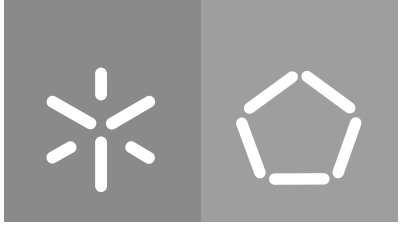
Universidade do Minho
Escola de Engenharia

Ricardo Alexandre Gonçalves Carotta Martins

Emotional state detection through text analysis

February 2022

Ricardo Martins **Emotional state detection through text analysis** UMinho | 2022



Universidade do Minho
Escola de Engenharia

Ricardo Alexandre Gonçalves Carotta Martins

Emotional state detection through text analysis

Doctoral Thesis
Doctoral Program in Informatics

Work developed under the supervision of
Paulo Novais
Pedro Henriques

February 2022

Acknowledgements

First of all, I would like to thank the professors Pedro Henriques, Paulo Novais and José João Almeida for all support, discussions and good times during this work. Also, I would like to thank the ISLab members, for all lunches, soccer, and good ideas that enriched this work.

For my familiy: Luciana - love of my life, Manuela - my little princess, thank you to embrace one more time another crazy idea of mine. Without you it couldn't be possible !

For my parents and brother, thank you for all positive words and support during difficult times.

At last, thank you for all friends - I will not cite individually here because I would be unfair if forget someone - whoo cheered for this work. In many times you gave me inspiration to keep in the right path.

STATEMENT OF INTEGRITY

I hereby declare having conducted this academic work with integrity. I confirm that I have not used plagiarism or any form of undue use of information or falsification of results along the process leading to its elaboration.

I further declare that I have fully acknowledged the Code of Ethical Conduct of the Universidade do Minho.

Braga

(Place)

Signature Not Verified

Assinado por: **RICARDO ALEXANDRE GONÇALVES CAROTTA MARTINS**

Num. de Identificação: BI32243997

Data: 2022.03.08 09.31.34 GMT Standard Time

CHAVE MÓVEL

(Ricardo Alexandre Gonçalves Carotta Martins)

"Só levo a certeza de que muito pouco sei." (Almir Sater)

Resumo

Detecção do estado emocional através de análise de textos

Atualmente, as pessoas são submetidas a rotinas intensas e muitas vezes exaustivas para que possam acomodar todas as expectativas e realizar seus desejos, sejam pessoais ou profissionais. Trabalhadores que exercem suas profissões diurnas e durante a noite tornam-se estudantes universitários, mães que têm que acomodar sua jornada profissional com as tarefas domésticas e jovens que dividem seu tempo em vários estudos escolares e profissionais - muitas vezes tendo que ajudar nos os lares - são alguns exemplos de perfis de pessoas que correm o risco de ter um problema emocional.

Apesar de atingir um grande número de pessoas, não é trivial saber quando alguém está a atingir seu limite, pois é impossível conectar fios e dispositivos que coletem dados por alguns dias para identificar problemas futuros. Além disso, o acesso a esse tipo de equipamento exige dinheiro e tempo, o que não é para todos. A abordagem proposta aqui é coletar dados para inferir o estado emocional atual, tais como (*stress*, fadiga, ansiedade, etc.) de forma não invasiva, transparente, barata, simples de usar e de fácil integração com outras sistemas, através da análise de textos curtos, como a troca de mensagens, redigidos através dos mecanismos de comunicação rápida hoje em voga (chats de e-mails e redes sociais, blogs, fóruns de discussão, SMS, etc.).

Para isso, a ideia é ensinar o computador a identificar as pistas deixadas nas mensagens de texto que revelem o estado emocional do autor. Usando técnicas de aprendizado de máquina (*machine learning*) e mineração de texto, várias mensagens previamente coletadas de diferentes fontes serão analisadas a fim de criar um modelo que classifique o estado emocional. Posteriormente, usando esse modelo de classificação, novas mensagens de texto podem ser analisadas para inferir o estado emocional atual do autor.

Após utilizar diferentes técnicas para extrair as emoções de textos, essas informações foram sintetizadas na criação um perfil emocional que foi utilizado em tarefas de classificação para identificar doenças como depressão, e prever comportamentos tanto individuais como coletivos, atingindo 98% de precisão na detecção de depressão.

Palavras-chave: Análise de Sentimentos, Aprendizado de Máquina, Processamento de Linguagem Natural

Abstract

Emotional state detection through text analysis

Daily a large number of people end up suffering car accidents, heart attacks and emotional problems due to stress. Some reasons like sedentary lifestyle and the poor quality of life experienced currently are easily associated with them in the literature.

Despite reaching a high number of people, it is non-trivial to figure out when someone is in his emotional threshold; especially when connecting body sensors, there is no alternative to collect data. Furthermore, monitoring emotional state conditions continually requires money and time. Our approach to collect human data is the analysis of text messages gathered from email or social networks chats, blogs, SMS's, and other fast communication mechanisms popular at present. This approach here proposed and discussed is useful to measure up the current emotional state (stress, fatigue, etc.) of a person in a non-invasive manner, transparent, cheap, simple to use, and easy to carry around through mobile devices usage.

To achieve this objective, the idea is to *teach* the computer to identify cues left in text messages which reveal the emotional state of the author. By using machine learning and text mining, several messages previously collected from different sources are analysed to create a model which classifies the emotional state. Later, using this classification model, new text messages can be analysed to classify the current emotional state.

After performed different techniques to extract emotions from texts, these information were used to create an emotional profile that was used in classification tasks to identify diseases such as depression, or predict individual or group behaviours, achieving 98% precision in depression detection.

Keywords: Machine Learning, Natural Language Processing, Sentiment Analysis

Contents

List of Figures

List of Tables

1 Introduction

Every day, a vast number of people take photos, make videos and send texts using their mobile equipment (smartphones, tablets, etc.). Businesses around the world collect data on consumer preferences, purchases and trends. Supported by appropriate regulations, governments collect all sorts of data concerning traffic over the internet to incident reports in police departments. This amount of data is growing fast. According to [33], in 2020 for each minute of the day, users share 150000 messages in Facebook, Amazon ships 6659 packages, WhatsApp users share 41666667 messages and users apply for 69444 jobs in LinkedIn. And this only a slice of the daily data production. The same company predicted in 2018 that in 2020 every person on earth will produce 1.7 Mb of data each second [32].

This massive amount of data is generated from several sources (and not restricted to): a single person, social groups or companies. For instance, personal smartwatches can measure and store the heartbeat; smartphones can calculate travelled distances, iPods can estimate the number of calories burned during the running, websites can store the user's navigation information. Regarding the data generated by a person and among many other sorts, all texts produced in the context of the so-called Computer Mediate Communications—like blogs, comments as replies to Social Network posts or Newspaper articles, chats, tweets , and so on — can contribute as a source of information characterizing the author.

All this individual/personal generated data, does not reveal, in a primary analysis, relevant information. However, after an in-depth analysis, it can provide relevant knowledge about the person involved.

Using data to infer information about users is not new. Researching on customer buying behaviour over time can reveal some unsuspected patterns – the most famous example is the relation between diapers and beer sold in larger than usual quantities on Friday nights in a retailer. The hypothesis was that husbands had to shop for diapers for the weekend while in their way home from work - and while there, they would pick up beer for the weekend sports on TV. This relationship allowed the store to locate diapers and beers close to each other so that more husbands might be reminded of the one they were most likely to forget.

In science areas like psychology and medicine, the textual information produced by a user can also be an essential source to identify patterns. For [62], expressing emotions can be done through writing, body language, or talking with other people. [48] link negative emotions to increased stress.

The automatic detection of emotions in texts is becoming essential in many different areas such as educational/edutainment games to collect feedback from users, financial to predict stock market prices and healthcare to detect the mood of patients while in treatment. When handling with emotions, there is common thinking that exists a straight connection between emotions, mood and personal profile, and it is a half-truth because mood expresses emotions. However, the mood is different from the emotional profile (or personality) while the emotional profile of a person stays rigid throughout our life, the mood is more open to change, influenced by the current emotions. In addition the emotional state is a snapshot of the mood in a specific moment, being the emotional states' collection along the time similar to the emotional profile.

1.1 Motivation

Emotions are present in most of our everyday occasions and manifestations during our life, such as in decision making and social relations. The research contributions in this area are significant and can provide theoretical and practical advances in human-machine interaction, and a better understanding of the technology influence has on human development [155], allowing the computer to adapt to people and not the opposite. The study of the emotions identification in texts is inserted in a multidisciplinary context, going from Psychology, through the Mining of Texts and Pattern Recognition to Human-Computer Interaction. The research area, known as Opinion Mining, has been experiencing growth, with many researchers working in automatic opinion evaluation on e-commerce and Opinion portals.

The PhD work here discussed aims at presenting a method for the identification of the emotional state through the processing of texts written in English. The emotions to be identified in the texts refer to the emotions proposed in the Plutchik's wheel of emotions [157]. Neutral texts (without emotion) are also identified by the method. For the evaluation of the proposed method, will be constructed a corpus of text messages of people demonstrating different levels of sentiment and a tool that allows the conduction of experiments with different configurations.

The motivation for this work is creating software that analyses the content of messages, extracting the emotions contained into these messages to infer the author's emotional profile and - using this information - classify the emotional state that the author presents at the writing time.

This classifier can be used in many different situations (and not limited to these ones): in a psychological treatment scenario, where a patient needs to be assessed constantly; in the scholarly context, identifying cues about bullying based on the student's emotional profile changing; in the labour context, helping to identify the workers under risk of stress.

However, this follow-up implies constant data collection, which can be considered somewhat invasive. In times when the limits of privacy and how a person's data are treated are discussed, it is necessary that the people whose texts are under analysis are aware and in agreement with this monitoring.

1.2 Objective and Research Questions

The objective of the doctoral project here discussed is to resort to techniques like text mining and machine learning to extract and classify emotional information, from unstructured documents, in order to identify the sentiment contained in the text that can be used to characterise the text's author emotional profile.

Thus, the research questions that will guide the execution of the work are:

A. Do emotional labels impact on the accuracy of classifications of the emotional states?

B. Is there an abstract model which differentiate the emotions considering personal perspectives ?

C. Based on the emotional state of a person, would be possible to predict his actions, choices or some situations which inspire concerning ?

To guide the answers for these questions, it were formulated these hypothesis:

A. The current emotional state of a person is expressed in his texts when writing;

B. The set of emotions detected in a large collection of texts of an author during a large interval represent his emotional profile;

C. The emotional state can be classified according to variations with the emotional profile.

The main contribution of this project will be an emotional state classifier through text analysis, natural language processing and machine learning algorithms.

At a formal description level, the proposal is the creation of an emotional state classifier ES:

$$ES = f(T) \tag{1.1}$$

where:
$T = \{txt_1, txt_2, ..., txt_n\}$ is a set of texts
and
$TXT = \{w_1, w_2, ..., w_n\}$ is a set of words, where w can contain emotional information.

1.3 Document Structure

This thesis is organized in 4 parts and begins with an introduction to the context, motivation and objectives in Chapter 1.

Part I includes Chapters 5 to 2 and presents the State of the art.

Part II just includes one chapter intended to present the PhD proposal.

Part III includes Chapters 7 to 12, aiming at presenting and discussing the case studies to which the proposed emotional classifier was applied.

Part IV presents a summary presented and some conclusions as well as and some directions for future work are presented.

Part I

State of the Art

This part introduces the state of art of the theories and technologies that were used during the work. First, it is presented abstract concepts and theories from psychology, such as emotions and the different models to represent them, the human personality and the personal vocabulary as source of personality differentiation.

Later, the a set of techniques from Natural Language Processing are presented, in order to explain how the information can be retrieved from texts and modelled to represent the abstract concepts presented earlier.

2

Emotions

Although emotions have been studied in several areas of knowledge such as Psychology, Neuroscience, Philosophy and Artificial Intelligence, there is no consensus on the definition of emotion because some issues such as its subjective nature, the divergence of researchers as to its origin, and the term used to describe a wide range of cognitive and physiological states [57]. This lack of consensus reinforces the idea of [47] which claimed that "everyone knows what emotion is until asked to give a definition. Then, it seems no one knows."

For [96], the concept of emotion can vary among several definitions, depending on the area of knowledge from which they arise. In the psychological and behavioural area, emotions can be understood as systemic responses that occur when highly motivated actions are delayed or inhibited. Thus, emotions concern the execution of something relevant to the organism.

[29] highlighted the fundamental role of emotions in the process of adaptation of living beings. According to the author, a facial expression indicating emotional states is a form of communication that is efficient and understandable to people, regardless of culture, and these striking characteristics are adaptive in all life forms.

In cognitive sciences, according to [28], emotions are body movements or actions visible to third parties. On the other hand, the sentiments are hidden in the organism where they occur, invisible to the public. Sentiments can arise by merely thinking about an event, and imagining what would happen if it would occur.

The differentiation presented above allows noticing that emotions in cognition have a much more visible, as well as conscious, role in the day-to-day. For example, the mood is widely used in educational settings as a way of motivating attention, developing affective sentiments towards the taught content, and promoting a more enjoyable learning experience [140].

Different scientific theories of emotion have been created over the years of research in cognitive sciences, each trying to explain the diversity of affect phenomena. These theories gave rise to three main models of emotions: discrete models, also called categorical ones; dimensional models; and models based on appraisal theory (cognitive models).

2.1 Discrete Models

Discrete model theories propose the existence of basic emotions (happiness, anger, sadness, surprise, disgust, and fear, for instance) that are universally displayed and recognised [38, 88], grouping emotions into categories and assuming that they are independent.

Many researchers have attempted to identify some basic universal emotions which are familiar to all people and differ one from another in significant ways. A famous example is a cross-cultural study by Paul Ekman and his colleagues, in which they concluded that the six basic emotions are anger, disgust, fear, happiness, sadness, and surprise [36]. One of the main advantages of discrete models is that, through psychophysical experiments, the perception of emotions by human beings is discrete. In this way, these models can easily associate emotions with facial expressions that represent them. The six basic emotions of Ekman are represented by six universal facial expressions, since they are understandable by people in different locations, regardless of culture. The facial expressions of the six basic emotions are shown in Figure 1.

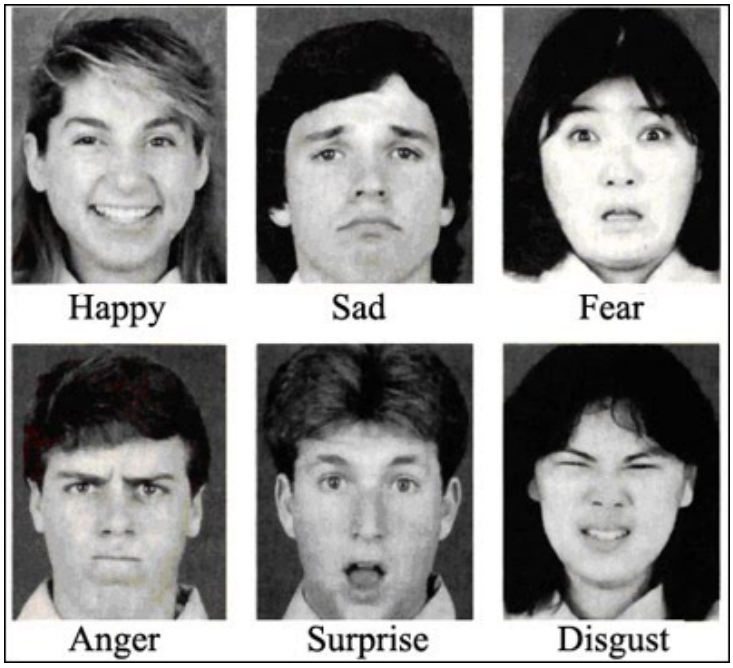


Figure 1: The six major emotions

A significant part of the work on emotion mining and classification from text has been adopted these basic emotion set [75, 160, 187]. For example, to model public mood and emotion, [75] extracted from Twitter six dimensions of mood including tension, depression, anger, vigour, fatigue and confusion. [187] created a significant data set with six basic emotions: anger, disgust, fear, joy, sadness and surprise.

However, there is no consensus on which human emotions should be categorised as basic and be included in the basic emotion set. Moreover, the emotional disambiguation is a contested issue in emotion

research. For instance, it is unclear if "surprise"should be considered an emotion since it can assume negative, neutral or positive valence.

2.2 Dimensional Models

Dimensional theories try to explain emotions regarding two or three dimensions. The most frequent dimensional characterisation of emotions uses two dimensions: arousal (intensity) and valence. Valence is related to positive or negative evaluation and is associated with the feeling state of pleasure (vs displeasure). Arousal reflects the general degree of intensity felt. Low arousal is associated with less energy and high arousal with more energy. However, using this two-dimensional is challenging to differentiate emotions that share the same values of valence and arousal, as *anger* and *fear*. For this reason, a third dimension (e.g. intensity) is often added. According to [101] "the third view emphasises the distinct component of emotions, and is often termed the componential view."

Among the dimensional models in 2D, one of the most known is the circumplex model of affect, proposed by [169], in which the emotions are related to each other, organised in a circle and represented in two dimensions: valence (pleasant versus unpleasant sentiment) and activation (awake versus sleepy). Followers of this affective model suggest that each affective experience is a consequence of a linear combination of these two dimensions, resulting in a particular emotion, as presented in Figure 2. Fear, for example, is conceptualised as a neurophysiological state involving the combination of negative valence and increased activation in the central nervous system.

Figure 2: The circumflex model of affect

In his study, [169] presented a list of 28 words that people use to describe sentiment, moods and emotions.

Later, this model was refined by [202] and is presented in Figure 3. The proximity or the distance between the emotions represented in the circumference presupposes the similarity or the difference between the emotions. This model defines that emotions are less positively related when they are spaced approximately 90 degrees apart. At 90 degrees of estrangement, two affective states must be little or nothing related. In turn, at 180 degrees of estrangement, affective states must be negatively related.

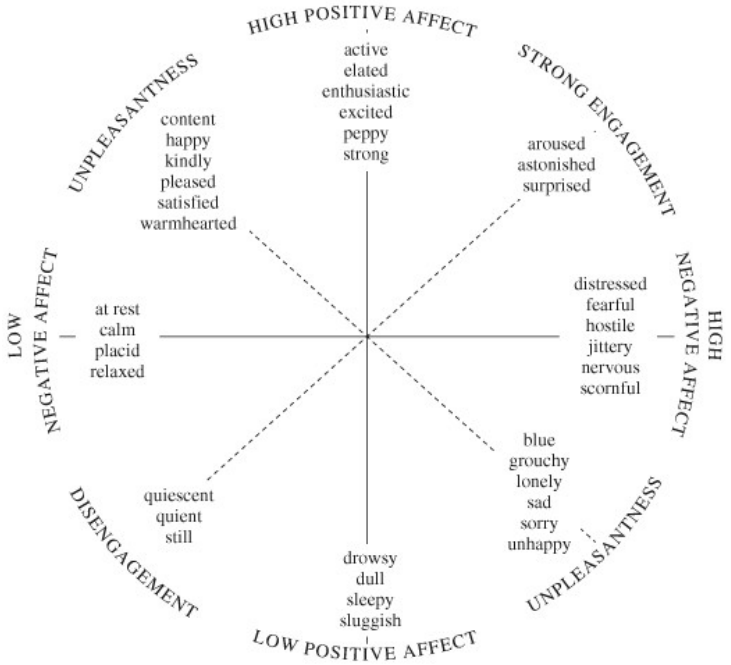


Figure 3: The Watson's two-dimensional structure of affect

In 3D dimensional models, the model proposed by [157] stands out. The model, as presented in Figure 4, proposes eight primary emotions: joy, sadness, anger, fear, disgust, surprise, anticipation, and trust. In this three-dimensional model, it is possible to verify the combination of emotions to generate others. The vertical dimension of the cone represents the intensity, and the circle represents the degree of similarity among the emotions. The eight sectors are designed to indicate that there are eight dimensions of primary emotions. Emotions in the blanks are the combination of two primary emotions. The emotional disposition is defined as presented in the [169], where the emotions are displayed in the wheel 180 degrees far from their opposite.

2.3 Cognitive Models

Emotional cognitive psychologists focus their studies mainly on the appraisal process. According to [179], the central idea is that emotions are triggered and differentiated by subjective analysis of an event, situation or object.

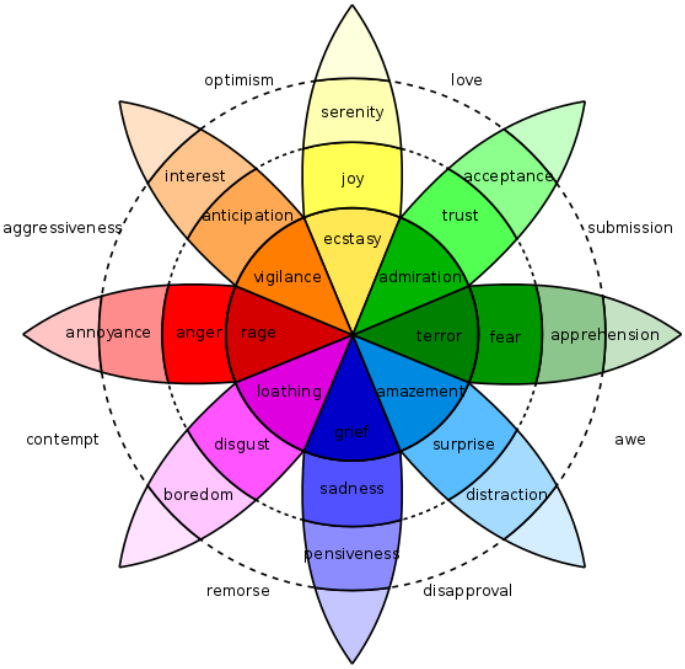


Figure 4: Plutchik's wheel of emotions. Extracted from [157]

This cognitive assessment performed personally is called appraisal. For instance, Paul and John are watching a basketball game where their favourite teams are playing. John's team wins (event). Paul's appraisal is that an undesirable event happened: his team lost. He is sad. For John, the situation's appraisal is that the event is desirable and he is happy. So, emotion and reason are not disconnected. Emotions require cognitive processes to generate or retrieve preferences and meanings. Emotions are triggered by the personal interpretation of the annoying or cheerful aspects of an event, the appraisal. Also, is the appraisal, a cognitive process, that triggers the emotions.

Based on appraisal theories concepts, the OCC Model [144] is an exciting model of emotions that provides a structure of the emotion-eliciting conditions and the variables that affect their intensities.

This model, so called by combining the initial letters of its authors, Ortony, Clore and Collins was organized in a structure with 22 types of emotions, listed below and kept in the English language: happy for, resentment, gloating, pity, joy, distress, pride, shame, admiration, reproach, love, hate, hope, fear, satisfaction, fears-confirmed, relief, disappointment, gratification, remorse", gratitude and anger. The OCC model is shown in Figure 5

As already said above, based on the cognitive theory of emotions, the OCC model considers that emotion arises from a cognitive evaluation, called "appraisal", generated by a person and based on the three aspects of the world: events, agents, and objects. Events are the sense, the occurrences by which people perceive what happens around them and their perception of the world in the face of consequences. Agents can be people, animals or, in some cases, other objects, or abstracted as institutions. So finally,

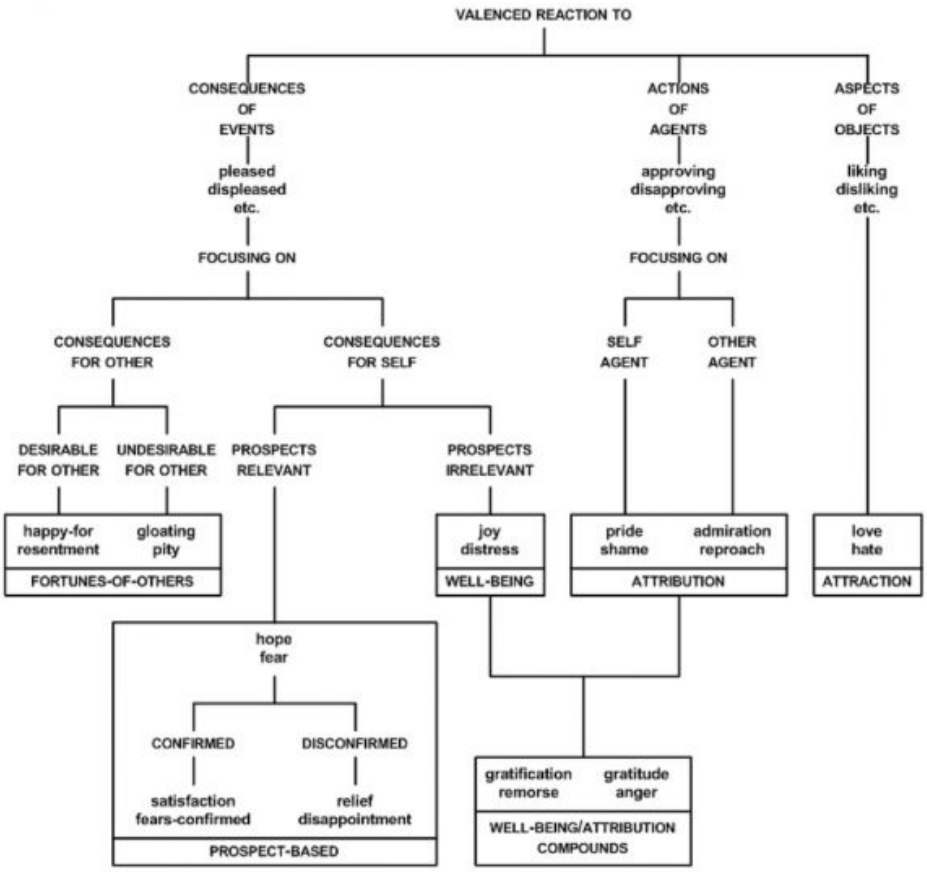


Figure 5: OCC Model

objects, which are perceived by people as objects present in the world, exerting an attraction, generating a positive or negative sentiment.

According to the cognitive appraisal structure, three interconnected components underlie the perceptions of good or bad: goals, standards and attitudes. The variable associated with reactions to events is desirability, that is, whether they promote or prevent someone from achieving their goals. The actions of an agent are analysed according to their compliance with standards and norms. So finally, objects are analysed as attractive according to the compatibility of their attributes as to the preference of someone [81].

Briefly, according to the OCC model, emotions can arise from:

A. Analysing the consequences of events according to their desirability about a person's goals;

B. Analysing (approving or disapproving) the actions of a person according to an individual's moral standards;

C. the attractive appeal of objects aspects.

.

The emotions of joy, sadness, fear, hope, satisfaction, relief, frustration and fear-confirmed arise by assessing the desirability of the consequences of an event for oneself.

Joy (happiness) can arise due to the occurrence of some desired event. On the other hand, the sadness occurs by the occurrence of an inconvenient fact or event. Also, when the event is expected, i.e. there is an expectation from the individual that the event happens or not, the emotions satisfaction, frustration, hope, fear, fear-confirmed and relief may arise.

Fear and hope can arise when there is no confirmation about the event happened, i.e. when the individual is still experiencing expectation. Fear occurs when the consequences of the event are undesirable according to the person's goals, and hope, otherwise. When this expected event happens, satisfaction or fear-confirmed emotions may arise. **Satisfaction** happens when an expected event that has already occurred is desirable, already **fear-confirmed** when the event is undesirable. When the expected event turns out to be unfulfilled, there may be frustration or relief. **Frustration** happens when there is confirmation that the expected and popular event did not happen. The **relief** happens in the same situation, but the event is not desirable.

All the emotions explained so far occur by assessing the event desirability by itself. However, events can also be assessed as undesirable or desirable to other people, and in this case, the individual may feel happy-by-other emotions, resentment, sorrow, or pleasure-for-evil-of-the-other.

When the event is desirable for another person, an individual may feel **joy for the other person** or **resentment**. When the event is undesirable for another, he may experience **pleasure from the evil of others** or pity.

When the object of evaluation is the action of a person, pride, shame, admiration or disapproval may occur.

Pride and **shame** happens when the individual approves (or disapproves) their action. When another person's action is approved (or disapproved), **admiration** or **disapproval** can be experienced.

There is yet another category of emotions that arise when a person analyses the actions of another or oneself about the interference in the accomplishment of its objectives. In this way, these emotions result in focusing simultaneously on an agent's action and the resulting event and its consequences.

When the individual analyses his actions, he can feel **gratification** when he approves of his action and he has positive consequences for himself, and **remorse** when he disapproves of his action and it has negative consequences. When he evaluates another person's actions, he may feel **gratitude** for that person when he approves that other person's actions, and also when the resulting event has positive consequences for him or her. Likewise, **anger** will arise when it disapproves of another's action and it still has negative consequences for you.

Finally, an individual may or may not like aspects of an object. When the object exerts a definite attraction, it feels **affection** to the object; otherwise, it feels **aversion**.

2.4 Affective Computing

The concept of Affective Computing (AC) was introduced by [155], who defined "the computing that relates to, arises from or deliberately influences emotions." The affective computing focus is on establishing models, based on physiological and behavioural signals collected by sensors and techniques to perceive, recognise and understand human emotions to provide better feedback. Emotion identification is used in the field of cognitive science [144] having a connection to affective computing enabling computers to recognise emotions [155].

According to [69], the arising of AC is related to the needs to put computers interacting directly, thinking, receiving and transmitting people's personalities. [155] and [69] emphasise AC as a research area which examines how computational systems can identify, classify, and prove human personality as well as group knowledge in other areas, such as psychology and cognitive science.

To achieve this objective, AC in computer investigates how computers could model, recognise, and respond to human behaviours, and thus how to express through an interface computational interaction. The purpose of promoting this characterisation is to contribute to increase the consistency, coherence and credibility of the reactions and computational responses provided during human interaction via human-computer interface [155].

To improve studies, [164] analysed the behaviour of people through computational agents, using the models of psychologists such as [144], [178] and [167], containing emotional characteristics. These characteristics have contributed to the consistency, coherence, and prediction of emotional response in computer responses.

For [137], after [155] the research on performing the analysis of emotions computationally increased significantly, even more when a great variety of environments of interaction between humans and computers arose, such as social media, chatbots, among others.

[105] defines Sentiment Analysis (SA) which is embedded in the Affective Computing as the area that analyses the opinions, feelings, evaluations, attitudes and emotions of the people expressed in written text. Under the SA concept, it is possible to find also many different names and tasks, such as Opinion Mining, Opinion Analysis, Opinion Extraction, Sentiment Mining, Subjectivity Analysis, Affect Analysis, Emotion Analysis and Review Mining. In industry, the term Sentiment Analysis is most commonly used, and in academic research both Sentiment Analysis and Opinion Mining are often employed. Regardless, they represent, basically, the same field of study. The term Sentiment Analysis appeared for the first time in [138], as the term Opinion Mining was introduced by [30].

2.5 Detecting Emotions Challenges

For the five human senses (sight, smell, touch, taste and hearing), two of them - sight and hearing - are responsible to transmit the emotions among persons. This transmission uses environments such as texts,

speeches and images, and according to the environment used, different techniques must be applied to detect these emotions.

2.5.1 Detecting Emotions in Texts

Lexicon-based and Machine Learning (ML) approaches are commonly used to solve SA problems in texts. Figure 6 shows the two main approaches and how they are subdivided.

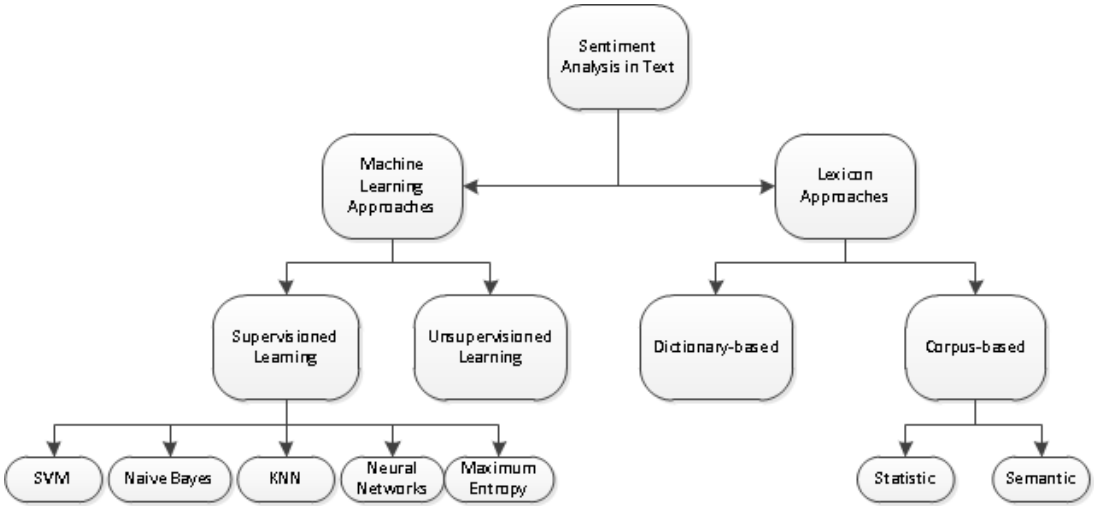


Figure 6: Approaches for Sentiment Analysis in text. Source: author

It is usual to find in the literature hybrid works that use both approaches to identify emotions in texts. [59] presents a work which uses lexicon and ML to identify the six basic emotions. [163] use a lexicon to help in the selection of features and ML algorithms to identify the six basic emotions in Spanish texts.

The ML-based approach is divided into supervised ML and unsupervised AM. Supervised methods require a large number of labelled texts to perform the training. Unsupervised methods are an option when there is difficulty in obtaining labelled text. The main algorithms based on supervised ML that are most commonly used in the classification of emotions in texts are presented in Section 2.5.1.2.

The lexical approach depends on the availability of an emotional terms lexicon. This approach is subdivided into two other approaches: dictionary-based and corpus-based. In the dictionary-based approach, initially, a small set of emotional terms is collected manually where each term has its associated emotion. This set of words, called seeds, is used to search in dictionaries such as WordNet Affect [188] or SentiWordNet [43] to explore semantic links, synonyms and antonyms and newly discovered terms are added to the seed list. The iterative process ends when no new terms are found [103].

In the corpus-based approach, there is also a list of emotional terms used to find other emotional terms in a large domain corpus. The objective is to aid in emotional terms searches with specific context guidelines. This can be done using statistical or semantic methods [103].

2.5.1.1 Lexicon-based approaches

For [66], in NLP-context, lexicons are component of a system that contains information (semantic and/or grammatical) about words or expressions, whereas the term dictionary usually refers to objects (printed books or electronic) intended for human readers, but also accessible by computers.

For [6] a lexicon is derived from the examination of a corpus, which in turn is defined in the Oxford English dictionary as an "organ, collection of writings."There is no minimum or maximum size for corpora, or any specification what it should contain.

The manual construction of a lexicon is an arduous task due to the large volume of information and the amount of time that is spent to carry out the steps. For this task, there are efforts in the creation of lexicons through computational techniques, for example, as presented by [149]. Another method for the construction of computational lexicons is the analysis and improvement of existing lexicons.

The initial point of any approach to study emotion in a text is the use of specific affective lexicons.[23] are one of the first researchers targeting the problem of the referential structure of the affective lexicon.

Affective lexicons are subsets of lexicons which are constructed based on different methodologies and provide bases for most machine learning algorithms. In our comprehension, these subsets are responsible for carrying emotional labels for the words. This vision is shared by Mohammad [134], that defines an affective lexicon as "a list of emotions and words that are indicative of each emotion."

[145] argue that affective lexicons do not only contain terms related to emotion but have other terms and affective conditions (affection, mood and sentiment). Terms such as "affection"and "emotion"are used, sometimes as synonyms. The distinction occurs when the term affection refers to anything whose valence value is positive or negative. Affection has a broader category when compared to emotion. Types of affective condition cause emotions, but not all affective conditions are emotions.

At the beginning of studies in affective lexicons, [8] analysed the data selected and considered having affective connotations from [4]. The objective was to develop a method, called "semantics", which would map a universe of words with affective characteristics. However, were not all words included in the analysis that had affectivity, which "justify that any division between affective concepts is necessarily vague and arbitrary"[8].

There is no pre-defined model for the construction of an affective lexicon. Most of the works have created defined steps and goals from studies to achieve the goal. Other works take lexicons already developed and implemented and make improvements and extensions. The following is a description of some more important lexicons and affective lexicons available:

WordNet WordNet[1] is an extensive English-language lexical database developed by George A. Miller, combining lexicographical information (lexical and semantic relations used to represent the lexical knowledge organisation) and computational resources [51]. [131] defines the vocabulary of a language with a set W with pairs (f, s), where a form f is a character string over a finite alphabet and s is a sense from a

[1] http://wordnet.princeton.edu

set of meanings. Each form with a sense in a language is called a word in that language. In WordNet, a sense is represented by a set of one or more synonyms. The base has more than 118,000 different words and more than 90,000 words [131].

WordNet is composed of nouns, verbs, adjectives and adverbs grouped into sets of synonyms (synsets), each expressing a distinct concept. These are organised into a set of lexicographers by syntactic category and by other organisational criteria. Adverbs are kept in only one file, while nouns and verbs are grouped according to semantics. Adjectives are divided into descriptive and relational adjectives [130].

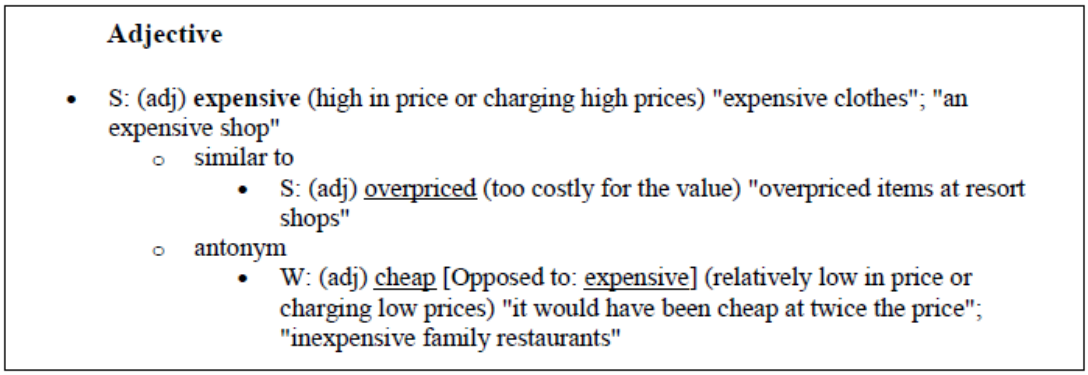


Figure 7: WordNet online search

Figure 7 illustrates the result of an online WordNet query for the word *expensive*. In the figure, it is possible to visualise the lexical relationship for the word expensive. In *similar to*, there is the term *overpriced*, which has the same synset and *antonym* corresponds to the opposite of the searched word. The word *expensive*, in the example, has only one sense in WordNet.

WordNet Affect WordNet-Affect [188] is an extension of the WordNet database [129], which includes a subset of synsets suitable to represent affective concepts. In particular, it allows WordNet synset to be assigned with one or more affective labels (a-labels). There are also a-labels for concepts representing moods, situations eliciting emotions, or emotional responses. Table 1 show some example of synsets.

It was extended with a set of additional a-labels (called emotional categories), hierarchically organised to specialise synsets with a-label emotion according to emotional valence: positive, negative, ambiguous, and neutral.

The positive valence corresponds to positive emotions, defined as emotional states characterised by the presence of positive hedonic signals. It includes synsets such as joy#1 or enthusiasm#1. Similarly, the negative a-label identifies negative emotions characterised by negative hedonic signals, for example, anger#1 or sadness#1. Synsets representing affective states whose valence depends on semantic context (e.g. surprise#1) were marked with the tag ambiguous. Finally, synsets referring to mental states that are considered affective but are not characterised by valence were marked with the tag neutral.

Another valuable property for affective lexicon is the stative/causative dimension. An emotional adjective is considered causative if it refers to some emotion that is caused by the entity represented by the

Table 1: Example of synsets

A-label	Example of synsets
EMOTION	noun 'anger', verb 'fear'
MOOD	noun 'animosity', adjective 'fear'
TRAIT	noun 'aggressiveness', adjective 'competitive'
COGNITIVE state	noun 'confusion', adjective 'dazed'
PHYSICAL state	noun 'illness'
HEDONIC signal	noun 'hurt', noun 'suffering'
Emotion-eliciting SITUATION	noun 'awkwardness'
Emotional RESPONSE	noun 'cold sweat', verb 'tremble'
BEHAVIOUR	noun 'offence', adjective 'inhibited'
ATTITUDE	noun 'intolerance', noun 'defensive'
SENSATION	noun 'coldness', noun 'feel'

modified noun (e.g. annoying movie). In the same way, an emotional adjective is said stative if it refers to the emotion owned or felt by the subject denoted by the modified noun (e.g. cheerful/happy boy).

SentiWordNet SentiWordNet (SWN) is another lexicon, developed by [43] explicitly for collaborate on applications for Data Mining and Opinion Classification. This lexical resource is the result of automated notations in all synsets of WordNet 3.0, with a degree of positivity, negativity and neutrality. Each synset is associated with three numeric values, Pos(s), Neg(s) and Obj(s) that indicate how positive, negative, or objective (neutral) the term contained in the synset is. Each of the three values varies in the range [0.0, 1.0] and their sum is 1.0 for each synset, so Obj (s) + Pos (s) + Neg (s) = 1.

Figure 8 illustrates the graphical representation adopted by SWN Representing properties related to synset. The edges of the triangle represent one of three classifications (positive, negative and objective) and a pointer (Synset position) points to the highest value classification.

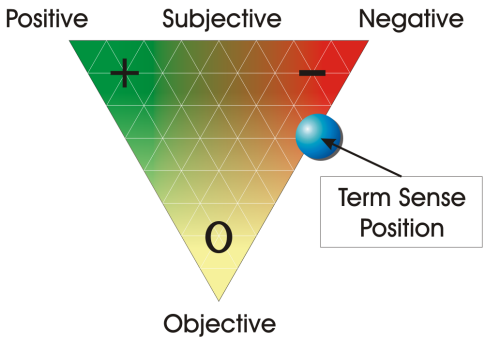


Figure 8: SentiWordNet Synset's properties. Adapted from https://ontotext.fbk.eu/sentiwn.html

The method used to develop SWN is derived from the work of [44, 45], based on the quantitative

analysis of terms associated to synsets and on the use of vector representation of resulting terms for semi-supervised classification of synsets. The scoring trio in the SWN is derived from the results combination produced by a committee of eight ternary classifiers, with similar levels of precision, but with different classification composition. Each subject classifier differs from the training set and the learning adopted for this set, thus producing distinct classifications resulting from each WordNet synset.

The SWN score is given by the ratio of classifiers assigned the corresponding label to the synset. If all ternary classifiers result in assigning the same label to a synset, the labelling will have the highest score for the synset but will have a score proportional to the number of classifiers that assigned it [9, 43].

The SWN data lexicon is also available as a data file (.txt). The Figure 9 show some examples of the records in SentiWordNet 3.0:

```
# POS   ID  PosScore    NegScore    SynsetTerms Gloss
a   00004171    0   0   moribund#2  being on the point of death; breathing your last; "a moribund patient"
a   00004296    0   0   last#5  occurring at the time of death; "his last words"; "the last rites"
a   00004413    0   0   abridged#1  (used of texts) shortened by condensing or rewriting; "an abridged version"
a   00004615    0   0   shortened#4 cut#3   with parts removed; "the drastically cut film"
a   00004723    0   0   half-length#2   abridged to half its original length
a   00004817    0   0   potted#3    (British informal) summarized or abridged; "a potted version of a novel"
a   00004980    0   0   unabridged#1    (used of texts) not shortened; "an unabridged novel"
```

Figure 9: SentiWordNet records

The POS-ID identifies the synset, *PosScore* and *NegScore* values correspond to positivity and negativity pointed out by SWN for the defined synset. The value of objectivity is given by: Obj(s) = 1 - (Pos(s) + Neg(s)). The *SynseTerms* column corresponds to the spaced terms in the synset, with the grammar class and number corresponding to the sense. The Gloss column describes the meaning of the term.

SentiStrength SentiStrength is a lexicon created by [192] aimed to identify sentiment in short texts, containing 2310 sentiment words and word stems obtained from the LIWC, the General Inquirer list of sentiment terms [186] and some ad-hoc additions.

It uses a Kleene Star implementation to identify the lexicon with a wildcard at the end of a word. For instance, “amaz*”matches all words starting with “amaz”, such as amazed and amazing.

For each token or stem in a text, SentiStrength returns a positive and negative score, ranging from 1 to 5 and -1 to -5 respectively. Matching this, each token or stem in the dictionary receives a positive or negative score within one of these two ranges.

Using the tool, the sentence “I really like you but dislike your cold sister ”has as results:

I really love **[3] [+1 booster word]**
you but dislike **[-3]**
your cold **[-2]** sister

So, the final results score 4 - strong positive sentiment, and score -5 - strong negative sentiment. All words in lexicon have theirs scores previously classified, analysed and evaluated, and the final score sentence is a sum of the individual words' scores for each polarity. The words that appear in the text, and are not contained in the lexicon, are not analysed and consequently do not interfere with the final classification.

For a lexicon sentiment weight scores fine-tune, SentiStrength uses a machine learning approach implementing a specific and proprietary algorithm. The reason for this kind of input for the sentiment weights is that the low frequency of many terms in texts would contribute to a wrong machine learning weights assignment.

ANEW *American Norms for English Words*, or ANEW [14], is a set of 1034 words which measures three emotional dimensions: valence, alertness and dominance. The valence consists in how pleasant or unpleasant a stimulus is perceived. The second dimension, alertness, consists of how stimulated or relaxed a stimulus makes us. The third dimension, dominance, consists in how much in control of a stimulus or dominated by it is perceived.

The value for each word in each dimension was obtained from ratings made on a scale of 1 to 9, made by undergraduate students, where a rating of 1 denoted highly negative and 9 denoted highly positive. Table 2 demonstrates an example of how the ANEW values are stored.

Table 2: ANEW values example

Description	Valence	Mean (SD)	Arousal	Mean (SD)	Dominance	Mean (SD)
abduction	2.33	(2.13)	6.09	(2.72)	2.76	(2.49)
abortion	3.15	(2.39)	5.85	(3.09)	4.00	(2.55)
absurd	4.70	(1.30)	4.65	(1.93)	4.70	(1.42)
abundance	6.55	(1.90)	5.10	(2.53)	5.55	(2.11)
abuse	1.46	(1.03)	6.88	(2.83)	3.81	(3.12)
acceptance	8.18	(1.56)	4.86	(3.00)	6.55	(1.99)
accident	2.08	(1.29)	6.48	(2.62)	3.92	(2.29)
ace	6.14	(2.10)	4.95	(2.33)	6.33	(2.08)

EmoLex The EmoLex Word-Emotion Association Lexicon (EmoLex) is a lexicon created by [134] composed of a list of more than 14000 English words and their associations with eight basic emotions (anger, fear, anticipation, trust, surprise, sadness, joy, and disgust) and two sentiments (negative and positive).

The annotations were collected using a crowdsourcing Mechanic Turk, and different than the other lexicons, EmoLex shows association scores for the emotions and sentiments as 0 (not associated) and 1 (associated). An example as EmoLex is composed is presented in Table 3.

Table 3: EmoLex lexicon

Word	Positive	Negative	Anger	Anticipation	Disgust	Fear	Joy	Sadness	Surprise	Trust
aback	0	0	0	0	0	0	0	0	0	0
abacus	0	0	0	0	0	0	0	0	0	1
abandon	0	1	0	0	0	1	0	1	0	0
abandoned	0	1	1	0	0	1	0	1	0	0
abandonment	0	1	1	0	0	1	0	1	1	0
abate	0	0	0	0	0	0	0	0	0	0
abatement	0	0	0	0	0	0	0	0	0	0
abba	1	0	0	0	0	0	0	0	0	0
abbot	0	0	0	0	0	0	0	0	0	1

2.5.1.2 Machine Learning algorithms for sentiment extraction

In Machine Learning literature, there are several algorithms developed to deal for different situations, each one having their pros and cons. Below are presented the most used algorithms to handle with sentiment analysis:

Naive Bayes The Naive Bayes classifier is widely used in classifying texts because of its computational efficiency and good predictive performance.

The Naive Bayes classifier has the "naive"because it assumes that the probabilities are combined independently of each other, that is, the probability that a term in the document is in a specific category is not related to the likelihood of other terms being in this category [181].

Bayesian classifiers use Bayes' Theorem to classify data. If X is the set of characteristics and Y is the class variable, it is possible to determine their relationship using probabilistically:

$$P(Y|X) = \frac{P(X|Y).P(Y)}{P(X)}$$

According to [151], to perform text classification is needed to assign a class c to a document d.

$$P(c|d) = \frac{P(d|c).P(c)}{P(d)}$$

Since *P(d)* does not disturb in the selection of c and to calculate *P(d|c)*, is considered that the probabilities of each characteristic are independent of the class.

Although the hypothesis of independence is necessary for the algorithm formulation, Naive Bayes works appropriately in several applications which such hypothesis cannot be verified, having success in part of the consequence of the versatility. As the hypothesis of independence loses power in a given model, the adjustment of the assumed distribution worsens. However, if both estimated and real distributions agree on the most likely class, the classifier will still perform well [165].

Support Vector Machines The classification task involves a set of training data. Each instance of the training set contains a "target value"that refers to the class or label. The objective of the Support Vector Machines (SVM) is to produce a model, based on the training data, which predicts the target values (class) of the new data [74].

The main idea of SVM-based classifiers is to construct an optimal hyperplane so that it can separate different classes of data with as much margin as possible. The SVM hyperplanes are determined by a relatively small subset of training examples, which are called support vectors. Figure 10 illustrates an optimal hyperplane for a set of linearly separable data.

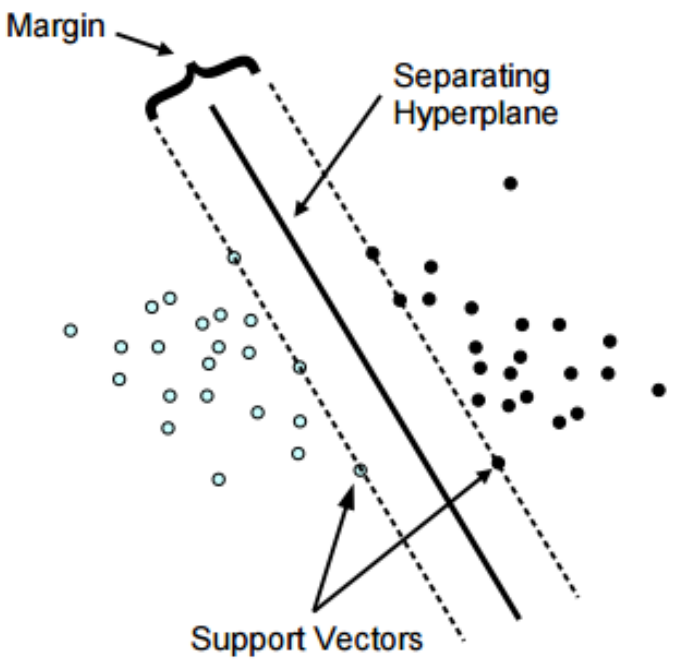


Figure 10: Support Vector and Margin identification. Adapted from https://medium.com/mlearning-ai/support-vector-machine-svm-algorithm-a5acaa48fe3a

Another relevant feature of SVM is the ability to classify data sets with complex scopes, as in Figure 11. For this, mapping the problem to a different space allows a hyperplane to perform class separation (Figure 12). To the transformation be simplified and computationally less costly, the "kernel trick"[73] is performed, with the user having to choose some transformation kernel functions.

Figure 11: Non-linearly separable data. Source: author

The SVM classifier presents an intrinsic limitation: the binary segmentation. It is not possible, through the usual methodology, to classify more than two types of elements into a heterogeneous set. To solve this problem, the solution is successive cascade classifications.

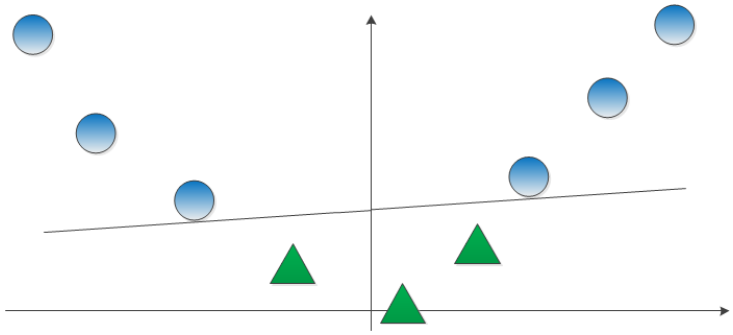


Figure 12: Non-linearly separable data. Source: author

In this approach, known as "SVM multiclass", the elements to be classified can be, at each iteration of the SVM, removed one by one from the original group, as shown in Figure 13. In this one-on-one confrontation process, the SVM lists a suitable match for each iteration, yielding the final one, a result that is qualitatively similar to other multiclass classifiers.

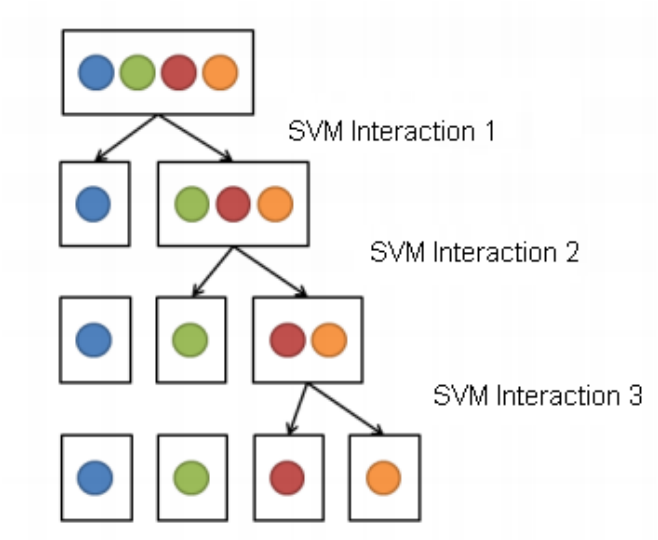


Figure 13: SVM Multiclass classification process. Source: author

Maximum Entropy Model A maximum entropy model classifier is a probabilistic model that favours classes distribution more uniform which adhere to a specific set of constraints determined by training data. [27] introduce an example in text categorization domain. Given a document d containing three classes c_1, c_2, c_3, if no information is known about the document from the training data, it is possible to claim that the probability of classifying d in each of the classes is equal, that is, $1/3$. However if it has "learned"from the training data that $2/3$ of the documents containing the word "sports"belong to class c_1 and the document d contains the word "sports", it is possible to claim that the document d classification probability for class c_1 is $2/3$ and the for classes c_2 and c_3 is equal to $1/3$ each. The sample data provide such characteristics and they will be used to determine the constraints.

kNN The idea of the *K* Nearest Neighbour (kNN) algorithm is finding all training examples that are relatively similar to the test example characteristics. A closer neighbour classifier represents each example as a data point in a two-dimensional space, where *d* is the number of characteristics. Given a test example, its proximity to the rest of the data points in the training set is calculated using a proximity measure. The *k* nearest neighbours of a given example *z* refer to the *k* points closest to *z* [190].

The algorithm calculates the distance between each test sample $z = (x', y')$ and all training examples (x, y) to determine its nearest neighbour list. This calculation can be costly if the number of training examples is large, so indexing techniques are used to reduce the number of calculations needed to find the nearest k for an example [190].

According to [83], *k*NN is a simple method to categorize text and has already been used in several works, such as [204], [68] and [64]. However, the algorithm has some disadvantages, like computational complexity and decreasing performance due to noise training samples. Some researchers look for ways to reduce the complexity of *k*NN, which can usually be divided into three methods: reducing the text vector dimensions [199], decreasing the number of training examples [206] and accelerating the process of finding the closest K neighbours [1].

Artificial Neural Networks Neural networks implement mathematical models that attempt to resemble biological neural structures. So, they can adapt their parameters as a result of interaction with the external environment, gradually improving their performance when solving a given problem [52].

According to [180], "a neural network for text classification is a network of units, where the input units represent the terms, the output units represent the category or categories of interest, and the weights at the connection ends are units representing dependency relationships. To classify a test document, the term weights are loaded into the input units; the activation of these units is propagated forward through the network, and the value of the decision output unit determines the classification. A typical form of the using artificial neural networks is backpropagation, where the training document weights are loaded into the input units, and if a classification error occurs, the error is propagated back in order to change the parameters on the network to eliminate or minimise the error.

There are many efforts using artificial neural networks. Some examples in the categorization of texts are presented by [205], [168] and [171], and in the classification of emotions by [182], [198] and [71].

2.5.2 Detecting Emotions in Speeches

When two persons talk to each other they can easily recognize the emotion applied in the speech used by the other person. So, the objective of an emotion recognition system is to reproduce this human characteristic and identify the emotions contained in the speech.

According to [99], "the importance of emotion recognition from human speech has increased significantly with the need to improve both the naturalness and efficiency of spoken language human-machine interfaces".

For [11], the "identification of emotion can be done by extracting the features or different characteristics from the speech and a training is needed for a large number of speech database to make the system accurate". Yet, according to the author, in order to build an emotion recognition system is mandatory two steps: an emotional speech corpora where emotion specific features are extracted from those speeches and a classification model to recognize the emotions.

[18], claims that "most researchers have used global suprasegmental/prosodic features as their acoustic cues for emotion recognition, in which utterance-level statistics are calculated". Examples of features widely used are: mean, standard deviation, maximum, and minimum of pitch contour and energy in the utterances. Yet according to the author, "the main limitation of those global-level acoustic features is that they cannot describe the dynamic variation along an utterance. To address this, for example, dynamic variation in emotion in speech can be traced in spectral changes at a local segmental level, using short-term spectral features".

According to [41], recognizing emotions in speech is still quite challenging, mainly for two reasons:

- It is unclear which speech features are relevant in distinguishing emotions - obstacles as the acoustic variability caused by the existence of different sentences and styles of speech, the speech rates, the existence of more than one emotion in the same sentence and which emotion corresponds to a different portion of the sentence;
- The way how the speaker express certain emotions - that is usually influenced by the culture and the environment in which he lives.

2.5.3 Detecting Emotions in Images

The beginning of emotion detection in images can be dedicated to Charles Darwin, who demonstrated in 1872 the universality of facial expressions in man and animals. Later, as mentioned earlier in section 2.1, in 1972 Ekman distinguished six primary emotions - also called basic emotions - that are common regardless ethnicities and cultures.

The emotion detection from visual content has become even more prominent in popular social media such as Twitter[2], WhatsApp [3] and Instagram [4] where the textual description is limited due to restrictive devices capabilities. In such cases, extracting information from visual content is critical to understand the emotions, affect, or sentiments conveyed in the image that they arouse in humans.

According to [5], to detect emotion in images, prior is mandatory to follow the steps:

A. Face detection;

B. Face aspect detection;

[2] http://www.twitter.com
[3] http://www.whatsapp.com
[4] http://www.instagram.com

C. Expression classification.

To achieve this objective, the author claims that there are two different approaches: image classification and psychological models.

Image classification consists in the construction of a machine learning classifier trained with thousands of images containing emotions and some other relevant information, such as *Facial Characteristic Points*, as presented by [63]. Psychological models, as *Facial Action Coding System* (FACS) [39], identify the emotions through muscle contractions, resulting in *Action Units* (AU) used to represent levels of contractions of face expressions. The occurrence of an emotion is achieved when a combination of certain (AUs) is detected.

2.6 Summary

This chapter introduced different approaches used to model the emotions and the challenges to identify emotions in different communication channels, such as texts, speeches and images. Despite of all differences between the models, cognitive models do not represent the best approach for this work, because the subjectivity considered in this model is impossible to identify on textual analysis. Moreover, dimensional models can lead to different perspectives when compared to discrete models, because they can incorporate the concepts of basic emotions and also have a new dimension of study - generally the intensity of the emotion - which is an important source of information in this work.

So, in this work I defined to use the dimensional model proposed by Plutchik because it is easier to represent in a textual approach, makes possible to identify new emotions based on the 8 basic emotions - which allows to create new dimensions - and is compatible with EmoLex lexicon.

3

Human Behaviour

It is undeniable that psychology is one of the sciences that most help people, even in things you would never imagine as influencing. From that commercial on television to the decoration of a work office or a hospital, for example, there is a lot of work concerned with the understanding of human behaviour attempting to adapt the place to humans or adapt the behaviour to the place. Human behaviour, according to [85], is considered as "the potential and expressed capacity for physical, mental, and social activity during the phases of human life".

Thus, the psychology of human behaviour is a very interesting area that opens the door to countless answers, but also to many questions.

Some of these questions are:

- Is the human behaviour defined by a set of characteristics?
- Is it possible to predict human behaviour just knowing these characteristics?
- Could these characteristic be identified individually through texts?

Human behaviour has been studied in areas such as computer vision and pattern recognition for years, such as the studied presented by [142], [152] and [122]. These studies resulted in several applications of different areas, UX[1] & UI[2] design, security, etc. These studies have in common the aim to satisfy the user needs. This information is a valuable source that can provide adequate responses to several different problems.

But, what determines human behaviour, and thus, how can it be explained?

According to [60], human behaviour can be influenced by some factors at some levels. These levels can be divided into: individual level, group level (as a neighbourhood) and society level. As presented in Section 2.3, these influences trigger emotions that contribute in the decision-making process.

The identification of a society behaviour can be done by statistics because it is impossible to represent the decision-making process of their all components and how an individual in this society influences the

[1] User Experience
[2] User Interface

society as a whole, remaining for studies only the results of their decisions. For small groups, the studies generally represent them as individuals and apply the science behind the decision-making modelling for individuals.

For [89], the perspectives and factors which can influence human behaviour are:

- **People as information processors** - People process pieces of information about the environment, process with their situation and past experiences to decide the actions to take;
- **Human motivation** - According to [123], the human motivation can be described as a pyramid, where the order of the levels represent the importance of the human needs (from physiological to self-realization);
- **Human behaving rationally** - For [24] in his Rational Choice Theory, the human behaves in a way to maximize their benefits or/and minimize their costs, following logical processes according to the information that they have to analyse;
- **Human behaving emotionally/intuitively/unconsciously** - For discrete emotions theorists as [37] and [157], there is a set of basic emotions. These emotions are individual fingerprints in normal situations. They impact on mood and affect unconsciously human's behaviour (for instance, acting under a depressive behaviour). In a long-term analysis, it allows to identify mood and problems that affect the behaviour, like depression;
- **Human behaving socially** - When considering the social relations, the human behaviour is influenced by two aspects:
  A. Third part objectives - ss presented in the "Theory of Mind" [35], which considers what others have as objective and what they are thinking and feeling;
  B. Group behaviour - Combination with the behaviour of other components of the group, as discussed by [97].

The aim of this work is to identify the emotional state through text messages. On account of that, the discussion about techniques to model the emotional state will be restricted to the "humans behaving emotionally/intuitively/unconsciously" factor at small groups level.

3.1 Identification of human personality

In the dictionary, the word "resilience" means "the capacity to recover quickly from difficulties". This characteristic is one of the most required soft-skills for job-seekers. But, not all are or can be resilient all the time. For a long period, it is impossible to be resilient if it is not contained in the personality. According to [112], "personality is a combination of an individual's behaviour, emotion, motivation, and

thought pattern characteristics". The individual personality affects decisions, health, and preferences, distinguishing the individuals among them. So, be resilient is directly influenced by personality, just like emotions.

In psychology, there are different models to represent the human personality, such as Eysenck's traits, Cattel's traits and Cloninger's temperament and character traits [25]. However, the most widely accepted model of personality is the Five Factor Model (or Big Five) [196]. This model is used for different purposes, and by different companies, such as tests for psychological diagnosis by psychiatrists, even content analysis, such as the IBM Watson Tone Analyzer[3].

The Five Factor Model defines the individual personality in a set of opposite values for five characteristics:

A. Extroversion - extroverted x introverted;

B. Neuroticism - nervous and sensitive x confident and secure;

C. Agreeableness - generous and trustworthy x boastful and unreliable;

D. Conscientiousness - efficient and organized x careless;

E. Openness - inventive and curious x dogmatic and cautious.

It is undeniable that the words chosen for speaking and writing reveal much of the author's personality. [177] claimed that "if the eyes are the window to the soul, then words are the gateway to the mind". For [19], "the more distinct and less frequently used content words tell what the author is saying, the more common particle words tell how the author is speaking". This statistical analysis provides cues about personalities and changing ideals for all kind of people, as discussed by [34]. [184] and [92] argue that texts often bring the emotions that reflect the author's personality.

In NLP domain, there are many efforts to identify personality through texts. [112] created an approach using Deep Learning to identify the Big Five's personalities characteristics. After preprocessing the input texts by cleaning irrelevant data, they represented the texts as a sequence of word vectors pretrained using Word2Vec **??** embeddings. They created five similar models - a 1-dimension Convolutional Neural Network (CNN) for each Big Five's trait - to calculate its existence's percentage in the texts, in order to infer the author personality.

[203] used the personality to improve the quality of video game recommender systems. Based on the idea that people with similar personalities have the same interests, they used the Steam network to collect review from games and their author's. After preprocessing, all reviews were classified using the Personality Recognizer tool [111], which identified the correlation between Big Five's traits and the text analysis provided by LIWC[4]. The user's personality traits were defined as the average of each personality

[3] http://www.ibm.com/watson/developercloud/tone-analyzer.html
[4] http://liwc.wpengine.com/

trait identified in the set of all user's reviews. For the recommendation, the average of each personality trait of all games' reviews was calculated for each game. Finally, the best game recommendations were ranked by a cosine distance from the user's personality traits.

Other approach using NLP to identify personality was presented by [174], where texts were gathered from Facebook posts in Brazilian Portuguese and the texts author's personality was identified based on Big Five's traits. Using preprocessing steps, the texts were represented in vectors, containing the frequency of the words in each Big Five's trait, according to LIWC. These vectors fed a MultiLayerPerceptron classifier to evaluate the results, and these results are compared to a previous performed analysis, in order to evaluate the algorithm.

3.2 Personal lexicon

It is known that for all languages there is a vast set of words - the lexicon - which people can explore and enrich dialogue in a variety of ways. However, when people talk, they use a "personal lexicon", which is exactly the words that within their language, are chosen by the person who is expressing themselves. Within the vocabulary, there are synonyms, which are different words, but which have the same meaning. In some cases, there are dozens of different words, but with the same meaning. The main question is that for some reason, some people prefer to use some words or others, and what drives these people to make that choice decision? Which leads them to choose the word "interesting" instead of "cool", for example. Why when someone who is going to travel to a small city prefers to say that it is "quiet" or "comfortable" even "divine place". It depends on each person and that in some cases, some words remind us of someone, due to the number of times that person uses that word.

It is interesting to note how different personal vocabulary is related to people, as there are preferences in choosing words in dialogue. In a conversation, each word is important, so that all the words spoken are not in vain, have great value. They are pronounced precisely because they are the best, the most pleasant or the best way for a person to express themselves. This is like a fingerprint, where the frequency of using those words says a lot about the personality of the user.

This personal lexicon is important because it helps to differentiate individuals according to the frequency of use of words. However, this is not the only information that can be extracted from a personal lexicon. In general, through the use of words, it is possible to identify a person's origin, gender or even where a conversation occurred. In Portuguese, the word "*obrigada*" (thanks) is used only the author is female. If the author is male, the word must be "*obrigado*" (thanks).

Moreover, according to the context (country, place, company, etc) where a conversation occurs, it is possible to identify differences in the interpretation of the words used, that is valid only in this context, as presented by [120]. This is the regionalism.

Several examples exhibit this situation: the word "*fato*" in Brazilian Portuguese refers to the word "fact" while in Portuguese spoken in Portugal, "*fato*" means "coat". The Spanish word "*tinto*" in Colombia is

about “coffee” where in Spain, the word is regarding to “red wine”.

Due to these differences, in text interpretation in NLP, it is not secure (and certainly impossible) to have an universal lexicon to consult all words, leading to the risk of misunderstanding. To overcome this issue, a useful approach is perform a **lexicon expansion**.

3.3 Summary

In this chapter we presented the concepts of personality and personal lexicon. According to the personality of the person, his vocabulary can vary when comparing to the vocabulary of other person. A calm person certainly use calm word more frequently when compared to an anxious person. For this reason it is important not to generalize the vocabulary for all person, i.e. having a personal vocabulary for each person, representing their basic personality’s bricks.

In this work will be used a personal (expanded) representation of an emotional lexicon, in order to individualize the emotions applied into the sentences.

4

Text Mining and NLP

Text mining, also referred as text data mining or knowledge discovery [46], is an area of computer science research which refers to the process of extracting relevant information from unstructured text documents using techniques from data mining, natural language processing, machine learning, knowledge management and information retrieval.

It can be defined as a process to seek and extract useful information from unstructured data sources such as e-mails, HTML files, etc. through the identification and exploration of interesting patterns [50]. Figure 14 gives the overview of text mining process.

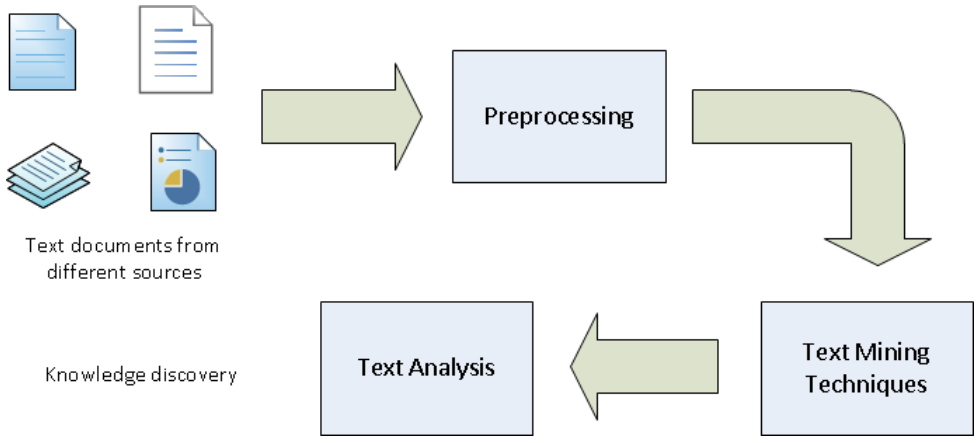


Figure 14: Text Mining overview. Source: author

These data sources are document collections, and the frequently interesting information patterns are found in the unstructured textual data. So, it is a straight deduction that text mining and data mining have similarities. For instance, both systems have features as pre-processing routines, pattern-discovery algorithms, and presentation-layer elements, however, while text mining works on unstructured data, data mining works on data with known and fixed structure.

4.1 Applications of Text Mining

While much of the pre-processing tasks in data mining focus in data cleaning, normalizing and creating extensive numbers of table joins in a structured database, in the context of text mining systems, pre-processing operations focus in pattern identification and extraction of representative features for natural language documents. These pre-processing operations transform the unstructured data contained in the documents into a structured intermediate representation of the original document [95]. So, application of Text Mining can enrich some research areas, such as *Information Retrieval, Information Extraction, Categorization and Natural Language Processing*.

4.1.1 Information Retrieval

According to [114] information retrieval (IR) "is finding material (usually documents) of an unstructured nature (usually text) that satisfies an information need from within large collections (usually stored on computers)."

In other words, an IR model selects or ranks the set of documents regarding to an user query. Both result text in documents and queries can be formalized by a function that returns a Retrieval Status Value (RSV) for each document of the collection. Most IR systems represent document contents by a set of descriptors, called terms, belonging to a vocabulary V.

According to [106], main IR models define the query-document matching a function that weights the query terms occurring in a document according to four main approaches:

- The estimation of the probability of user's relevance rel for each document d and query q concerning a set R_q of training documents $Prob(rel|d, q, R_q)$;
- The computation of a similarity function between queries and documents in a vector space $SIM(d, q)$;
- The estimation of the probability of retrieving the document d given a query q, $p(d|q)$;
- The information carried by the query terms in the document, that is the number of bits necessary to code X_i occurrences of the query terms $t_i \in$ q in the document: $-log_2 Prob(d|X_1, ..., X_q)$.

An information retrieval system is not a database system. Despite of having close functionalities, mainly the ability to look for answers to the user given queries searching in large information stores, there are significant differences between them, such as:

- Database systems

  Concurrency control;

  Recovery;

  Transaction management;

  Update;

- Information retrieval

  Unstructured documents;

  Search based on keywords;

  Concept of relevance.

Due to the large amount of information available in text sources, there are many applications for information retrieval, such as on-line library catalogue systems, online document management systems, and, as mentioned by [65], Web search engines as Google Search Engine.

4.1.2 Information Extraction

Extraction process is responsible for identifying existent keywords and relationships in a text. According to [82], "the general goal of information extraction is to discover structured information from unstructured or semi-structured text."

It is done through a process called **pattern matching**, which searches for predefined sequences in the text. The software infers the relationships between all the identified data to give a meaningful information.

For example, given the sentence:

In 1993, Palhinha, Toninho Cerezo and Muller scored the goals of São Paulo in the interclub world cup against Milan

we can extract the following information,

PlayerOf(Palhinha, São Paulo)
PlayerOf(Toninho Cerezo, São Paulo)
PlayerOf(Muller, São Paulo)

Such information can be presented to an end user, or can be used as source for other systems such as search engines and database management systems to provide better services to end users.

This process is depicted in Figure 15.

4.1.3 Categorization

Categorization involves identifying the main themes of a document and organizing them into a fixed number of predefined categories. Using machine learning techniques, the system "learns"how to classify from examples, performing the category assignment automatically. This is a supervised learning problem.

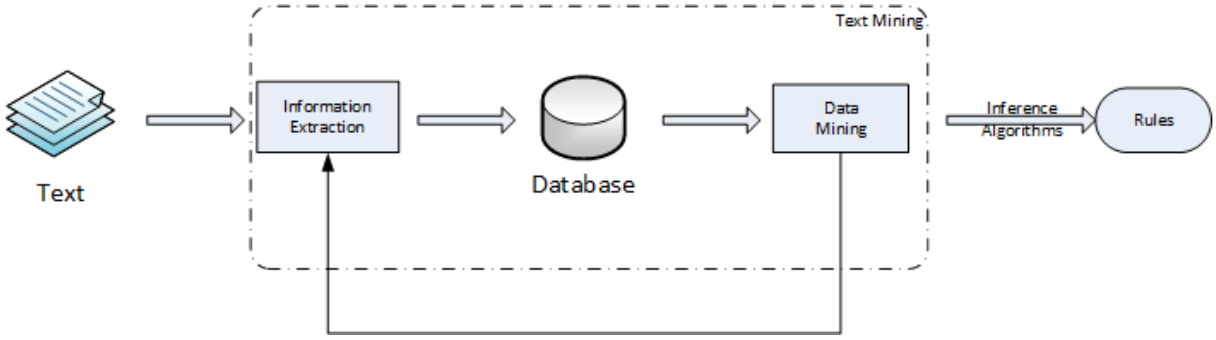


Figure 15: Process of Information Extraction from Texts. Source: author

During the categorization process, a computer program will see the document as a "stemmed bag of words", i.e., a set of individual words where each word is "stemmed"(as will be described at subsection 4.2.2), suitable for the learning algorithm and the classification task. This process of changing the document is part of preprocessing techniques, that is described at section 4.2.

Different from information extraction, according to [199] "categorization only counts words that appear and then identifies the main topics that the document covers. Categorization often relies on a glossary for which topics are predefined, and relationships are identified by looking for large terms, narrower terms, synonyms, and related terms."

A good example of classification can be seen in Gmail[1] or other mailboxes, where the messages are automatically classified in some different categories such as spam, social and promotions.

4.1.4 Natural Language Processing

Natural Language Processing (NLP) consists of the development of computational models to perform tasks that rely on informations expressed in some natural language (for example, translation and interpretation of text and human-machine interface) [170].

According to [26], the research in NLP is focused, essentially, on three aspects of communication in natural language:

- Sound: phonology;
- Structure: morphology and syntax;
- Meaning: semantic and pragmatic.

Phonology is related to the recognition of sounds that compound the words of a language. Morphology recognizes words regarding primitive units that comprise it (for example "*watched*"→ watch + ed). The syntax defines the structure of a sentence, based on how the words are related in the sentence, as

[1] http://www.gmail.com

illustrated in Figure 16. The semantics associates are a meaning with a syntactic structure, regarding the meanings of the words composing it. Finally, pragmatics show how context contributes to meaning (e.g. in the context blockbuster, "*classic*"→ refers to old film which won prizes).

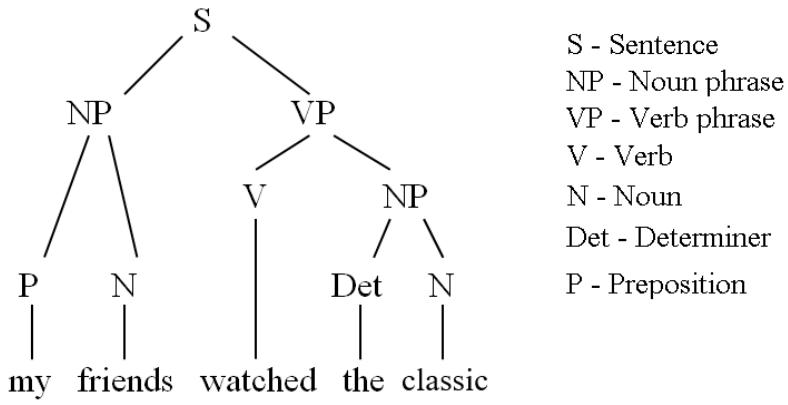


Figure 16: Syntactic Tree

4.2 Techniques of Text Preprocessing

In order to have assertive results in text mining operations, it is necessary to extract the relevant information and avoid the irrelevant information. For this reason, text mining uses various preprocessing techniques to identify and extract structured representations from raw unstructured data sources.

According to [50] "preprocessing tasks generally convert the information from each original data source into a canonical format before applying various types of feature extraction methods against these documents to create a new collection of documents fully represented by concepts."

Despite classifying text mining preprocessing technique, the use of most algorithms in text mining preprocessing activities is not restricted to particular tasks, and several quite different algorithms can solve a major part of the problem. Often, the same algorithm is used alone or in combination with other algorithms for different tasks, constituting different techniques. For instance, Hidden Markov models (HMMs) can be used for part-of-speech (POS) tagging and named-entity recognition (NER).

In general, each preprocessing technique receives an unstructured document as input and proceeds to add information to the structure by analysis the present features. When finishing, the most important information about meaning-representing features are used for the text mining, whereas the rest is discarded. The major difference between preprocessing techniques is related to the nature of the input representation and the output features.

While NLP-based techniques use and produce domain-independent linguistic features, text categorization and information extraction (IE) techniques deal with the domain-specific knowledge. One still unsolved problem is to combine the processes of different techniques instead of combining the results. For instance, POS ambiguities generally can be solved by searching at the syntactic roles of the words. In the same way,

structural ambiguities can be resolved through domain-specific information. Furthermore, a large part of the text in any document does not contain relevant information but must be processed before being considered useless and discarded in the final stage. It turns the process extremely inefficient. Thus, the processes must be executed in parallel, exchanging information with each other. However, because the algorithms were created for different tasks, it is very difficult to redesign them to run simultaneously. There are some attempts to find an algorithm, or a set of algorithms to perform most of the preprocessing task in a single large step.

Preprocessing is a very important step in text mining processes and applications. It is the first step not only for text mining approaches but also in data mining. There are several preprocessing techniques used to extract information from text, and their usage is according to the characteristics of the information desired. Despite some techniques were created in data mining, they are used in text mining approaches, since the same technique can be used for both information extraction, information retrieval, or combined. Some most used techniques are described below.

4.2.1 Tokenization

Tokenization is considered one of the grand challenges of NLP and is conventionally interpreted as breaking up "natural language text [...] into distinct meaningful units (or tokens)"[87].

In order to occur more sophisticated processing, the text stream must be split into tinier meaningful constituents. Documents can be split into chapters, sections, paragraphs, sentences, words, and even syllables or phonemes, according to the needs.

The most used approach in text mining systems consists in breaking the text into sentences and words, which is called **tokenization**. Possibly, the most difficult task in identifying sentence in the English language is distinguishing the difference between a period that signals the end of a sentence and a period that is part of a previous token like Mr., Dr., etc. and others.

It is common for the tokenizer also to extract token features. These are usually simple categorical functions of the tokens describing some superficial property of the sequence of characters that make up the token. Among these features are types of capitalization, the inclusion of digits, punctuation, special characters, and so on.

However, when tokenizing a text, it is important to be aware of some problems as:

- Different languages - Languages that do not mark word boundaries (like white spaces) to differentiate the tokens, as Chinese and German, present a challenge because an original word would be broken in 2 or more tokens;

- Punctuation - Unless text provide some punctuation, it is hard to find the end-of-sentence. Interpret "apples, bananas, grapes"must be the same as "apples bananas grapes";

- Hyphens - Words that contain hyphens must be considered as a single word, as middle-office while words connected by grammatically required hyphens, such as computer-based, should be considered multiple tokens;

- Apostrophes - Different than hyphens, apostrophes must be considered as 2 different words (*"we're here"* $\Longrightarrow$ *"we are" "here"*), however, for possessives, the rule is considered as a single word. It would be done carefully to avoid understand mistakes, as *"it's"* and *"its"*;

- Tokens that are not words - Sometimes is needed to interpret other text different than words. For instance, license plate or ZIP codes. In these cases, it will be necessary to customize to handle the situations.

4.2.2 Stemming

A stemming algorithm is responsible for obtaining the stem of a word, which is, its morphological root, through clearing the parts of the word that bring grammatical or lexical information. In both cases, these parts do not change the concept which word is related to as the semantic informality has been proven in the literature, especially in languages that are highly inflective [158] and in short documents [93], regarding recall and precision.

The purposes of a stemming algorithm can be classified into 3 different approaches:

A. Clustering words according to their topic - Words which are derivations from the same stem belong to the same concept from the stem (e.g., drive, driven, driver). These derivations are generated through appended affixes (prefixes, infixes, and/or suffixes) but, in general, and more specifically in English, only suffixes are considered, as generally prefixes and infixes modify the meaning of the word, and stripping them would lead to errors of bad topic determination [77];

B. The possibility of IR process improvement - Expanding the query to obtain more precise results allows it to be refined by replacing the contained terms to their related topics present in the collection, or adding these topics to the original query. This process can be done automatically and transparently to users or the system can propose one or more improved formulations of the query to users letting them decide if any of them is more specific and defines better their information needs. According [200], "even if interactive query expansion is better in principle because the user has more feedback on what is happening, typically it cannot be done directly with the result of stemming, as stems usually are not understandable by humans";

C. Reduction of the dictionary - Once the whole vocabulary contained in the original unprocessed collection of documents can be reduced to a set of topics or stems, it would require less space to store the structures used by an information retrieval system and consequently also would light the computational load of the system.

In the literature, it is possible to find several different implementations of stemming algorithms. The most known stemming algorithms used are: Lovins stemmer [109], Dawson stemmer [31], Porter stemmer [159], Paice/Husk stemmer [147].

4.2.2.1 Lovins stemmer

The Lovins stemmer algorithm is composed of two steps: suffix removal and treatment of the remaining stem. The suffix identification is made by matching the word's termination with the longest suffix from a list of 294 suffixes. After matching the suffix, it is applied one of the 35 associated application rules, and the remaining stem is treated to solve some linguistic exceptions (like ending in double d or double t).

For example, the word nationally finishes with **ationally** and is associated with condition B, which relates to "minimum stem length = 3."If removing **ationally** would leave a stem of length 1 and consequently would be rejected. But it also has ending **ionally** with associated condition A. Condition A is "no restriction on stem length", so **ionally** is removed, leaving "nat."

4.2.2.2 Dawson stemmer

Dawson stemmer algorithm is an extension of the Lovins algorithm which has a wider list of suffixes [185]. In addition to Lovins stemmer, Dawson stemmer also has a single pass stemmer which makes it performs fast. The suffixes are structured and stored by their lengths and last letters. In fact, they are structured as a set of separated character trees for quick access. So, the benefit of Dawson stemmer is it covers more suffixes compared to Lovins stemmer and Dawson stemmer. Besides that, it also performs fast. However, the weaknesses of Dawson stemmer include its complexity and lacks standard reusable implementation.

4.2.2.3 Porter stemmer

Porter stemming algorithm probably is the most known stemming methods. Simple, it has 60 suffixes, two transformation rules, and context-sensitive rule to determine whether a suffix should be removed or not.

The algorithm is composed of 5 steps, each one containing specific rules for removing suffixes or transforming the words: the first is responsible for handling inflectional suffixes; the second, third and fourth are responsible for handling derivational suffixes while the fifth is responsible for recoding.

In a formal model, their rules are simple can be seen like this:

$$NS = f(C, S) \tag{4.1}$$

where:

NS = new suffix
C = Condition

S = suffix

For example, a rule as “if the word has, at least, one vowel and consonant and ends in EED, transform the ending to EE”. So “guaranteed”will be “guarantee”while “bleed”stays unchanged.

According to [146], Porter stemmer has a lower error percentage than the Lovins stemmer. On the other hand, according to [183] “Lovins stemmer is a noticeably bigger stemmer that produces better data reduction results. It comprises of 294 extensive endings list (list of suffixes), which give it an advantage over other algorithms regarding speed. Lovins algorithm has large suffix set and works with only two major steps to strip a suffix, compared with the five steps of the Porter algorithm.”

4.2.2.4 Paice/Husk stemmer

The Paice/Husk stemmer is an algorithm which contains a table with 120 rules indexed by the last letter of a suffix. At each iteration, it tries to find an appropriate rule according to the last character of the word. Each rule identifies either removal or replacement of an ending. If no rule is found, the process is dismissed. It also terminates if a word starts with a vowel and if there are only two letters left on the word or if a word starts with a consonant and there are only three letters left. Otherwise, the rule is used and the process repeats.

The advantages of Paice/Husk stemmer are it is in the simple form and every iteration can treat of both removal and replacement as each of the rules applied. However, the disadvantage of Paice/Husk stemmer is it has a very heavy algorithm. As a result, the over stemming process may occur.

4.2.3 Lemmatization

As the stemming process, the lemmatization is responsible to convert the different variations of a word to a normalized form (known as lemma). However, differently than the stemming, the lemmatization does not “truncate” the word to produce a stem, but converts the suffix of the word to a normalized infinitive form.

For [22], “Lemmatization for languages with rich inflectional morphology is one of the basic, indispensable steps in a language processing pipeline”. Its use decreases the number of words to be processed because it converts all forms/conjugations of a word into a single word. Also, it increases the chance of identifying words in lexicons, since this type of resource generally stores words in a normalized form.

Because lemmatization produces a more sophisticated response when compared to stemming, it requires more information to work. In order to identify the lemma of a word, it needs to know the sentence’s part of speech which contains the word. Besides, to identify the variances of a word, it is necessary lots of labelled texts to input a Machine Learning model trained to identify the relations of words, and parts of speech to predict better resolutions - like resolving “is” and “are” are variations to “be”).

There are different libraries that provide lemmatization for different languages, such as NLTK[2], SpaCy[3] and Gensin[4] for different programming languages, as Python, Java, R and others.

4.2.4 Part of Speech

Part of Speech (POS) tagging is a process of labelling textual elements - typically words and punctuation - to evidence the grammatical structure of a particular piece of text. In speech recognition and synthesis, it is useful for extracting terms, disambiguation, composition of new sentences and lexicographic research. It reads the text and assigns parts of speech to each word (and another token), such as noun, verb, adjective, etc., although computational applications use more fine-grained POS tags like "noun-plural."

For instance, applying Stanford CoreNLP POS Tagger [113] in the sentence "Four little monkeys jump on the bed", the POS annotations would be as:

Four/CD little/JJ monkeys/NNS jump/VBP on/IN the/DT bed/NN

where:

CD = cardinal
JJ = adjective
NNS = noun plural
VBP = verb in the third person of singular present
IN = preposition or subordinating conjunction
DT = determiner
NN = noun

POS tagging can be performed using different approaches as: rule-based models and stochastic models.

4.2.4.1 Rule-Based Models

Rule-based approaches use contextual information and morphological information to assign tags. Their process basically follows rules, like: "if an ambiguous/unknown word X is preceded by a determiner and followed by a noun, tag it as an adjective"or "if an ambiguous/unknown word ends in an -ing and is preceded by a verb, label it a verb."

Rule-based taggers most commonly require supervised training; but, very recently there has been a great deal of interest in the automatic induction of rules. One approach to automatic rule induction is to

[2] https://www.nltk.org/
[3] https://spacy.io/
[4] https://radimrehurek.com/gensim/

run an untagged text through a tagger and see how it performs. A human then goes through the output of this first phase and corrects any erroneously tagged words the properly tagged text is then submitted to the tagger, which learns correction rules by comparing the two sets of data. Several iterations of this process are sometimes necessary.

The most know rule-based model was proposed by [16].

4.2.4.2 Stochastic Models

According to [153], "like transformation-based tagging, statistical (or stochastic) part-of-speech tagging assumes that each word is known and has a finite set of possible tags. These tags can be drawn from a dictionary or a morphological analysis. When a word has more than one possible tag, statistical methods enable us to determine the optimal sequence of part-of-speech tags $T = \{t_1, t_2, t_3, ..., t_n\}$, given a sequence of words $W = \{w_1, w_2, w_3, ..., w_n\}$."

There are several approaches to algorithms implementing stochastic models for POS tagging. The most used are: Hidden Markov Model [162] and Viterbi Algorithm [54].

4.2.5 Word Sense Disambiguation

The objective of Word Sense Disambiguation (WSD) according to [79] is to distinguish the meaning of single words or phrases in a particular context. An example is the word "bass"which may have the senses "the lowest adult male singing voice"or the "the member with the lowest range of a family of musical instruments."

To achieve this objective, currently, there are two main methodological approaches in this area: knowledge-based and corpus-based methods.

Knowledge-based methods use external knowledge resources, like dictionaries and thesauri, which define explicit sense distinctions for assigning the correct sense of a word in context and words associated with a given sense.

Introduced by [100], these approaches have influenced many researchers to use machine-readable dictionaries (MRDs) as a structured source of lexical knowledge to deal with WSD. These approaches use the knowledge contained in the dictionaries to avoid the need for large amounts of training material. [3] distinguish ten different types of information that can be useful for WSD, those in a major part can be located in MRDs, and include part of speech, semantic word associations, syntactic cues, selectional preferences, and frequency of senses, among others.

In general, WSD techniques using pre-existing structured lexical knowledge resources differ in:

- The lexical resource used (monolingual and/or bilingual MRDs, thesauri, lexical knowledge base, etc.);
- The information contained in this resource, exploited by the method;

- The property used to relate words and senses.

Other approaches tend to measure the accordance between words, referenced by a structured semantic net. Thus, [189] uses the notion of conceptual distance between network nodes to improve precision during document indexing. [2] applied the notion of conceptual density for the resolution of the lexical ambiguity of nouns using the WordNet noun taxonomy. [7] proposed an approach combining a set of knowledge-based algorithms to disambiguate definitions of MRDs accurately.

Although knowledge-based approaches have been proven to be easy to use because they do not require sense-annotated data, for [135], in general, supervised, corpus-based methods have obtained a better precision than knowledge-based ones.

Corpus-based methods uses semantically annotated corpora to train machine learning (ML) algorithms to choose which word sense fits better in which context. The words contained in these corpora are tagged manually using semantic classes taken from a particular lexical semantic resource (usually WordNet[5]). Many standard ML techniques have been tried. [56] and [98] have presented works applying Bayesian learning; [139] did the first work on *k*NN for WSD; Decision Trees (DT) are used by [136]. However [173] in a comparative experiment with many ML algorithms for WSD concluded that decision trees are not among the top performing method.

4.2.6 TF-IDF

Term Frequency - Inverse Document Frequency (TF-IDF) is a metric commonly used to express the importance of the word within a collection of documents (corpus). As higher as the number of times a word appears in the document, it is more important in this document. But if this same word appears many times in various documents of the collection, it becomes less important within the whole collection. Thus, importance is attached to those words that appear several times in the document and that can be considered keywords (such as “recommendation”in an article on recommendation systems) and penalizes the words that appear in several documents of the set (such as “computing”in a set of computing articles where the recommendation systems article is inserted).

According to [172], “though TF-IDF is a relatively old weighing scheme, it is simple and effective, making it a popular starting point for other, more recent algorithms.”

The formal procedure for implementing TF-IDF works at a given document collection D, a word w, and an individual document d ∈ D, calculates

$$w_d = f_{w,d} * log(\frac{|D|}{f_{w,D}}) \tag{4.2}$$

where:

$f_{w,d}$ = the number of times w appears in d

$|D|$ = size of the corpus

[5] https://wordnet.princeton.edu/

$f_{w,D}$ = the number of documents in which w appears in D

Regarding of its statistical characteristics, TF-IDF can be used in different kind of applications, as summarization proposed by [70] and keyword extraction developed by [207].

4.2.7 Stopwords

In Information Retrieval (IR), words that do not transmit any information are known as stopwords. Examples of stop words in English are articles, prepositions, and pro-nouns, etc. that does not give the meaning of the documents. According to [17], the top 10 most frequently occurring words in English typically account for 20% to 30% of the tokens in a document.

Due to does not contain information, stop words are not measured as keywords in text mining applications [159].

To detect stopwords in texts, the effort is restricted to compare a previously compiled stopwords' list against all tokens existent into the text. Later, the found stopwords are removed and a new version of the text is ready to be processed.

Despite good results on removing texts, stopwords detection may become difficult over the time because writing styles change during the time. Moreover, the existence of information in a word depends on the context to which it is referring.

To solve these problems, there are efforts like presented by [107] to detect stopwords automatically in a text.

4.2.8 N-Grams

According to [116], "a N-Gram is a sequence of N words: a 2-gram (or **bigram**) is a two-word sequence of words like 'please turn', 'turn your', 'your homework', and a 3-gram (or **trigram**) is a three-word sequence of words like 'please turn your', or 'turn your homework'."

N-Grams are considered one of most important tools in natural language processing, being used for estimating probabilities of next words or of whole sequences of words.

One way to estimate these probabilities is from relative frequency counts. For example: consider a very large corpus, count the number of times that a specific phrase (e.g. *the book is on the*), and count the frequency of this phrase is followed by *table*.

So, it is possible to calculate the probability of *table* be the next word in the phrase, according the formula:

$$P(table|the\ book\ is\ on\ the) = \frac{Freq(the\ book\ is\ on\ the\ table)}{Freq(the\ book\ is\ on\ the)} \tag{4.3}$$

So, in a scenario where exists a very large corpus with text, such as web, it is possible to compute the frequencies and calculate the probabilities of any next word, sorting descending from the most suggested to less suggested.

Another approach using N-Gram is in Stopwords removal supporting. This support consists of identifying the most frequent unigrams, bigrams, trigrams, etc. It is common that as more frequent are the unigrams, bigrams, trigrams, more irrelevant they are. For example, the most frequent unigrams in English are: "*the*", "*of*", "*and*"and "*to*", where the bigrams "*of the*", "*and the*", "*this is*"and "*in a*"are equally frequent and irrelevant (regarding to information).

4.3 Summary

In this chapter, I presented different techniques used in Text Mining to extract information from unstructured text. Moreover, these techniques are a toolset for leverage different texts in a standard format which allows to identify emotional characteristics.

Despite of all presented techniques are useful for a specific task, it was defined a set of approaches here described to be used during a pipeline in this project, such as tokenization, lemmatization, part of speech and stopwords removal.

5

Word Representation

The representation of words in natural language processing (NLP) can be considered as one of the basic building blocks because makes possible the understanding of a human language by a machine. According to [197], "a word representation is a mathematical object associated with each word, often a vector. Each dimension's value corresponds to a feature and might even have a semantic or grammatical interpretation, so we call it a word feature".

5.1 Word Representation Approaches

In the last years, the approaches for word representation gained an increased importance in the NLP processes. From simple techniques of counting word frequencies to identifying the context of words, each approach has pros and cons in their utilization.

5.1.1 Dictionary Lookup

Dictionary Lookup is the simplest approach of word representation, consisting in a word ID lookup in a dictionary. Due to their simplicity, it is difficult to find a sophisticated definition about dictionary lookup, being essentially a set of key/value pairs, where the key is an unique word and the value is an unique ID which represents the word.

The words contained in the set of keys are collected from a corpus - which can be collection of words, sentences or texts - where the words are preprocessed (using some techniques presented in Section 4.2), hereafter called "*vocabulary*".

Then, for each given word, the dictionary returns the corresponding word's ID by looking it up in the dictionary. If the word is not present in the dictionary, the dictionary should return an "Out of Vocabulary"code. An example of this kind of dictionary is presented in Table 4.

Despite being easier and simple to implement, this approach has some important drawbacks which must be taken into consideration. By considering IDs as integer numbers, the model might assume the existence of relations just considering the order of this IDs. For example, if the dictionary contains entries

Table 4: Dictionary Lookup example

Word	ID
planet	1
water	2
motorcycle	3
goal	4
objective	5
Portugal	6

such as 1: “player”and 2: “referee”, the higher ID number might be incorrectly considered as “more important”by the Machine Learning models than the least ID numbers. On the other hand, using the IDs to represent information, such as size measures 1: “small”, 2: “medium”, 3: “large”is suitable for this case because there is a natural ordering in the data.

5.1.2 One Hot Encoding / Bag of Words

One hot encoding / Bag of words is a well-known process in NLP that maps the occurrence of all words existent in a corpus into a vector. Thus, it is possible to represent any sentence as a vector of integers.

This approach considers that a text can be represented as a list of sentences, and on the other hand, a sentence can be represented as a list of words (or tokens, as presented in Subsection 4.2.1).

So, the number of distinct words existent in the text, denotes the word vector size, where each position of the array represents a word at the same index in the set of distinct words (where this set is generally alphabetically sorted).

The resulting sentence’s representation is an array of integers containing at each position the frequency of the word in the sentence. This approach is known as **Bag of Words** model, because it loses the order of how the words appear in the sentence.

An example of the One Hot Encoding / Bag of Words is presented in Figure 17.

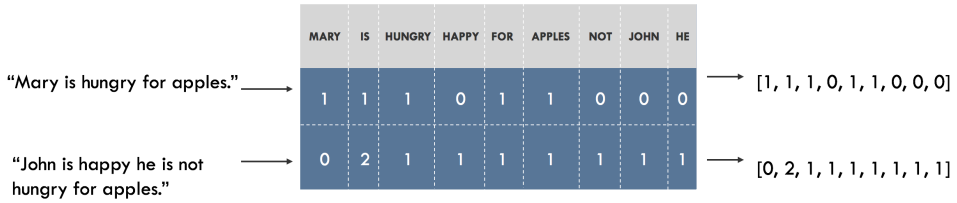


Figure 17: One Hot Encoding / Bag of Words Representation

A variation in One Hot Encoding / Bag of Words creation is identifying the relevance of each word in the sentence considering the existent corpus instead of their frequency. This can be achieved using different solutions, however, the most known is the *Term Frequency - Inverse Document Frequency* (TF-IDF), as presented in Section 4.2.6.

A drawback of this approach is that the representation of several sentences will produce immense and sparse vectors. When considering large texts, where the number of different words is higher, the representation of each word or sentence in vectors will require large memory for computation, because each vector has the same size as the number of distinct words. Moreover, once that each sentence has a tiny subset of all distinct words, the vectors will have in their majority zero values as dimensions.

Other weakness of this approach is due to the positional reference of the words. As the model does not take into consideration the relationship of precedence among the words, different sentences using the same words will be represented as the same vector. For example, the sentences "John loves chocolate and Anna hates to study"and "Anna loves chocolate and John hates to study"will be represented as the same vector.

5.1.3 Word Embeddings

The main problem on the representations based on vocabulary - as *dictionary lookup* and *one hot encoding* - is that all fail in not to capture the relational structure of the lexicon. The reason for this problem is that the approaches do not consider as relevant the word's position in the text. So, determining the context in which the words were presented is impossible. However, different from the approaches presented previously, the **Word Vectors** take into consideration the proximity among words in texts and similarity when representing them.

The characteristic of the Word Embeddings is to represent words as features in vectors, where each entry stands for one hidden feature inside the word meaning, allowing to reveal semantic or syntactic dependencies.

An initial approach to describe these vectors is creating a co-occurrence matrix, i.e., a matrix containing the number of counts of each token (word) and a flag indicating if the token is the next neighbour in the sentence.

For example, considering the sentence "*I love chocolates, but I hate sports*", a co-occurrence matrix of each word can be described as presented in Table 5.

Table 5: Co-occurrence matrix

Token	**I**	**love**	**chocolates**	**but**	**hate**	**sports**
I	0	1	0	1	1	0
love	1	0	1	0	0	0
chocolates	0	1	0	1	0	0
but	1	0	1	0	0	0
hate	1	0	0	0	0	1
sports	0	0	0	0	1	0

So, the word vectors for the sentence above would be:

I = [0, 1, 0, 1, 1, 0]
love = [1, 0, 1, 0, 0, 0]
chocolates = [0, 1, 0, 1, 0, 0]
but = [1, 0, 1, 0, 0, 0]
hate = [1, 0, 0, 0, 0, 1]
sports = [0, 0, 0, 0, 1, 0]

These results can provide some useful insights. For instance, the words "*love*" and "*hate*" are neighbours to the nouns "*chocolate*" and "*sports*" as well as is neighbour of "*I*". These proximities and sequences indicate that these words must be verbs.

When using word vectors to expand the datasets, some calculations can provide information about the words used in similar contexts - as presented in Subsection 5.2 - giving us insights to their semantic and syntactic relations.

According to [127], "we find that the learned word representations in fact capture meaningful syntactic and semantic regularities in a very simple way. Specifically, the regularities are observed as constant vector offsets between pairs of words sharing a particular relationship. For example, if we denote the vector for word i as xi, and focus on the singular/plural relation, we observe that xapple — xapples ≈ xcar — xcars, xfamily — xfamilies ≈ xcar — xcars, and so on. Perhaps more surprisingly, we find that this is also the case for a variety of semantic relations".

Yet, as the dimensionality of words increases according to the size of the corpus, proportionally, the matrices will grow in size - remaining sparse and compromising storage efficiency. This problem has been focus of research about optimizing the representation of word vectors. The most known approaches are Word2Vec and GloVe.

5.1.3.1 Word2Vec

Word2Vec is an open-source tool, developed by [128], which is used to calculate representations of words as vectors. According to [102], Word2Vec "is a popular choice for pre-training the projection matrix $W \in \mathbf{R}^{dx|V|}$ where d is the embedding dimension with the vocabulary V . As an unsupervised task that is trained on raw text, it builds word embeddings by maximizing the likelihood that words are predicted from their context or vice versa. As an unsupervised task that is trained on raw text, it builds word embeddings by maximizing the likelihood that words are predicted from their context or vice versa."

Word2Vec works by taking a corpus as input and returning word vectors as output. A vocabulary is constructed during the process of learning representations as vectors. The tool has defined in its structure two models to represent the vectors: Skip-Gram model and Continuous Bag of Word (CBOW) model, as presented in Figure 18.

Both models use a hidden layer neural network to generate output, and the back propagation algorithm to update, or just weigh, the parameter values while observe new examples.

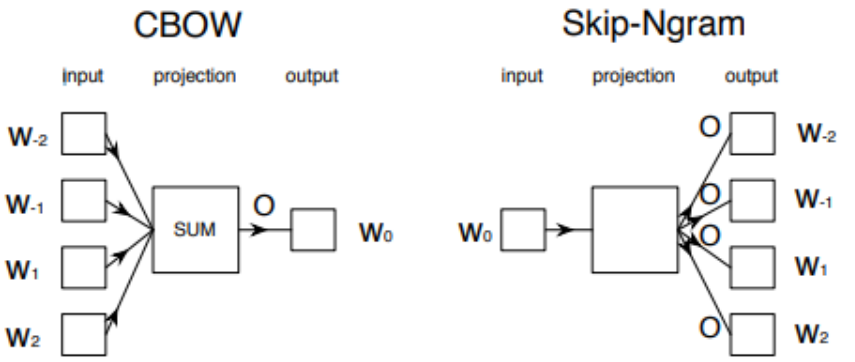


Figure 18: Illustration of the Continuous Bag-of-Word (CBOW) and Skip-Gram models. Source: [102]

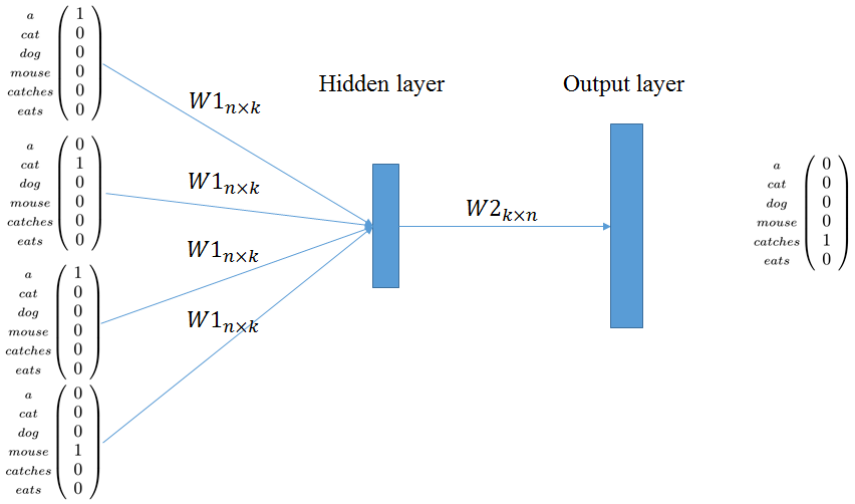


Figure 19: CBOW network. Source [53]

In the CBOW model the input is a context (surrounding words) and the output an omitted word, i.e., for this representation, the surrounding words are combined to predict the middle word. It is considered as context the words that are around the target word w. An example is to use as context the two words before w and the two words after. In the case of a sentence $w_1\ w_2\ w_3\ w_4\ w_5$ the context of the word w_3 is $w_1\ w_2\ w_4\ w_5$.

Different than the the Bag of Words approach, the CBOW can consider different contexts for a single target word. For example, in the sentences: “Ronaldo scored a goal”and “Neymar scored a goal”, both “Ronaldo”and “Neymar”are contexts for the word “scored”. The processing for different contexts is possible by processing N times the vector word, where N is the number of different contexts the word appears in the corpora. Each N word existent into the corpora will produce a vector $W1$, in the context k. These vectors will feed the hidden layer that will produce a $W2_{kxn}$ matrix to the output layer. In the output layer, a softmax function will calculate the probabilities for a resultant word.

An example of CBOW processing is presented in Fig. 19.

In the Skip-Gram model, the objective is the inverse of CBOW: given a vector representing a single word,

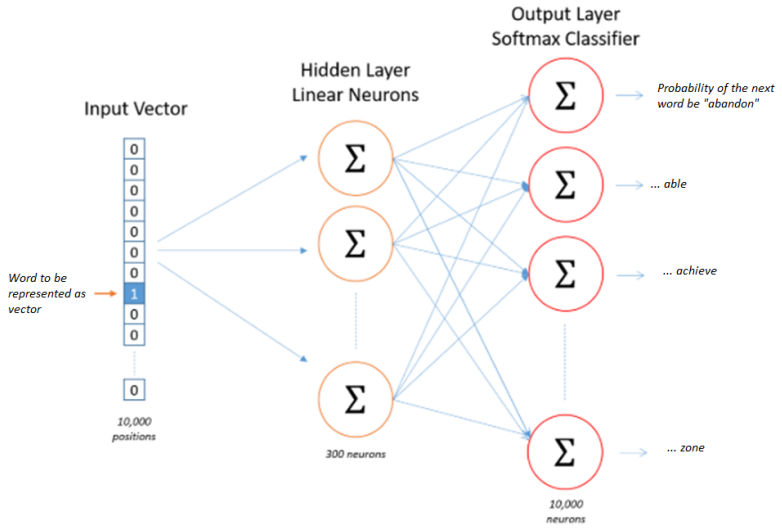


Figure 20: Skip-gram network. Source https://towardsdatascience.com/word2vec-skip-gram-model-part-1-intuition-78614e4d6e0b

it can predict its context. In this case, the input is a word that goes through a hidden layer and outputs the most likely context. The objective, in this case, is to identify the weights for each word calculated in the hidden layer given a word. Once the hidden layer uses a Softmax function as activation function, these weights will represent the probability of the words be the neighbours of a given word. A representation of the Skip-Gram model is illustrated in Fig. 20.

5.1.3.2 GloVe

According to [154], the Global Vectors for Word Representation (GloVe) "is an unsupervised learning algorithm for obtaining vector representations for words. Training is performed on aggregated global word-word co-occurrence statistics from a corpus, and the resulting representations showcase interesting linear substructures of the word vector space."

Despite both Word2Vec and GloVe generate word vectors based on word neighbourhood, their difference is that GloVe uses the word co-occurrence global yet to generate the word vectors, not relying only on local statistics (local context words) as Word2Vec.

The model is based on the idea that the proportions of probabilities of a word co-occurrence matrix can bring some form of meaning that can be translated as a vectors difference. Therefore, the objective is to identify the relation among the words and represent them as word vectors so that their scaled product is equal to the logarithm of the probability of the words' co-occurrence. As the logarithm of a ratio is equal to the difference in logarithms, this objective associates the proportions of co-occurrence probabilities with vector differences in the vector space of words. It creates word vectors that perform well in word analogy tasks and in similarity and named entity recognition tasks. An example of GloVe processing is presented in Fig.21.

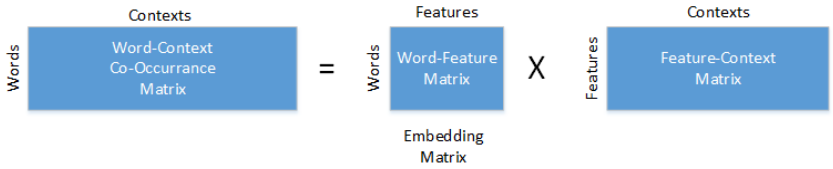


Figure 21: GloVe processing. Adapted from https://www.kdnuggets.com/2018/04/implementing-deep-learning-methods-feature-engineering-text-data-glove.html

5.2 Text similarity

In Natural Language Processing, one of the most important problems is how to handle the similarity of words and texts. Similar texts and words can represent the same information and emotions, and, once it is impossible to know all words for all contexts, knowing if an unknown word or sentence is similar to a known word or sentence increases the probability of the computer "understand" the meaning and emotion of it. So, an answer for the question **"how to determine how 'close' two texts or words are ?"** is fundamental for a good comprehension of a text.

A part of these problems occur due to the spoken language richness, where different words have the same meaning and a single word can assume different meanings. There are different reasons for that such as geographical distance and contextual situations. For example, speakers from Portugal and Brazil share the same language - Portuguese - but some words as *rapariga* (*girl* in Portugal and *hooker* in Brazil) and *fato* (coat in Portugal and fact in Brazil) illustrate the effect of *distance* between the location of the speakers. In the art scenario in Brazil, the expression *"quebre a perna"* (break the leg) is equivalent to *"good luck"*, however, in other different contexts, this expression is considered an offensive wish.

Other potential source of problems occur when considering the distribution of the words in sentences to evaluate if they are similar. For example, the sentences **the dog broke the cat's bed** and **the cat slept at the dog's bed** are very similar, because they share 4 out of 6 words. However, it does not take into consideration the meaning of the words or the entire sentence.

Thus, this closeness must be evaluated both in a surface closeness (considering a lexical similarity) and meaning closeness (semantic similarity).

In the literature there are several approaches to measure the similarity between words and texts. In the following sections, some well known techniques to measure the similarities between texts and words will be described.

5.2.1 Jacquard Coefficient

According to [191], "the Jacquard coefficient measures the similarity between finite sample sets, and is defined as the size of the intersection divided by the size of the union of the sample sets".

When transposing this definition to NLP scenario, it is straightforward to identify that each sample set represents a sentence and their content is composed of the set of unique words existent on each sentence.

For example, in the sentences:

- Sentence A - *"The boy loves play soccer"*
- Sentence B - *"The girl wants to play her videogame"*

we have
A = {*the, boy, loves, play, soccer*}
B = {*the, girl, wants, to, play, her, videogame*}

So, the Jacquard coefficient for these sentences would be:

$$JacquardCoefficient_{(A,B)} = \frac{A \cap B}{A \cup B} = \frac{2}{10} = 0.2$$

resulting in a similarity of 20% between the sentences.

Despite its ease of use, the Jacquard coefficient is considered a "weak" measure of similarity. The weakness of the approach happens because its application is based only on the frequency of words in the texts. However, different words can have the same meaning in a sentence and; even if applied pre-processing techniques of text, such as Stemming and Lemmatization - as presented in sections 4.2.2 and 4.2.3 respectively - the synonyms will not be identified.

Furthermore, this approach does not capture the context of the sentences. The sentence *"I saw a man on a hill with a telescope"* can be interpreted in different ways according to the context which is related to. An example of different interpretations of this sentences are:

- The man seen had a telescope;
- I used a telescope to saw the man.

So, discarding the context when considering the similarity of sentences can bring misunderstandings.

5.2.2 Euclidean Distance

The Euclidean Distance is - possibly - the simplest approach to measure the similarity between two texts. It uses Pythagorean Theorem to determine the distance between two points in a plane or in a n-dimensional space.

In a generalization for an n-dimension, the Euclidean Distance can be represented as the formula:

$$Dist_{(p,q)} = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2}$$

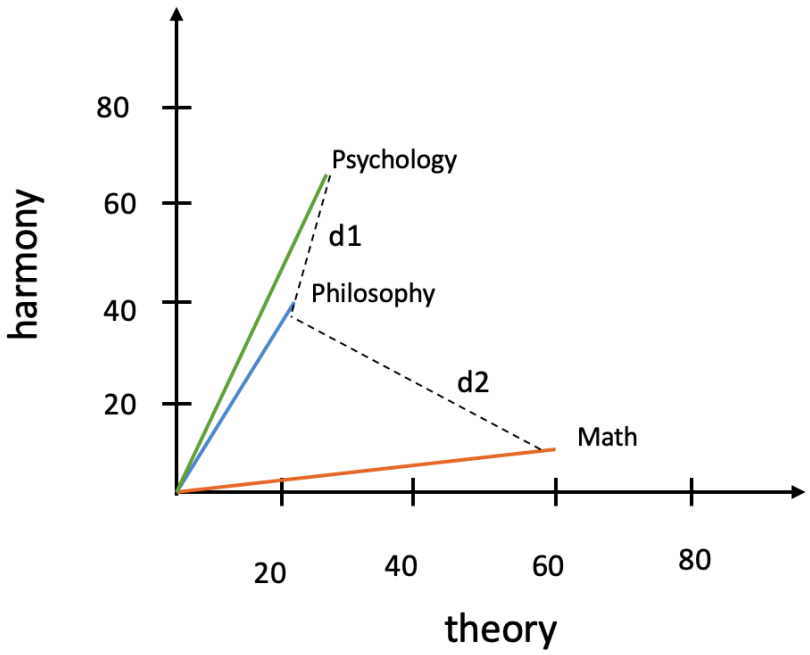


Figure 22: Euclidean Distance. Source https://morioh.com/p/05256ee15f96

where n is the number of dimension existent and p and q represent points in the space.

When considering the Euclidean Distance for textual similarity measurement, the most common approach is to define the set of words in the text as dimensions and the number of their occurrences as the values in the dimension. However, this approach fails because the texts generally have different sizes and words - and consequently different dimensions. So, calculating the Euclidean Distance between texts with different sizes and words will produce unrealistic values.

For example, suppose that in 3 different scholar books (Math, Psychology and Philosophy), the words *harmony* and *theory* occurs 10, 40 and 70 and 60, 20 and 25 respectively. In a 2D-representation, it is possible to translate this information as presented in Fig. 22.

Finally, the Euclidean Distance can be measured as the distance between the points.

5.2.3 Cosine Similarity

The Cosine Similarity is one of the most used technique to measure the similarity between texts, being used together with word embeddings. The concept is based on the idea of all texts are represented as vectors - as presented in sections 5.1.2 and 5.1.3, and their similarities are calculated as a product of these vectors.

For example, let's consider two vectors $\overrightarrow{a}$ and $\overrightarrow{b}$ representing two words respectively. These vectors can be represented as $\overrightarrow{a} = (a_1, a_2, a_3, ...)$ and $\overrightarrow{b} = (b_1, b_2, b_3, ...)$, where a_n and b_n are components of the vector. The value of each component can be the occurrence frequency of each word, the TF-IDF value for each word or the word embedding vector and n is the dimension of the vector.

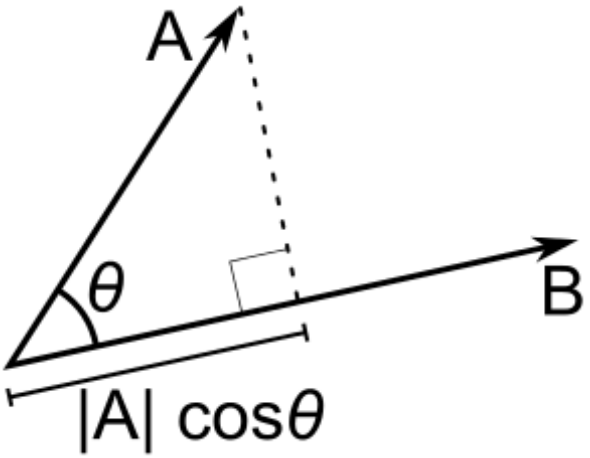


Figure 23: Right-Angled Triangle

Based on the mathematical principle that if two elements have some degree of similarity, they are multiple by some scale, it is possible to infer that the similarity between the vectors $\overrightarrow{a}$ and $\overrightarrow{b}$ is the product between these two vectors, i.e., the dot product $\overrightarrow{a}.\overrightarrow{b}$.

The geometric definition of dot product, is:

$$\overrightarrow{a}.\overrightarrow{b} = ||\overrightarrow{a}||.||\overrightarrow{b}|| \cos\theta$$

When analysing a right-angled triangle, as presented in Fig. 23, being A and B represented by the directions of $\overrightarrow{a}$ and $\overrightarrow{b}$ and an angle θ, it is possible to identify that the projection of $\overrightarrow{a}$ into $\overrightarrow{b}$ is a multiple of $\overrightarrow{b}$.

The value of $|A|cos\theta$ is called cosine similarity between $\overrightarrow{a}$ and $\overrightarrow{b}$. So, as higher is the θ angle, the lesser will be the projection of $\overrightarrow{a}$ into $\overrightarrow{b}$.

In a situation where the $\theta = 90\circ$, there will not be a projection from $\overrightarrow{a}$ into $\overrightarrow{b}$, so it is possible to claim that the vectors (and the texts whose they represent) do not have a relationship, as if $\theta > 90\circ$, the projection will result in a negative value.

Thus, the range of values provided by the cosine similarity is from -1 to 1, which is the same range of Pearson's correlation coefficient (r^2). Once the r^2 provides a scale of interpretation of how data are related, as presented in Table 6, it is possible to use the same scale to interpret how similar two words are.

For example, in Section 5.1.3, the vectors for the words love and hate are:

love = [1, 0, 1, 0, 0, 0]
hate = [1, 0, 0, 0, 0, 1]

To identify how similar the words are, it is necessary to check the cosine similarity between the words - which is 0.5 - and interpret this value according to the Table 6.

Table 6: Levels of relationship

Cosine Similarity	Interpretation
0.00 - 0.19	Very weak relationship
0.20 - 0.39	Weak relationship
0.40 - 0.69	Moderated relationship
0.70 - 0.89	Strong relationship
0.90 - 1.00	Very strong relationship

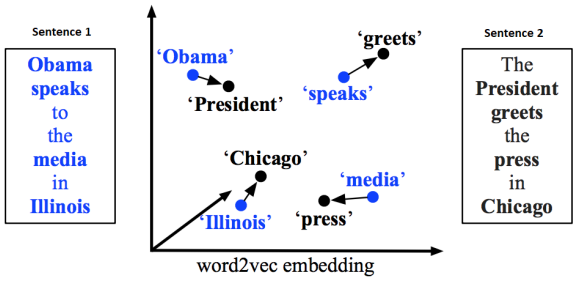


Figure 24: WMD Example. Adapted from [94]

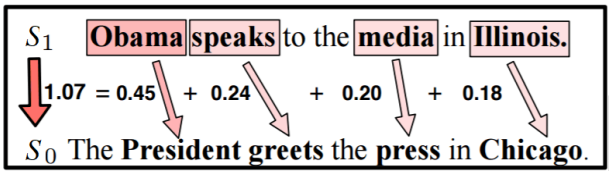


Figure 25: Similarity values. Adapted from [94]

5.2.4 Word Mover's Distance

The Word Mover's Distance (WMD) aims to solve the problem where two sentences have no common words and a cosine similarity of zero, but they are similar concerning their meanings.

Proposed by [94], the WMD uses word embeddings to identify the word similarities and take into account this information to calculate the text similarities. The similarity is calculated using the minimum distance between the words in the first sentence and the words in the second sentence.

For example, when considering the sentences *"Obama speaks to the media in Illinois"* and *"The president greets the press in Chicago"*, there are no common word among the sentences and the cosine similarity is zero. So, to identify the similarity between the sentences, the WMD will consider the minimum distance of the words to determine the similarity, as presented in Fig. 24, resulting in the value presented in Fig 25.

5.3 Summary

In this chapter we introduced the different approaches to represent words and how to identify the similarity between words. Unfortunately there is no a “silver bullet”to be used in all cases. It must be analysed according to the situation and used the most appropriated. In this work, in some situations the representation of a one hot encoding is the most indicated, while in other situation the use of word embeddings is recommended. For this reason we used both approaches, and using the cosine similarity as technique to calculate similarity.

The choice for cosine similarity is justified by the possibility better results for n-dimensions, when comparing to other approaches such as Euclidean Distance.

Part II

Application & Architecture

This part presents the architecture of the Emotional State Classifier and the process flow between their modules.

6

Emotional State Classifier

Sentiments - as presented in Chapter 2 - are defined by [20] as “an acquired and relatively permanent major neuropsychic disposition to react emotionally, cognitively, and conatively toward a certain object (or situation) in a certain stable fashion, with awareness of the object and the manner of reacting.”

Emotional state, on the other hand, according to [176] “may be considered a function of state of physiological arousal and of a cognition appropriate to this state of arousal.”

At a formal description level, the objective of this doctoral work is the creation of an emotional state classifier f where:

$$ES_T = f_{WE_T} \tag{6.1}$$

$T = \{w_1, w_2, w_3, ..., w_n\}$, set of words

$WE = \{we_{1_T}, we_{2_T}, we_{3_T}, ..., we_{n_T}\}$, set of emotions contained in the text’s words.

6.1 Architecture

To achieve the objective of creating an emotional classifier, this work is structured in 3 different parts, as presented in Fig. 26:

- Learning module - which collects the messages and preprocess them to extract the emotions;
- Models - that are the formula responsible provide the classifications;
- Presentation module - which is responsible to return the information to the user.

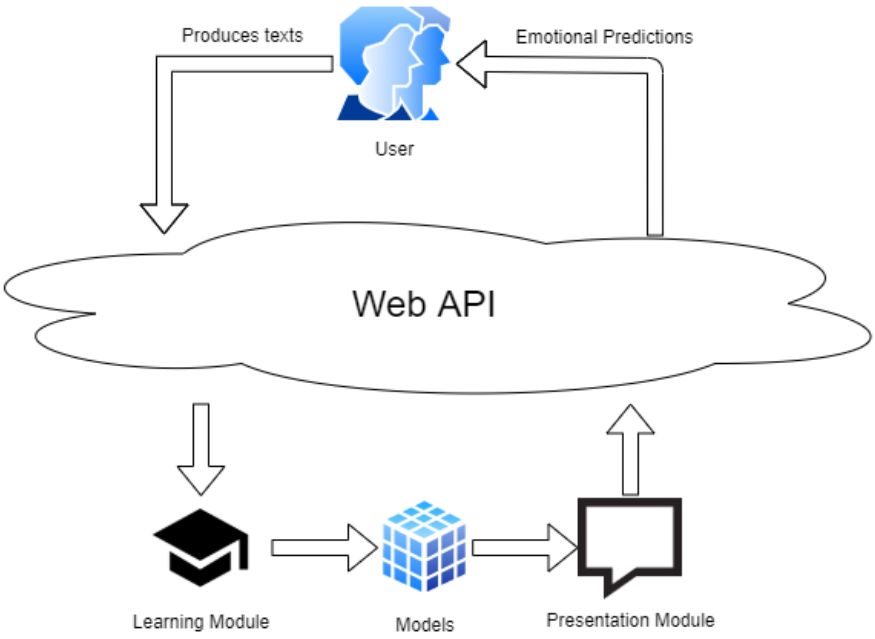


Figure 26: Model's architecture

6.1.1 Modules

The **Learning Module** is the module responsible to transform the text messages in a format that the models can understand. As presented in Fig. 27, it is divided in 2 different stages: the analysis stage and the learning stage.

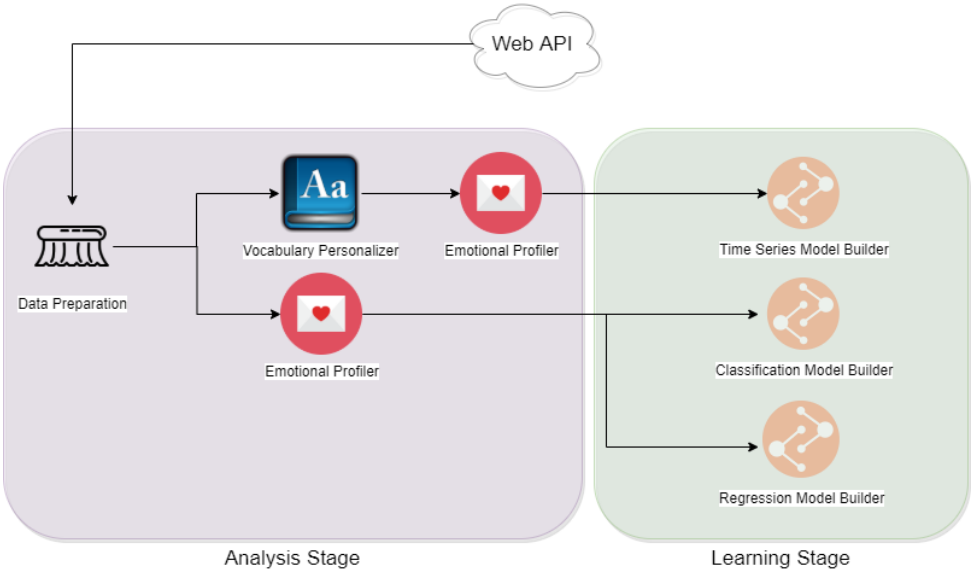


Figure 27: Learning Module

The analysis stage is responsible to receive all texts and transform them in a standard format which will be used by machine learning algorithms. It performs the following actions:

A. Collect messages;

B. Preprocess all messages;

C. Creates and updates a personal vocabulary;

D. Calculate the emotional distribution of the messages according to the vocabulary.

At the learning stage, the processes use the emotional distribution of the messages to create machine learning **models**.

The **models** created (one per author in Time Series models and all-authors models in Classification and Regression models) will feed the **Presentation Module** and provide the output format for the Web API requests.

6.1.2 Web API

Initially, the architecture was planned to have front-end and back-end modules, however, during its development, it was realized that there is no need for a front-end module. This led to the creation of a single module to provide the emotional state detection service. Thus, it was decided to create a Web API to provide the detection of the emotional state, enabling any software developer to consume that service and provide this functionality on his software, regardless of the type of device under which the software is running.

So, the Web API has methods that receive texts independently from their origins, such as files, social media, smartphone logs, SMS messages, etc. In order to increase the flexibility, both input and response data will communicate in an interoperable format. For this reason, all methods in these categories use JSON as format type for inputs and outputs, enabling the software developer to use the operating system as programming language.

After those considerations, four Web API methods were created:

- InputTrainingData - Is the entrance door to input data for training;
- RequestClassification - Is the method responsible to provide the emotional state classification based on a set of given messages;
- RequestRegression - Is the method responsible to return a set of emotional state percentages based on a set of given messages;
- PredictEmotionalLevels - Is the method that predicts and returns the level of each Pluchik's 8 basic emotion, taking into consideration the past level of the emotions.

The Web API provides a communication channel with the client software and for this reason, the format of both request and response methods must be known. According to the Web API method, the JSON formats that must be used are different, as presented in next sections.

6.1.3 InputDataTraining

The *InputDataTraining* is the API method responsible to allow the input of labelled texts from different authors and sources to train the models to predict the emotional state and future emotional levels. The input can be used to provide new data or add data to the existing one, regarding some specific text writer (the author).

Its flow contains three different paths that occur separately, and is exactly the learning phase, being responsible to provide all data needed to feed the models to be trained. The flow starts when the messages are received; the those messages are preprocessed by a Data Preparation task to remove unnecessary information and simplify the texts to be processed.

After the preprocessing, the flow is divided into 2 paths: the first one starts with the Vocabulary Personalization Module to create / update the personal lexicon of the texts writer (the author). Later, the text messages have their Emotional Profile identified according to the personal vocabulary and finally the information flows to the creation of a Time Series Model, where a time-series model for each author's emotions is created.

Following the second path, information (text messages) flows directly to the Emotional Profile identification to calculate the distribution of each basic emotion in each sentence. These distributions will feed 2 paths: one responsible to create a classification model; and the other responsible to create a regression model.

Along the steps described above, the data computed are saved in a database for future researches and follow up the authors emotions over the time.

6.1.3.1 Input and output data formats

The *InputTrainingData* method receives a list $Messages$ of structures

$$Text(txt, label, author, source, datetime)$$

as parameter to train the Classification Model, the Regression Model and the Time Series Emotional Model. In a formal definition, it receives:

$Messages = \{text_1, text_2, text_3, ...text_n\}$
$text_i =< txt, label, author, source, datetime >$ where:

txt is the text to be analyzed;
$label$ corresponds to the author's emotional state when produced the text;
$author$ is an id reference for the author of the text;
$source$ is an identification of the source from where the data was inputted;
$datetime$ is the date & time when the message was produced.

An example of the *InputTrainingData's* input format is presented below:

```
{
   "Messages": {
      "Text": [
      {
         "datetime": "2020-04-20 15:32:35",
         "source": "1",
         "label": "Depression",
         "author": "1",
         "text": "Sometimes I want to cry without reason"
      },
      {
         "datetime": "2020-04-22 17:58:10",
         "source": "1",
         "label": "Anxiety",
         "author": "2",
         "text": "I just want to work after the lockdown"
      }
      ]
   }
}
```

The *InputTrainingData's* method returns a message in JSON format indicating the status of the process, as presented below:

```
{"response": "Sucess"}
```

6.1.4 RequestClassification

The *RequestClassification* method is the core of this work. It receives a set of text messages and returns the emotional state related to the emotions detected in the texts.

In order to identify the author's emotional state, the pipeline - as presented in Fig. 28 - performs a sequence of 6 steps to classify the emotional state according to the texts received.

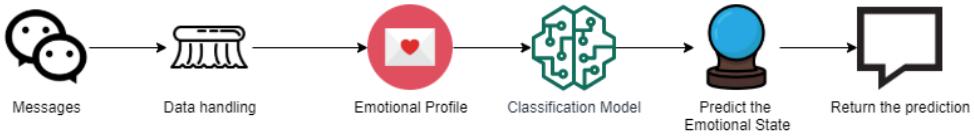


Figure 28: RequestClassification pipeline

These steps are:

1) The method receives a list M of messages to be analysed;
2) The Data Preparation preprocess the data;
3) The Emotional Profiler identifies the emotions for each sentence contained in M;
4) The source's classification model is loaded;
5) The emotions are grouped and normalized and the emotional state is predicted using the source's classification model;
6) The information is returned.

6.1.4.1 Input and output data formats

The *RequestClassification* method, receives a list M of structures $T(text, source)$ as parameter to predict the emotional state. In a formal definition, it receives:

$M = \{t_1, t_2, t_3, ...t_n\}$
$t_i =< txt, source >$
where:

txt is the original text to be analyzed;
$source$ is an identification of the source from where the data was inputted.

An example of the *RequestClassification's* request is presented below:

```
{
   "Messages": {
      "Text": [
      {
         "source": "1",
         "text": "Nothing at all in my life is running ok"
      },
      {
         "source": "1",
         "text": "I really want to be happy"
      }
      ]
   }
}
```

The *RequestClassification's* response returns a message in JSON format indicating the status of the process, as illustrated below:

```
1 {"classification": "Depression"}
```

6.1.5 RequestRegression

The *RequestRegression* method is an auxiliary method to provide a more detailed information about the classification. It receives a set of text messages and returns a list of all emotional states and their respective percentage, according to the emotions detected in the texts.

In order to identify the percentage of each emotional state in the messages, the pipeline - as presented in Fig. 29 - performs a sequence of 6 steps to calculate the emotional state's percentage according to the texts received.

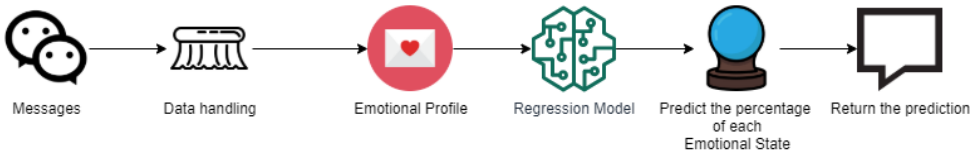


Figure 29: RequestRegression pipeline

These steps are:

1) The method receives a list M of messages to be analysed;
2) The Data Preparation preprocesses the data;
3) The Emotional Profiler identifies the emotions for each sentence contained in M;
4) The source's classification model is loaded;
5) The emotions are grouped and normalized and the emotional state is predicted using the source's classification model;
6) The information is returned.

6.1.5.1 Input and output data formats

The *RequestRegression* method, receives a list $Messages$ of structures $Text(txt, source)$ as parameter to predict levels of each emotional state. In a formal definition, it receives:

$$Messages = \{text_1, text_2, text_3, ...text_n\}$$
$$Text_i =< txt, source >$$

where:

txt is the text to be analysed;

$source$ is an identification of the source from where data was inputted.

An example of the *RequestRegression's* request is presented below:

```
{
   "Messages": {
      "Text": [
      {
         "source": "1",
         "text": "Nothing at all in my life is running ok"
      },
      {
         "source": "1",
         "text": "I really want to be happy"
      }
      ]
   }
}
```

The *RequestRegression's* response returns a message in JSON format indicating the level of the each emotional state, in a range from 0 up to 1, as illustrated below:

```
{
   "EmotionalStates": {
      "EmotionalState": [
      {
         "Name": "Depression",
         "Level": "0.84"
      },
      {
         "Name": "Happiness",
         "Level": "0.04"
      },
      {
         "Name": "Anxiety",
         "Level": "0.12"
      }
      ]
```

```
17      }
18  }
```

6.1.6 PredictEmotionalLevels

The *PredictEmotionalLevels* provides information about an author's future emotional levels, taking into consideration the previous emotional levels. This is a relevant information because according to the psychological bibliography, some diseases - such as depression - need to be followed up for a long time prior to be diagnosed. However, it is possible an individual begins to demonstrate depressive signals in the last messages, but he won't have his emotional state classified as depressive because of the lack of information in the time. For this reason, this method predicts the emotional levels of an individual, allowing to identify individuals with risk of future diseases.

In order to predict each basic emotion of an author, the pipeline - as presented in Fig. 30 - performs a sequence of 5 steps to calculate the emotional state's percentage according to the texts received.

Figure 30: PredictEmotionalLevels pipeline

These steps are:

1) The method receives the author id;

2) The Time Series personal model is loaded;

3) The most recent last 12 weeks emotions are loaded from the database;

4) The emotions are predicted for the next 4 weeks;

5) The information is returned.

6.1.6.1 Input and output data formats

The *PredictEmotionalLevels* method, receives a structure $Author(source, id)$ as parameter to predict the user's future emotional levels. In a formal definition, it receives:

$$Author = (source, id)$$

where:

$source$ is the source (Twitter, SMS, WhatsApp) from where the data was inputted;
id is the author id that the predictions refers to.

An example of the *PredictEmotionalLevels's* request is presented below:

```
{
    "source" : "4",
    "id" : "3"
}
```

The *PredictEmotionalLevels's* response returns a message in JSON format containing the distribution of each basic emotion up to 4 weeks, as presented below:

```
{
    "Week1": {
        "Anger": "0.15",
        "Anticipation": "0.15",
        "Disgust": "0.22",
        "Fear": "0.14",
        "Joy": "0.06",
        "Sadness": "0.18",
        "Surprise": "0.07",
        "Trust": "0.07"
    },
    "Week2": {
        "Anger": "0.05",
        "Anticipation": "0.13",
        "Disgust": "0.26",
        "Fear": "0.13",
        "Joy": "0.07",
        "Sadness": "0.21",
        "Surprise": "0.08",
        "Trust": "0.7"
    },
    "Week3": {
        "Anger": "0.12",
        "Anticipation": "0.17",
        "Disgust": "0.17",
        "Fear": "0.16",
```

```
        "Joy": "0.09",
        "Sadness": "0.19",
        "Surprise": "0.02",
        "Trust": "0.08"
    },
    "Week4": {
        "Anger": "0.08",
        "Anticipation": "0.10",
        "Disgust": "0.28",
        "Fear": "0.17",
        "Joy": "0.08",
        "Sadness": "0.16",
        "Surprise": "0.06",
        "Trust": "0.07"
    }
}
```


6.2 Tasks

Each method defined in the Web API needs to invoke some tasks to produce the results. Here is presented an overview about these tasks.

The most relevant tasks used by the emotional state classification system are:

- Data Preparation;
- Emotional Profiler;
- Vocabulary Personalizer;
- Time Series Model Builder;
- Classification Model Builder;
- Regression Model Builder.

They will be described in the next subsections.

6.2.1 Data Preparation

The objective of Data Preparation is to preprocess the texts received, reducing the set of original words to a set containing only emotional words. This task reduces significantly the size of the dataset to be processed.

In general, preprocessing is the first step in NLP approaches being used to extract relevant information from text, relationships and other useful insights which the words can carry on.

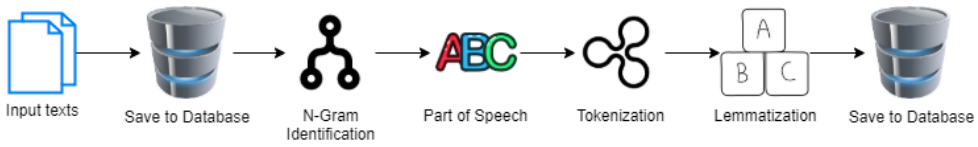


Figure 31: Data Preparation pipeline

The Data Preparation was idealized to perform a pipeline to clean the input text prior to analyse it. The pipeline, as presented in Fig. 31, begins when the data is received. In a formal description, the data received is a set DT of sentences ST where:

$DT = \{st_1, st_2, ..., st_n\}, n > 0$
$ST = \{w_1, w_2, ..., w_k\}, k > 0$
W_k is a word at position k

The next step in the pipeline saves DT in the database to keep its original text prior the processing. At the third step, all messages are analysed to detect known N-Grams (bigrams and trigrams essentially). The objective of this step is avoid that known expressions (such as “be right back”) with more than one word be handled as a set of single words. Through a list of known expressions, all messages are analysed in order to identify the existence of a known expression. If identified, the expression in the sentence is updated to its “ngrammed” version.

Then, the sentences are analysed through a part of speech tagger. This process identifies the grammatical categories of the words contained in a sentence. The idea in this process is to keep only nouns, verbs, adverbs and adjectives. This is important because only these grammatical categories can bring emotional information. Moreover, this approach helps to decrease the processing time, because the step of stopwords removing is not necessary since the grammatical category of the most known stopwords are different than nouns, verbs, adverbs and adjectives used in this approach.

Later, the remaining sentences are tokenized, i.e., all sentence is considered as an array of words. For each word in this array of words, it lemmatizes the word to a normalized form. This process uses lemmatization instead of stemming because the lexicon’s words rate identification is higher when compared to stemming. So, this step is more important to us because it increases the rate of identification of words in the emotional lexicon.

Finally the new sentence containing only lemmatized emotional words is saved into the database, enabling a comparison between the original text and the preprocessed text.

6.2.2 Emotional Profiler

The Emotional Profiler is the central core. It is responsible for the identification of the emotions contained in the input texts and extract these emotions in a personal distribution that represents the author's emotional characteristics when writing.

When considering the claim of [177] ("if the eyes are the window to the soul, then words are the gateway to the mind"), it is possible to identify a relation between this claim and the works of [184] and [92] that claim that texts reflect their author's personality embedding their emotions when writing.

The existence of emotional lexicons linking words to emotions - as the EmoLex lexicon - provides a key to extract emotions from texts and identify the emotional distribution in each sentence. When analysing for long period messages from a single author, it is possible to extract the frequency of each basic emotion used in his sentences, and thus identify his emotional profile.

So, when taking into consideration the Plutchik's model of emotions, referred above, that proposes 8 basic emotions, it is possible to define that each emotional profile is composed of 8 basic emotions, distributed according to the author's personal characteristics.

So, it is possible to define an emotional profile as a 8-dimension tuple:

$$EP = \{EP_{Anger}, EP_{Anticipation}, EP_{Disgust}, EP_{Fear}, EP_{Joy}, EP_{Sadness}, EP_{Surprise}, EP_{Trust}\}$$

where:

$$EP_{Anger} + EP_{Anticipation} + EP_{Disgust} + EP_{Fear} + EP_{Joy} + EP_{Sadness} + EP_{Surprise} + EP_{Trust} = 1$$

and

$0 \leq EP_e \leq 1$, for e in $[Anger, ..., Trust]$

The process, as presented in Fig. 32 begins when a set of preprocessed sentences from an author is received. For each word in the sentences, it is checked their existence in the emotional lexicon (EmoLex or personal emotional lexicon based on EmoLex). If the word under consideration belongs to the emotional lexicon, the counters for each basic emotion flagged in the lexicon are incremented by 1. The result is a list of EP tuples (8 dimension values, as introduced above), where each element of this list represents a preprocessed sentence and each dimension represents the total of words flagged in the dictionary having the same emotion in the respective preprocessed sentence.

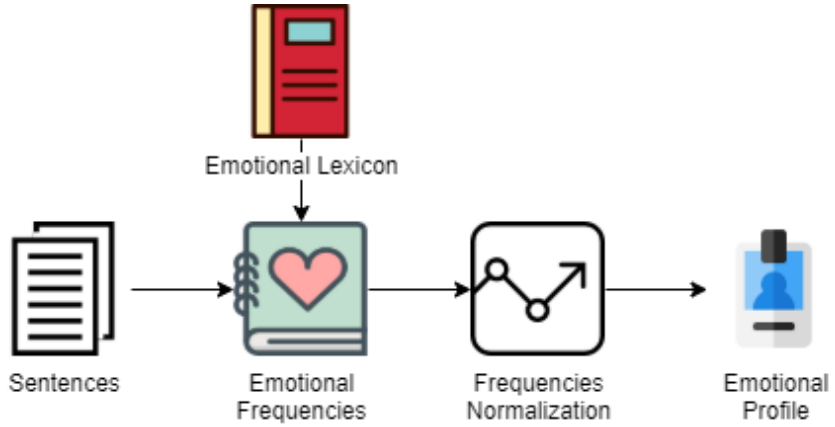


Figure 32: Emotional Profile pipeline

Later, the sums of all rows of each basic emotion are computed normalized to ensure that the sum of all basic emotions is 1. Finally, this distribution of normalized emotions represents the **author's emotional profile**. An example of how the Emotional Profile task works is presented in Fig. 33

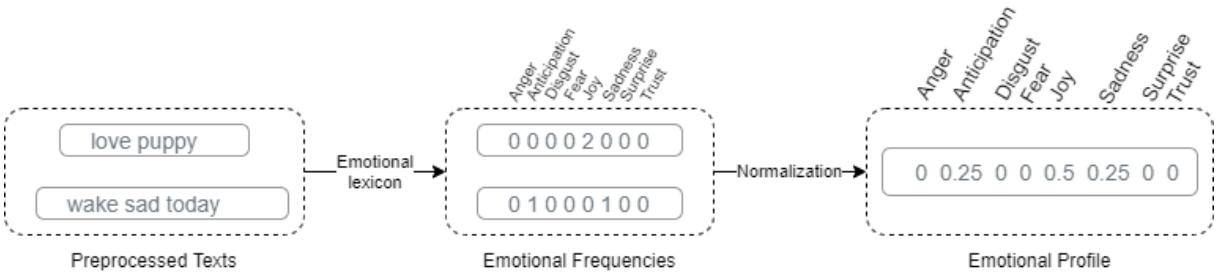


Figure 33: Emotional Profile pipeline example

In this step, must be verified the module's caller. If it was for Time Series Personal Module data training, the correct lexicon to be loaded is the author's personal lexicon, otherwise the original EmoLex lexicon must be loaded.

6.2.3 Vocabulary Personalizer

The Vocabulary Personalizer module aims to create an exclusive and personal emotional lexicon, according to the author's emotional profile to help on the forecasting of emotional levels.

This new lexicon is based on the NRC Word-Emotion Association Lexicon (EmoLex) which is a lexicon created by [134]. Its main characteristic when compared to other emotional lexicons is the relationship indication between the words and the Plutchik's basic emotions. This relationship is presented as binary values, being 0 considered as not related and 1 as strongly related. An example of EmoLex words is presented in Table 7.

Table 7: EmoLex lexicon snapshot

Word	Positive	Negative	Anger	Anticipation	Disgust	Fear	Joy	Sadness	Surprise	Trust
aback	0	0	0	0	0	0	0	0	0	0
abacus	0	0	0	0	0	0	0	0	0	1
abandon	0	1	0	0	0	1	0	1	0	0
abandoned	0	1	1	0	0	1	0	1	0	0
abandonment	0	1	1	0	0	1	0	1	1	0
abate	0	0	0	0	0	0	0	0	0	0
abatement	0	0	0	0	0	0	0	0	0	0
abba	1	0	0	0	0	0	0	0	0	0
abbot	0	0	0	0	0	0	0	0	0	1

The personal lexicon creation pipeline, as presented in Fig. 34, receives preprocessed texts from a single author. This text input is done by loading all preprocessed texts from the author saved during the Data Preparation execution. These preprocessed texts - words identified as nouns, verbs, adverbs and adjectives, will be converted into word vectors using an approach known as “word embeddings”.

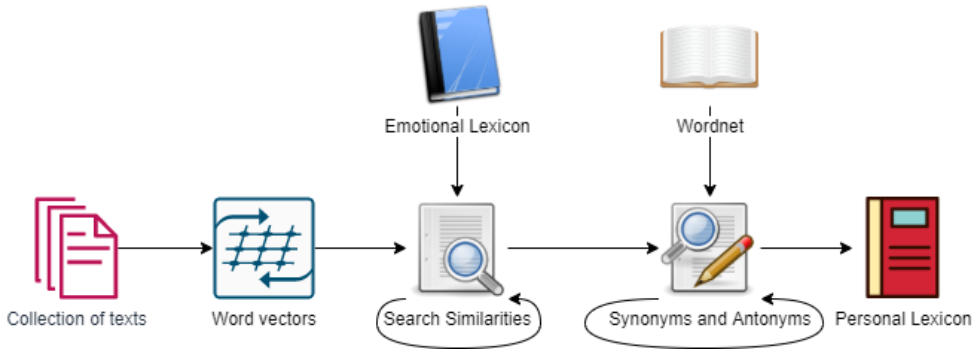


Figure 34: Personal lexicon creation pipeline

In order to avoid wrong interpretations in emotional words preceded by “no” and “not”, it was considered an approach to “neutralize” the emotional load of these words. It’s because - in our understandings - that the words “no” and “not” do not change the polarity of a word. The sentence “I don’t like soccer” is different from “I hate soccer”. The first one claims that the author does not have feelings about soccer. The second one claims that the author has negative feelings.

For this word vectors creation, the module uses the Word2Vec algorithm, as presented in subsection 5.1.3.1, considering all 5 words surrounding each specific word and a 50 as dimension size of each vector. The algorithm executes 50 epochs for training, and at the final, a set of word vectors having 50 dimensions is returned.

The next step is to identify the emotional similarities contained in the preprocessed texts. For this purpose, the module iterate over all words existent in the EmoLex lexicon and measures the similar words according to the word vectors (trained using Word2Vec), where only the similar words having at least 0.7 of similarity were considered.

After iterate over all words contained in EmoLex lexicon and find their similarities according to the word

vectors, the resulting intermediate document is a new lexicon having the same structure of EmoLex lexicon, containing the same words and the emotional flags as EmoLex lexicon. Besides, all words identified as similar with more than 70% of similarity are contained in the intermediate document, sharing the same emotional flag as its "referencer" in EmoLex lexicon.

Next, the intermediate document is iterated using the same process occurred previously in order to capture each word and compare the existence of antonyms in Wordnet [131]. If antonyms are find, the word is included in the intermediate document, but their opposite emotions are changed when compared to the "reference". For example, if the word "happy" (which contains a flag on the emotion **joy**) is contained in the intermediate document, its antonyms "upset" will be added to the intermediate document, but their flag will indicate the emotion **sadness**, which is opposite to joy according to the Plutchik's model.

Note that it was intended not to iterate as a recursive model - i.e. searching the similarities of words found as similar. This restriction was applied to speed up the entire processing.

The final result is a new lexicon, based on the EmoLex emotional lexicon and enriched with some contextual words and their relationships according to the author's texts. Once the word vectors contain only texts of a single author, and these word vectors are the source of similarities, this new lexicon can be considered as the author personal emotional vocabulary, containing the word's emotional reference used by the author.

6.2.4 Time Series Personal Model

A significant benefit of classifying author's emotional state is the possibility to forecast emotional states based on the emotion's historical to avoid potential risks to people's health. This possibility allows to predict the emotions of a person in a normal situation and use this information to forecast dangerous situations. For this reason, the Time Series Personal Model module takes into consideration the historic of emotions of an author to predict the emotions that will have.

According to Plutchik's model, the 8 basic emotions are opposite among them. So it is straightforward to conclude that when an emotion increases, its opposite emotion decreases. So, to predict the emotions based on their past emotions, we decided to handle this problem as a *Multivariate Time Series* problem. During the test phases, it was identified that each single emotion set is a stationary series, so, the use of these information in a time series is recommended and will not demand individual analysis.

Thus, to forecast the future emotional profile, it was applied a Vector Autoregression (VAR) algorithm, in a pipeline as presented in Fig. 35, which receives a list of the 8 basic emotions percentages, where each set of 8 basic emotion represents the author's emotional profile during a week. The training dataset was composed of the last author's 12-week emotional profiles, representing the average of all basic emotions, grouped by week, during the last 12 weeks - or 3 months, as presented in Table 8.

The reason for considering the last 12 weeks is because some diseases - as depression - need a long follow up to be diagnosed. And, for trending analysis, a projection of how will be the emotional profile is mandatory to define strategies to prevent future illnesses. For this reason the limit of 4 weeks

Table 8: Training dataset example

Author	Week	Anger	Anticipation	Disgust	Fear	Joy	Sadness	Surprise	Trust
1	1	0.08	0.20	0.12	0.25	0.25	0.01	0.06	0.03
1	2	0.08	0.18	0.08	0.27	0.22	0.02	0.05	0.10
1	3	0.11	0.20	0.18	0.02	0.07	0.18	0.10	0.10
1	4	0.21	0.14	0.00	0.02	0.10	0.05	0.13	0.34
1	5	0.17	0.15	0.10	0.14	0.05	0.29	0.03	0.09
1	6	0.07	0.05	0.04	0.21	0.20	0.18	0.20	0.05
1	7	0.08	0.21	0.24	0.23	0.01	0.00	0.16	0.07
1	8	0.01	0.16	0.08	0.12	0.02	0.17	0.19	0.25
1	9	0.06	0.13	0.26	0.06	0.01	0.19	0.13	0.16
1	10	0.07	0.18	0.19	0.12	0.30	0.02	0.0	0.12
1	11	0.47	0.03	0.14	0.06	0.14	0.06	0.03	0.07
1	12	0.21	0.12	0.01	0.07	0.13	0.15	0.04	0.27

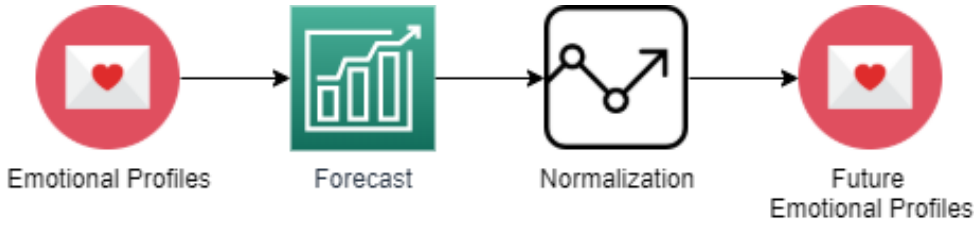


Figure 35: Time Series Personal Model Builder pipeline

for forecasting was considered for the model. After the model prediction, the values are normalized - i.e. having their values in a range between 0 and 1 - for each week to create the future emotional profiles.

6.2.5 Classification & Regression Model Builder

The main objective of these tasks is to create a machine learning model to classify the emotional state of an author. In order to achieve these purposes, the task performs a supervised learning taking into consideration the emotions detected in the texts. The result of the classification model is a model which allows to identify the emotional differences between a set of emotional states, and according to these differences, identify the current emotional state of the author. For the regression model, the result is a model which identifies the probability of a set of emotions be identified as a specific emotional state.

To create these models, it was defined a pipeline (the same for both models) as presented in Fig. 36, starts with a list of preprocessed texts from an author. The next step calculates the emotional profile - as introduced in Section 6.2.2 - for each message and the label indicating the emotional state, that will input the supervised learning. Later, the outliers of all emotions are ignored and all emotions are normalized to fit in a range from 0 to 1, indicating the percentage of each basic emotion in the author's profile.

Regarding to the outliers identification, for each emotion is calculated its mean ($\overline{x}$) and quantiles. Later, it is calculated the Inter Quantile Range (IQR), which is the difference between the quantile 3 and

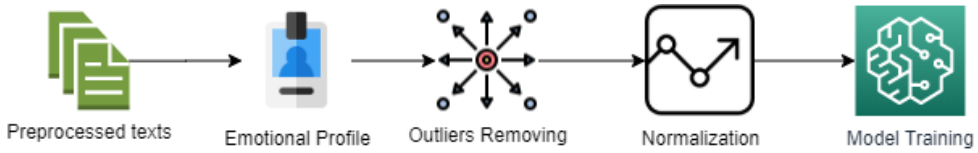


Figure 36: Classification Model module pipeline

1. The superior and inferior limits are identified through the formulas $L_{Sup} = \overline{x} + 1,5 * IQR$ and $L_{Inf} = \overline{x} - 1,5 * IQR$. If a sentence contains an emotion percentage higher than L_{Sup} or lower than L_{Inf}, it is discarded.

The training step uses a SVM algorithm to learn the distribution of the emotions with an input shape of (8,) - i.e. a specific value regarding to each basic emotion - and the classification output is a value indicating which emotional state was classified. For Regression Model, the decision function of SVM algorithm receives the indication of giving per-class scores probabilities.

The choice for SVM is because the algorithm had the best performance during the benchmark tests.

Part III

Case Studies

This part presents some case studies using emotional information, aiming at demonstrating different real applications of the emotional state classification. This part is a compilation of papers submitted and presented in conferences. Each chapter in this part corresponds to one of those papers, however to avoid repetitive information, the papers were not included directly; they were slightly adapted and the sections regarding to the state of the art and the approach proposed in this Ph.D. work were removed, because their content was presented with more detail in Part I; also suggestions for future work were removed from th last section. However, to keep the context and motivation, or the approach followed to deal with each case study, it is possible that some paragraphs may sound repetitive.

The case studies chosen and discussed in the next chapters aim at demonstrate:

- How emotional profile can differentiate people;
- How to create and represent a personal vocabulary and their individual emotions;
- How each one of us tend to be attracted and create good relationships with people with similar emotional profiles;
- How the emotions affect the daily routines, and how to detect and forecast them using texts;
- How to use emotions to predict a population's behaviour;
- How to use text to classify emotional state individually.

7

Case Study 1 - User identification by emotional profile

Since Barack Obama's election, the politicians are using social media to have a direct contact with their voters and increase its credibility with their posts and comments. On the other hand, this direct channel enables a correct perception by the voters about the politics, creating opinions about the subjects they consider important. This phenomenon is increasingly turning politicians into digital influencers. So, the way as they communicate in social media can be considered their "personal signature"; so their worries about the way how they can be interpreted are equally important.

With massive information from social media, the digital influencers and their legion of followers validate, reinforce and amplify news, many times faked. As the main objective of these individuals is be "liked, loved and shared", it is very important to choose correctly the words contained into their texts, in order to maximize the sentiment raised up in the readers.

So, the emotional characteristics contained in the messages make up an "emotional profile"about the author and which, along with the words used in the text, helps to determine the message's author profile while writing.

For example, the following posts are from different authors and deal the same theme - the Paris Climate Agreement - however, the writing styles are different and arouse different emotions. While the first balances positive and negative words in the text, the second mostly uses words with negative emotions:

> "Today marks a crucial step forward in the fight against climate change, as the historic Paris Climate Agreement officially enters into force. Let's keep pushing for progress"(Barack Obama);

> "I'm optimistic we can stop climate change and help those who are being hurt the most by it—all while meeting the world's energy needs"(Bill Gates).

In this case study, it is presented an approach using the author emotional profile in order to improve the authorship identification.

7.1 Data analysis

In order to predict the authors of a post based on the emotion contained in text, 2100 Facebook posts were collected from 8 different authors of different areas, as presented in Table 9. All data was collected at the same time span, reducing temporal situations interference in the text emotions. In order to compare all information, the posts were manually labelled into 2 categories: politicians and non-politicians.

Table 9: Posts authors

Author	Area	Category
Barack Obama	Politics	Politician
Bill Gates	Business	Non-Politician
Donald Trump	Business	Non-Politician
Hillary Clinton	Politics	Politician
Jeremy Corbyn	Politics	Politician
Leonardo Di Caprio	Entertainment	Non-Politician
Magic Johnson	Sports	Non-Politician
Theresa May	Politics	Politician

The task of predict the author of a text is composed of several intermediates steps. First, it was needed some preprocessing tasks in order to reduce data size by removing unnecessary text from the original message.

Preprocessing is a very important step in text mining processes and applications. It is the first step not only for text mining approaches but also in data mining. There are several preprocessing techniques useful in order to extract information from text, and their usage is according to the characteristics of the information desired. Despite of some techniques were created in data mining, they are useful in text mining approaches, since the same technique can be used for both information extraction, information retrieval, or combined

The preprocessing, after the tokenization, was divided in 3 parallel jobs, as showed in Fig. 37: Part of Speech Tagging (POS-T), Named Entity Recognition (NER) and Stopwords Removal. This strategy was used because both POS-T and NER need the text in the original format, in order to return the correct data from the analysis.

The POS-T process identifies the text grammatical structure. Concerning text cleaning, only nouns, verbs, adverbs and adjectives were preserved. This is important because only these grammatical categories can bring emotional information. So, in a more formal way, the Tokenization process converts the original text D in a set of tokens $T = \{t_1, t_2, ..., t_n\}$ where each element contained in T is part of the original document D. Later, the POS-T labels each token with a semantic information. Later, a process collects all nouns, verbs, adverbs and adjectives in a set P, where $P_T = \{p_{(T,1)}, p_{(T,2)}, ..., p_{(T,k)}\}$ and 0 $\leq$ k $\leq$ n and $P_T \subset T$.

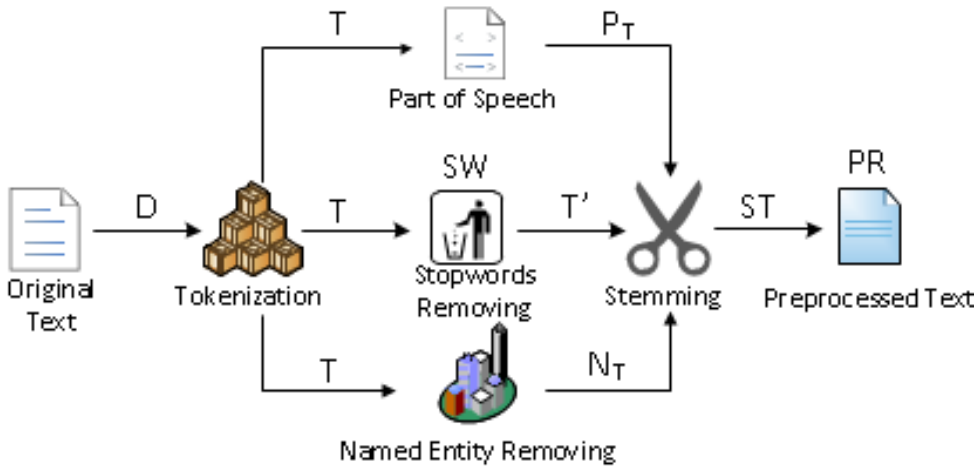


Figure 37: Preprocessing tasks

Similarly, NER process identifies names in 3 different categories: "Location", "Person"and "Organization". Once identified tokens in one of these categories, they are removed. As a result, a set $N_T = \{n_{(T,1)}, n_{(T,2)}, ..., n_{(T,j)}\}$ is constructed based on identified word category and where 0 ≤ j ≤ n and $N_T \subset T$. This step is important to be done in parallel with POS because some locations can be confused with nouns (as Long Beach, for instance).

The Stopwords list is a personal predefined set $SW = \{sw_1, sw_2, ...sw_y\}$ of words, manually created according to several similar lists available on the internet.

After the 3 preprocessing tasks finish, the result document ST must contain a set of words where $ST = T' \cap P_T \cap N_T$.

Later, in this set ST is applied a stemming process to reduce the words to their word stem in order to consider all inflected words as only one, producing the preprocessed text $PR = \{ST_1, ST_2, ..., ST_z\}$ ready to be analysed.

For all three tasks - POS-T, NER and Tokenization - the Stanford Core NLP [113] toolkit was used.

An example using a real post from Barack Obama of text preprocessing is presented in Fig. 38.

7.2 Polarity analysis

The first analysis made was aimed at determining the posts polarities. To achieve this objective, after the preprocessing, all sentences contained in PR were compared against EmoLex lexicon [133] in order to identify the positive and negative words contained in the text. This analysis did not take into account the intensity of the polarities neither the emotions.

When comparing the posts' polarities according to their author's category (politicians and non-politicians), the data did not reveal relevant differences between politician and non-politicians, as showed in the Fig. 39. The same analysis was confirmed using the chi-squared test, where was obtained a value $\chi^2 = 1$, indicating that both polarities data (politicians and non-politicians) are not independent.

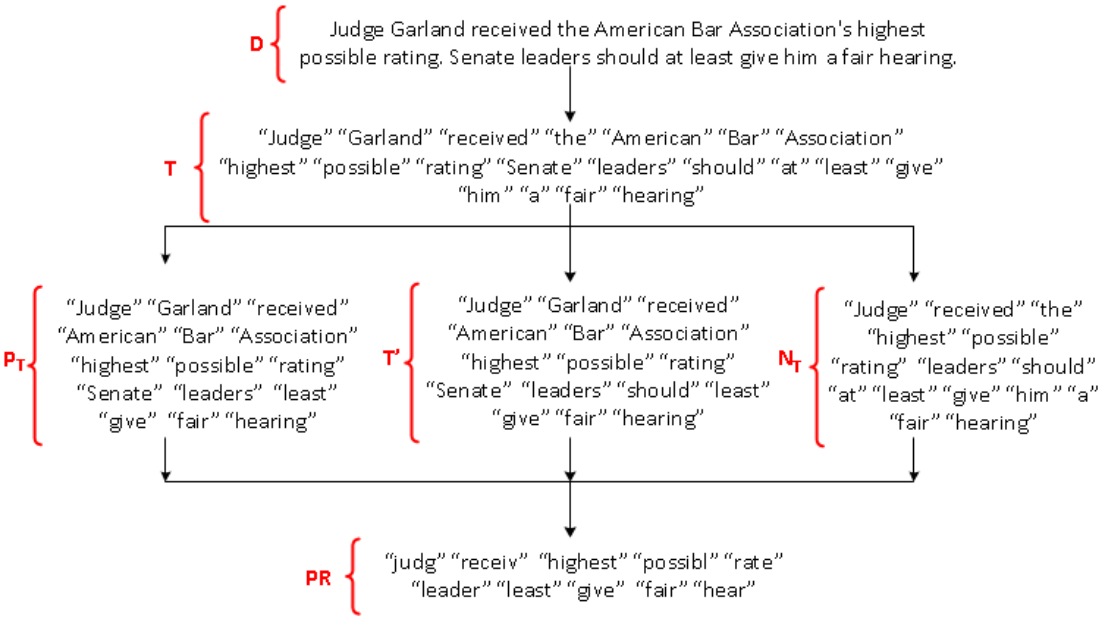


Figure 38: Preprocessing text example

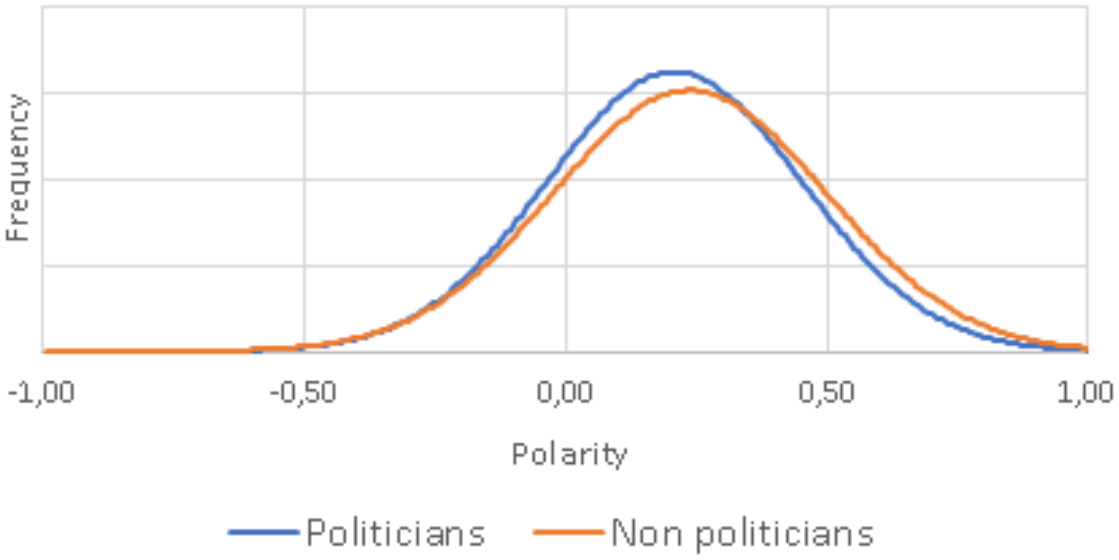


Figure 39: Polarities distribution by category

However, this interpretation may lead to a wrong understanding about the scenario. When comparing the polarities by author, according to Fig. 40, it is possible to conclude that while politicians tend to have their posts in the same area in a normal distribution, non-politicians tends to be in the extremes, i.e., they are blunter than politicians when expressing through Facebook and indicating that each author has its own "emotional signature"in his posts.

This information is confirmed in Table 10, which presents the positive and negative polarities by author.

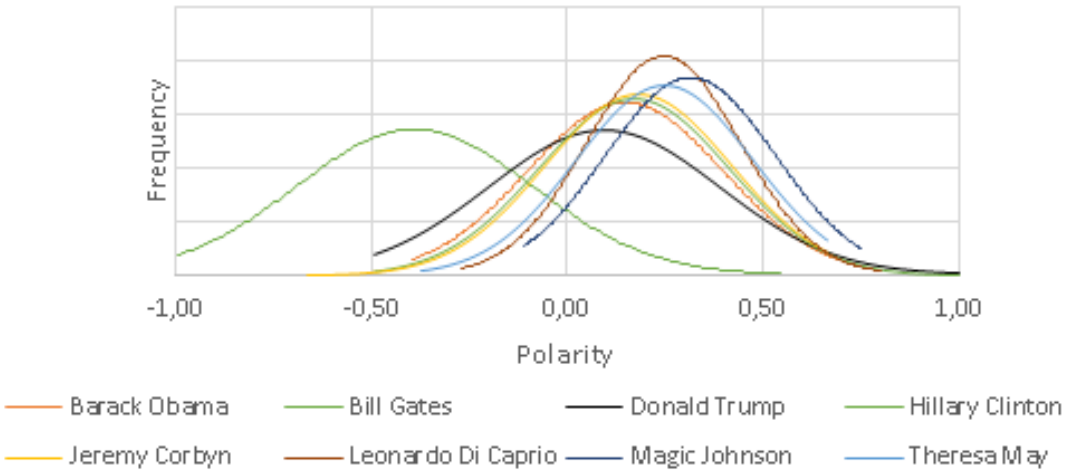


Figure 40: Polarities distribution by author

Table 10: Polarities by author

Author	Positive	Negative
Barack Obama	0.28	0.13
Bill Gates	0.30	0.11
Donald Trump	0.25	0.16
Hillary Clinton	0.35	0.17
Jeremy Corbyn	0.30	0,13
Leonardo Di Caprio	0.34	0.09
Magic Johnson	0.37	0.06
Theresa May	0.36	0.10

7.3 Lexicon-based emotion analysis

In order to analyse the emotions contained into the text, it was used a lexicon-based approach, which consists in comparing the labelled emotion contained into the EmoLex lexicon with the preprocessed texts described earlier. Using the emotions model proposed by Plutchik [157], where all sentiment is composed of a set of 8 basic emotions (*anger*, *anticipation*, *disgust*, *fear*, *joy*, *sadness*, *surprise* and *trust*), all posts where analysed according to this model and a list of emotions in each post was generated, according to Table 11.

Hence, when applying the Person's correlation coefficient (r^2) between polarities and basic emotions, as presented in Table 12, it is possible to point which emotions are related with polarities.

In a scale ranging from -1 to 1, emotions related with a high r^2 value indicates a strong relation with the polarity (as *Anger* and negative polarity), while high negative r^2 values indicates a strong inverse relationship (as *Fear* and positive polarity). In our approach, ambiguous emotions are classified when the standard deviation for r^2 polarity's value is less than 10% range (i.e. 0.2).

Table 11: Basic emotions average per author

Author	Anger	Anticipation	Disgust	Fear	Joy	Sadness	Surprise	Trust
Barack Obama	0.08	0.15	0.03	0.10	0.13	0.05	0.05	0.21
Bill Gates	0.06	0.14	0.04	0.08	0.15	0.06	0.06	0.14
Donald Trump	0.06	0.12	0.02	0.09	0.12	0.10	0.04	0.16
Hillary Clinton	0.14	0.26	0.02	0.07	0.22	0.15	0.12	0.30
Jeremy Corbyn	0.08	0.16	0.03	0.08	0.09	0.08	0.06	0.23
Leonardo Di Caprio	0.04	0.11	0.01	0.07	0.09	0.03	0.03	0.16
Magic Johnson	0.03	0.19	0.03	0.05	0.21	0.04	0.07	0.21
Theresa May	0.06	0.17	0.02	0.06	0.14	0.07	0.07	0.22

Table 12: Correlation between polarities and emotions

Polarity	Anger	Anticipation	Disgust	Fear	Joy	Sadness	Surprise	Trust
Positive	-0.10	0.49	-0,26	-0.90	0.48	-0.22	0.44	0.40
Negative	0.83	0.27	-0,08	0.60	0.01	0.89	0.34	0.37

In summary, positive and negative emotions are important to describe the author's emotional pattern, while the neutral emotions do not have significant contribution to achieve this objective. the emotions classified in text according to polarities are:

- Positive polarity - Joy;
- Negative polarity - Anger, Fear, Sadness;
- Ambiguous polarity - Anticipation, Disgust, Surprise, Trust.

7.4 Machine learning-based emotion analysis

Once identified the average of each emotion from author, the next analysis was to identify the emotional pattern of the author. To achieve this, it was used an approach based on machine learning (ML) techniques. The first attempt was aimed at identifying the lowest prediction rate author. For this, it was used the same messages with just preprocessing and the authors identification in a ML approach. Once this information was obtained only by texts, this value can be considered the lowest acceptable value, and, in case of decreasing this rate, it may be interpreted as a negative influence of emotions in the authors prediction.

In our initial tests, the best rate was presented by a SVM implementation through Weka [67] and a 10-fold cross validation in the whole dataset, with a correct prediction precision of 82% of when predicting authors.

When the lowest prediction rate was identified, the next step was to classify using the emotional information. Using the previous preprocessed texts, polarity values and each basic emotion rate, a new dataset was generated in order to be used in ML process. In our tests, it was used the most relevant algorithms for text classification, as SVM, Naive Bayes, Random Forests, however, using a Naive Bayes Multinomial implementation through Weka and a 10-fold cross validation in the whole dataset, returned a precision of 87.41% of correct predictions when predicting authors. Both results (non-preprocessed and preprocessed) are presented in Table 13.

Table 13: Detailed accuracy results for non-preprocessed and preprocessed texts

	Non-preprocessed texts			**Preprocessed texts**		
Author	**Precision**	**Recall**	**F-Measure**	**Precision**	**Recall**	**F-Measure**
Barack Obama	0.933	0.82	0.776	0.907	0.860	0.883
Bill Gates	0.887	0.874	0.836	0.882	0.944	0.912
Donald Trump	0.317	0.465	0.518	0.761	0.556	0.642
Hillary Clinton	0.571	0.686	0.693	0.676	0.762	0.716
Jeremy Corbyn	0.741	0.824	0.814	0.859	0.836	0.847
Leonardo Di Caprio	0.807	0.782	0.734	0.878	0.876	0.877
Magic Johnson	0.867	0.893	0.867	0.945	0.917	0.931
Theresa May	0.495	0.591	0.588	0.649	0.758	0.699

7.5 Conclusion

This case study presents a combination of lexicon-based and machine learning approaches to explore the emotions contained in a text through the best practices in sentiment analysis in order to increase the results' accuracy in authorship identification.

Everyone have particular characteristics of expressing themselves, and these personal characteristics can be expressed in their texts.

Once the author's writing style profile is known, by using the emotional information contained into text helps to increase the accuracy on authorship identification. This claiming is based on the successful predictions rate grown from 82% to 87.41% in our tests, besides the values of precision, recall and f-measure which have increased in the majority of the cases, when using emotional labelled data. This improvement can be interpreted as a very satisfactory result as our proposal.

8

Case Study 2 - Lexicon personalization

It is common in large countries that people share known words with different meanings. In Portugal, people from Lisbon order a draft beer as “imperial” while in Porto is “fino”, however, in Lisbon “fino” can be a polite person and in Porto “imperial” is about everything related to the Portuguese royalty. These words, pronounced by people from different locations express different emotions, and on the other hand, these emotions express the sentiment that the author wanted to transmit when pronounced them. So, it is essential to know what the author wants to transmit, in order to avoid misunderstandings - commonplace when we travel to different countries which speak the same language. If we do not know the author’s meaning for used words in the text, we primarily will “decode” the words according to our comprehension of them. So it could be a problem! In sentiment analysis, the same problem occurs when applying emotional lexicon to detect the emotions embedded in the text. Once one or more persons create the emotional lexicon, different emotional interpretation can be applied to the words, and other some other cannot be present in the lexicon. So, detecting emotions using an emotional lexicon as support is like a “pieces of different points of view” instead of author’s vision.

In this case study we present an approach for creating a personal emotional lexicon based on social media messages, using Natural Language Processing.

8.1 Related work

According to Dictionary.com, the term lexicon is defined as:

A. a wordbook or dictionary, especially of Greek, Latin, or Hebrew;

B. the vocabulary of a particular language, field, social class, person, etc.;

C. inventory or record.

There are several works of lexicon expansion for diversified objectives, and some are more relevant for the present paper as they have been used as a source for the idea proposed. In this section, we survey these inspiring works.

The idea of a personal emotional lexicon was inspired by the work of [118], which uses emotional labels to improve the authorship identification. This identification is made using social media posts from chosen personalities and an approach containing lexicon and machine learning approaches, where the usage of a personal lexicon of each author would increase the accuracy of the identification.

The work of [86] contributed to the idea of an unsupervised lexicon building method. Although this work handles only with polarities, the process of identifying individual words and share their polarities to other words is an essential issue in our approach.

On your hand, [15] inspired the use of texts from social media and the identification of their particularities, like hashtags, emoticons and neologisms.

8.2 Lexicon expansion process

The initial point for a lexicon expansion is to collect as many as possible texts from the authors, used to express their thoughts. Since people tend to express themselves differently, using specific words more or less frequently to designate affections, emotions, or even repudiation, the identification of this personal vocabulary will serve as a "fingerprint" of how the author expresses their perceptions.

For this reason, to create this "fingerprint" in our study, we collected comments from Twitter to create a personalised emotional lexicon which corresponds as the way as the author expresses. Twitter was elected as a source of information because differently than other social media because the comments are not restricted a topic, or comments about a post, so people tend to express more freely when there are no constraints in social media.

In our study, it was collected all published tweets from different authors. Due to space limitations, the analysis will present only the data for Donald Trump and Bill Gates.

The process of lexicon expansion, as presented in Figure 41, is composed of different steps, changing the the original and unstructured text in a format able to extract relevant information.

8.2.1 Corpus creation

After collecting tweets from the authors, all texts are processed in order to remove unusable information, remaining only the text message written. So, using the Stanford Core NLP [113] toolkit, the Part of Speech (POS) tagger identifies and removes all texts different than nouns, verbs, adverbs and adjectives. The remainder information is stored in a new file, hereafter called corpus.

8.2.2 Vocabulary creation

For this step, there are two main approaches: one hot encoding vectors - that creates vectors of 0's and 1's to represent the existence of words in a sentence - and word embeddings vectors - that takes in consideration the proximity of known keywords in a sentence. This study applied the word embeddings approach

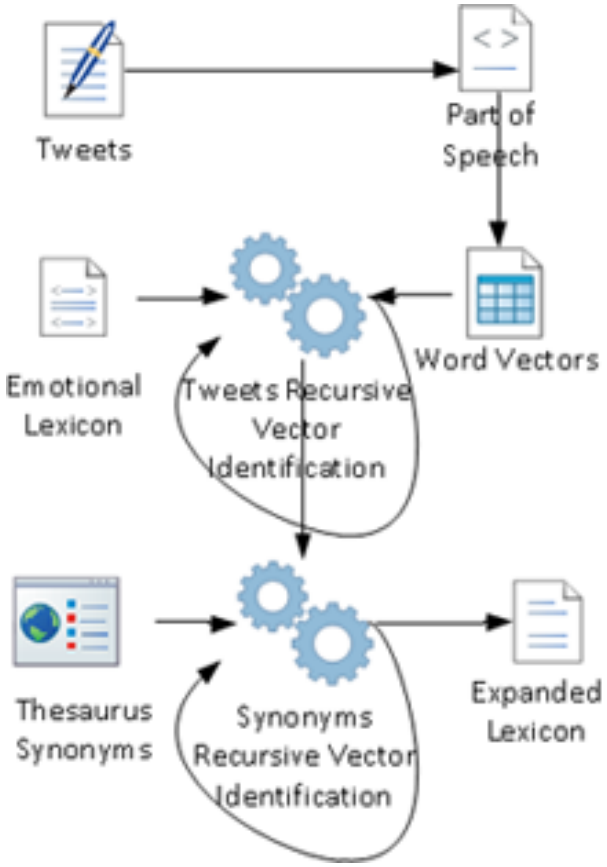


Figure 41: Lexicon expansion process

because according to [12], when the words in a corpus are distributed in a vector space - as word embeddings are - the similarity can be measured through the *Cosine Distance* between the words. Moreover, according to [90], has advantages in the identification of similar words, that is the core functionality in the lexicon expansion, that is the idea of our work.

For the word vector's creation, it was applied the Glove [154] using the corpus file of each author sources, resulting in 2 different lists of vectors, representing the authors' vocabulary.

8.2.3 Similarities

The next step is to analyse the similarity between words. Based on a set of known emotional seed words, the objective is to identify in the corpus the most similar words related to these seed emotional words. Once identified that words are similar, it is possible to claim that they have the same basic emotions values.

It is possible to find the same similar word related to different seed words (for example, "looking" is similar to seeing and seeming). In this case, "looking" must be disambiguated in order to have the emotions annotated correctly according to his meaning. However, it is not part of this work to handle with disambiguation process actually - it will be handled in future work -, so for the lexicon expansion process,

we consider only the similar word containing the higher cosine distance value.

In order to identify these similar words, we used them as emotional seed words the ones contained in EmoLex lexicon [133] while for the similar word's identification, it was created a process to iterate inside the lexicon and a recursive process to iterate into the authors word vectors' in order to detect the similar words. This recursivity allows to identify and associate deep levels of similar words higher than a predefined threshold - in our tests was applied 0.8 as a threshold -, based on the original corpus. If a word has a similarity higher than the threshold, it means that the similar word shares the same basic emotions values with the lexicon word. All identified similar words and basic emotions and their lexicon emotional words and their basic emotions are stored in order to input the next step, hereafter called "Similarities".

8.2.4 Synonyms

The next step detects the synonyms of each word in "Similarities". To reach this objective, all words in "Similarities" were analysed in the Wordnet [131] in order to identify all synonyms for the word. An important detail in this step is the attention with pre-existent words. Once the idea is detecting the emotions related to how the author expresses in a text, the most critical information is the words identified as similar. So, in the case of words identified as similar and synonyms, the synonym is discarded.

Like the previous step, after the synonyms identification was created a recursive process to iterate inside the synonyms and the author's word vectors' in order to detect the similar words between "Similarities" and synonyms. All identified similar words and their basic emotions and the synonym of the emotional words and their basic emotions are stored, resulting in the personal expanded lexicon.

8.3 Results

Once performed all process described in the previous sections, it was generated different lexicons, based on the author's vocabulary.

Using the original seed lexicon as a parameter, it is possible to compare the amount of information added for each author. Table 14 shows author's lexicon words increasing for each emotion and their respective rate when compared to the original lexicon.

Table 14: Lexicon comparison

		Total Words	Positive	Negative	Neutral	Anger	Anticipation	Disgust	Fear	Joy	Sadness	Surprise	Trust
	Original Lexicon	14182	2312	3324	8546	1247	839	1058	1476	689	1191	534	1231
Donald Trump Lexicon	**Words**	19194	3098	4638	11458	1679	1141	1549	1923	937	1510	750	1571
	Increasing Rate	35,34%	34,00%	39,53%	34,07%	34,64%	36,00%	46,41%	30,28%	35,99%	26,78%	40,45%	27,62%
Bill Gates Lexicon	**Words**	29360	5186	7461	16713	2901	1937	2482	3201	1616	2628	1237	2740
	Increasing Rate	107,02%	124,31%	124,46%	95,57%	132,64%	130,87%	134,59%	116,87%	134,54%	120,65%	131,65%	122,58%

Based on these pieces of information, according to Figure 43, Bill Gates lexicon has almost the same proportion of the basic emotions when compared to the original lexicon (2 of 8). On the other hand, Donald Trump lexicon presents proportional differences in 6 of 8 basic emotions when compared to the original lexicon. Also, as presented in Figure 42, Donald Trump lexicon has proportionally more positive and

negative words, and fewer neutral words, when compared to original and Bill Gates lexicon, demonstrating that the first author is blunter than the other author, raising more emotions in their speeches.

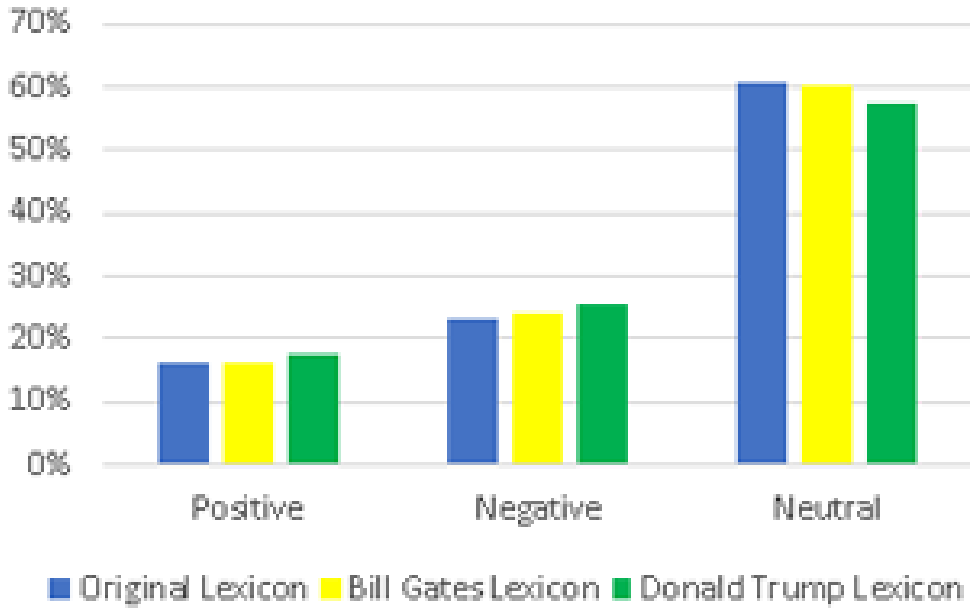


Figure 42: Lexicon polarities

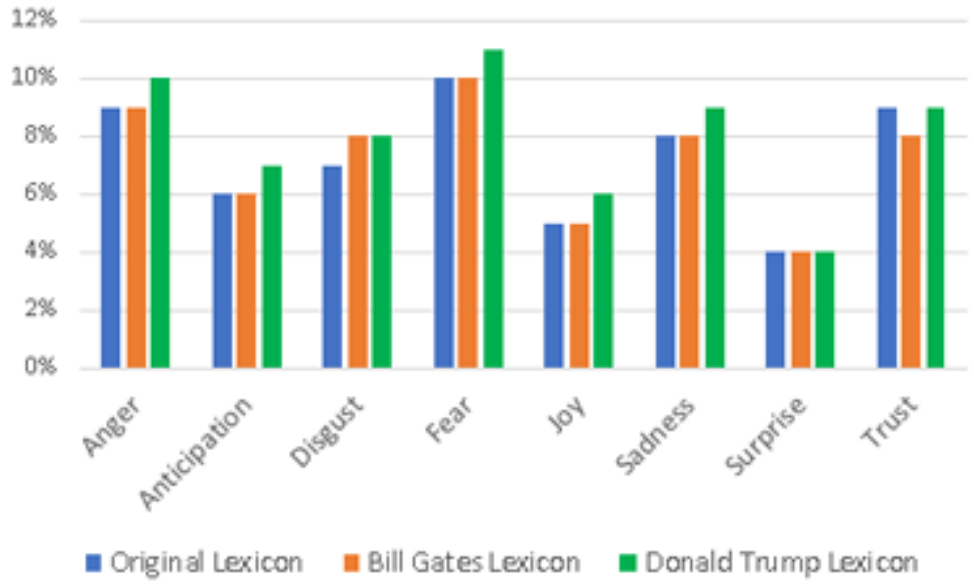


Figure 43: Proportions

Furthermore, different personal lexicon can represent an opportunity of interpreting texts as the lexicon author could interpret them. In order to perform this analysis, we choose two different texts from the authors: the Bill Gates' World Economic Forum speech (2008) and Donald Trump's United Nations speech (2017).

Each text was analysed using three different lexicons: original lexicon, Bill Gates lexicon and Donald Trump lexicon, in order to identify the existent words in the texts and their related emotions existent in each lexicon. Tables 15 and 16 present the proportion of each emotion in Bill Gates and Donald Trump speeches respectively.

An impressive result is that when we use a lexicon from another author to analyse the text, the sentiments *Anger, Fear* and *Sadness* are lower than the values obtained when using the author's lexicon. On the other hand, the *Joy* sentiment is higher in texts from authors different than analyses. In this case, it can be interpreted as the author can be more "complacent" with others and more "stern" with himself.

Table 15: Bill Gates' speech analysis

Emotion	Original	Donald Trump	Bill Gates
Anger	5.57%	4.09%	11.47%
Anticipation	18.38%	22.80%	13.09%
Disgust	4.18%	5.59%	6.48%
Fear	10.58%	7.31%	10.39%
Joy	17.55%	18.92%	12.28%
Sadness	6.41%	6.24%	12.42%
Surprise	7.52%	7.31%	5.94%
Trust	29.81%	27.74%	27.94%

Table 16: Donald Trump speech analysis

Emotion	Original	Donald Trump	Bill Gates
Anger	10.36%	11.44%	8.51%
Anticipation	15.00%	11.90%	12.71%
Disgust	6.18%	7.28%	7.01%
Fear	15.82%	13.49%	12.71%
Joy	12.82%	11.57%	17.52%
Sadness	9.18%	9.19%	7.71%
Surprise	4.73%	5.95%	5.31%
Trust	25.91%	29.17%	28.53%

8.4 Conclusion

This work presents an approach to create a personal lexicon based on the expansion of a seed lexicon through social media texts. This personal lexicon contains the vocabulary used for each author and the emotions associated with each word, according to what the author wanted to transmit. This solution decreases the problem of misunderstandings when interpreting texts because it helps to know the emotions that the author wanted to transmit in their text. Once the interpretation of emotions is mainly personal, knowing the word's meaning according to each author helps to interpret the text according to the author's point of view. Moreover, new expressions - even hashtags - and local expressions raise quickly, and "translating" its emotional meaning takes time when compared to traditional emotional lexicons.

The word's increasing of 35.34% and 107.02% of each personal lexicon, when compared to the original lexicon, shows that this solution can be used in sentiment analysis processes because it increases the possibility of text interpretation. Regarding Natural Language Processing, it contributes to provide a customised service to the end user, enabling to avoid misunderstandings when interpreting texts, analysing sentiments in an individual scale and increasing the level of accuracy about recommendations, based on their characteristics and personality.

9

Case Study 3 - Impact of emotions in usual tasks

In any kind of relationship, a golden rule to avoid problems is not taking decisions under emotional pressure. There are several strategies to do this: from counting to ten before responding an unpolished message until taking a break to "refresh the mind" before the decision.

However, there are situations, such as tests, where it is impossible to avoid emotional pressure and its consequences. When in a stressful situation, or under strong emotional conditions, people tend to make mistakes more frequently. This situation happens in any profession, and so being able to predict these errors that are consequences of emotional states is an important approach to plan a strategy to decrease or avoid them. For example, how important would it be for transportation companies to know the drivers' emotional state before travelling, aiming to reduce the risk of accidents? How important would be for a hospital to predict the errors of a doctor, based on his emotions?

Errors differ from profession to profession and also the effect of emotions over the work is different. Different data sets must be collected to identify these errors, and to correlate them with the emotional state of the worker.

As writers usually express their emotions in the texts they produce through the bag of words they use in each situations, and the typing errors they do along an editing session ca be measured, we intend to model the relation between emotions and errors, using the computer as a case of study. The purpose of the study here reported is to analyse a big collection of texts annotated with editing data to demonstrate that the relation between errors and emotions can be identified, and quantified in order to predict undesired situations.

In this paper, we present an approach using Sentiment Analysis and Machine Learning to characterize the impact of emotions in the number of errors during a typing process. After training the model, it will be used to predict new cases in order to assess it.

It is not our intention to claim that this approach is an alternative for predicting errors in all situations, however, we think that the approach will lead further future investigation in this relevant topic.

9.1 Theory of Basic Emotions

Basic emotion theorists explain that every human emotion is composed of a set of discrete basic emotions [36, 80, 156].

Many researchers have identified some basic universal emotions. One of the first attempts is a study by [40] which concluded that there are six basic emotions are *Dislike*, *Happiness*, *Sadness*, *Anger*, *Fear* and *Surprise*. His work is based on the theory that human faces can represent this basic emotion as universal pictures.

For [156], every sentiment is composed of a set of 8 basic emotions: *Anger*, *Anticipation*, *Disgust*, *Fear*, *Joy*, *Sadness*, *Surprise* and *Trust*, represented as a "wheel of emotions". Furthermore, the combination of basic emotions results in *dyads*. Plutchik created rules for building the *dyads*, defining the primary dyads emotions as the sum of two adjacent basic emotions, as *Optimism* = *Anticipation* + *Joy*. Meanwhile, secondary dyads emotions are composed of emotions that are one step apart on the "emotion wheel", as *Unbelief* = *Surprise* + *Disgust*. The tertiary emotions are generated from emotions that are two steps apart on the wheel, as *Outrage* = *Surprise* + *Anger*.

Other well-known emotions model is the Five Factor Model (as known as Big Five), introduced by [124] which suggests that the personality is composed of 5 independent factors:

A. **Openness to experience -** People with high scores like news and tend to be creative. At the other end of the scale are the conventional and orderly, those who like the routine and have a keen sense of right and wrong;

B. **Conscientiousness -** It measures the level of concentration. Those with high scores are highly motivated, disciplined, committed and trustworthy. Those with low results are undisciplined and easily distracted;

C. **Extroversion -** It measures the sense of well-being, the level of energy, and the ability in interpersonal relationships. High scores mean affability, sociability, and ability to impose oneself. Lows indicate introversion, reservation, and submission;

D. **Agreeableness -** It refers to how we relate to others. Many points indicate a compassionate, friendly and warm person. At the other end are the withdrawn, critical and egocentric;

E. **Neuroticism -** It measures emotional instability. People with high scores on this scale are anxious, inhibited, melancholic and have low self-esteem. Those that get low scores are easy to deal with, optimistic and well-liked with themselves.

In this case study, we adopt the Plutchik's model to represent emotions because we consider more realistic, easy to use and this model allows to represent several different emotions through dyads emotion. Moreover, there are some libraries and lexicons used in this work which represent and process the emotions according to this model.

9.2 Related work

There are several works using sentiment analysis for diversified objectives, and some are more relevant for the present paper as they have been used as a source for the idea proposed. In this section, we survey these inspiring works. However, predicting typing errors from an emotional analysis is an unexplored field and we have not found specific previous works to reference.

The usage of emotional labels for predictions was inspired by the work of [118], which uses emotional labels to improve the authorship identification. This is made using Facebook posts from personalities known and a hybrid approach containing lexicon and machine learning approaches.

Moreover, [193] have presented a work to extract sentiment strength from the informal English text, using new methods to exploit the de facto grammars and spelling styles of cyberspace, which contributed with the idea of extract sentiment polarities from text.

Finally, the work presented by [125] contributed to the idea of predicting writing performance using affective variables to relate to efficacy expectations.

9.3 Data creation

In order to analyse the impact of the emotions during the text writing process, it was necessary to analyse texts containing emotional load and meta-information about its creation. For this purpose, the dataset provided by [10] containing keystroke logs for opinion texts about gun control[1] was used as the basis for a new dataset creation. To perform the intended analysis we actually needed a text repository with emotional load and editing numerical data for each written piece. The option of a keystroke log is justified by the necessity of gather information about the text creation process. So, in our study all texts are considered written "from beginning to end", i.e., the first typing step without a posterior text revision phase. This is important for levelling possible errors and editing in a same identifiable pattern.

The process of the new datasets creation is performed in 2 steps: Meta-information Creation and Emotional Analysis, as presented in Figure 44.

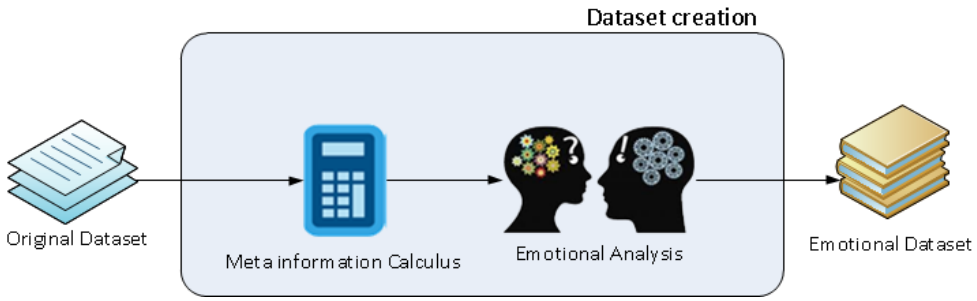


Figure 44: Dataset creation process

[1] This is a hot, sensible, topic provoking emotive reactions on commenters.

9.3.1 Meta-information

The Meta-information Creation step analyses the keystroke log and calculates metrics about the text creation. For analysis purposes, we defined some metrics considered important in this study. These metrics are:

- **TimeText** - It represents the time spent during the text writing. It is the amount of time span in milliseconds between press and release for each character and white space key in the keyboard. Punctuations, numbers, and others are discarded;
- **AveragePerWord** - Is the average between the total words in the text and the time spent during typing process;
- **AmountErrors** - It is the number of errors during the typing process. It is important to emphasize that due to dataset limitations that do not store mouse movements or selections, it is impossible to detect all forms of removing characters (for example, single character or block removing). For convenience of this study, it is considered an *error* each *backspace occurrence*[2];
- **AverageErrors** - It is the average of the total words in the text and the number of errors during typing process;
- **TimeBetweenKeys** - It is the average time span between the keyboard press;
- **RepeatedCharacterFrequence** - It is the frequency of a character is repeated in the text. A repeated character is considered the same character those have been pressed immediately before and the time between them is at least 15% lower than TimeBetweenKeys. This is important to detect situations when a key is pressed for a long time, repeating the character.

9.3.2 Emotional Analysis

The Emotional Analysis step is responsible for identification of each basic emotion according to Plutchik's model [156]. This model was chosen because there are many libraries to process information according to it and lexicons which contain the basic emotions for each word. To achieve this objective, all sentences are analysed using the EmoLex [133] lexicon.

For this analysis, all texts have been submitted to a preprocessing pipeline; at the end of this phase, only the relevant information remained.

This pipeline was composed of n-Gram identification, tokenization, stopwords removal, part of speech tagging and named entity removal, as presented in Figure 45.

[2]We are aware that counting backspaces is not the more adequate way to count errors, because other reasons can lead the writer to backspace and delete characters, and also many errors are made without being detected and corrected. Anyway we feel that there is a clear relation between both.

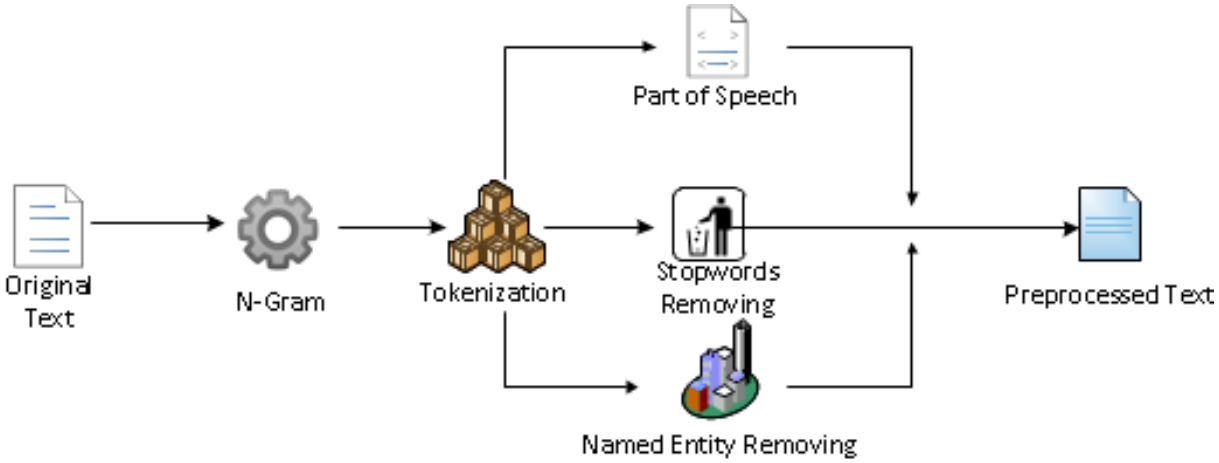


Figure 45: Preprocessing pipeline

Using the Stanford Core NLP toolkit [113] for these tasks, the preprocessing is divided into 3 parallel tasks. This is important because both Part of Speech Tagging and Named Entity Recognition need the text in the original format in order to identify the information.

The preprocessing begins with the N-Gram identification, where a predefined set of n-grams are identified in the text and labelled to be interpreted as a single word. Later, the tokenizer splits the text in a list of words (tokens) and these tokens are syntactically analysed in Part of Speech, where the nouns, verbs, adverbs, and adjectives are identified and stored for future purposes. In parallel, the tokens identified in a predefined stopwords list are removed and the tokens in named entity process are analysed in order to identify names (persons, locations or organizations) and discard them.

Later, the common tokens in these processes are stored and the emotions from each preprocessed text are identified through a process in R which queries the EmoLex lexicon [133] and identifies the basic emotions using the Syuzhet package [84].

Finally, all information is stored in a new dataset containing the opinion and preprocessed text (from the original dataset), the 6 metrics created in Meta-Information Creation step, 8 basic emotions percentages and 2 polarities identified in the Emotional Analysis step.

9.4 Data analysis

The objective of this analysis is to find some evidence that emotions influence the writing process. In order to achieve this objective, some experiments were performed to relate emotions and writing patterns.

9.4.1 Emotional correlations

As initial step, the values for some Plutchik's basic emotions and defined dyad emotions [156] were calculated in order to provide more sources of information to analyse the data. To calculate these emotions,

we used the package Syuzhet in R, which analyses the text provided and returns the values of each basic emotion contained into the text, according to the EmoLex lexicon [133]. The dyad emotions were calculated according to the formula below:

- Optimism = Anticipation + Joy;
- Disapproval = Surprise + Sadness;
- Hope = Anticipation + Trust;
- Unbelief = Surprise + Disgust;
- Anxiety = Anticipation + Fear;
- Outrage = Surprise + Anger;
- Love = Joy + Trust;
- Remorse = Sadness + Disgust;
- Guilt= Joy + Fear;
- Delight = Joy + Surprise;
- Pessimism = Sadness + Anticipation;
- Curiosity = Trust + Surprise;
- Awe = Fear + Surprise;
- Despair = Fear + Sadness;
- Pride = Anger + Joy;
- Shame = Fear + Disgust;

Later, the Pearson correlation (r^2) was applied to both each basic emotion and dyad emotions, to obtaining the correlation between the number of backspaces in the writing process[3] (*AmountErrors* as presented in subsection 9.3.1) and the emotions, according to Table 17.

Table 17: Correlations between *AmountErrors* and emotions

Emotion	r^2	**Emotion**	r^2	**Emotion**	r^2	**Emotion**	r^2
Anger	0.35	Optimism	0.30	Pessimism	0.42	Trust	0.36
Anticipation	0.31	Hope	0.39	Awe	0.38	Curiosity	0.37
Disgust	0.26	Anxiety	0.52	Despair	0.38	Pride	0.38
Fear	0.38	Love	0.35	Shame	0.38	Surprise	0.19
Joy	0.23	Guilt	0.41	Disapproval	0.30	Remorse	0.31
Sadness	0.30	Delight	0.25	Unbelief	0.27	Outrage	0.34

Despite no single emotion having a strong correlation, it is possible to identify that among all emotions analysed, anxiety is the most relevant for the number of errors during typing, having a moderate correlation.

9.4.2 Machine learning predictions

A machine learning analysis was applied to determine the influence of the meta-information and emotional labels on the number of errors prediction. For this purpose, 5 different scenarios were considered:

- Scenario A - Only text and opinion - no meta-information neither emotional labels;
- Scenario B - Only meta-information and emotional labels;

[3] Remember that we are using this measure to compute the errors number.

- Scenario C - Only meta-information;
- Scenario D - Only emotional labels;
- Scenario E - All dataset information - text, opinion, meta-information and emotional labels.

The dataset used for both training and validation is the same created in subsection 9.3.2. In this analysis, the meta-information *AverageErrors* was discarded because it has strong correlation with *AmountErrors*, induced by $AmountErrors \approx AverageErrors * TimeText$.

For each scenario, the following machine learning algorithms were applied: Linear Regression, SVM, Random Forest and Decision Table.

All tests were performed using a 10-fold cross-validation in Weka; the correlation coefficients obtained with each algorithm between AmountErrors and the dimensions analysed in each scenario are shown in Table 18.

Table 18: Algorithms correlations and their Root Mean Squared Error(RMSE)

Scenario	**Linear Regression**	**RMSE**	**SVM**	**RMSE**	**Random Forest**	**RMSE**	**Decision Table**	**RMSE**
Scenario A	0.171	391.28	0.347	208.62	0.429	145.36	0.321	104.78
Scenario B	0.774	99.62	0.769	103.32	0.791	96.67	0.739	106.37
Scenario C	0.777	99.14	0.770	103.08	0.780	98.66	0.739	106.37
Scenario D	0.525	134.10	0.528	137.73	0.492	141.70	0.426	143.54
Scenario E	0.320	437.12	0.696	131.79	0.586	127.88	0.591	129.18

After the tests, the best correlation coefficient for predicting the number of errors was obtained with the Random Forest algorithm for Scenario B (only meta-information and emotional labels).

Once identified that the meta-information and emotional labels are an adequate to predict the number of errors, the next step was to identify the patterns for these predictions. For this purpose, all values were discretized into 4 groups and a K-Means algorithm was used to cluster the data (meta-information and emotional labels) into 4 different clusters representing respectively 39%, 17%, 23% and 21% of the information available.

In a preliminary analysis, all meta-information was identified with the same value range, and for this reason, it was considered irrelevant for this objective and removed from the visualization. Also, as the purpose of this analysis is to measure the emotional influence in the *AmountErrors* values, the dimensions *Positive* and *Negative* were removed too. Then the relevant information remaining was grouped by *AmountErrors* and is presented in Table 19, where the range in *AmountErros* refers to the number of errors identified, while the range for each emotion refers to the number of words containing the emotion in the text.

Having in mind the results in Table 19, it is possible to conclude that in general, as higher the emotions are, higher is the impact on the *AmountErrors* number.

Table 19: Ranges of *AmountErrors* per emotions

	Emotions							
AmountErrors	**Anger**	**Anticipation**	**Disgust**	**Fear**	**Joy**	**Sadness**	**Surprise**	**Trust**
0.0-204.5	2.5-3.5	0.5-1.5	0.5-1.5	0.0-4.5	0.0-0.5	0.0-1.5	0.0-0.5	2.5-4.5
204.5-256.5	3.5-5.5	1.5-2.5	0.0-0.5	5.5-7.5	1.5-2.5	2.5-3.5	0.5-1.5	4.5-∞
256.5-340	3.5-5.5	0.5-1.5	1.5-2.5	5.5-7.5	0.5-1.5	3.5-∞	0.0-1	2.5-4.5
340-∞	5.5-∞	2.5-∞	2.5-∞	7.5-∞	2.5-∞	3.5-∞	2.5-∞	4.5-∞

9.5 Conclusion

A typist certainly will have fewer errors than a normal person when typing. However, even this typist will do more mistakes if he is, for instance, anxious or feeling guilty.

As a first step in a research direction we want to further explore — *the sentiment analysis in different tasks to understand the effect of the worker's emotional state on his performance, to reduce mistakes* — this paper presents a combination of lexicon-based and machine learning approaches to correlate the number of typing errors based on the emotional labels and metrics associated with the text creation (text first typing/editing). That model can be used to predict errors based on the information of the emotional state; in this way we will have a rigorous criterion to recommend people to stop doing some task under some personal states to avoid dangerous faults.

Everyone has particular characteristics of expressing himself and these personal characteristics can of course influence that prediction. Maybe on account of that, the study results so far obtained were a bit surprising, and the measured influence of emotions on users tendency to make mistakes is *moderate* (we were expecting bigger values). In our tests, the best approaches for predicting errors based on human behaviour were obtained using emotional information (emotions inferred from the text lexical analysis)and meta-information (metrics evaluated based on the text creation process) collected during text typing. Clustering the data revealed how the emotions can affect the number of errors. It is a promising result.

10

Case study 4 - Determining emotional profiles based in textual analysis

Probably, one of the well known and used proverbs is: “Birds of a feather, flock together”. However, what does it mean? In general meaning, it refers that people with common traits, interests and tastes tend to associate and relate with each other, in the same way as birds of the same species flock together. It can be observed in several different human behaviours, where people with common personalities tend to relate to each other.

Psychodynamic researchers claim that personality structure is set in childhood. For [175], the individual personality is formed around 2 or 3 years old, mostly through child training practices. [55] argues that when the Oedipal complex is resolved, all basic structures of personality - the id, ego, and superego - are fully developed in opposition to [42] and [108], which believe that personality continues to develop later in life. Sharing the same vision of Erikson and Loevinger, the motivational speaker [166] claimed that “you are the average of the five people you spend the most time with”.

Through social media usage - in general microblogging - people (authors) can express their opinion, desires and thoughts to a broad audience - from friends to unknown followers - keeping proximity despite physical distance. However, is this audience interested in the author’s posts because they share the same sentiment, mood or emotions? Also, since software has no childhood, neither id, ego, and superego, is it possible to create an emotional profile based on existing ones, enabling the software “learn” how to have a personality?

In this case study, we present an approach for emotional profile creation based on existent emotional profiles, using emotion-based analysis to determine the proximity of the author’s emotional and grammatical writing style with their audience on microblogging.

10.1 Emotion theories

In the literature, there are several models that attempt to explain the emergence of emotions and their associated behaviours. The main research theories here surveyed to serve as background to our analytical work are discrete, dimensional and appraisal theories.

Discrete emotional theories propose the existence of basic emotions that are universally displayed and recognized, grouped into categories and independent. An example of discrete emotional theory is proposed by [157], where all sentiment is composed of a set of 8 basic emotions (*anger*, *anticipation*, *disgust*, *fear*, *joy*, *sadness*, *surprise* and *trust*).

On the other hand, **dimensional theories** characterize emotions regarding two or three dimensions, generally "arousal" and "valence." Valence is related to a positive or negative evaluation and is associated with the feeling state of pleasure (vs displeasure). Arousal reflects the general degree of intensity felt. However, using this two-dimensional is confusing to different emotions that share the same values of valence and arousal, as *anger* and *fear*. For this reason, it is common to add a third dimension to support this differentiation, as intensity. According to [101] "the third view emphasises the distinct component of emotions, and is often termed the componential view."

Emotional-cognitive psychologists focus their studies mainly on the **appraisal process**. According to [179], the central idea is that emotions are triggered and differentiated by subjective analysis of an event, situation or object. For instance, Bill and Mike are watching a football game where their teams are playing. Bill's favourite team wins (event). Mike's appraisal is that an undesirable event happened. For Bill, the appraisal is that the event is desirable. So, the same event has produced opposite appraisals. In fact, emotions are triggered by the personal interpretation of the annoying or cheerful aspects of an event, the appraisal.

10.2 Related work

Due to the extensive usage, sentiment analysis on microblogs can be considered an opinion-rich resource and has been gaining popularity and attracting researchers from other areas to correlate information about specific events (e.g. Christmas, football matches, elections) with the sentiment contained in posts.

To perform sentiment analysis on microblogging, according to [150], a straightforward approach is to exploit traditional sentiment analysis models. However, such methods are inefficient because they ignore some unique characteristics of microblog's data, as emoticons representations. Moreover, there are lots of colloquial terms, abbreviations and misspelt words used in microblogs which leads to heavy preprocessing tasks in order to identify its occurrences and "translate" them to a canonical form to be interpreted correctly. Due to such properties, several models have been developed especially for microblogs sentiment analysis recently.

An example of correlations between events and sentiments was proposed by [75], which measures the sentiments on Twitter during a period and compares the correlation between sentiments contained in the text and significant events, including the stock market, elections and Thanksgiving. Also, [91] examined a dataset containing tweets about Michael Jackson's death in order to analyse how emotion is expressed on Twitter. [141] have analysed the sentiments about politicians, detecting a strong correlation between the aggregated sentiment and manually collected poll ratings.

[76] predict the individual well-being, as measured by a life satisfaction scale, through the language people used on social media. This is made using randomly selected posts from Facebook and a lexicon-based approach to identify the text words polarities.

A different approach of sentiment analysis using Twitter posts was presented by [148], which consists of a linguistic analysis of the collected corpus to build a sentiment classifier. This classifier can determine positive, negative and neutral sentiments for a document.

[61] proposed a framework which interprets the emoticons in tweets as noisy labels using supervised learning. However, as [104] describes, there are some disadvantages when using only emoticons as noisy labels. A reason for this is because it is difficult to collect a large number of tweets with emoticons because they are time-related, dynamic and region-related. For [110], "usually we can only exploit topic-independent tweets with emoticons. That is to say, in topic-dependent datasets which focus on one given topic, the performance boost brought by emoticons is not significant enough. Besides emoticons, rich topic-dependent unlabelled data can be exploited better."

Despite vast works about sentiment analysis in microblogs, none concerns on the study of the relationship between emotional profile and the writing style similarities among authors and their audiences.

10.3 Data analysis

In order to analyse the correlation between author's emotional writing style and their audience, we collected 2500 recent Twitter tweets from 6 different aleatory authors from different areas, as presented in Table 20:

Table 20: Tweets authors

Author	Area	Origin
Elon Musk	Business	Press office
Katy Perry	Entertainment	Press office
Donald Trump	Business	By himself
Alan Shipnuck	News	By himself
Michele Dauber	Education	By herself
Floyd Mayweather	Sports	Press office

Although it is clear that three authors do not post in Twitter - i.e. it is a press office representing them - the idea is of this paper is analyse the emotional, grammatical and textual proximity between authors and audiences, even if an author and/or an audience is a press office. In a different point of view, it can highlight an "press office emotional style", which can inform even where the conversation has occurred, as presented by [117].

All tweets were gathered using the package TwitteR [58] for R [161]. Additionally, the tweets were labelled with an annotation indicating if the message was produced by a press office or by the author

himself. During the gathering processes, we considered only the tweets and discarded the re-tweets. This decision wad adopted to avoid that texts from other author, like digital influencers or unknown viral texts, biased the individual analysis.

The task of analysing the emotional profile can be split into some intermediate steps: first, it was necessary for some preprocessing tasks in order to reduce data size by removing unnecessary text from the original message; Later, the relevant remaining text was analysed in order to evidence the author's polarities and the author's emotional style.

10.3.1 Preprocessing

Preprocessing is a data mining technique that involves transforming raw data into an intelligible form. In the literature, several preprocessing techniques are available to extract information from text, and their usage is according to the characteristics of the information desired.

In our analysis, the preprocessing pipeline begins with tokenization and in subsequent starts three parallel jobs, as shown in Fig. 46: Part of Speech Tagging (POS-T), Named Entity Recognition (NER) and Stopwords Removal. This strategy was used because both POS-T and NER need the text in the original format, in order to return the correct data from the analysis.

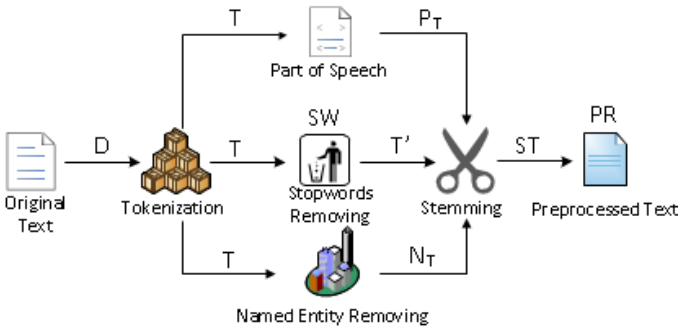


Figure 46: Preprocessing tasks

The POS-T process identifies the text grammatical structure and preserves nouns, verbs, adverbs and adjectives. The reason for this approach is because only these grammatical categories can bring emotional information. In a formal description, the Tokenization process converts the original text D in a set of tokens $T = \{t_1, t_2, ..., t_n\}$ where each element contained in T is part of the original document D. Later, the POS-T labels each token with semantic information and the process keeps all nouns, verbs, adverbs and adjectives in a set set P_T, where $P_T = \{p_{T_1}, p_{T_2}, ..., p_{T_k}\}$ and 0 ≤ k ≤ n and $P_T \subset T$.

Like POS-T, NER process identifies names in 3 different categories: "Location", "Person" and "Organization" and removes all tokens related with these categories. As a result, a set $N_T = \{n_{(T_1)}, n_{(T_2)}, ..., n_{(T_j)}\}$ is constructed based on identified word category where $\forall j, cat(N_j) = "O"$. This step is important to be done in parallel with POS-T because some locations can be confused with some grammatical structure (as Long Beach or Crystal Lake, for instance).

The Stopwords list is a predefined set $SW = \{sw_1, sw_2, ...sw_y\}$ of words, available in R through the package **tm** [126]. This step will return a set $T' = t'_1, t'_2, ..., t'_n$, where $T' \cap SW = \emptyset$.

After the 3 preprocessing tasks finish, the outcoming set ST is defined as $ST = T' \cap P_T \cap N_T$.

Later, a stemming algorithm is responsible for obtaining the stem of a word. For this task, we adopted an implementation of the Lovins stemmer [109], resulting in a set of stemmed words $PR = \{ST_1, ST_2, ..., ST_z\}$ ready to be analysed.

For all three tasks - POS-T, NER and Tokenization - the Stanford Core NLP [113] toolkit was used. An example of how the steps change the information was previously presented in Fig. 38.

10.3.2 Polarity analysis

In order to determine the author's polarity style, after the preprocessing all sentences contained in PR were compared against EmoLex lexicon [133] in order to identify the positive and negative sentiment of the entire text. Later, it was collected and analysed tweets from the top 5 most contacted audience from the author, in order to analyse the proximity of their polarities tweets and author, as presented in Table 21.

Table 21: Polarities analysis from authors and their top 5 audience

	Author		**Audience's average**	
	Positive	**Negative**	**Positive**	**Negative**
Elon Musk	0,68	0,26	0,68	0,28
Donald Trump	1,13	0,76	1,05	0,57
Katy Perry	1,01	0,27	0,59	0,18
Alan Shipnuck	0,45	0,27	0,57	0,38
Michele Dauber	0,68	0,65	0,83	0,70
Floyd Mayweather	0,62	0,13	0,61	0,27

When applying the Pearson's correlation coefficient (r^2) between polarities authors and their respective top audience give $r^2 = 1$ for all results, indicating a **very strong** correlation between author's polarities and top audience's polarities.

Another analysis made was creating the sets:

$A = \{a1, ..., a6\}$ of authors,

$AP = \{ap_{a_1}, an_{a_1}, ..., ap_{a_6}, an_{a_6}\}$ of polarities where ap is the author positive polarity, an is the author negative polarity,

$C_A = \{c_{a_1,1}, ..., c_{a_i,j}\}$ of author's topmost contacts, where $0 \leq i \leq 6$ and $1 \leq j \leq 5$,

$CP = \{\overline{cp_i}, \overline{cn_i}\}$ of polarities, where $\overline{cp}$ is the average of audience's positive polarities, $\overline{cn}$ is the average of audience's negative polarities and $0 \leq i \leq 6$.

When applying the correlation coefficient between AP and CP, the result is $r^2 = 0,85$, indicating a strong correlation between authors' polarities their audiences polarities.

10.3.3 Emotional analysis

In order to analyse the emotions contained into the text, it was used a lexicon-based approach provided by Syuzhet package in R, in order to identify the emotions contained in text according to the model proposed by [157], where all sentiment is composed of a set of 8 basic emotions (*anger*, *anticipation*, *disgust*, *fear*, *joy*, *sadness*, *surprise* and *trust*).

After all author's tweets analysis, the distribution of each basic emotion introduces a specific emotional profile for each author, defined as "emotional writing style", which is presented Table 22.

Using this information, the next step was to determine the average of each basic emotion for the audience's author. To achieve this objective, we used the same strategy used for the polarities, resulting in the audience's emotional writing style, according to Table 23.

Table 22: Basic emotions per author

Author	Anger	Anticipation	Disgust	Fear	Joy	Sadness	Surprise	Trust
Elon Musk	0,11	0,34	0,06	0,14	0,23	0,10	0,12	0,38
Donald Trump	0,34	0,53	0,20	0,36	0,45	0,40	0,33	0,83
Katy Perry	0,12	0,52	0,04	0,13	0,67	0,16	0,25	0,43
Alan Shipnuck	0,11	0,20	0,09	0,14	0,20	0,15	0,11	0,26
Michele Dauber	0,37	0,32	0,23	0,34	0,20	0,31	0,10	0,48
Floyd Mayweather	0,11	0,44	0,02	0,10	0,43	0,08	0,18	0,37

Table 23: Basic emotion's frequency average of audience's author

Average audience of	Anger	Anticipation	Disgust	Fear	Joy	Sadness	Surprise	Trust
Elon Musk	0,13	0,38	0,08	0,20	0,20	0,13	0,13	0,30
Donald Trump	0,25	0,48	0,12	0,36	0,40	0,34	0,33	0,80
Katy Perry	0,10	0,35	0,05	0,09	0,42	0,11	0,18	0,34
Alan Shipnuck	0,18	0,31	0,13	0,22	0,25	0,21	0,18	0,33
Michele Dauber	0,45	0,36	0,26	0,53	0,22	0,36	0,17	0,64
Floyd Mayweather	0,19	0,34	0,08	0,14	0,35	0,12	0,16	0,33

Hence, when applying the Pearson's correlation coefficient (r^2) between basic emotions from authors and the average of their audiences, it is possible to verify that in significant part they are strongly correlated, as presented in Table 24.

Table 24: Correlation between basic emotion's authors and frequency average basic emotion's top audiences

Author	Correlation	Author	Correlation
Elon Musk	0,93	Donald Trump	0,99
Katy Perry	0,99	Alan Shipnuck	0,96
Michele Dauber	0,95	Floyd Mayweather	0,98

Moreover, in order to establish Jim Rohn's statement, the analysis was expanded to verify the emotional profile of 100 most frequent contacts from each author. According to Fig. 47, the correlation between

authors' emotions and most frequent contacts emotions' average decreases when the number of contacts increases, supporting that "you are the average of 5 people you spend the most time with."

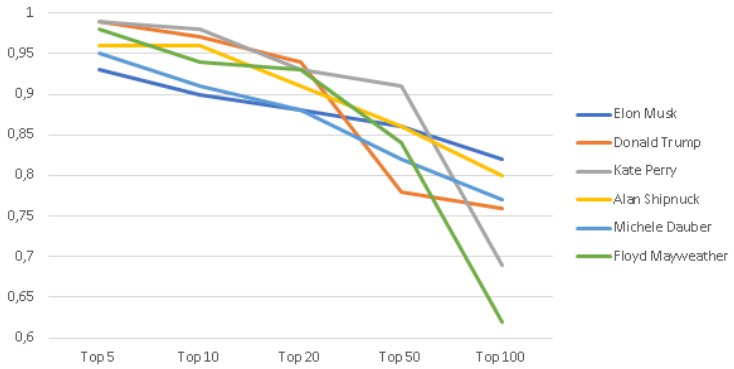


Figure 47: Correlations by author

A new point identified during the analysis, as showed in Table 25, is that the correlations tend to be higher within authors from the same origin, indicating a "press office emotional pattern."

Table 25: Correlation between basic emotion's authors

	Elon Musk	**Donald Trump**	**Katy Perry**	**Alan Shipnuck**	**Michele Dauber**	**Floyd Mayweather**
Elon Musk	1,00	0,92	0,77	0,94	0,48	0,89
Donald Trump	0,92	1,00	0,61	0,95	0,64	0,71
Katy Perry	0,77	0,61	1,00	0,79	-0,04	0,97
Alan Shipnuck	0,94	0,95	0,79	1,00	0,49	0,84
Michele Dauber	0,48	0,64	-0,04	0,49	1,00	0,11
Floyd Mayweather	0,89	0,71	0,97	0,84	0,11	1,00

10.3.4 Grammatical analysis

Another approach used was to determine if both authors and audiences share the same grammatical style when writing. For grammatical style, we understand the distribution of grammatical categories of words in sentences. To achieve this objective, both authors and audiences had their tweets labelled according to Part of Speech Penn Treebank [115] tags using Stanford Core NLP [113]. The next step was to determine the average of each Part of Speech tag for each author and their audience, resulting in the grammatical style, according to Table 26.

Hence, when applying the Pearson's correlation coefficient (r^2) between the grammatical style of authors and their audience, it is possible to verify that they are strongly correlated, as presented in Table 27.

10.3.5 Similarity analysis

The objective of the similarity analysis is to quantify the level of similarity of the author's texts and their respective audiences' texts. For this analysis, we collected the last 1000 tweets for each author and the

Table 26: Grammatical style for authors and audiences

	Part of Speech	Elon Musk	Donald Trump	Kate Perry	Alan Shipnuck	Michele Dauber	Floyd Mayweather
Author	CC	0,38	0,61	0,48	0,24	0,46	0,26
	CD	0,27	0,28	0,2	0,12	0,15	0,48
	DT	0,91	1,66	1,08	0,96	1,32	0,68
	IN	1,12	2,07	1,37	0,87	1,39	0,9
	JJ	1,05	1,39	1,09	0,83	1,11	0,63
	JJR	0,06	0,07	0,04	0,02	0,06	0,02
	JJS	0,06	0,07	0,03	0,03	0,01	0,04
	MD	0,28	0,31	0,1	0,15	0,21	0,05
	NN	2,94	3,18	3,58	2,56	3,73	2,72
	NNS	0,61	0,96	0,96	0,48	0,69	0,39
	POS	0,03	0,05	0,06	0,08	0,07	0,02
	RB	0,94	0,93	0,57	0,65	0,92	0,29
	RP	0,03	0,06	0,03	0,04	0,02	0,03
	TO	0,3	0,58	0,22	0,18	0,38	0,29
	VB	0,64	0,89	0,48	0,43	0,77	0,66
	VBD	0,18	0,4	0,25	0,25	0,46	0,12
	VBG	0,25	0,51	0,54	0,19	0,4	0,18
	VBN	0,21	0,35	0,12	0,12	0,26	0,05
	VBP	0,31	0,43	0,69	0,31	0,55	0,23
	VBZ	0,38	0,47	0,49	0,37	0,55	0,2
	WP	0,04	0,12	0,14	0,03	0,06	0,02
	WRB	0,03	0,05	0,06	0,04	0,11	0,03
Audience	CC	0,29	0,35	0,15	0,26	0,35	0,25
	CD	0,46	0,23	0,25	0,38	0,25	0,31
	DT	0,95	1,21	0,66	1	1,39	0,93
	IN	1,33	1,67	0,98	1,13	1,61	1,07
	JJ	1,02	0,9	0,62	0,84	1,13	0,82
	JJR	0,03	0,06	0,03	0,05	0,06	0,03
	JJS	0,05	0,05	0,03	0,03	0,03	0,03
	MD	0,2	0,14	0,05	0,18	0,2	0,13
	NN	3,65	3,93	2,92	2,96	3,68	3,33
	NNS	0,62	0,91	0,42	0,58	0,77	0,51
	POS	0,11	0,18	0,05	0,09	0,11	0,08
	RB	0,7	0,39	0,43	0,72	0,94	0,56
	RP	0,06	0,08	0,08	0,06	0,06	0,08
	TO	0,28	0,48	0,24	0,26	0,44	0,27
	VB	0,63	0,67	0,44	0,52	0,83	0,55
	VBD	0,27	0,32	0,14	0,37	0,35	0,25
	VBG	0,3	0,41	0,22	0,28	0,42	0,26
	VBN	0,23	0,28	0,09	0,19	0,25	0,17
	VBP	0,41	0,32	0,26	0,4	0,5	0,25
	VBZ	0,44	0,52	0,23	0,43	0,65	0,32
	WP	0,05	0,09	0,04	0,05	0,09	0,06
	WRB	0,08	0,06	0,05	0,08	0,11	0,05

Table 27: Correlation of grammatical frequency average between authors and audience

Author	Correlation	Author	Correlation
Elon Musk	0,99	Donald Trump	0,95
Katy Perry	0,98	Alan Shipnuck	0,99
Michele Dauber	0,99	Floyd Mayweather	0,99

last 1000 tweets for the same audiences used before, in order to identify the similarity between their texts.

Table 28: Similarities between authors and audiences

	Audiences							
Author	**(1)**	**(2)**	**(3)**	**(4)**	**(5)**	**(6)**	**Mean**	**Standard Deviation**
Elon Musk (1)	**0,681%**	0,605%	0,629%	0,650%	0,542%	0,536%	0,607%	0,058%
Donald Trump (2)	0,728%	**1,068%**	0,840%	0,833%	0,893%	0,687%	0,842%	0,135%
Katy Perry (3)	0,712%	0,613%	0,999%	0,811%	0,623%	**1,096%**	0,809%	0,201%
Alan Shipnuck (4)	0,553%	0,531%	**0,738%**	0,664%	0,557%	0,573%	0,603%	0,081%
Michele Dauber (5)	0,739%	0,745%	0,731%	0,824%	**0,926%**	0,672%	0,773%	0,089%
Floyd Maywheather (6)	0,605%	0,542%	0,846%	0,566%	0,473%	**1,430%**	0,744%	0,360%

Before analysing the texts, they were preprocessed using the same pipeline described in Section 10.3.1 in order to keep the texts in the same structure in for the different analysis.

Using the Jaccard distance as metric to analyse the similarity among the texts; initially, we analysed the similarity between each author's texts and the texts of all audiences, in order to identify which audience is more similar to the author. Once identified the text's similarity percentage, we calculated the average of each audience, according to the formula:

$$\frac{\sum_{i=1}^{n} SM1_i + \sum_{i=1}^{n} SM2_i + \sum_{i=1}^{n} SM3_i + \sum_{i=1}^{n} SM4_i + \sum_{i=1}^{n} SM5_i}{n}$$

Later, for each author, we calculated the mean and standard deviation for the similarity between him and the audiences, as presented in Table 28.

This information allowed to identify that, in most cases, the highest similarity average was between the author and his audience. Moreover, the cases where it did not occur, the similarities values of the author's audience added with standard deviation indicates that the audience's value is close to the highest value.

10.4 Conclusion

This paper presented an analysis of the emotional and grammatical writing styles similarity from authors and their most frequent audiences on microblogs. This approach used lexicon-based techniques to explore the emotions contained in tweets and NLP techniques to identify grammatical excerpts.

Once the emotional and grammatical writing styles have very high values, indicating a strong correlation between authors and audiences, it is possible to conclude that both authors and audiences share the same writing style. Moreover, the correlation between authors emotions and the most frequent audiences emotions exhibited in Fig. 47 is high, and as the size of this audience increases, the lower the correlation becomes, confirming Jim Rohn's claiming.

This is a crucial issue because it enables the possibility of chatbots to create an emotional profile based on the interactions received from people or even other systems, creating an identity and interacting with the end user in smooth communication. Combining Generative Adversarial Networks (GANs) and emotional profiles, a new generation of chatbots can create its own "personality" and generate textual responses that fit its emotional profile.

11

Case Study 5 - Prediction of election results according to emotional analysis

It is undeniable that social media have changed how people contact to each other, enabling them to maintain relationships that previously would be difficult to maintain for various reasons, such as distance, the passage of time, and misunderstandings.

Barack Obama used social media as the main platform of his presidential campaign in the United States in 2008, and since then, it is well established that social media created a strong influence on voters' decisions in that election and many others. Facebook and Twitter are now considered essential tools for any political campaign, along with other social media platforms that have been created since then.

The influence of social media increases when candidates with low funding levels and low levels of traditional media exposure try to defeat adversaries with more resources. Social media allow the candidates to post their political platforms as well as inflammatory posts against their opponents. In many cases, political candidates use social media mainly to carry provocative attacks against their opponents. Their followers, in turn, use social media to broadcast their reactions of anger and satisfaction in posts that can be shared thousands of times.

In electoral campaigns highly marked by their massive presence on social networks - as was the Brazilian presidential election in 2018 – by using Natural Language Processing (NLP) and Machine Learning (ML) it is possible to identify what voters think and feel about the candidates and their proposals for the various areas of the government. Thus, some interesting research question that arises are: "How do the emotions about each candidate influence the electoral decision?"and "Is it possible to predict the result of an election only by knowing what voters "feel" about a candidate?"

In this case study, we present an approach to predict the results of the election. As a case study, we collected messages from social media about the Brazilian presidential election of 2018, where we used Sentiment Analysis, NLP and ML to identify which emotions dominated the electorate and their correlations with the number of messages about the candidates and finally predict the results.

11.1 Related work

The analysis of emotions on Twitter to explain elections is not a new approach. Several researchers have already done work in this area, each with different approaches and results. [201] developed a system to analyze the tweets about presidential candidates in the 2012 U.S. election as expressed on Twitter. His approach analyzes the text's polarities (positive, neutral or negative), the volume of posts and the word most used. This is the same approach used by [72], that trained a Convolutional Neural Network using an annotated lexicon - Sentiment140 - to detect the text's polarities.

These approaches do not go deeper on the reasons of polarities, and thus, does not identify which sentiments influence the voters' decision, which is the objective of our work.

[195] has developed an analysis of tweets for the German elections which, similar to Wang's work, used the polarities of phrases to analyze the messages. However, unlike the previous work, Tumasjan's study relates the volume of messages to the final result of the election. This approach is highly influenced by financial questions (as richer the candidate is, more publicity about him can be posted in social media), and for this reason, we did not consider trustworthy. [13] used this same approach, using the data from the Irish general election of 2011, but, different than Tumasjan, he has expanded the model for training by using polls as parameters for training the predictions, which has inspired our work in the training dataset creation.

While the existing works focused only on the aspect of the text's polarities, the work of [121] inspired our decision to consider the basic emotions contained in the text as a factor of influence in the decision of the voter, and use it to predict results.

11.2 Dataset creation

Initially, to draw from the data a general idea of the Brazilian voter, it would be necessary to generalize the emotional profile of these voters, regardless of their region. The idea is that, according to the emotions contained in the texts, it would be possible to determine relevant information about the characteristics of the Brazilian voters. Thus, a pipeline was created for dataset creation, as presented in Fig. 48.

This pipeline begins with a collection of messages about the candidates. For this purpose, we collected tweets from 145 cities in Brazil, with each state being represented by at least its four largest cities, and delimiting a radius of 30 km for each city. The option for geolocated tweets is to avoid posts from countries different than Brazil, where the author probably would be not able to vote in Brazilian's election. To select what would be considered relevant or not, we defined that only the tweets containing the main candidates' names would be collected. Tweets containing two or more candidates' names were analyzed for all candidates mentioned in the text. Moreover, we considered relevant only tweets - not retweets. This decision was inspired by the necessity to avoid viral posts or the ones from digital influencers. In other words, we wanted to know the opinion from the message's author about a candidate, not the opinion of an author who the author likes.

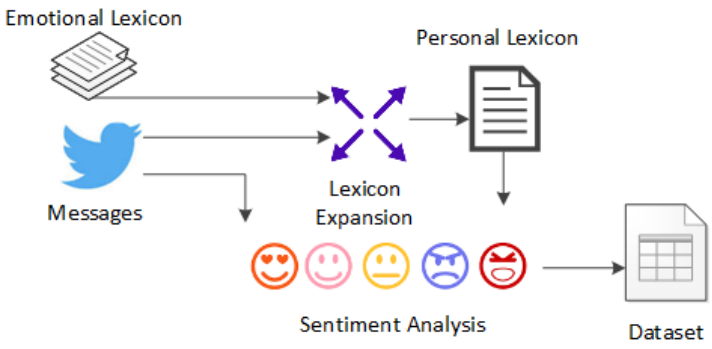


Figure 48: Dataset's creation pipeline

For gathering the tweets, we developed a script in Python using the official API provided by Twitter, and collected all geolocated messages from the 145 cities mentioned earlier in the period from May 2018 to October 2018, which contained at least one of the following names in their texts: "Bolsonaro", "Ciro Gomes", "Marina Silva", "Alckmin", "Amoêdo", "Álvaro Dias", "Boulos", "Meirelles"and "Haddad", divided in two groups: first round and second round.

11.2.1 Out of scope

After an initial analysis of the messages collected, we decided not to handle the ambiguity in the texts during the analysis. The reason for choosing not to address this problem was justified by the tiny amount of messages that could lead to erroneous interpretations. Since it was a mandatory requirement for the messages to contain the name of at least one candidate, the nature of the Twitter messages - which limits each post by 280 characters - already considerably inhibited this type of problem.

Furthermore, this initial analysis showed that the existence of the candidate's name in the text made the context of the message as political and egalitarian in the emotional sense, as derogatory nicknames for the leading candidate candidates bring negative emotions. So, we avoided that exacerbated emotional expressions of some candidates affect others.

11.2.2 Lexicon expansion

When working with sentiment analysis, a common approach is to use a dictionary-based algorithm to identify the emotional words in texts. However, according to Feldman [49], "the main disadvantage of any dictionary-based algorithm is that the acquired lexicon is domain-independent and hence does not capture the specific peculiarities of any specific domain."Thus, it is essential to know some particularities about the domain which the texts represent, to avoid misunderstandings and enable analysts to make a better classification of the sentiments contained in the texts.

With this problem in mind, we adapted the solution presented by Martins [119], where the texts were represented by a vector of words and these vectors were used to analyze the similarities of the words

contained in an emotional lexicon, to expand it. A major concern when creating these vectors was about the polarization among the candidates. Our idea was that the texts about a candidate do not influence the emotional words of other candidates. For this reason, we adopted the strategy of creating a personal emotional lexicon for each candidate and thus analyzing the candidate's sentiments individually according to their respective emotional lexicon.

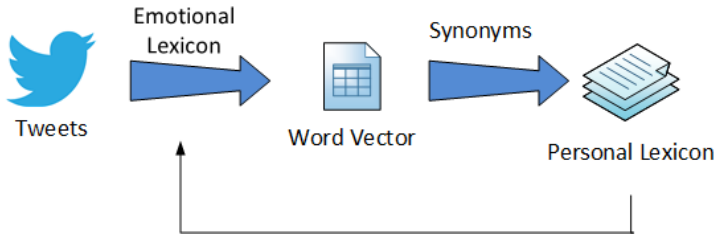


Figure 49: Lexicon expansion process

An overview of the entire process of lexicon expansion is presented in Fig. 49.

11.2.2.1 Word Vectors

The process of lexicon expansion begins with grouping all tweets collected by the candidate's name, removing their stopwords and creating the word vectors. For this purpose, we developed a script in Python using the Word2Vec algorithm, presented by Mikolov [128], having as parameters: size of 50; window 5 and trained for 200 epochs. Later, the emotional lexicon is introduced, to feed the word vectors with emotional seed words. For this step, we used the EmoLex lexicon [133] to provide the emotional words for the word vectors. The reason for this lexicon's choice is that it provides emotional words in Portuguese and also contains indications for polarities (positive and negative) and annotations for the eight basic emotions according to Plutchik's theory [157], which defines sentiments as *anger, anticipation, disgust, fear, joy, sadness, surprise* and *trust*.

Each word in the emotional lexicon was analyzed in the word vector of each candidate, to identify similarities. For all similar words found with a value higher than 0.7, these words inherited the emotional values from the lexicon's word and - recursively - were analyzed in the word vector to search for new similarities. For each similar word found in the word vectors, we added this word in a "new emotional lexicon"containing the original emotional lexicon and their respective similarities and emotional annotations according to the word vectors.

An important issue to emphasize in this approach is that when finishing the creation of this lexicon, we have a contextual emotional lexicon, because contextual words and its similarities were used in its creation. This context is provided because all messages contain at least one candidate's names. Thus, the context of the lexicon is about politics.

11.2.2.2 Results

When the lexicon expansion process was finished, the result was a set of personal lexicons about politics, containing the basic lexicon data increased by similarities found in the text and the synonyms of the words. The characteristics of each personal lexicon are presented in Table 29. Due to space limitation, Table 29 only presents the top 5 most known politicians, but in our study, all politicians that were candidate were considered in this analysis.

Table 29: Characteristics of the personal lexicon created

Lexicon	Words	Anger	Anticipation	Disgust	Fear	Joy	Sadness	Surprise	Trust	Positive	Negative
Original Lexicon	13911	8,85%	5,92%	7,48%	10,44%	4,88%	8,46%	3,76%	8,72%	16,36%	23,47%
Geraldo Alckmin	15251	8,89%	5,91%	7,50%	10,39%	4,82%	8,45%	3,79%	9,14%	16,56%	23,45%
Jair Bolsonaro	22082	9,16%	5,73%	7,63%	10,50%	5,01%	9,24%	3,38%	10,10%	17,49%	24,11%
Ciro Gomes	15316	8,95%	6,00%	7,47%	10,31%	4,99%	8,65%	3,77%	9,11%	16,78%	23,50%
Fernando Haddad	18264	9,27%	5,70%	7,47%	10,65%	4,87%	8,67%	3,46%	9,19%	17,26%	23,37%
Marina Silva	14842	8,76%	5,88%	7,40%	10,37%	4,85%	8,44%	3,76%	8,79%	16,38%	23,32%

11.2.3 Preprocessing

After creating a personal lexicon for each candidate, the next step consisted of creating a text preprocessing pipeline to remove unnecessary information from the texts. This pipeline, as presented in Fig. 50, begins with tokenization, which converts the texts into a list of single words, or *tokens*. Then, the process is divided into two parallel tasks: Part of Speech Tagging (POS-T) and Stopwords Removal. The POS-T process is responsible for identifying grammatical pieces of information for each word in the text, such as adjectives, adverbs and nouns, while the Stopword Removal removes any occurrence in the text of a defined word or list of words.

This strategy of paralleling POS-T and Stopwords removal was used because POS-T needs the text in the original format, to classify the words in their respective grammatical categories correctly.

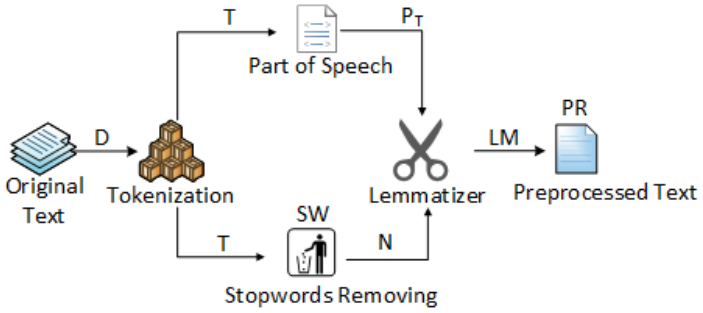


Figure 50: Preprocessing tasks

Concerning text cleaning, in POS-T, every word in a grammatical category other than noun, verb, adverb or adjective is discarded. This is important because only these grammatical categories carry emotional information that can be used in further steps. So, more formally, the tokenization process

converts the original text D in a set of tokens $T = \{t_1, t_2, ..., t_n\}$ where each element contained in T is part of the original document D. These tokens will feed the POS-T, which will label each token with semantic information. Finally all nouns, verbs, adverbs and adjectives will be collected in a set P, where $P_t = \{p_{(t,1)}, p_{(t,2)}, ..., p_{(t,k)}\}$ and 0 ≤ k ≤ n and $P_t \subset T$.

The Stopwords list is a manual and predefined set $SW = \{sw_1, sw_2, ...sw_y\}$ of words, intended to avoid the analysis of common and irrelevant words. There are many examples of Stopwords lists on the internet and in libraries for Natural Language Processing (NLP). In our approach, after the Stopwords Removing process, the result list is a set $N = T - SW$.

After the parallel preprocessing tasks finish, the result document ST must contain a set of words where $ST = P \cap N$.

Later, in LM a lemmatizer process reduces the words to their lemma. This step is important because allows considering all inflected words as only one, producing the set of preprocessed texts $PR = \{LM(ST_1), LM(ST_2), ..., LM(ST_z)\}$.

The final result of this pipeline is a new emotional lexicon that considers the similarities of words used in expressions that cite the candidates, and their synonyms, and that is a personal representation of the sentiments about each candidate.

For this preprocessing step, we developed a Python module using Spacy[1] for automatizing the Tokenization, POS-T, Stopwords Removal and Lemmatization processes. We chose to use this toolkit in the development because it provides support for Brazilian Portuguese in all steps described earlier.

11.2.4 Sentiment Analysis

Once we created new personal lexicons for all candidates, the next step in the dataset creation was to analyze the emotions contained in the texts about each candidate.

For this purpose, we developed a tool that counts the frequency of each emotional word in a text. The result of this analysis is the final dataset, containing the emotional analysis of each Twitter message for each candidate, on a scale from 0 to 100 for each Plutchik's primary emotion.

This approach - a bag-of-word approach - was adopted because we intended to identify which emotions were more relevant to the voters when deciding their candidate, besides to generate a "candidate fingerprint"through the words used to describe them. Moreover, the absence of emotional corpora about politics in Portuguese restricted the possibility of using other techniques to identify the emotions in our texts.

11.3 Data analysis

Once the dataset was created, we attempted to use data analysis to identify some particularities about the data, and how these particularities could explain the results of the elections. We used several techniques

[1] https://spacy.io/

to identify correlations between the results of the elections and the data analyzed.

Once we identified the emotions that influenced the first-round results and how they did so, the next objective was to predict the results for the second round. To reach this objective, we decided to use an approach based on machine learning. The goal of this approach was to train a model that could accurately predict the percentage of votes for each candidate based on the emotions previously identified, the percentage of votes cast for each candidate in the first round, and the emotions contained in tweets on the day of the second-round vote.

11.3.1 Training dataset

During the creation of a dataset for training the model, an important issue was identified: how to "translate"the emotions into a percentage of votes. Once the first round results were known, the relationship between candidates' emotional profiles and the percentage of votes cast on that day could also be determined. However, it was necessary to obtain many more examples to train a model. To bypass this obstacle, we chose to use the public information on electoral polls available from public institutes. The chosen institutes were: Instituto Brasileiro de Opinião e Pesquisa (Ibope) [2], Instituto Datafolha [3], Vox Populi [4] and Paraná Pesquisas [5] which are the most important polling institutes in Brazil.

To create the training dataset, we collected the voting intention results of 42 polls for candidates that we had in the dataset, which resulted in 324 examples for training, with 107,039.52 tweets analyzed. Later, knowing the period of each poll, we analyzed the average of each basic emotion for each candidate in the same period from the database. We then transferred these emotions to a new file, indicating the candidate's name, the number of candidates in the poll, period, emotions, and institute.

Despite the number of Tweets messages of each candidate are different, this new dataset contains each candidate's grouped emotions during the period of each poll. Thus, all candidates had the same number of registers in the dataset. This approach ensured that the most cited candidates in messages did not bias the dataset.

11.3.2 Predicting results

After creating the training dataset, the next step was to train a model for predicting the results for the second round. For this purpose, we analyzed five different machine learning algorithms, to identify the best correlation between data and results. In all cases, the dataset was separated 70% for training and 30% for testing, using the Mean Absolute Error (MAE) as errors standard measure to have a comparison basis between traditional pools and twitter Sentiment Analysis.

[2] http://www.ibope.com.br
[3] http://datafolha.folha.uol.com.br/
[4] http://www.voxpopuli.com.br
[5] http://www.paranapesquisas.com.br/

The algorithms (using implementations for R and all tuned for the best fit) chosen for the analysis and their results after training the models are presented in Table 30.

Table 30: Algorithms evaluation

Algorithm	Correlation	MAE
Simple Linear Regression	0,2639	10,6302
SVM	0,3677	9,3608
Decision Table	0,5385	9,226
Extreme Gradient Boost	**0,9096**	**0,9787**
Random Forest	0,8088	6,322

The best result was obtained by an Extreme Gradient Boost algorithm, which had a correlation of 0,9096 (very strong correlation) between the results in the polls and the emotions in the same period, and a mean average error of 0,9787.

Once the model was created, the goal was to predict the percentage of votes for each candidate in the second round. In our experiment, we decided to predict the values based only on the emotions expressed on the day of second-round voting until 17:00. This time limitation is because voters could vote only until 17:00. Voters made their decisions based on their emotions during the voting process. Therefore, emotions that were expressed before the second-round election day were not crucial in this analysis.

After collecting the tweets for each second-round candidate - Jair Bolsonaro and Fernando Haddad - on October 20 and analyzing tweets about them using the same process presented in section 11.2.3, we got the values presented in Table 31.

Table 31: Emotions in second round's day

Candidate	Anger	Anticipation	Disgust	Fear	Joy	Sadness	Surprise	Trust
Jair Bolsonaro	13,14%	11,14%	8,18%	12,71%	12,55%	14,80%	6,97%	20,50%
Fernando Haddad	12,39%	13,50%	6,41%	10,67%	14,95%	14,22%	7,57%	20,29%

When applying these values to the model created previously, we got a prediction of **54,58% for Jair Bolsonaro and 43,98% for Fernando Haddad and 0,9787% of MAE** that can be considered as a **correct prevision** because the official results for the second round were 55,15% for Jair Bolsonaro and 44,87% for Fernando Haddad, whose values are in the accepted error margin.

11.4 Conclusion

Social media have changed the way people interact and express their thoughts about everything. Because of this changing reality and the vast quantity of data available, sentiment analysis is becoming a powerful, fast, and relatively inexpensive tool that is extremely useful for analyzing many different types of scenarios and predicting future results.

Furthermore, although there have been many studies about the influence of social media on elections, there are not approaches using sentiment analysis to identify voters' emotions and predict future election results, while taking into account the results of previous studies.

The correlation between voters' emotions and the percentage of votes shows how vital is to know the audience's sentiments to plan effective strategies for interacting with them. Moreover, the very strong correlations that we found between the basic emotions and poll results', as well our model's successful prediction of the second-round results of Brazilians elections, strongly suggest that sentiment analysis can become a viable and reliable alternative to traditional opinion polls, with the advantages of being much faster and less expensive.

However, it is important to emphasize that there are no studies yet published about what the acceptable threshold is for replacing traditional polls with sentiment analysis. Also, it has not yet been established how many tweets must be analysed to replace a traditional opinion poll with a sufficient degree of certainty.

12

Case Study 6 - Depression classification based social media analysis

In our social relationships, the chance of knowing someone who is suffering or suffered from depressive disorders is high. According to United Nations (UN), the worldwide population is about 7.6 billion[1] habitants, when compared to 300 million[2] of depressive people according to World Health Organization [143], draws attention to the fact that every 25 people in the world, 1 suffers from depression. Yet according to WHO, "depressive disorders are characterized by sadness, loss of interest or pleasure, feelings of guilt or low self-worth, disturbed sleep or appetite, feelings of tiredness, and poor concentration. Depression can be longlasting or recurrent, substantially impairing an individual's ability to function at work or school or cope with daily life. At its most severe, depression can lead to suicide."

Having the words of [177] in mind ("if the eyes are the window to the soul, then words are the gateway to the mind"), the symptoms associated with depressive disorders are evident in texts produced by depressive people. So, Twitter, Facebook and web forums are an excellent source of information to collect clues about the messages authors' mental wellness. Automated analysis of social media can provide detection methods, and, once identified an individual as depressive, an assessment, support and treatment can be provided sooner.

In this paper, we present an approach based on Machine Learning, Sentiment Analysis and Natural Language Processing to identify depressive profiles on Twitter, based on the messages posted. It is not our intention to identify different levels of depression which authors can have. To reach this objective, we adopted the emotion model defined by [157], because we consider more realistic due to it differentiate the emotions in more categories than [36], despite easy to use and able to represent different emotions through dyads emotion.

[1] https://news.un.org/pt/story/2017/06/1589091-populacao-mundial-atingiu-76-bilhoes-de-habitantes
[2] https://www.who.int/news-room/fact-sheets/detail/depression

12.1 Related work

Identifying depression using artificial intelligence is not a new research area. During the last years, works that handled aspects of depression, such as identification, degree estimation and treatment monitoring, through Machine Learning or Deep Learning techniques have been produced.

[21] was a pioneer to describe how to construct a classifier to identify Twitter depressive users based on the analysis of their activities in social media. That research introduced social media as a source of data for recognizing symptoms of depression in a user, using measures as user engagement and emotion, egocentric social graph, linguistic style, depressive language use, and mentions of antidepressant medications, reaching 70% accuracy and precision of 0.74.

The work of [194] introduced an approach to identify the level of depression in Twitter users with an accuracy of 69%. Based on the user's historical activities, he constructed an SVM model fed by topics inferred from word frequencies. These words were supplied by volunteers, which filled a questionnaire to analyse the degree of depression, and the history activities were gathered through the Twitter API.

[78] presented an approach using the CLPsych 2015 Shared task data that considers Twitter posts from authors to detect depression. His approach created different word vectors approaches (Skip-Gram, CBOW, Optimized and Random) and applied these word vectors to different Deep Learning architectures, such as a Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN), reaching an accuracy of 87.957% and precision of 87.435% as better results.

All these works inspired and contributed with some thoughts to our solution, but they do not consider emotions as important information to identify depression, neither the mood fluctuation over time, that is our different approach.

12.2 Identification of depressive profiles

People have in their personalities, characteristics that distinguish them from others. Some people are shy, others more enthusiastic or sarcastic in its remarks. All these characteristics determine their personalities, and according to discrete emotional theorists, this personality is present in our expressions - facial, speech, dress - and can be described as a set of independent and basic emotions.

In the literature, a well-known model proposed by [36] states that all emotion is composed by 6 basic emotions: happiness, sadness, fear, anger, surprise, and disgust. For Plutchik, any sentiment is a combination of 8 basic emotions: *Anger*, *Anticipation*, *Disgust*, *Fear*, *Joy*, *Sadness*, *Surprise* and *Trust*, and can be mapped as a wheel in emotions". Additionally, Plutchik introduces a dimension of *intensity* which differentiates all basic emotions according to the intensity degree, introducing a 3D emotional model, as exhibited in Fig. 51.

.

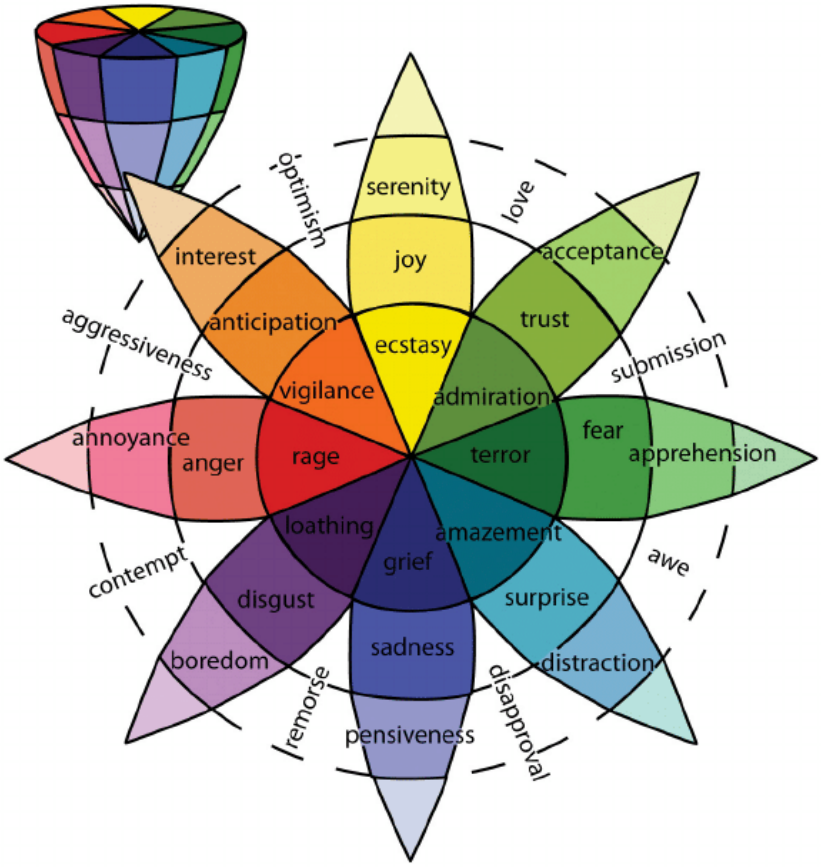


Figure 51: Plutchik's wheel of emotions

12.2.1 Emotional profile

When handling with emotions, there is common thinking that exists a straight connection between emotions and mood, and it is true because mood expresses emotions. However, the mood is different from the emotional profile. While the emotional profile of a person stays rigid throughout our life, the mood is more open to change, influenced by the current emotions. Therefore, it is not possible to consider that the emotional profile could be detected in a short period. Moreover, when reminding the definition of WHO about depression (*depression can be longlasting or recurrent*), we must consider a relevant time interval to detect the emotional profile of an individual, i.e. the depression affects the emotional profile of a person, not his mood.

In this work, we considered the emotional profile of an individual as the set of the eight Plutchik's basic emotions, in a distribution where the sum of all these emotions results in 100%. We decided to use Plutchik's model because we believe that it represents better a way to handle the emotions and provides mechanisms to differentiate levels of each emotion.

12.2.2 Data sample creation

Dealing with depression in social media requires the ability to identify a pattern in the writings of someone who is suffering. However, as well as depression is known as the "silent killer" illness, it is very difficult to find data on social media posted by clinically identified people with depression. For this reason, we have adopted the strategy of searching in Twitter all messages containing the text **"I have depression"** in English. The objective of this approach is to collect messages from authors self-declared depressives for training a model using Natural Language Processing and Deep Learning to identify the depressive emotional profile.

The results were manually analysed to identify in which context the sentence was mentioned, the authors of these sentences classified as "depressives", and they had all their tweets collected in a time interval of 9 months.

During the author's tweets gathering, we discarded all re-tweets and tweets containing mentions or links. This is because we intend to detect the emotional profile of the depressive author with no interferences, i.e., we wanted the messages those are not a chat with others, advertisements, neither posts from an influencer (in case of re-tweets). This approach resulted in a set of 200 authors (94 classified as non-depressive and 106 classified as depressive) and 492178 tweets collected.

12.2.3 Tweets preprocessing

After collecting a set of tweets messages from depressive authors, the next step consisted of a text preprocessing pipeline to remove unnecessary information from the texts. This pipeline, as presented in Fig. 52 have their steps as:

- Tokenization - converts the texts into a list of single words, or *tokens*;
- Part of Speech - processes all messages for grammatical identification and cleaning all tokens those grammatical categories are different from nouns, verbs, adverbs and adjectives;
- Lemmatization - identifies the lemma for each token and saves it for later use;
- Undesired words removal - removes all words contained in a "blacklist" (*stopwords*) and those with 3 or fewer characters.

The final result of this pipeline (therefore called preprocessed text)is a new sentence containing only the lemmas of words contained in tweets texts classified as verbs, adverbs, adjectives and nouns, containing at least 4 characters.

All preprocessing steps were developed under Python using the Spacy[3] module for automatizing the processes.

[3] https://spacy.io/

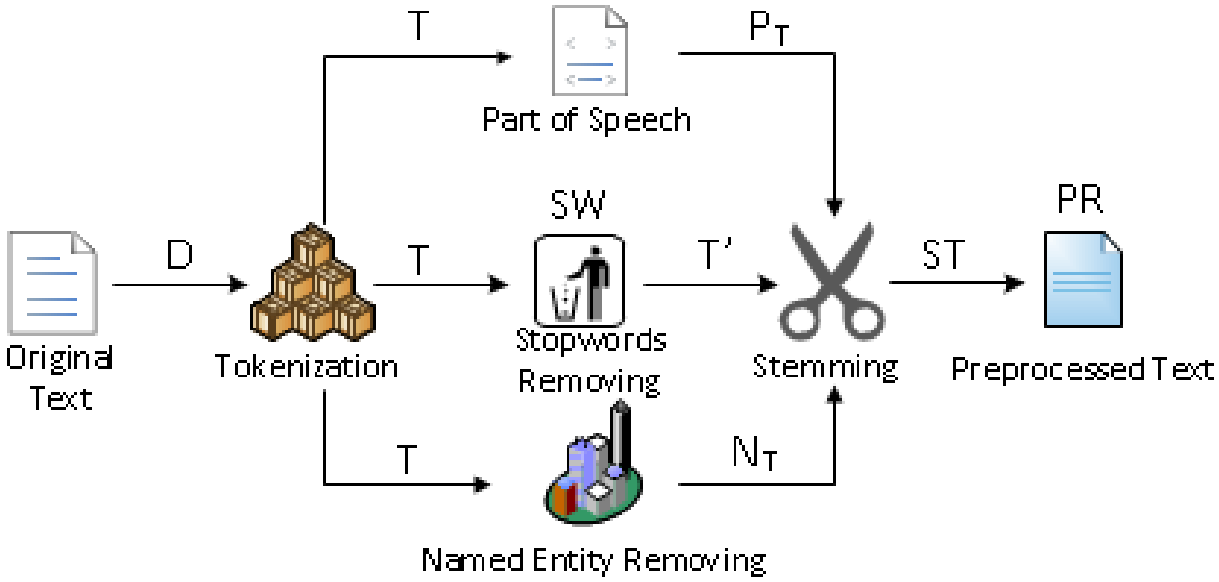


Figure 52: Preprocessing tasks

12.2.4 Sentiment Analysis

When the preprocessing is finished, the following step in the dataset creation was to analyse the emotions contained in the preprocessed texts. Regarding this purpose, we developed a tool in Python that identifies the frequency of each word on the preprocessed text and compares it with an emotional lexicon. The existence of the word into the lexicon results in the information of the word has an emotional context. Using the EmoLex Emotion lexicon [133] and EmoLex Affect Intensity lexicon [132], we identified the frequencies of all emotional words according to Plutchik's emotional definition and extracted the distribution of each emotion in a text. Regarding the affect intensity, the approach was to sum all intensities of each emotion for each preprocessed text.

Finally, we created a dataset containing the mean of all emotions and intensities grouped by author and trimester, where each author must have posted at least 150 messages. This minimal limit of 150 messages is because it is necessary to collect as many messages as possible, so, authors having less than this amount could bias the analysis due to less information about their emotions.

This approach - known as bag-of-words - was adopted because we intended to identify which emotions and intensities are relevant to detect depression, besides creating a "depressive emotional fingerprint"through the words used in the texts.

The final result was a dataset containing 125 registers, divided into 68 non-depressive and 57 depressive authors, indicating the average of emotions and intensities of all your posts during the trimester. An overview of the dataset is presented in Fig. 53.

```
Depressive,IAnger,IFear,IJoy,ISadness,Anger,Fear,Joy,Sadness,Anticip,Disgust,Surprise,Trust
1,0.307881,0.178609,0.290166,0.195166,0.1241,0.1633,0.1453,0.1495,0.1294,0.0923,0.0944,0.1018
1,0.338536,0.23027,0.263468,0.193108,0.1584,0.1463,0.1318,0.1143,0.1191,0.1227,0.0828,0.1245
0,0.206103,0.174085,0.23,0.151455,0.0976,0.144,0.1424,0.1392,0.1472,0.0592,0.1232,0.1472
0,0.220147,0.205588,0.297647,0.189265,0.0985,0.1286,0.1472,0.1031,0.1796,0.0823,0.0904,0.1703
0,0.236933,0.244311,0.289778,0.192222,0.1059,0.1187,0.1257,0.1129,0.1816,0.0931,0.092,0.17
1,0.28194,0.169526,0.283966,0.173276,0.1187,0.1288,0.1602,0.1216,0.1416,0.0973,0.0687,0.1631
```

Figure 53: Dataset created

12.3 Data analysis

The data analysis was divided into two different analysis: Exploratory Analysis, and Deep Learning-based analysis

12.3.1 Exploratory Analysis

Exploratory analysis is the approach aimed at analysing datasets to summarize their main features, often with visual methods, aiming to find pieces of information hidden in the data.

In this work, the initial approach in the exploratory analysis was to identify the outliers in each emotion and intensity to remove them from the sample. For this reason, we created a boxplot to identify visually, as presented in Fig. 54, the outliers for each emotion and intensity.

Regarding outliers handling, our approach was to change each outlier value for the mean of each emotion or intensity category (depressive or non-depressive). The reason was to avoid that outliers in some emotion (for example) affect others in case of removing the entire register.

The next step was to identify the relationship between the dataset dimensions and the depression indication. For this purpose, we measured the correlation between the depressive flag and the emotions, as presented in Table 32.

Table 32: Correlations between depressive status and basic emotions

Basic emotions							
Anger	**Anticipation**	**Disgust**	**Fear**	**Joy**	**Sadness**	**Surprise**	**Trust**
0.47	-0.43	0.17	0.45	-0.56	0.32	-0.28	-0.19

These outcomes show that negative emotions (anger, fear and sadness) have a moderated correlation (r^2) with depression, whereas positive emotions (in this case, anticipation and joy) have an inverse moderated correlation. The same analysis was performed for the emotional intensities, as presented in Table 33, confirming the same results.

Despite the data values analysed in the correlation are different in their characteristics - emotions have continuous values while the information about depression has a nominal value - the Pearson correlation can be applied because these data can be interpreted as a **biserial correlation**. This is justified because the depression indicative can only have 2 values, assuming binary characteristics.

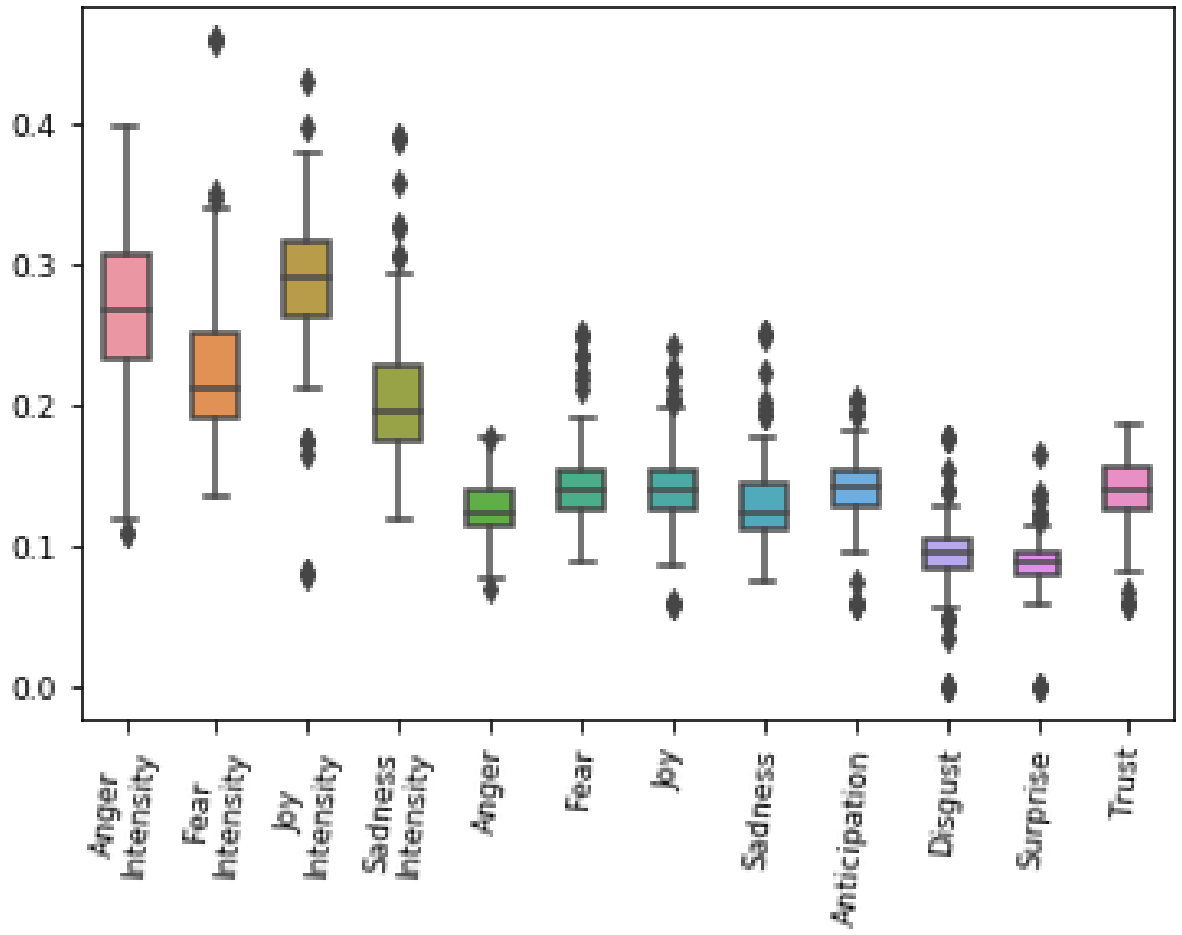


Figure 54: Outliers identification

Table 33: Correlation between depressive status and intensities

	Intensities			
Anger	**Fear**	**Joy**	**Sadness**
0.49	0.49	-0.43	0.41

12.3.2 Clustering analysis

The objective of the Clustering Analysis is to perform data transformations to understand if and how the data can be grouped. To achieve this, the initial step was to transform the 12-dimension dataset into a 2-dimension dataset for visualizing the data as a scatter plot, and this would be impossible using 12-dimension data.

Initially, we performed a Principal Component Analysis (PCA) to reduce the dataset (with no information about depressive classification) dimensionality. The PCA algorithm identified that the most relevant dimensions in the dataset were *Fear* and *Disgust*, which can contain 72.33% of the information.

Next, the dataset resultant from PCA analysis fed the KMeans algorithm used to cluster into 2 categories the data and generated a scatter plot. The resultant graphic is presented in Figure 55.

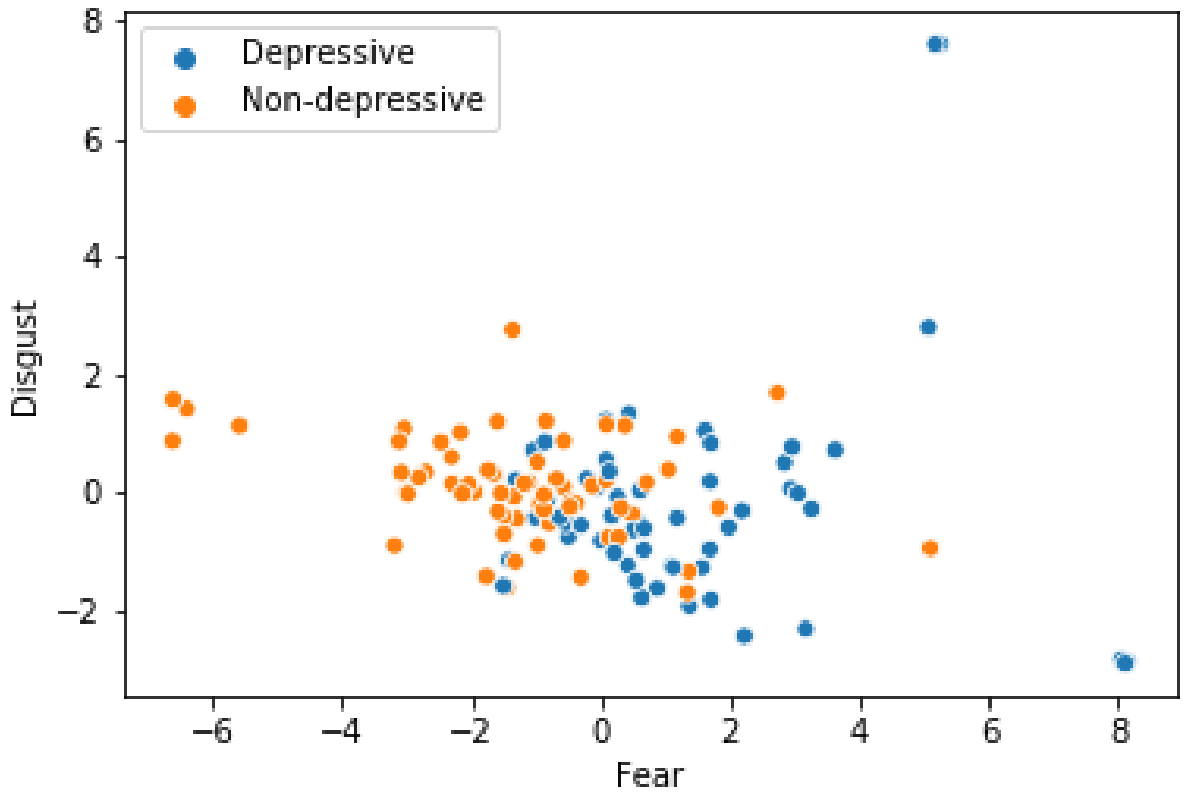


Figure 55: Clusters of information

This information is important because it shows that the emotional data about depressive and non-depressive can be divided into 2 distinct categories, and visually distant.

12.3.3 Machine and Deep Learning analysis

The next analysis consisted in the creation of a classification model able to identify the depressive emotional profile in messages. To achieve this objective, we used the dataset created in section 12.2.4 in several Machine Learning and Deep Learning algorithms, aiming to identify the best model to classify depression.

Some characteristics from the problem (depression classification) and the dataset - as few examples - were relevant in the choice of the algorithms analysed. Algorithms like LDA (Linear Discriminant Analysis) and Fisher Linear Discriminant were discarded because we believe that the problem is not represented by a linear function. Furthermore, due to the few data for training, Machine Learning algorithms based on neural networks and boosting were not considered because they tend to have a better performance when using a big amount of data. However, we analysed these network models, through a Dense Neural Network (DNN) and Convolutional Neural Network (CNN) using a Deep Learning approach.

So, in our analysis were considered the following algorithms: Support Vector Machines (SVM), Random Forests, Naive Bayes, DNN and CNN. In our analysis we just considered the algorithm's results after a tuning process, i.e., each algorithm presents its better results using the best parameters. For the Deep Learning models, we created several different model architectures to evaluate the better results. For DNN, the architecture whose got the best accuracy was a 5-tier neural model, having a dropout of 0.5 between each tier to avoid overfitting, respectively. Regarding CNN architectures, the most accurate was a 5-tier 1-D Convolutional Neural Network, using the same strategy of dropout 0.5 on each tier to avoid overfitting.

The accuracy and mean squared error for each algorithm and network architecture is presented in Table 34.

Table 34: Benchmark of Machine and Deep Learning algorithms

Algorithm	Accuracy	Mean Squared Error
SVM	0.984	0.126
Random Forest	0.8	0.176
Naive Bayes	0.76	0.45
DNN	0.939	0.076
CNN	0.915	0.071

These results show that some Machine Learning algorithms such as SVM and Deep Learning algorithms as DNN can identify the depressive emotional pattern with good accuracy. These results are better than the results observed in Section 12.1, reinforcing that the emotional text analysis to identify depressives can be a promising alternative to help people that are suffering silently.

12.4 Conclusion

Day after day, depression is becoming an epidemic disease that affects people of different social levels, cultures and ethnicities. Due to the nature of silence, identifying people who ask for help because of this illness but cannot verbalize that request is quite complicated, and often goes unnoticed even by the person suffering from depression.

The use of textual sentiment analysis can help identify the disease as it is a noninvasive method that can be continuously monitored. This is a huge help in the war against depression because it enables us to identify periods of wellness and sadness without a necessity to visit a psychologist, enabling a quick action when necessary.

Despite many works in this research area, the results obtained for this approach are promising when compared to the previous efforts, principally when the accuracy of 0.984 on depression classification is presented. However, once the information was data collected from social media, we cannot discard the hypothesis of biased data, because it is not possible to assure that the authors were true when writing their posts.

Part IV

Summary, Contributions and Future Work

13

Conclusion

One of the main characteristics that differentiate Human Being from other species in the animal kingdom is his ability to transmit his knowledge over time. Man has always used the elements around him to express his environment, whether in drawings made in caves thousands years ago, in hieroglyphics as in ancient Egypt, or using the existing alphabets. Over time, different ways of communicating were structured and thus the different languages (such as Latin, Coptic, and Ancient Greek) emerged, which in turn evolved and gave rise to other languages, with different textual representations, such as Hindi, Kandi, Cyrillic, Latin, Greek and Arabic (among several other languages), being used today by people over all the world still with the same purpose of the caveman: to transmit knowledge.

Through this knowledge transmitted over time, we were able to learn about the people's life and their beliefs, customs, fears and joys. We learn about Homer's Odyssey from the Greeks, about the teachings contained in the Hebrew Bible through the Dead Sea Scrolls, about the funeral rituals of the ancient Egyptians in the tombs of the pharaohs.

However, when learning through reading, the reader will inevitably be subject to the point of view of the author of the text, which is somewhat dangerous when it comes to fake news. To illustrate this point of view, at the time of writing the author tries to recreate the whole scenario so that the information is more easily understood by the reader through language, using words to describe the facts, emotions and senses in which he is feeling or trying to convey. Thus, when writing a text, the author transmits his emotions in this process.

Changing to the present, people have never produced so many texts as today. We are surrounded by textual information that come from websites, blogs, social media, etc. Moreover, the social relations became more physically distant - even more in pandemic times - but closer due to the use of apps such as WhatsApp. And the biggest part of these communication is done by text.

So, these texts can be a valuable source of information to identify what people are feeling and classify their emotional state according to their emotions expressed in texts.

Observing the basic emotions model (namely the Plutchik's model), where each individual has a set of 8 basic and universal emotions, when identifying the frequency of use of these emotional words, it was possible to draw a **emotional profile** of each author.

Some issues were taken into account during this work in relation to the effectiveness of this emotional profile to classify the emotional state. First, it was necessary to know whether through the use of this profile it would be possible to differentiate people (in this case, using texts by their respective authors). Once the effectiveness of the emotional profile for differentiating people was verified, it was necessary to improve the model for identifying emotional words, as each author has different emotional characteristics, therefore, it would not be possible to consider that the same word had the same emotional charge for all.

Then, we identified that the affinity between people is also identified through emotional profiles. Through the emotional profile, we identified that it was possible to predict people's behaviours, such as their impacts on daily activities and decisions taken under emotional context.

Finally, in the last step, we analysed a set of messages from people with depression together with a set of people who did not have the disease. Through the use of Machine Learning and all steps previously developed, a model was created capable of learning the emotional profile of a depressed person, and thus classifying their emotional state based only on text analysis.

13.1 Research questions & Results and contributions

Regarding the main objective that guided this PhD research, which was to create a classifier to identify an author's emotional state from a set of texts of his authorship, in Section 1.2 were presented some research questions that are revisited and analysed here:

- **Do emotional labels impact on the accuracy of classifications?**

  Yes. The use of emotional labels in machine learning models increased the accuracy of classification, as demonstrated in the Chapters 7, 10 and 12, showing that this information is useful to detect the emotional state. To achieve this conclusion, it was reproduced experiments using as baselines the results obtained from classifications with no emotional labels. Later, the experiments were redone considering the emotional labels and both results were compared;

- **Is there an abstract model able to represent emotional information from a personal perspective?**

  Yes. The combination of the 8 basic emotions in a normalized distribution - the emotional profile - demonstrated that it is possible to be considered as a personal representation of the author's emotional characteristics, being used to represent them, as demonstrated in the Chapters 7, 8 and 10. This emotional profile is a set of 8 basic emotions according to Plutchik's model, where each emotion is in a range of 0 and 1 and the sum of all 8 emotions is 1. The emotional profile represents the author's emotional characteristics expressed in their texts and is different from an author to others.

- **Based on the emotional state of a person, would be possible to predict his actions or choices?**

Yes. It was identified that people that share similar emotional profile tend to do similar errors and attitudes, as demonstrated in Chapter 9 and 11. It was possible because people tend to be repetitive in their habits and routines. The study to predict actions was based on the numbers of typing errors when an author types about an emotional topic. It was considered the emotions contained in the texts and the number of errors was correlated with emotions, bringing some interesting results. The prediction of choices was studied when creating an emotional profile of each candidate in Brazilian's elections using posts in Twitter, and making a regression to predict the percentage of votes.

- **Through emotional information, is there a correlation between personal writing style and emotional state?**

  Yes. The analysis of the writing style - the emotions contained in the vocabulary used by the author and the grammatical characteristics of the text - have a high correlation with the emotional state of the author, as demonstrated in Chapter 10 and 12. This was possible because the author expresses their emotions in their texts, like a fingerprint. Through the emotional profile identified in texts, is was possible to create a classifier to identify depressive profiles - i.e. authors that have depression - and apply this classifier to identify if a new emotional profile of an author is regarding a depressive or not.

For these questions, the development of this PhD research followed some steps, which were portrayed in the case studies.

By studying the way in which an author usually expresses himself, it is possible to identify in his texts personality traits that reflect his way of being. In this light, we compared the words frequently used by a given author with an emotional lexicon (under the premise that the author was not masking his emotions), enabling the identification of the most frequently used emotional words.

Based on the initial objectives and the achieved results, the main contribution of this PhD work is an approach to classify the emotional state based on text messages.

13.2 Publications

During this PhD work, there were published some preliminary efforts which contributed to this work. The publications that outcome from this doctoral work are listed below:

- *Determining Emotional Profile Based on Microblogging Analysis* - Proposal of an approach to define the emotional profile based on text messages.

  Martins R., Henriques P., Novais P. (2019) Determining Emotional Profile Based on Microblogging Analysis. In: Moura Oliveira P., Novais P., Reis, L. P. (eds) 19th EPIA Conference on Artificial Intelligence. EPIA 2019, Springer - Progress in Artificial Intelligence, Part 2, ISBN 978-3-030-30244-3, pp 159-171, 2019. https://doi.org/10.1007/978-3-030-30244-3_14. SCImago SJR IF (2019) 0.65 (Q2 - Artificial Intelligence), Indexed: Scopus;

- *A sentiment analysis approach to increase authorship identification* and *Hate Speech Classification in Social Media Using Emotional Analysis* - Proposal of an approach using emotional information to improve classifications.

  Martins R., Almeida J.J., Henriques P., Novais P., A sentiment analysis approach to increase authorship identification. Expert Systems, WILEY-BLACKWELL, Special Issue. ISSN 0266-4720, 2019. https://doi.org/10.1111/exsy.12469. ISI JCR (2017) IF: 1,430 (Q2 - COMPUTER SCIENCE, THEORY & METHODS). SCImago SJR (2017) IF: 0,429 (Q2 - Artificial Intelligence)

  Martins R., Gomes M., Almeida J. J., Novais P., Henriques P. (2018), Hate Speech Classification in Social Media Using Emotional Analysis. 7th Brazilian Conference on Intelligent Systems. BRACIS 2018, IEEE - Proceedings - 2018 Brazilian Conference on Intelligent Systems, BRACIS 2018, pp 61-66, 2018. https://doi.org/10.1109/BRACIS.2018.00019. SCImago SJR IF (2020) 0.22, Indexed: IEEE;

- *Creating a social media-based personal emotional lexicon* - Proposal of a tool for expansion of emotional lexicons.

  Martins R., Almeida J. J., Novais P., Henriques P. (2018), Creating a Social Media-based Personal Emotional Lexicon. 24th Brazilian Symposium on Multimedia and the Web. WebMedia 2018, ACM - Proceedings of the 24th Brazilian Symposium on Multimedia and the Web, pp 261-264, 2018. http://doi.acm.org/10.1145/3243082.3264668. SCImago SJR IF (2020) 0.12, Indexed: ACM;

- *Domain Identification Through Sentiment Analysis* - Proposal of an approach to identify the context where a conversation occurs, based on emotional analysis.

  Martins R., Almeida J. J., Henriques P., Novais P. (2018) Domain Identification Through Sentiment Analysis. In: De La Prieta F., Omatu S., Fernández-Caballero A. (eds) 15th International Symposium on Distributed Computing and Artificial Intelligence. DCAI 2018, Springer - Advances in Intelligent Systems and Computing, volume 800, ISSN 2194-5357, pp 276-283, 2018. https://doi.org/10.1007/978-3-319-94649-8_33. SCImago SJR IF (2018) 0.174 (Q3 - Computer Science), Indexed: Scopus;

- *Predicting Performance Problems Through Emotional Analysis* - Proposal of an approach to predict problems based on text analysis and emotional analysis.

  Martins R., Almeida J. J., Henriques P., Novais P. (2018), Predicting Performance Problems Through Emotional Analysis (short paper). In: Henriques P., Leal P., Leitão A. M., Guinovart X. G. (eds) 7th Symposium on Languages, Applications and Technologies. SLATE 2018, Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, OpenAccess Series in Informatics (OASIcs), volume 62, ISBN 978-3-95977-072-9, pp 19:1-19:9, 2018. https://doi.org/10.4230/OASIcs.SLATE.2018.19, SCImago SJR IF (2018) 0.206, Indexed: DBLP;

- *Predicting an Election's Outcome Using Sentiment Analysis* - An approach to predict group decisions based on emotional analysis.

  Martins R., Almeida J. J., Henriques P., Novais P. (2020) Predicting an Election's Outcome Using Sentiment Analysis. In: Rocha A., Adeli H., Reis, L. P., Costanzo S., Orovic I., Moreira F. (eds) 8th World Conference on Information Systems and Technologies. WorldCist 2020, Springer - Advances in Intelligent Systems and Computing, volume 1159, ISBN 978-3-030-45688-7, pp 134-143, 2020. https://doi.org/10.1007/978-3-030-45688-7_14. SCImago SJR IF (2019) 0.184 (Q3 - Computer Science), Indexed: Scopus;

- *Identifying Depression Clues using Emotions and AI* - A case of study about emotional state identification using emotional information.

  Martins R., Almeida J. J., Henriques P., Novais P. (2021) Identifying Depression Clues using Emotions and AI. In: Rocha A. P., Steels L., Herik H. J. (eds) 13th International Conference on Agents and Artificial Intelligence. ICAART 2021, SciTePress - Proceedings of the 13th International Conference on Agents and Artificial Intelligence, volume 2, ISBN 978-989-758-484-8, pp 1137-1143, 2021. https://doi.org/10.5220/0010332811371143. SCImago SJR IF (2020) 0.13, Indexed: Scopus;

13.3 Future Work

Since the approach here proposed to identify emotional states is a totally non-invasive technique, it allows the creation of a system to *monitor* people at risk, such as depressive crises, panic attacks or anxiety, as it can be associated with the tools that surround people today, like smartphones, computers and personal assistants.

If, on the one hand, access to information about people's emotional state is an important tool for predicting and preventing diseases, on the other hand it opens up a huge potential for actions that are not as noble, such as targeted marketing. Therefore, it is necessary that measures are taken to prevent the collection of this information in an irregular manner, as well as its exploitation. Fortunately, there are several government initiatives on data collection and use that provide greater security in this regard.

As a future work, it is possible to point out some promising directions, namely the crossing of information obtained about emotional states with laboratory analyses, in order to identify the impacts of the levels of hormones, vitamins, etc. in people's emotional profile.

Another line of research to be explored from this work is the prediction of human behaviour based on people's comments, networks and interactions. Thus, by knowing the emotional and social aspects that influence people in their decision-making, it may be possible to predict what behaviours a population can take.

Bibliography

[1] Z. A. Aghbari. “Array-index: a plug&search K nearest neighbors method for high-dimensional data”. In: *Data & Knowledge Engineering* 52.3 (2005), pp. 333–352. issn: 0169-023X. doi: https://doi.org/10.1016/j.datak.2004.06.015. url: https://www.sciencedirect.com/science/article/pii/S0169023X04001260 (cit. on p. 24).

[2] E. Agirre and G. Rigau. “Word Sense Disambiguation Using Conceptual Density”. In: *Proceedings of the 16th Conference on Computational Linguistics - Volume 1*. Copenhagen, Denmark: Association for Computational Linguistics, 1996, pp. 16–22. doi: 10.3115/992628.992635. url: https://doi.org/10.3115/992628.992635 (cit. on p. 43).

[3] E. Agirre and M. Stevenson. “Knowledge Sources for WSD”. In: *Word Sense Disambiguation: Algorithms and Applications*. Ed. by E. Agirre and P. Edmonds. Dordrecht: Springer Netherlands, 2006, pp. 217–251. isbn: 978-1-4020-4809-8. doi: 10.1007/978-1-4020-4809-8_8 (cit. on p. 42).

[4] G. W. Allport and H. S. Odbert. “Trait-names: A psycho-lexical study.” In: *Psychological monographs* 47.1 (1936), p. i (cit. on p. 16).

[5] M. J. V. Amorim and M. Bercht. “O uso da webcam na educação”. In: *RENOTE* 7.3 (2009), pp. 519–529 (cit. on p. 25).

[6] G. Aston and L. Burnard. *The BNC handbook: exploring the British National Corpus with SARA*. Capstone, 1998 (cit. on p. 16).

[7] J. Atserias et al. “Combining Multiple Methods for the Automatic Construction of Multilingual WordNets”. In: *CoRR* cmp-lg/9709003 (1997). url: http://arxiv.org/abs/cmp-lg/9709003 (cit. on p. 43).

[8] J. R. Averill. *A semantic atlas of emotional concepts*. American Psycholog. Ass., Journal Suppl. Abstract Service, 1975 (cit. on p. 16).

[9] S. Baccianella, A. Esuli, and F. Sebastiani. "SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining." In: vol. 10. Valletta, Malta: European Language Resources Association (ELRA), May 2010, pp. 2200–2204 (cit. on p. 19).

[10] R. Banerjee et al. "Keystroke Patterns as Prosody in Digital Writings: A Case Study with Deceptive Reviews and Essays". In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 1469–1473. doi: 10.3115/v1/D14-1155. url: https://www.aclweb.org/anthology/D14-1155 (cit. on p. 96).

[11] S. Basu et al. "A review on emotion recognition using speech". In: *2017 International Conference on Inventive Communication and Computational Technologies (ICICCT)*. 2017, pp. 109–114. doi: 10.1109/ICICCT.2017.7975169 (cit. on p. 25).

[12] R. J. Bayardo, Y. Ma, and R. Srikant. "Scaling up All Pairs Similarity Search". In: *Proceedings of the 16th International Conference on World Wide Web*. Banff, Alberta, Canada: Association for Computing Machinery, 2007, pp. 131–140. isbn: 9781595936547. doi: 10.1145/1242572.1242591. url: https://doi.org/10.1145/1242572.1242591 (cit. on p. 89).

[13] A. Bermingham and A. Smeaton. "On Using Twitter to Monitor Political Sentiment and Predict Election Results". In: *Proceedings of the Workshop on Sentiment Analysis where AI meets Psychology (SAAIP 2011)*. Chiang Mai, Thailand: Asian Federation of Natural Language Processing, Nov. 2011, pp. 2–10. url: https://www.aclweb.org/anthology/W11-3702 (cit. on p. 112).

[14] M. M. Bradley and P. J. Lang. *Affective norms for English words (ANEW): Instruction manual and affective ratings*. Tech. rep. Technical report C-1, the center for research in psychophysiology, University of Florida, 1999 (cit. on p. 20).

[15] F. Bravo-Marquez, E. Frank, and B. Pfahringer. "Positive, Negative, or Neutral: Learning an Expanded Opinion Lexicon from Emoticon-Annotated Tweets". In: *Proceedings of the 24th International Conference on Artificial Intelligence*. Buenos Aires, Argentina: AAAI Press, 2015, pp. 1229–1235. isbn: 9781577357384 (cit. on p. 88).

[16] E. Brill. "A Simple Rule-based Part of Speech Tagger". In: *Proceedings of the Third Conference on Applied Natural Language Processing*. ANLC '92. Trento, Italy: Association for Computational Linguistics, 1992, pp. 152–155. doi: 10.3115/974499.974526. url: http://dx.doi.org/10.3115/974499.974526 (cit. on p. 42).

[17] R. Burchfield. "Frequency Analysis of English Usage: Lexicon and Grammar. By W. Nelson Francis and Henry Kučera with the assistance of Andrew W. Mackie. Boston: Houghton Mifflin. 1982. x + 561". In: *Journal of English Linguistics* 18.1 (1985), pp. 64–70. doi: 10.1177/007542428501800107. eprint: https://doi.org/10.1177/007542428501800107. url: https://doi.org/10.1177/007542428501800107 (cit. on p. 44).

[18] C. Busso et al. “Analysis of Emotion Recognition Using Facial Expressions, Speech and Multimodal Information”. In: *Proceedings of the 6th International Conference on Multimodal Interfaces*. ICMI '04. State College, PA, USA: Association for Computing Machinery, 2004, pp. 205–211. isbn: 1581139950. doi: 10.1145/1027933.1027968. url: https://doi.org/10.1145/1027933.1027968 (cit. on p. 25).

[19] R. S. Campbell and J. W. Pennebaker. “The Secret Life of Pronouns”. In: *Psychological Science* 14.1 (Jan. 2003), pp. 60–65. doi: 10.1111/1467-9280.01419. url: https://doi.org/10.1111/1467-9280.01419 (cit. on p. 29).

[20] R. B. Cattell. “Sentiment or attitude? The core of a terminology problem in personality research”. In: *Journal of Personality* 9.1 (1940), pp. 6–17. doi: https://doi.org/10.1111/j.1467-6494.1940.tb02192.x. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1467-6494.1940.tb02192.x. url: https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-6494.1940.tb02192.x (cit. on p. 60).

[21] M. D. Choudhury et al. “Predicting Depression via Social Media”. In: *Proceedings of the Seventh International Conference on Weblogs and Social Media, ICWSM 2013, Cambridge, Massachusetts, USA, July 8-11, 2013*. Ed. by E. Kiciman et al. The AAAI Press, 2013. url: http://www.aaai.org/ocs/index.php/ICWSM/ICWSM13/paper/view/6124 (cit. on p. 121).

[22] G. Chrupała. “Simple Data-Driven Context-Sensitive Lemmatization”. In: *Procesamiento del Lenguaje Natural* 37.0 (2006). issn: 1989-7553. url: http://journal.sepln.org/sepln/ojs/ojs/index.php/pln/article/view/2741 (cit. on p. 40).

[23] G. L. Clore, A. Ortony, and M. A. Foss. “The psychological foundations of the affective lexicon”. In: *Journal of personality and social psychology* 53.4 (1987), p. 751 (cit. on p. 16).

[24] J. S. Coleman. *Foundations of social theory*. Harvard university press, 1994 (cit. on p. 28).

[25] P. J. Corr and G. Matthews. *The Cambridge handbook of personality psychology*. Cambridge University Press New York, 2009 (cit. on p. 29).

[26] M. A. Covington, D. Nute, and A. Vellino. *Prolog programming in depth*. Scott, Foresman & Co., 1987 (cit. on p. 35).

[27] N. V. Cuong et al. “A Maximum Entropy Model for Text Classification”. In: *The International Conference on Internet Information Retrieval 2006*. 2006, pp. 134–139 (cit. on p. 23).

[28] A. Damásio. “Ao encontro de Espinosa: as emoções sociais ea neurologia do sentir”. In: *Mem Martins: Publicações Europa-América* (2003) (cit. on p. 7).

[29] C. Darwin and P. Prodger. *The expression of the emotions in man and animals*. Oxford University Press, USA, 1998 (cit. on p. 7).

[30] K. Dave, S. Lawrence, and D. M. Pennock. “Mining the Peanut Gallery: Opinion Extraction and Semantic Classification of Product Reviews”. In: *Proceedings of the 12th International Conference on World Wide Web*. WWW '03. Budapest, Hungary: Association for Computing Machinery, 2003, pp. 519–528. isbn: 1581136803. doi: 10.1145/775152.775226. url: https://doi.org/10.1145/775152.775226 (cit. on p. 14).

[31] J. Dawson. “Suffix removal and word conflation”. In: *ALLC bulletin* 2.3 (1974), pp. 33–46 (cit. on p. 39).

[32] Domo. *Data Never Sleeps 6.0*. https://web-assets.domo.com/blog/wp-content/uploads/2018/06/18-domo-data-never-sleeps-6.png. (Accessed on 11/22/2020). June 2018 (cit. on p. 1).

[33] Domo. *Data Never Sleeps 8.0*. https://https://web-assets.domo.com/blog/wp-content/uploads/2020/08/20-data-never-sleeps-8-final-01-Resize.jpg. (Accessed on 11/22/2020). Aug. 2020 (cit. on p. 1).

[34] J. Dönges. “You Are What You Say”. In: *Scientific American Mind* 20.4 (July 2009), pp. 14–15. doi: 10.1038/scientificamericanmind0709-14. url: https://doi.org/10.1038/scientificamericanmind0709-14 (cit. on p. 29).

[35] R. Dunbar. *The human story*. Faber & Faber, 2011 (cit. on p. 28).

[36] P. Ekman. “An argument for basic emotions”. In: *Cognition & emotion* 6.3-4 (1992), pp. 169–200 (cit. on pp. 8, 95, 120, 121).

[37] P. Ekman. “Basic emotions”. In: *Handbook of cognition and emotion* 98 (1999), pp. 45–60 (cit. on p. 28).

[38] P. Ekman and W. V. Friesen. “The repertoire of nonverbal behavior: Categories, origins, usage, and coding”. In: *semiotica* 1.1 (1969), pp. 49–98 (cit. on p. 8).

[39] P. Ekman and E. L. Rosenberg. *What the face reveals: Basic and applied studies of spontaneous expression using the Facial Action Coding System (FACS)*. Oxford University Press, USA, 1997 (cit. on p. 26).

[40] P. E. Ekman and R. J. Davidson. *The nature of emotion: Fundamental questions*. Oxford University Press, 1994 (cit. on p. 95).

[41] M. El Ayadi, M. S. Kamel, and F. Karray. “Survey on speech emotion recognition: Features, classification schemes, and databases”. In: *Pattern Recognition* 44.3 (2011), pp. 572–587. issn: 0031-3203. doi: https://doi.org/10.1016/j.patcog.2010.09.020. url: https://www.sciencedirect.com/science/article/pii/S0031320310004619 (cit. on p. 25).

[42] E. H. Erikson. *Childhood and society*. WW Norton & Company, 1993 (cit. on p. 102).

[43] A. Esuli. "Automatic Generation of Lexical Resources for Opinion Mining: Models, Algorithms and Applications". In: vol. 42. 2. New York, NY, USA: Association for Computing Machinery, Nov. 2008, pp. 105–106. doi: 10.1145/1480506.1480528. url: https://doi.org/10.1145/1480506.1480528 (cit. on pp. 15, 18, 19).

[44] A. Esuli and F. Sebastiani. "Determining Term Subjectivity and Term Orientation for Opinion Mining". In: *11th Conference of the European Chapter of the Association for Computational Linguistics*. Trento, Italy: Association for Computational Linguistics, Apr. 2006. url: https://www.aclweb.org/anthology/E06-1025 (cit. on p. 18).

[45] A. Esuli and F. Sebastiani. "Determining the Semantic Orientation of Terms through Gloss Classification". In: *Proceedings of the 14th ACM International Conference on Information and Knowledge Management*. CIKM '05. Bremen, Germany: Association for Computing Machinery, 2005, pp. 617–624. isbn: 1595931406. doi: 10.1145/1099554.1099713. url: https://doi.org/10.1145/1099554.1099713 (cit. on p. 18).

[46] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth. "From Data Mining to Knowledge Discovery in Databases". In: *AI Magazine* 17.3 (Mar. 1996), p. 37. doi: 10.1609/aimag.v17i3.1230. url: https://ojs.aaai.org/index.php/aimagazine/article/view/1230 (cit. on p. 32).

[47] B. Fehr and J. A. Russell. "Concept of emotion viewed from a prototype perspective." In: *Journal of Experimental Psychology: General* 113.3 (1984), pp. 464–486. doi: 10.1037/0096-3445.113.3.464. url: https://doi.org/10.1037/0096-3445.113.3.464 (cit. on p. 7).

[48] P. J. Feldman et al. "Negative emotions and acute physiological responses to stress2". In: *Annals of Behavioral Medicine* 21.3 (Sept. 1999), pp. 216–222. issn: 0883-6612. doi: 10.1007/BF02884836. url: https://doi.org/10.1007/BF02884836 (cit. on p. 1).

[49] R. Feldman. "Techniques and Applications for Sentiment Analysis". In: *Commun. ACM* 56.4 (Apr. 2013), pp. 82–89. issn: 0001-0782. doi: 10.1145/2436256.2436274. url: https://doi.org/10.1145/2436256.2436274 (cit. on p. 113).

[50] R. Feldman and J. Sanger. *The Text Mining Handbook*. Cambridge University Press, 2006. doi: 10.1017/cbo9780511546914. url: https://doi.org/10.1017/cbo9780511546914 (cit. on pp. 32, 36).

[51] C. Fellbaum. *WordNet*. Wiley Online Library, 1998 (cit. on p. 16).

[52] E. Ferneda. "Redes neurais e sua aplicação em sistemas de recuperação de informação". In: *Ciência da Informação* 35.1 (2006) (cit. on p. 24).

[53] L. Ferrone and F. M. Zanzotto. “Symbolic, Distributed, and Distributional Representations for Natural Language Processing in the Era of Deep Learning: A Survey”. In: *Frontiers in Robotics and AI* 6 (2020), p. 153. issn: 2296-9144. doi: 10.3389/frobt.2019.00153. url: https://www.frontiersin.org/article/10.3389/frobt.2019.00153 (cit. on p. 50).

[54] G. Forney. “The viterbi algorithm”. In: *Proceedings of the IEEE* 61.3 (1973), pp. 268–278. doi: 10.1109/PROC.1973.9030 (cit. on p. 42).

[55] S. Freud. *The ego and the id*. WW Norton & Company, 1962 (cit. on p. 102).

[56] W. A. Gale, K. W. Church, and D. Yarowsky. “A method for disambiguating word senses in a large corpus”. In: *Computers and the Humanities* 26.5 (Dec. 1992), pp. 415–439. issn: 1572-8412. doi: 10.1007/BF00136984. url: https://doi.org/10.1007/BF00136984 (cit. on p. 43).

[57] M. Gazzaniga, T. Heatherton, and D. Halpern. *Ciência psicológica*. Artmed Editora, 2005 (cit. on p. 7).

[58] J. Gentry. *twitteR: R Based Twitter Client*. R package version 1.1.9. 2015. url: https://CRAN.R-project.org/package=twitteR (cit. on p. 104).

[59] D. Ghazi, D. Inkpen, and S. Szpakowicz. “Prior and Contextual Emotion of Words in Sentential Context”. In: *Comput. Speech Lang.* 28.1 (Jan. 2014), pp. 76–92. issn: 0885-2308. doi: 10.1016/j.csl.2013.04.009. url: https://doi.org/10.1016/j.csl.2013.04.009 (cit. on p. 15).

[60] K. Glanz and B. Rimer. *Theory at a Glance: A Guide for Health Promotion Practice*. NIH publication. U.S. Department of Health and Human Services, 1997. url: https://books.google.com.br/books?id=rUXiaSxFT48C (cit. on p. 27).

[61] A. Go, R. Bhayani, and L. Huang. “Twitter Sentiment Classification using Distant Supervision”. In: *Processing* (2009), pp. 1–6. url: http://www.stanford.edu/~alecmgo/papers/TwitterDistantSupervision09.pdf (cit. on p. 104).

[62] D. Goleman. *Emotional intelligence : why it can matter more than IQ*. English. Bloomsbury London, 1996, p. 352. isbn: 0747529825 (cit. on p. 1).

[63] H. Gu, G. Su, and C. Du. “Feature points extraction from faces”. In: *Image and vision computing NZ* 26 (2003), pp. 154–158 (cit. on p. 26).

[64] G. Guo et al. “Using kNN model for automatic text categorization”. In: *Soft Computing* 10.5 (Mar. 2006), pp. 423–430. issn: 1433-7479. doi: 10.1007/s00500-005-0503-y. url: https://doi.org/10.1007/s00500-005-0503-y (cit. on p. 24).

[65] V. Gupta, G. S. Lehal, et al. “A survey of text mining techniques and applications”. In: *Journal of emerging technologies in web intelligence* 1.1 (2009), pp. 60–76 (cit. on p. 34).

[66] L. Guthrie et al. “The Role of Lexicons in Natural Language Processing”. In: *Commun. ACM* 39.1 (Jan. 1996), pp. 63–72. issn: 0001-0782. doi: 10.1145/234173.234204. url: https://doi.org/10.1145/234173.234204 (cit. on p. 16).

[67] M. Hall et al. “The WEKA Data Mining Software: An Update”. In: *SIGKDD Explor. Newsl.* 11.1 (Nov. 2009), pp. 10–18. issn: 1931-0145. doi: 10.1145/1656274.1656278. url: https://doi.org/10.1145/1656274.1656278 (cit. on p. 85).

[68] E.-H. (Han, G. Karypis, and V. Kumar. “Text Categorization Using Weight Adjusted k-Nearest Neighbor Classification”. In: *Advances in Knowledge Discovery and Data Mining*. Ed. by D. Cheung, G. J. Williams, and Q. Li. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, pp. 53–65. isbn: 978-3-540-45357-4 (cit. on p. 24).

[69] M. H. M. Hassin, A. A. Aziz, and N. M. Norwawi. “Affective computing: knowing how you feel”. In: *IN: The National Seminar of Science Technology and Social Science (STSS ‘04), UiTM Pahang*. 2004 (cit. on p. 14).

[70] V. Hatzivassiloglou et al. “Simfinder: A flexible clustering tool for summarization”. In: Columbia University, 2001. doi: 10.7916/D87S7X4R. url: https://academiccommons.columbia.edu/doi/10.7916/D87S7X4R (cit. on p. 44).

[71] N. A. Hendy and H. Farag. “Emotion recognition using neural network: A comparative study”. In: *Proceedings of World Academy of Science, Engineering and Technology*. 2 75. World Academy of Science, Engineering and Technology (WASET). 2013, p. 791 (cit. on p. 24).

[72] B. Heredia, J. D. Prusa, and T. M. Khoshgoftaar. “Location-based twitter sentiment analysis for predicting the us 2016 presidential election”. In: *The Thirty-First International Flairs Conference*. 2018 (cit. on p. 112).

[73] M. Hofmann. “Support vector machines-kernels and the kernel trick”. In: *An elaboration for the Hauptseminar Reading Club SVM* (2006) (cit. on p. 22).

[74] C.-W. Hsu, C.-C. Chang, and C.-J. Lin. *A Practical Guide to Support Vector Classification*. Tech. rep. Department of Computer Science, National Taiwan University, 2003. url: http://www.csie.ntu.edu.tw/~cjlin/papers.html (cit. on p. 22).

[75] X. Hu et al. “Exploiting Social Relations for Sentiment Analysis in Microblogging”. In: (2013), pp. 537–546. doi: 10.1145/2433396.2433465. url: https://doi.org/10.1145/2433396.2433465 (cit. on pp. 8, 103).

[76] X. Hu et al. “Exploiting social relations for sentiment analysis in microblogging”. In: *Proceedings of the sixth ACM international conference on Web search and data mining*. ACM. 2013, pp. 537–546 (cit. on p. 104).

[77] D. A. Hull. “Stemming Algorithms: A Case Study for Detailed Evaluation”. In: *J. Am. Soc. Inf. Sci.* 47.1 (Jan. 1996), pp. 70–84. issn: 0002-8231 (cit. on p. 38).

[78] A. Husseini Orabi et al. “Deep Learning for Depression Detection of Twitter Users”. In: *Proceedings of the Fifth Workshop on Computational Linguistics and Clinical Psychology: From Keyboard to Clinic*. New Orleans, LA: Association for Computational Linguistics, June 2018, pp. 88–97. doi: 10.18653/v1/W18-0609. url: https://www.aclweb.org/anthology/W18-0609 (cit. on p. 121).

[79] N. Ide and J. Véronis. “Introduction to the special issue on word sense disambiguation: the state of the art”. In: *Computational linguistics* 24.1 (1998), pp. 2–40 (cit. on p. 42).

[80] C. E. Izard. *Human emotions*. Springer Science & Business Media, 2013 (cit. on p. 95).

[81] P. A. Jaques and R. M. Vicari. “Estado da arte em ambientes inteligentes de aprendizagem que consideram a afetividade do aluno”. In: *Revista informática na educação: teoria & prática* 8.1 (2005), pp. 15–38 (cit. on p. 12).

[82] J. Jiang. “Information Extraction from Text”. In: (2012). Ed. by C. C. Aggarwal and C. Zhai, pp. 11–41. doi: 10.1007/978-1-4614-3223-4_2. url: https://doi.org/10.1007/978-1-4614-3223-4_2 (cit. on p. 34).

[83] S. Jiang et al. “An Improved K-Nearest-Neighbor Algorithm for Text Categorization”. In: *Expert Syst. Appl.* 39.1 (Jan. 2012), pp. 1503–1509. issn: 0957-4174. doi: 10.1016/j.eswa.2011.08.040. url: https://doi.org/10.1016/j.eswa.2011.08.040 (cit. on p. 24).

[84] M. L. Jockers. *Syuzhet: Extract Sentiment and Plot Arcs from Text*. 2015. url: https://github.com/mjockers/syuzhet (cit. on p. 98).

[85] B. M. H. Kagan J. and L. R. M. “Human Behaviour”. In: *Encyclopedia Britannica*. 2020 (cit. on p. 27).

[86] H. Kanayama and T. Nasukawa. “Fully Automatic Lexicon Expansion for Domain-Oriented Sentiment Analysis”. In: *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*. Sydney, Australia: Association for Computational Linguistics, 2006, pp. 355–363. isbn: 1932432736 (cit. on p. 88).

[87] R. M. Kaplan. “A method for tokenizing text”. In: *Inquiries into words, constraints and contexts* (2005), p. 55 (cit. on p. 37).

[88] D. Keltner and P. Ekman. “Emotion: an overview”. In: *Encyclopedia of psychology* 3 (2000), pp. 162–167 (cit. on p. 8).

[89] W. G. Kennedy. “Modelling Human Behaviour in Agent-Based Models”. In: *Agent-Based Models of Geographical Systems*. Ed. by A. J. Heppenstall et al. Dordrecht: Springer Netherlands, 2012, pp. 167–179. isbn: 978-90-481-8927-4. doi: 10.1007/978-90-481-8927-4_9. url: https://doi.org/10.1007/978-90-481-8927-4_9 (cit. on p. 28).

[90] D. Kiela, F. Hill, and S. Clark. "Specializing Word Embeddings for Similarity or Relatedness". In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Lisbon, Portugal: Association for Computational Linguistics, Sept. 2015, pp. 2044–2048. doi: 10.18653/v1/D15-1242. url: https://www.aclweb.org/anthology/D15-1242 (cit. on p. 89).

[91] E. Kim et al. "Detecting sadness in 140 characters: Sentiment analysis of mourning michael jackson on twitter". In: *Web Ecology* 3 (2009), pp. 1–15 (cit. on p. 103).

[92] E. Komulainen et al. "The Effect of Personality on Daily Life Emotional Processes". In: *PLoS ONE* 9.10 (Oct. 2014). Ed. by J. Yuan, e110907. doi: 10.1371/journal.pone.0110907. url: https://doi.org/10.1371/journal.pone.0110907 (cit. on pp. 29, 72).

[93] R. Krovetz. "Viewing Morphology as an Inference Process". In: Pittsburgh, Pennsylvania, USA: Association for Computing Machinery, 1993, pp. 191–202. isbn: 0897916050. doi: 10.1145/160688.160718. url: https://doi.org/10.1145/160688.160718 (cit. on p. 38).

[94] M. J. Kusner et al. "From Word Embeddings to Document Distances". In: *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*. ICML'15. Lille, France: JMLR.org, 2015, pp. 957–966 (cit. on p. 56).

[95] T. K. Landauer, P. W. Foltz, and D. Laham. "An introduction to latent semantic analysis". In: *Discourse Processes* 25.2-3 (1998), pp. 259–284. doi: 10.1080/01638539809545028. eprint: https://doi.org/10.1080/01638539809545028. url: https://doi.org/10.1080/01638539809545028 (cit. on p. 33).

[96] P. J. Lang. "The emotion probe: studies of motivation and attention." In: *American psychologist* 50.5 (1995), p. 372 (cit. on p. 7).

[97] B. Latané. "The psychology of social impact." In: *American psychologist* 36.4 (1981), p. 343 (cit. on p. 28).

[98] C. Leacock, G. Towell, and E. M. Voorhees. "Towards Building Contextual Representations of Word Senses Using Statistical Models". In: *Corpus Processing for Lexical Acquisition*. Cambridge, MA, USA: MIT Press, 1996, pp. 97–113. isbn: 026202392X (cit. on p. 43).

[99] C. M. Lee and S. Narayanan. "Toward detecting emotions in spoken dialogs". In: *IEEE Transactions on Speech and Audio Processing* 13.2 (2005), pp. 293–303. doi: 10.1109/TSA.2004.838534 (cit. on p. 24).

[100] M. Lesk. "Automatic Sense Disambiguation Using Machine Readable Dictionaries: How to Tell a Pine Cone from an Ice Cream Cone". In: *Proceedings of the 5th Annual International Conference on Systems Documentation*. Toronto, Ontario, Canada: Association for Computing Machinery, 1986, pp. 24–26. isbn: 0897912241. doi: 10.1145/318723.318728 (cit. on p. 42).

[101] H. Leventhal and K. Scherer. "The Relationship of Emotion to Cognition: A Functional Approach to a Semantic Controversy". In: *Cognition and Emotion* 1.1 (1987), pp. 3–28. doi: 10.1080/02699938708408361. eprint: https://doi.org/10.1080/02699938708408361. url: https://doi.org/10.1080/02699938708408361 (cit. on pp. 9, 103).

[102] W. Ling et al. "Two/Too Simple Adaptations of Word2Vec for Syntax Problems". In: *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Denver, Colorado: Association for Computational Linguistics, May 2015, pp. 1299–1304. doi: 10.3115/v1/N15-1142 (cit. on pp. 49, 50).

[103] B. Liu. "Sentiment Analysis and Opinion Mining". In: *Synthesis Lectures on Human Language Technologies* 5.1 (May 2012), pp. 1–167. doi: 10.2200/s00416ed1v01y201204hlt016. url: https://doi.org/10.2200/s00416ed1v01y201204hlt016 (cit. on p. 15).

[104] K.-L. Liu, W.-J. Li, and M. Guo. "Emoticon Smoothed Language Models for Twitter Sentiment Analysis". In: *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*. Toronto, Ontario, Canada: AAAI Press, 2012, pp. 1678–1684 (cit. on p. 104).

[105] L. Liu. *Opinions, Sentiment, and Emotion in Text*. Cambridge University Press, 2015, p. 381 (cit. on p. 14).

[106] L. Liu and M. T. Ozsu. *Encyclopedia of database systems*. Vol. 6. Springer Berlin, Heidelberg, Germany, 2009 (cit. on p. 33).

[107] R. T.-W. Lo, B. He, and I. Ounis. "Automatically building a stopword list for an information retrieval system". In: *Journal on Digital Information Management: Special Issue on the 5th Dutch-Belgian Information Retrieval Workshop (DIR)*. Vol. 5. 2005, pp. 17–24 (cit. on p. 44).

[108] J. Loevinger. "The meaning and measurement of ego development". In: *American Psychologist* 21.3 (1966), p. 195 (cit. on p. 102).

[109] J. B. Lovins. "Development of a stemming algorithm". In: *Mech. Translat. & Comp. Linguistics* 11.1-2 (1968), pp. 22–31 (cit. on pp. 39, 106).

[110] T.-J. Lu. "Semi-supervised microblog sentiment analysis using social relation and text similarity". In: *2015 International Conference on Big Data and Smart Computing (BIGCOMP)*. 2015, pp. 194–201. doi: 10.1109/35021BIGCOMP.2015.7072831 (cit. on p. 104).

[111] F. Mairesse et al. "Using Linguistic Cues for the Automatic Recognition of Personality in Conversation and Text". In: *J. Artif. Int. Res.* 30.1 (Nov. 2007), pp. 457–500. issn: 1076-9757 (cit. on p. 29).

[112] N. Majumder et al. "Deep Learning-Based Document Modeling for Personality Detection from Text". In: *IEEE Intelligent Systems* 32.2 (2017), pp. 74–79. doi: 10.1109/MIS.2017.23 (cit. on pp. 28, 29).

[113] C. Manning et al. “The Stanford CoreNLP Natural Language Processing Toolkit”. In: *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*. Baltimore, Maryland: Association for Computational Linguistics, June 2014, pp. 55–60. doi: 10.3115/v1/P14-5010. url: https://www.aclweb.org/anthology/P14-5010 (cit. on pp. 41, 82, 88, 98, 106, 108).

[114] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval*. USA: Cambridge University Press, 2008. isbn: 0521865719 (cit. on p. 33).

[115] M. P. Marcus, M. A. Marcinkiewicz, and B. Santorini. “Building a Large Annotated Corpus of English: The Penn Treebank”. In: *Comput. Linguist.* 19.2 (June 1993), pp. 313–330. issn: 0891-2017 (cit. on p. 108).

[116] J. H. Martin and D. Jurafsky. *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition*. Pearson/Prentice Hall Upper Saddle River, 2009 (cit. on p. 44).

[117] R. Martins, P. Almeida José João an Henriques, and P. Novais. “Domain Identification Through Sentiment Analysis”. In: *Distributed Computing and Artificial Intelligence, 15th International Conference*. Ed. by F. De La Prieta, S. Omatu, and A. Fernández-Caballero. Cham: Springer International Publishing, 2019, pp. 276–283. isbn: 978-3-319-94649-8 (cit. on p. 104).

[118] R. Martins et al. “A sentiment analysis approach to increase authorship identification”. In: *Expert Systems* n/a.n/a (). e12469 10.1111/exsy.12469, e12469. doi: https://doi.org/10.1111/exsy.12469. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/exsy.12469. url: https://onlinelibrary.wiley.com/doi/abs/10.1111/exsy.12469 (cit. on pp. 88, 96).

[119] R. Martins et al. “Creating a Social Media-based Personal Emotional Lexicon”. In: *Proceedings of the 24th Brazilian Symposium on Multimedia and the Web*. Salvador, BA, Brazil: ACM, 2018, pp. 261–264. isbn: 978-1-4503-5867-5. doi: 10.1145/3243082.3264668. url: http://doi.acm.org/10.1145/3243082.3264668 (cit. on p. 113).

[120] R. Martins et al. “Domain Identification Through Sentiment Analysis”. In: *Distributed Computing and Artificial Intelligence, 15th International Conference*. Ed. by F. De La Prieta, S. Omatu, and A. Fernández-Caballero. Cham: Springer International Publishing, 2019, pp. 276–283. isbn: 978-3-319-94649-8 (cit. on p. 30).

[121] R. Martins et al. “Predicting Performance Problems Through Emotional Analysis (Short Paper)”. In: *7th Symposium on Languages, Applications and Technologies, SLATE 2018, June 21-22, 2018, Guimaraes, Portugal*. Ed. by P. R. Henriques et al. Vol. 62. OASICS. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018, 19:1–19:9. doi: 10.4230/OASIcs.SLATE.2018.19. url: https://doi.org/10.4230/OASIcs.SLATE.2018.19 (cit. on p. 112).

[122] P. Martiskainen et al. “Cow behaviour pattern recognition using a three-dimensional accelerometer and support vector machines”. In: *Applied Animal Behaviour Science* 119.1 (2009), pp. 32–38. issn: 0168-1591. doi: https://doi.org/10.1016/j.applanim.2009.03.005. url: https://www.sciencedirect.com/science/article/pii/S0168159109000951 (cit. on p. 27).

[123] A. H. Maslow. “A theory of human motivation.” In: *Psychological review* 50.4 (1943), p. 370 (cit. on p. 28).

[124] R. R. McCrae and O. P. John. “An Introduction to the Five-Factor Model and Its Applications”. In: *Journal of Personality* 60.2 (1992), pp. 175–215. doi: https://doi.org/10.1111/j.1467-6494.1992.tb00970.x. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1467-6494.1992.tb00970.x. url: https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-6494.1992.tb00970.x (cit. on p. 95).

[125] S. Meier, P. R. McCarthy, and R. R. Schmeck. “Validity of self-efficacy as a predictor of writing performance”. In: *Cognitive Therapy and Research* 8.2 (Apr. 1984), pp. 107–120. issn: 1573-2819. doi: 10.1007/BF01173038. url: https://doi.org/10.1007/BF01173038 (cit. on p. 96).

[126] D. Meyer, K. Hornik, and I. Feinerer. “Text mining infrastructure in R”. In: *Journal of statistical software* 25.5 (2008), pp. 1–54 (cit. on p. 106).

[127] T. Mikolov, W.-t. Yih, and G. Zweig. “Linguistic Regularities in Continuous Space Word Representations”. In: *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Atlanta, Georgia: Association for Computational Linguistics, June 2013, pp. 746–751 (cit. on p. 49).

[128] T. Mikolov et al. “Distributed Representations of Words and Phrases and Their Compositionality”. In: *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*. Lake Tahoe, Nevada: Curran Associates Inc., 2013, pp. 3111–3119 (cit. on pp. 49, 114).

[129] G. Miller and C. Fellbaum. *Wordnet: An electronic lexical database*. 1998 (cit. on p. 17).

[130] G. A. Miller et al. “Introduction to WordNet: An on-line lexical database”. In: *International journal of lexicography* 3.4 (1990), pp. 235–244 (cit. on p. 17).

[131] G. A. Miller. “WordNet: A Lexical Database for English”. In: *Commun. ACM* 38.11 (Nov. 1995), pp. 39–41. issn: 0001-0782. doi: 10.1145/219717.219748. url: https://doi.org/10.1145/219717.219748 (cit. on pp. 16, 17, 75, 90).

[132] S. Mohammad. "Word Affect Intensities". In: *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. Miyazaki, Japan: European Language Resources Association (ELRA), May 2018. url: https://www.aclweb.org/anthology/L18-1027 (cit. on p. 124).

[133] S. M. Mohammad and P. D. Turney. "CROWDSOURCING A WORD-EMOTION ASSOCIATION LEXICON". In: *Computational Intelligence* 29.3 (Sept. 2012), pp. 436–465. doi: 10.1111/j.1467-8640.2012.00460.x. url: https://doi.org/10.1111/j.1467-8640.2012.00460.x (cit. on pp. 82, 90, 97–99, 106, 114, 124).

[134] S. M. Mohammad and P. D. Turney. "Emotions Evoked by Common Words and Phrases: Using Mechanical Turk to Create an Emotion Lexicon". In: *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*. CAAGET '10. Association for Computational Linguistics, 2010, pp. 26–34 (cit. on pp. 16, 20, 73).

[135] A. Montoyo. "Método basado en marcas de especificidad para WSD". In: *Procesamiento del Lenguaje Natural* 26 (2000) (cit. on p. 43).

[136] R. J. Mooney. "Comparative experiments on disambiguating word senses: An illustration of the role of bias in machine learning". In: *Proceedings of the 1st Conference on Empirical Methods in Natural Language Processing (EMNLP), Philadelphia, U.S.A.* (1996), pp. 82–91 (cit. on p. 43).

[137] M. Munezero et al. "Are They Different? Affect, Feeling, Emotion, Sentiment, and Opinion Detection in Text". In: *IEEE Transactions on Affective Computing* 5.2 (2014), pp. 101–111. doi: 10.1109/TAFFC.2014.2317187 (cit. on p. 14).

[138] T. Nasukawa and J. Yi. "Sentiment Analysis: Capturing Favorability Using Natural Language Processing". In: *Proceedings of the 2nd International Conference on Knowledge Capture*. K-CAP '03. Sanibel Island, FL, USA: Association for Computing Machinery, 2003, pp. 70–77. isbn: 1581135831. doi: 10.1145/945645.945658. url: https://doi.org/10.1145/945645.945658 (cit. on p. 14).

[139] H. T. Ng and H. B. Lee. "Integrating Multiple Knowledge Sources to Disambiguate Word Sense: An Exemplar-Based Approach". In: *Proceedings of the 34th Annual Meeting on Association for Computational Linguistics*. ACL '96. Santa Cruz, California: Association for Computational Linguistics, 1996, pp. 40–47. doi: 10.3115/981863.981869. url: https://doi.org/10.3115/981863.981869 (cit. on p. 43).

[140] A. Nijholt. *Humor and embodied conversational agents*. Centre for Telematics and Information Technology, University of Twente, 2003 (cit. on p. 7).

[141] B. O'Connor et al. "From tweets to polls: Linking text sentiment to public opinion time series." In: *ICWSM* 11.122-129 (2010), pp. 1–2 (cit. on p. 103).

[142] N. Oliver, B. Rosario, and A. Pentland. "A Bayesian computer vision system for modeling human interactions". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22.8 (2000), pp. 831–843. doi: 10.1109/34.868684 (cit. on p. 27).

[143] W. H. Organization et al. *Depression and other common mental disorders: global health estimates*. Tech. rep. World Health Organization, 2017. url: https://apps.who.int/iris/bitstream/handle/10665/254610/WHO-MSD-MER-2017.2-eng.pdf (cit. on p. 120).

[144] A. Ortony, G. L. Clore, and A. Collins. *The cognitive structure of emotions*. Cambridge university press, 1990 (cit. on pp. 11, 14).

[145] A. Ortony, G. L. Clore, and M. A. Foss. "The Referential Structure of the Affective Lexicon". In: *Cognitive Science* 11.3 (1987), pp. 341–364. doi: https://doi.org/10.1207/s15516709cog1103_4 (cit. on p. 16).

[146] C. D. Paice. "An Evaluation Method for Stemming Algorithms". In: *SIGIR '94*. Ed. by B. W. Croft and C. J. van Rijsbergen. London: Springer London, 1994, pp. 42–50. isbn: 978-1-4471-2099-5 (cit. on p. 40).

[147] C. D. Paice. "Another Stemmer". In: vol. 24. 3. New York, NY, USA: Association for Computing Machinery, Nov. 1990, pp. 56–61. doi: 10.1145/101306.101310 (cit. on p. 39).

[148] A. Pak and P. Paroubek. "Twitter as a Corpus for Sentiment Analysis and Opinion Mining". In: *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*. Valletta, Malta: European Language Resources Association (ELRA), May 2010. url: http://www.lrec-conf.org/proceedings/lrec2010/pdf/385_Paper.pdf (cit. on p. 104).

[149] M. Palmer, D. Gildea, and P. Kingsbury. "The proposition bank: An annotated corpus of semantic roles". In: *Computational linguistics* 31.1 (2005), pp. 71–106 (cit. on p. 16).

[150] B. Pang and L. Lee. "Opinion Mining and Sentiment Analysis". In: *Found. Trends Inf. Retr.* 2.1–2 (Jan. 2008), pp. 1–135. issn: 1554-0669. doi: 10.1561/1500000011. url: https://doi.org/10.1561/1500000011 (cit. on p. 103).

[151] B. Pang, L. Lee, and S. Vaithyanathan. "Thumbs up? Sentiment Classification Using Machine Learning Techniques". In: *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing - Volume 10*. EMNLP '02. USA: Association for Computational Linguistics, 2002, pp. 79–86. doi: 10.3115/1118693.1118704. url: https://doi.org/10.3115/1118693.1118704 (cit. on p. 21).

[152] V. M. Papadakis et al. "A computer-vision system and methodology for the analysis of fish behavior". In: *Aquacultural Engineering* 46 (2012), pp. 53–59. issn: 0144-8609. doi: https://doi.org/10.1016/j.aquaeng.2011.11.002. url: https://www.sciencedirect.com/science/article/pii/S014486091100080X (cit. on p. 27).

[153] "Part-of-Speech Tagging Using Stochastic Techniques". In: *An Introduction to Language Processing with Perl and Prolog: An Outline of Theories, Implementation, and Application with Special Consideration of English, French, and German*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 163–184. isbn: 978-3-540-34336-3. doi: 10.1007/3-540-34336-9_7. url: https://doi.org/10.1007/3-540-34336-9_7 (cit. on p. 42).

[154] J. Pennington, R. Socher, and C. Manning. "GloVe: Global Vectors for Word Representation". In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 1532–1543. doi: 10.3115/v1/D14-1162. url: https://www.aclweb.org/anthology/D14-1162 (cit. on pp. 51, 89).

[155] R. W. Picard. *Affective computing*. Vol. 252. MIT press Cambridge, 1997 (cit. on pp. 2, 14).

[156] R. Plutchik. "Chapter 1 - A General Psychoevolutionary Theory of Emotion". In: *Theories of Emotion*. Academic Press, Jan. 1980, pp. 3–33. isbn: 978-0-12-558701-3. url: https://www.sciencedirect.com/science/article/pii/B9780125587013500077 (cit. on pp. 95, 97, 98).

[157] R. Plutchik. "Emotions: A general psychoevolutionary theory". In: *Approaches to emotion* 1984 (1984), pp. 197–219 (cit. on pp. 2, 10, 11, 28, 84, 103, 107, 114, 120).

[158] M. Popovič and P. Willett. "The effectiveness of stemming for natural-language access to Slovene textual data". In: *Journal of the American Society for Information Science* 43.5 (1992), pp. 384–390. doi: https://doi.org/10.1002/(SICI)1097-4571(199206)43:5<384::AID-ASI6>3.0.CO;2-L (cit. on p. 38).

[159] M. F. Porter. "An algorithm for suffix stripping". In: *Program* 14.3 (1980), pp. 130–137 (cit. on pp. 39, 44).

[160] M. Purver and S. Battersby. "Experimenting with Distant Supervision for Emotion Classification". In: *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*. Avignon, France: Association for Computational Linguistics, 2012, pp. 482–491. isbn: 9781937284190 (cit. on p. 8).

[161] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria, 2017. url: https://www.R-project.org/ (cit. on p. 104).

[162] L. Rabiner and B. Juang. "An introduction to hidden Markov models". In: *IEEE ASSP Magazine* 3.1 (1986), pp. 4–16. doi: 10.1109/MASSP.1986.1165342 (cit. on p. 42).

[163] F. Rangel Pardo and P. Rosso. "On the identification of emotions and authors'gender in Facebook comments on the basis of their writing style". In: vol. 1096. Jan. 2013, pp. 34–46 (cit. on p. 15).

[164] B. Reeves and C. Nass. *How people treat computers, television, and new media like real people and places*. CSLI Publications and Cambridge university press, 1996 (cit. on p. 14).

[165] I. Rish. "An empirical study of the naive Bayes classifier". In: *IJCAI 2001 workshop on empirical methods in artificial intelligence*. Vol. 3. IBM New York. 2001, pp. 41–46 (cit. on p. 21).

[166] E. J. Rohn and E. J. Rohn. *The treasury of quotes*. Health Communications, 1994 (cit. on p. 102).

[167] I. J. Roseman and C. A. Smith. "Appraisal theory". In: *Appraisal processes in emotion: Theory, methods, research* (2001), pp. 3–19 (cit. on p. 14).

[168] M. E. Ruiz and P. Srinivasan. "Automatic text categorization using neural networks". In: *Proceedings of the 8th ASIS SIG/CR Workshop on Classification Research*. 1998, pp. 59–72 (cit. on p. 24).

[169] J. A. Russell. "A circumplex model of affect." In: *Journal of Personality and Social Psychology* 39.6 (1980), pp. 1161–1178. doi: 10.1037/h0077714. url: https://doi.org/10.1037/h0077714 (cit. on pp. 9, 10).

[170] S. Russell and P. Norvig. "Intelligence Artificial: A modern approach". In: *Artificial Intelligence* 25 (1995), p. 27 (cit. on p. 35).

[171] S.Ramasundaram and S.P.Victor. "Article:Text Categorization by Backpropagation Network". In: *International Journal of Computer Applications* 8.6 (Oct. 2010). Published By Foundation of Computer Science, pp. 1–5. doi: 10.5120/1217-1754 (cit. on p. 24).

[172] G. Salton and C. Buckley. "Term-Weighting Approaches in Automatic Text Retrieval". In: *Inf. Process. Manage.* 24.5 (Aug. 1988), pp. 513–523. issn: 0306-4573. doi: 10.1016/0306-4573(88)90021-0. url: https://doi.org/10.1016/0306-4573(88)90021-0 (cit. on p. 43).

[173] S. L. Salzberg. *C4.5: Programs for Machine Learning by J. Ross Quinlan. Morgan Kaufmann Publishers, Inc., 1993*. Vol. 16. 3. Sept. 1994, pp. 235–240. doi: 10.1007/BF00993309. url: https://doi.org/10.1007/BF00993309 (cit. on p. 43).

[174] W. R. dos Santos, R. M. S. Ramos, and I. Paraboni. "Computational personality recognition from Facebook text: psycholinguistic features, words and facets". In: *New Review of Hypermedia and Multimedia* 25.4 (2019), pp. 268–287. doi: 10.1080/13614568.2020.1722761. eprint: https://doi.org/10.1080/13614568.2020.1722761. url: https://doi.org/10.1080/13614568.2020.1722761 (cit. on p. 30).

[175] E. Sapir. "Personality//Encyclopedia of the social sciences". In: *Seligman E., Johnson A* (1934), pp. 85–88 (cit. on p. 102).

[176] S. Schachter and J. Singer. "Cognitive, social, and physiological determinants of emotional state." In: *Psychological review* 69.5 (1962), p. 379 (cit. on p. 60).

[177] J. Schafer. *Reading People by the Words They Speak | Psychology Today*. https://www.psychologytoday.com/us/blog/let-their-words-do-the-talking/201106/reading-people-the-words-they-speak. (Accessed on 04/13/2020). June 2011 (cit. on pp. 29, 72, 120).

[178] K. R. Scherer. "Studying the emotion-antecedent appraisal process: An expert system approach". In: *Cognition & Emotion* 7.3-4 (1993), pp. 325–355 (cit. on p. 14).

[179] K. R. Scherer, T. Dalgleish, and M. Power. *Handbook of Cognition and Emotion*. Ed. by T. Dalgleish and M. J. Power. John Wiley & Sons, Ltd, Feb. 1999. doi: 10.1002/0470013494. url: https://doi.org/10.1002/0470013494 (cit. on pp. 10, 103).

[180] F. Sebastiani. "Machine Learning in Automated Text Categorization". In: *ACM Comput. Surv.* 34.1 (Mar. 2002), pp. 1–47. issn: 0360-0300. doi: 10.1145/505282.505283. url: https://doi.org/10.1145/505282.505283 (cit. on p. 24).

[181] T. Segaran. *Programming collective intelligence: building smart web 2.0 applications*. "O'Reilly Media, Inc.", 2007 (cit. on p. 21).

[182] Y.-S. Seol, D.-J. Kim, and H.-W. Kim. "Emotion recognition from text using knowledge-based ANN". In: *ITC-CSCC: International Technical Conference on Circuits Systems, Computers and Communications*. 2008, pp. 1569–1572 (cit. on p. 24).

[183] E. T. Al-Shammari. "Towards an Error-Free Stemming". In: *IADIS European Conf. Data Mining*. 2008, pp. 160–163 (cit. on p. 40).

[184] R. A. Sherman et al. "The independent effects of personality and situations on real-time expressions of behavior and emotion." In: *Journal of Personality and Social Psychology* 109.5 (2015), pp. 872–888. doi: 10.1037/pspp0000036. url: https://doi.org/10.1037/pspp0000036 (cit. on pp. 29, 72).

[185] S. R. Sirsat, V. Chavan, and H. S. Mahalle. "Strength and accuracy analysis of affix removal stemming algorithms". In: *International Journal of Computer Science and Information Technologies* 4.2 (2013), pp. 265–269 (cit. on p. 39).

[186] P. J. Stone, D. C. Dunphy, and M. S. Smith. *The general inquirer: A computer approach to content analysis*. MIT press, 1966 (cit. on p. 19).

[187] C. Strapparava and R. Mihalcea. "Learning to Identify Emotions in Text". In: *Proceedings of the 2008 ACM Symposium on Applied Computing*. Fortaleza, Ceara, Brazil: Association for Computing Machinery, 2008, pp. 1556–1560. isbn: 9781595937537. doi: 10.1145/1363686.1364052. url: https://doi.org/10.1145/1363686.1364052 (cit. on p. 8).

[188] C. Strapparava and A. Valitutti. "WordNet Affect: an Affective Extension of WordNet". In: *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*. Lisbon, Portugal: European Language Resources Association (ELRA), May 2004. url: http://www.lrec-conf.org/proceedings/lrec2004/pdf/369.pdf (cit. on pp. 15, 17).

[189] M. Sussna. "Word Sense Disambiguation for Free-Text Indexing Using a Massive Semantic Network". In: *Proceedings of the Second International Conference on Information and Knowledge Management*. Washington, D.C., USA: Association for Computing Machinery, 1993, pp. 67–74. isbn: 0897916263. doi: 10.1145/170088.170106. url: https://doi.org/10.1145/170088.170106 (cit. on p. 43).

[190] P.-N. Tan, M. Steinbach, and V. Kumar. *Introdução ao datamining: mineração de dados*. Ciência Moderna, 2009 (cit. on p. 24).

[191] J. G. Thanikkal and M. Danish. "A Novel Approach to Improve Spam Detection using SDS Algorithm". In: *International Journal for Innovative Research in Science & Technology* 1.12 (), pp. 306–310. url: http://www.ijirst.org/articles/IJIRSTV1I12160.pdf (cit. on p. 52).

[192] M. Thelwall. "The Heart and Soul of the Web? Sentiment Strength Detection in the Social Web with SentiStrength". In: (2017). Ed. by J. A. Holyst, pp. 119–134. doi: 10.1007/978-3-319-43639-5_7. url: https://doi.org/10.1007/978-3-319-43639-5_7 (cit. on p. 19).

[193] M. Thelwall et al. "Sentiment Strength Detection in Short Informal Text". In: *J. Am. Soc. Inf. Sci. Technol.* 61.12 (Dec. 2010), pp. 2544–2558. issn: 1532-2882 (cit. on p. 96).

[194] S. Tsugawa et al. "Recognizing Depression from Twitter Activity". In: *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. CHI '15. Seoul, Republic of Korea: Association for Computing Machinery, 2015, pp. 3187–3196. isbn: 9781450331456. doi: 10.1145/2702123.2702280 (cit. on p. 121).

[195] A. Tumasjan et al. "Predicting Elections with Twitter: What 140 Characters Reveal about Political Sentiment". In: vol. 4. 1. May 2010. url: https://ojs.aaai.org/index.php/ICWSM/article/view/14009 (cit. on p. 112).

[196] E. C. Tupes and R. E. Christal. "Recurrent Personality Factors Based on Trait Ratings". In: *Journal of Personality* 60.2 (June 1992), pp. 225–251. doi: 10.1111/j.1467-6494.1992.tb00973.x. url: https://doi.org/10.1111/j.1467-6494.1992.tb00973.x (cit. on p. 29).

[197] J. Turian, L. Ratinov, and Y. Bengio. "Word Representations: A Simple and General Method for Semi-Supervised Learning". In: *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. Uppsala, Sweden: Association for Computational Linguistics, 2010, pp. 384–394 (cit. on p. 46).

[198] M. S. Unluturk, K. Oguz, and C. Atay. "Emotion recognition using neural networks". In: *Proceedings of the 10th WSEAS International Conference on Neural Networks, Prague, Czech Republic*. 2009, pp. 82–85 (cit. on p. 24).

[199] S. Vijayarani, M. J. Ilamathi, and M. Nithya. "Preprocessing techniques for text mining-an overview". In: *International Journal of Computer Science & Communication Networks* 5.1 (2015), pp. 7–16 (cit. on pp. 24, 35).

[200] E. M. Voorhees. "Query Expansion using Lexical-Semantic Relations". In: *SIGIR '94*. Ed. by B. W. Croft and C. J. van Rijsbergen. London: Springer London, 1994, pp. 61–69. isbn: 978-1-4471-2099-5 (cit. on p. 38).

[201] H. Wang et al. "A System for Real-Time Twitter Sentiment Analysis of 2012 U.S. Presidential Election Cycle". In: *Proceedings of the ACL 2012 System Demonstrations*. Jeju Island, Korea: Association for Computational Linguistics, 2012, pp. 115–120 (cit. on p. 112).

[202] D. Watson et al. "The two general activation systems of affect: Structural findings, evolutionary considerations, and psychobiological evidence". In: *Journal of Personality and Social Psychology* 76.5 (1999), pp. 820–838. doi: 10.1037/0022-3514.76.5.820. url: https://doi.org/10.1037/0022-3514.76.5.820 (cit. on p. 10).

[203] H.-C. Yang and Z.-R. Huang. "Mining personality traits from social messages for game recommender systems". In: *Knowledge-Based Systems* 165 (2019), pp. 157–168. issn: 0950-7051. doi: https://doi.org/10.1016/j.knosys.2018.11.025. url: https://www.sciencedirect.com/science/article/pii/S095070511830577X (cit. on p. 29).

[204] Y. Yang. "An Evaluation of Statistical Approaches to Text Categorization". In: *Inf. Retr.* 1.1–2 (May 1999), pp. 69–90. issn: 1386-4564. doi: 10.1023/A:1009982220290. url: https://doi.org/10.1023/A:1009982220290 (cit. on p. 24).

[205] L. L. Yin and D. Savio. "Learned text categorization by backpropagation neural network". In: *Hong Kong University Thesis* (1996), pp. 41–59 (cit. on p. 24).

[206] F. Yuan, L. Yang, and G. Yu. "A New Density-Based Method for Reducing the Amount of Training Data in k-NN Text Classification". In: *2007 International Conference on Machine Learning and Cybernetics*. Vol. 6. 2007, pp. 3372–3376. doi: 10.1109/ICMLC.2007.4370730 (cit. on p. 24).

[207] M. Zhao et al. "Chinese Document Keyword Extraction Algorithm Based on FP-growth". In: (2016), pp. 202–205. doi: 10.1109/ICSCSE.2016.0062 (cit. on p. 44).