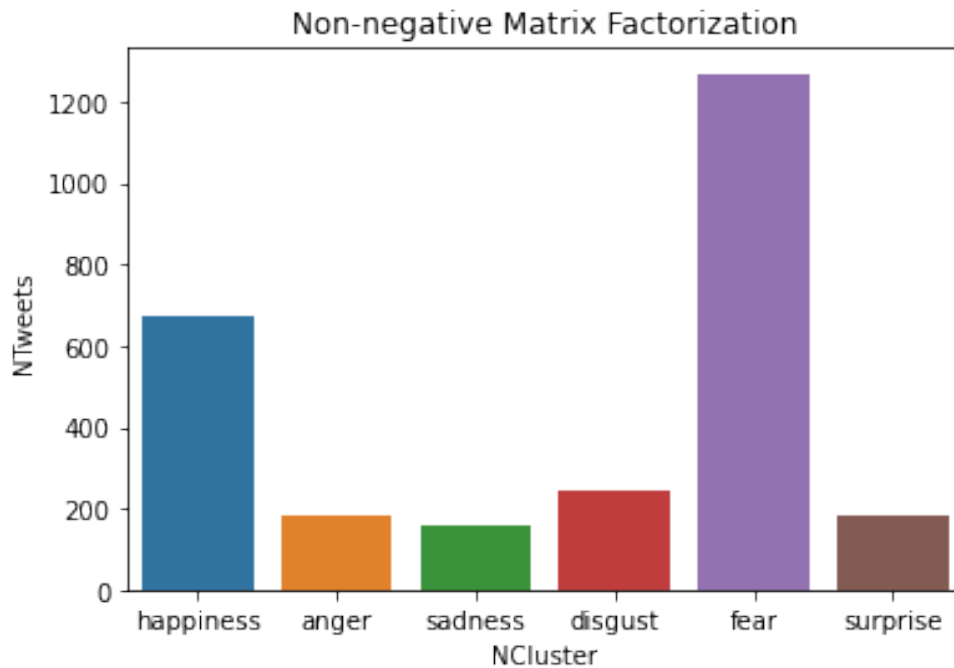


Figure 5.19: Emoji lexicon

In the second scenario, Figure 5.20, when the words were translated using Google Translator, distinguishing sentiments for each category is more challenging and may be less accurate than in the first. As a result, we concentrated on several terms that are commonly used to express a specific emotion, such as crying, which signifies sadness.

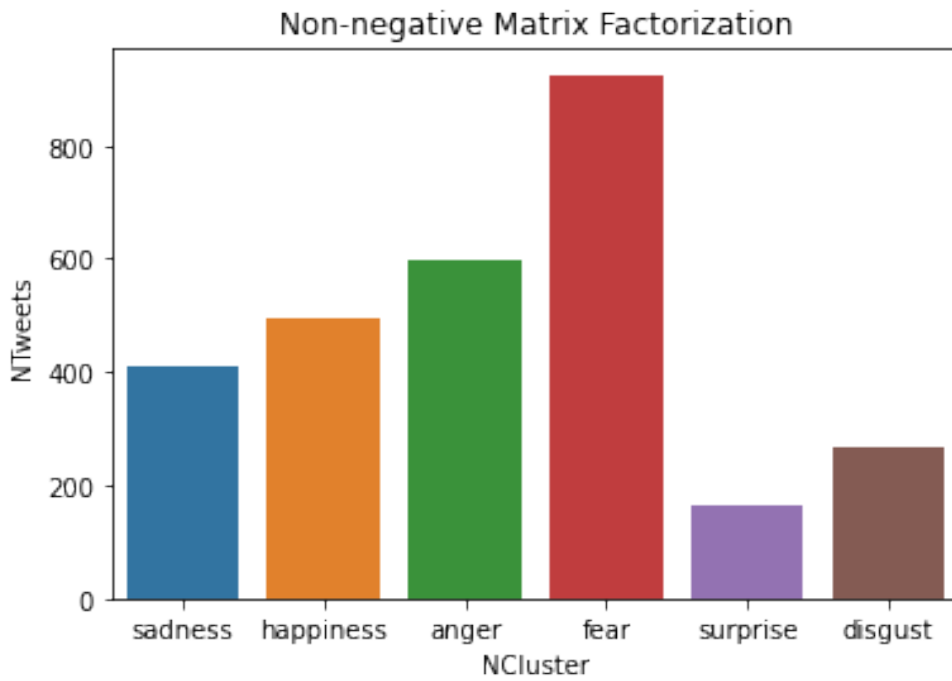


Figure 5.20: Emoji google Translator

The last scenario, Figure 5.21, which concentrated solely on the words of emotion without the requirement for additional words, may be better than the previous one in defining the hidden emotion in each category, but there is still the first technique is better for clarifying the emotion of each category.

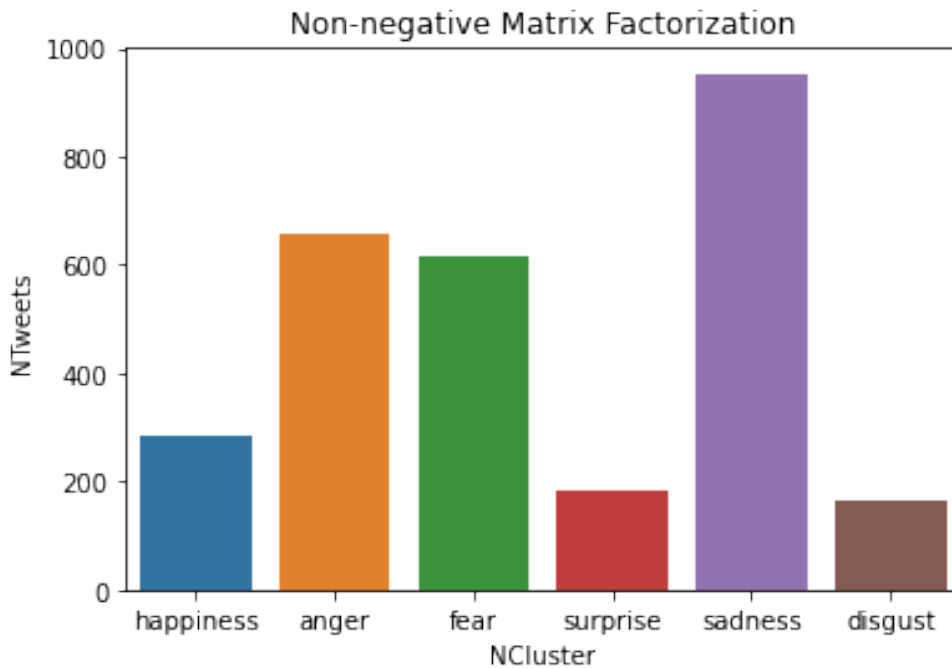


Figure 5.21: Emoji Subjective translation

5.9 Conclusion

After the presentation of our perspective to text clustering problem in the previous chapter. In this last chapter we displayed the implementation of our application in both phases preprocessing and processing (non matrix factorization algorithm application). In addition, we showed and discussed the obtained results.

Conclusion

Social media use has become a crucial everyday activity in today's culture. Social media is commonly used for social interaction, news and information access, decision making, and opinion expression. The excessive use of social media by people generates a huge amount of data that contains a huge amount of information. This information can be extracted and employed in several tasks. Text data is an important type of this data that appears in a form of natural language.

Natural language is the way humans express themselves and communicate with each other. The objective of understanding natural language by machines pushed researchers to apply automated techniques to the language. Natural language processing (NLP) is a branch of computer science and, more specifically, a branch of artificial intelligence (AI). NLP provides machines with the capacity to interpret text and natural language in the same manner that humans do. NLP uses machine learning techniques to achieve this goal. Text classification is an NLP task. This task has two forms: supervised (text categorization) and unsupervised (text clustering).

Although there was an orientation to apply NLP techniques to the Arabic text, and with less attention to the Algerian dialect, There is no access to labelled datasets to feed supervised approaches to text classification, especially in the Algerian dialect.

In our work we presented a tool for text clustering for Arabic text (Algerian dialect) in the context of emotion detection for the Algerian dialect. This tool can be used to transform raw text data to labelled dataset for six emotions (Happiness, Anger, Fear, Surprise, Sadness, Disgust). this dataset will be employed as a training dataset for supervised approaches.

For future work, text clustering evaluation techniques can be applied and discussed to provide an approximate standard for text clustering.

Bibliography

- [1] Taweh Beysolow II. *Applied Natural Language Processing with Python: Implementing Machine Learning and Deep Learning Algorithms for Natural Language Processing*. Apress, 2018.
- [2] Hannes Hapke, Cole Howard, and Hobson Lane. *Natural Language Processing in Action: Understanding, analyzing, and generating text with Python*. Simon and Schuster, 2019.
- [3] Kees Versteegh. *Arabic language*. Edinburgh University Press, 2014.
- [4] Salima Harrat, Karima Meftouh, Mourad Abbas, Walid-Khaled Hidouci, and Kamel Smaili. An algerian dialect: Study and resources. *International journal of advanced computer science and applications (IJACSA)*, 7(3):384–396, 2016.
- [5] Ahmed Abbasi and Hsinchun Chen. Applying authorship analysis to extremist-group web forum messages. *IEEE Intelligent Systems*, 20(5):67–75, 2005.
- [6] Mita K Dalal and Mukesh A Zaveri. Automatic text classification: a technical review. *International Journal of Computer Applications*, 28(2):37–40, 2011.
- [7] Rudolph Russell. *Machine Learning Step-by-Step Guide To Implement Machine Learning Algorithms with Python*. Rudolph Russell, 2018.
- [8] John Paul Mueller and Luca Massaron. *Machine learning for dummies*. John Wiley & Sons, 2021.
- [9] Charu C Aggarwal and ChengXiang Zhai. A survey of text clustering algorithms. In *Mining text data*, pages 77–128. Springer, 2012.
- [10] Michael Steinbach, George Karypis, and Vipin Kumar. A comparison of document clustering techniques. 2000.
- [11] Chiara Zucco, Barbara Calabrese, and Mario Cannataro. Sentiment analysis and affective computing for depression monitoring. In *2017 IEEE international conference on bioinformatics and biomedicine (BIBM)*, pages 1988–1995. IEEE, 2017.
- [12] M Emre Celebi. *Partitional clustering algorithms*. Springer, 2014.
- [13] Yan Zheng, Xiaochun Cheng, Ronghuai Huang, and Yi Man. A comparative study on text clustering methods. In *International Conference on Advanced Data Mining and Applications*, pages 644–651. Springer, 2006.
- [14] Tamanna Siddiqui and Parvej Aalam. Short text clustering; challenges solutions: A literature review. 06 2015.
- [15] Geli Fei and Bing Liu. Social media text classification under negative covariate shift. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2347–2356, 2015.
- [16] Sergio G Burdisso, Marcelo Errecalde, and Manuel Montes-y Gómez. A text classification framework for simple and effective early depression detection over social media streams. *Expert Systems with Applications*, 133:182–197, 2019.
- [17] KeYuan Wu, MengChu Zhou, Xiaoyu Sean Lu, and Li Huang. A fuzzy logic-based text classification method for social media data. In *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 1942–1947. IEEE, 2017.
- [18] Julia Ive, George Gkotsis, Rina Dutta, Robert Stewart, and Sumithra Velupillai. Hierarchical neural model with attention mechanisms for the classification of social media text related to mental health. In *Proceedings of the Fifth Workshop on Computational Linguistics and Clinical Psychology: From Keyboard to Clinic*, pages 69–77, 2018.

- [19] Manzhu Yu, Qunying Huang, Han Qin, Chris Scheele, and Chaowei Yang. Deep learning for real-time social media text classification for situation awareness—using hurricanes sandy, harvey, and irma as case studies. *International Journal of Digital Earth*, 12(11):1230–1247, 2019.
- [20] Mohammed Ali Al-Garadi, Yuan-Chi Yang, Haitao Cai, Yucheng Ruan, Karen O’Connor, Gonzalez-Hernandez Graciela, Jeanmarie Perrone, and Abeer Sarker. Text classification models for the automatic detection of nonmedical prescription medication use from social media. *BMC medical informatics and decision making*, 21(1):1–13, 2021.
- [21] Parisa Hajibabae, Masoud Malekzadeh, Mohsen Ahmadi, Maryam Heidari, Armin Esmaeilzadeh, Reyhaneh Abdolazimi, and H James Jr. Offensive language detection on social media based on text classification. In *2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC)*, pages 0092–0098. IEEE, 2022.
- [22] Kan Liu and Lu Chen. Medical social media text classification integrating consumer health terminology. *IEEE Access*, 7:78185–78193, 2019.
- [23] Akshi Kumar, Vikrant Dabas, and Parul Hooda. Text classification algorithms for mining unstructured data: a swot analysis. *International Journal of Information Technology*, 12(4):1159–1169, 2020.
- [24] Ricardo Martins, Marco Gomes, Jose Joao Almeida, Paulo Novais, and Pedro Henriques. Hate speech classification in social media using emotional analysis. In *2018 7th Brazilian Conference on Intelligent Systems (BRACIS)*, pages 61–66. IEEE, 2018.
- [25] Hissah ALSaif and Taghreed Alotaibi. Arabic text classification using feature-reduction techniques for detecting violence on social media. *International Journal of Advanced Computer Science and Applications*, 10(4), 2019.
- [26] Shailendra Kumar Singh and Manoj Kumar Sachan. Sentiverb system: classification of social media text using sentiment analysis. *Multimedia Tools and Applications*, 78(22):32109–32136, 2019.
- [27] Liping Jing, Lixin Zhou, Michael K Ng, and J Zhexue Huang. Ontology-based distance measure for text clustering. In *Proc. of SIAM SDM workshop on text mining, Bethesda, Maryland, USA*, 2006.
- [28] Shi Zhong. Efficient streaming text clustering. *Neural Networks*, 18(5-6):790–798, 2005.
- [29] Fasheng Liu and Lu Xiong. Survey on text clustering algorithm—research present situation of text clustering algorithm. In *2011 IEEE 2nd International Conference on Software Engineering and Service Science*, pages 196–199. IEEE, 2011.
- [30] Jiaming Xu, Peng Wang, Guanhua Tian, Bo Xu, Jun Zhao, Fangyuan Wang, and Hongwei Hao. Short text clustering via convolutional neural networks. In *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*, pages 62–69, 2015.
- [31] Wen Zhang, Taketoshi Yoshida, Xijin Tang, and Qing Wang. Text clustering using frequent itemsets. *Knowledge-Based Systems*, 23(5):379–388, 2010.
- [32] Aniket Rangrej, Sayali Kulkarni, and Ashish V Tendulkar. Comparative study of clustering techniques for short text documents. In *Proceedings of the 20th international conference companion on World wide web*, pages 111–112, 2011.
- [33] Antonio R Damasio. Emotion in the perspective of an integrated nervous system. *Brain research reviews*, 26(2-3):83–86, 1998.
- [34] Shiv Naresh Shivhare and Saritha Khethawat. Emotion detection from text. *arXiv preprint arXiv:1205.4944*, 2012.
- [35] Ekman P. *Basic emotions. Handbook Cognition and Emotion*. 1999.
- [36] Robert Plutchik. A general psychoevolutionary theory of emotion. In *Theories of emotion*, pages 3–33. Elsevier, 1980.
- [37] Andrew Ortony, Gerald L Clore, and Allan Collins. *The cognitive structure of emotions*. Cambridge university press, 1990.
- [38] James A Russell. A circumplex model of affect. *Journal of personality and social psychology*, 39(6):1161, 1980.

- [39] James A Russell and Albert Mehrabian. Evidence for a three-factor theory of emotions. *Journal of research in Personality*, 11(3):273–294, 1977.
- [40] Maryam Hasan, Elke Rundensteiner, and Emmanuel Agu. Automatic emotion detection in text streams by analyzing twitter data. *International Journal of Data Science and Analytics*, 7(1):35–51, 2019.
- [41] Ayushi Mitra. Sentiment analysis using machine learning approaches (lexicon based on movie review dataset). *Journal of Ubiquitous Computing and Communication Technologies (UCCT)*, 2(03):145–152, 2020.
- [42] Mohamed Osman Hegazi, Yasser Al-Dossari, Abdullah Al-Yahy, Abdulaziz Al-Sumari, and Anwer Hilal. Preprocessing arabic text on social media. *Heliyon*, 7(2):e06191, 2021.
- [43] Amr Al-Khatib and Samhaa R El-Beltagy. Emotional tone detection in arabic tweets. In *International Conference on Computational Linguistics and Intelligent Text Processing*, pages 105–114. Springer, 2017.
- [44] Malak Abdullah, Mirsad Hadzikadicy, and Samira Shaikhz. Sedat: sentiment and emotion detection in arabic text using cnn-lstm deep learning. In *2018 17th IEEE international conference on machine learning and applications (ICMLA)*, pages 835–840. IEEE, 2018.
- [45] Khaled Mohammad Alomari, Hatem M ElSherif, and Khaled Shaalan. Arabic tweets sentimental analysis using machine learning. In *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*, pages 602–610. Springer, 2017.
- [46] Mohcine Maghfour and Abdeljalil Elouardighi. Standard and dialectal arabic text classification for sentiment analysis. In *International conference on model and data engineering*, pages 282–291. Springer, 2018.
- [47] Yassir Matrane, Faouzia Benabbou, and Nawal Sael. Sentiment analysis through word embedding using arabert: Moroccan dialect use case. 07 2021.
- [48] Salima Mdhaffar, Fethi Bougares, Yannick Esteve, and Lamia Hadrich-Belguith. Sentiment analysis of tunisian dialects: Linguistic resources and experiments. In *Third Arabic Natural Language Processing Workshop (WANLP)*, pages 55–61, 2017.
- [49] Ghadah Alqahtani and Abdulrahman Alothaim. Emotion analysis of arabic tweets: Language models and available resources. *Frontiers in Artificial Intelligence*, 5, 2022.
- [50] Omneya Rabie and Christian Sturm. Feel the heat: Emotion detection in arabic social media content. In *The International Conference on Data Mining, Internet Computing, and Big Data (BigData2014)*, pages 37–49. Citeseer Kuala Lumpur, Malaysia, 2014.
- [51] Amira F El Gohary, Torky I Sultan, Maha A Hana, and MM El Dosoky. A computational approach for analyzing and detecting emotions in arabic text. *International Journal of Engineering Research and Applications (IJERA)*, 3(3):100–107, 2013.
- [52] Samar Al-Saqqa, Heba Abdel-Nabi, and Arafat Awajan. A survey of textual emotion detection. In *2018 8th International Conference on Computer Science and Information Technology (CSIT)*, pages 136–142. IEEE, 2018.
- [53] Nourah Alswaidan and Mohamed El Bachir Menai. Hybrid feature model for emotion recognition in arabic text. *IEEE Access*, 8:37843–37854, 2020.
- [54] Assia Soumeur, Mheni Mokdadi, Ahmed Guessoum, and Amina Daoud. Sentiment analysis of users on social networks: Overcoming the challenge of the loose usages of the algerian dialect. *Procedia computer science*, 142:26–37, 2018.