1 2 9 0

UNIVERSIDADE Ð COIMBRA

Tiago Emanuel Pacheco Caldeira Conceição

Natural Language Processing In Citizen Messages About Forest Fires

Dissertation in the context of Master in Data Science and Engineering, advised by Professor Catarina Silva, co-advised by Professor Hugo Oliveira, and Professor Alberto Cardoso, and presented to the Department of Informatics Engineering of the Faculty of Sciences and Technology of the University of Coimbra.

September 2023

1 2 9 0

FACULDADE DE
CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE Ð
COIMBRA

DEPARTMENT OF INFORMATICS ENGINEERING

Tiago Emanuel Pacheco Caldeira Conceição

Natural Language Processing In Citizen Messages About Forest Fires

Dissertation in the context of Master in Data Science and Engineering, advised by Professor Catarina Silva, co-advised by Professor Hugo Oliveira, and Professor Alberto Cardoso, and presented to the Department of Informatics Engineering of the Faculty of Sciences and Technology of the University of Coimbra.

September 2023

1 2 9 0

FACULDADE DE
CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE Ð
COIMBRA

DEPARTAMENTO DE ENGENHARIA INFORMÁTICA

Tiago Emanuel Pacheco Caldeira Conceição

Processamento de Linguagem Natural em mensagens de cidadãos relativos a incêndios

Dissertação no âmbito do Mestrado em Engenharia e Ciência de Dados, orientada pela Professora Catarina Silva, co-orientado pelo Professor Hugo e Professor Alberto Cardoso e apresentada ao Departamento de Engenharia Informática da Faculdade de Ciências e Tecnologia da Universidade de Coimbra.

Setembro 2023

Acknowledgements

I would like to express my gratitude to the advisor Professor Catarina Silva, and also to the co-advisors Professor Hugo Oliveira and Professor Alberto Cardoso, for their guidance, mentorship, and support throughout this journey. Their expertise has been fundamental for the shaping of this thesis.

I would like to acknowledge the support received from my family, whose encouragement, and understanding have been a constant source of motivation. The fact that they believe in my abilities and the constant availability and support throughout my academic path.

I am grateful to my friends who always motivated me to continue in the face of adversity, and who had the ability to distract me at times when the tension level was higher.

Finally, I extend my thanks to all the individuals who provided assistance and made themselves available to help me throughout the development of the thesis, thus contributing to the improvement of its quality.

To all my eternal thanks.

Abstract

With the recent increase in catastrophic events around the world, there is a need to monitor them. As a result of this growth and taking advantage of the technology that currently exists it is necessary to create a system that receives from citizens reports of these events. If the text after passing through this system comes up in a structured and concrete way, more information can be obtained and subsequently communicated with the relevant authorities so that they can use this knowledge to act accordingly.

As such, there is a project called FireLoc, which aims to streamline the communication process between citizens and the authorities, allowing information to be transmitted in the best possible way and mitigating localization and accuracy failures. This allows a viable data source for communication between citizens and the responsible authorities.

The approach taken to carry out this thesis involves collecting text from citizens via social media that corresponds to a contribution that will be considered a report. With the current capabilities of natural language processing and after the text has been captured, this content will be leveraged to capture extra information, using well-known methods such as sentiment analysis, event classification, and clustering events. Through sentiment analysis we can take advantage of the text to obtain the sentiment, thus allowing us to understand the severity of the event based on confidence. Another way of obtaining information is through event classification, which allows you to classify the event to see if it actually corresponds to an event, allowing you to filter out only the relevant reports. The last method that is also used in the approach is clustering events, which allows events to be grouped according to certain characteristics, providing extra information on the sentiment of each grouping of data and also information on how the events are related geographically. Finally, in order to complement all the information captured, it is necessary to apply an advanced visualization method that allows data to be collected in a visual format and made easier to understand.

The conclusion that can be drawn from the approach taken is that, in fact, a lot of information can be gathered through the text, which can prove to be an asset. And since the text is about reports of a catastrophic event, all the information is relevant in order to act accordingly. In this way, and given that the development of this thesis comes in the context of the FireLoc project, which acts as a direct communication channel between citizens and the competent authorities, it is hoped that it will contribute value. Is expected to contribute value to the competent authorities.

Keywords

Catastrophic Events, Citizenship, Natural Language Processing, Transformers, Machine Learning, Sentiment Analysis, Event Classification, Visualization Tools, Clustering Algorithms

Resumo

Com o recente aumento de eventos catastróficos em todo o mundo, existe a necessidade de os monitorizar. Como resultado deste crescimento e tirando partido da tecnologia atualmente existente é necessário criar um sistema que receba dos cidadãos relatos destes acontecimentos. Se o texto, depois de passar por este sistema, surgir de forma estruturada e concreta, é possível obter mais informação e, posteriormente, comunicá-la às autoridades competentes para que estas possam utilizar esse conhecimento para atuar em conformidade.

Neste sentido, existe um projeto denominado FireLoc, que visa agilizar o processo de comunicação entre os cidadãos e as autoridades, permitindo que a informação seja transmitida da melhor forma possível e mitigando falhas de localização e precisão. Este facto permite uma fonte de dados viável para a comunicação entre os cidadãos e as autoridades responsáveis.

A abordagem adoptada para a realização desta tese passa pela recolha de texto dos cidadãos através das redes sociais que corresponde a uma contribuição que será considerada um relatório. Com as capacidades actuais de processamento de linguagem natural e após a captura do texto, este conteúdo será aproveitado para capturar informação extra, utilizando métodos bem conhecidos como a análise de sentimento, a classificação de eventos e o agrupamento de eventos. Através da análise de sentimentos, podemos tirar partido do texto para obter o sentimento, o que nos permite compreender a gravidade do evento com base na confiança. Outra forma de obter informação é através da classificação de eventos, que permite classificar o evento para ver se corresponde efetivamente a um evento, permitindo filtrar apenas os relatórios relevantes. O último método que também é utilizado na abordagem é o clustering de eventos, que permite agrupar os eventos de acordo com determinadas características, fornecendo informação extra sobre o sentimento de cada agrupamento de dados e também informação sobre a forma como os eventos estão relacionados geograficamente. Finalmente, para complementar toda a informação captada, é necessário aplicar um método de visualização avançado que permita recolher os dados num formato visual e facilitar a sua compreensão.

A conclusão que se pode retirar da abordagem adoptada é que, de facto, é possível recolher muita informação através do texto, o que se pode revelar uma mais-valia. E uma vez que o texto trata de relatos de um acontecimento catastrófico, toda a informação é relevante para agir em conformidade. Desta forma, e dado que o desenvolvimento desta tese se insere no contexto do projeto FireLoc, que funciona como um canal de comunicação direto entre os cidadãos e as autoridades competentes, espera-se que seja uma mais-valia. Espera-se que esta tese contribua com valor para as autoridades competentes.

Palavras-Chave

Eventos Catastroficos, Cidadania, Processamento de Linguagem Natural, Machine Learning, Transformers, Análise de Sentimento, Classificação de eventos, Algotitmos de Clustering

Contents

Acronyms

AI Artificial Intelligence.

ALBERT A Lite BERT.

API Application Programming Interface.

BERT Bidirectional Encoder Representations from Transformers.

CPD Categorical Proportional Difference.

CPPD Categorical Probability Proportional Difference.

DT Decision Tree.

ICNF Instituto da Conservação da Natureza e das Florestas.

IR Information Retrieval.

LSTM Long Short Term Memory.

NB Naive Bayes.

NER Named Entity Recognition.

NLP Natural Language Processing.

NLTK Natural Language Toolkit.

PoS Part-of-speech tagging.

RNN Recurrent Neural Networks.

RoBERTa Robustly-optimized BERT approach.

SentiWordNet Sentiment Lexicon based on WordNet.

SVM Support Vector Machines.

XLNet eXtreme MultiLabel Text Classification Neural Network.

List of Figures

List of Tables

Chapter 1

Introduction

This chapter introduces the work of this thesis developed in the context of the Dissertation enrolled in the Master's Degree in Data Science and Engineering course, providing context and main objectives together with expected contributions.

1.1 Context

One of the problems in the world is the yearly temperature increase resulting in an escalation of forest fires and other fire-related problems.

Identifying a forest fire in an early stage can help the responsible authorities, in particular, the fire department authority for emergency and civil protection, to have a faster response, making the population and the planet benefit from it.

In the majority of the scenarios the early detection, of a fire, can prevent its spread. Early detection will make it possible to save the surroundings, whether, for example, houses or gas stations are present. However, besides buildings, this kind of disaster puts human lives at risk.

Every year, the number of occurrences of forest fires in Portugal is very high. According to the report provided by Instituto da Conservação da Natureza e das Florestas (ICNF), in 2022, Portugal had 10,449 forest fires recorded totaling 110,007 hectares [1]. These numbers are alarming and do not tend to decrease in the years to come. One of the main reasons why this happens is miscommunication and the need for clarity regarding the data provided on the fire to the responsible authorities.

Currently, one of the projects that try to mitigate the most common human inaccuracies is FireLoc [2]. The FireLoc project aims to contribute to some of the problems mentioned by identifying, locating, and monitoring the fires. With the tools developed in this project, the citizen can take a photo of the fire using a

[1] https://www.icnf.pt/florestas/gfr/gfrgestaoinformacao/grfrelatorios/areasardidaseocorrencias

[2] https://fireloc.org

smartphone, and by capturing this event, it is defined if it is effectively a fire. Geolocation is also obtained facilitating the access of the event location. The project relies on crowdsourced data. As many more contributions, more precise analysis is collected, which can determine whether or not there is a fire in some locations.

The system filters all the contributions, separating the false reports from the real ones, which allows faster responses and more accurate communication. All the information that the FireLoc report provides can help responsible entities understand the intervention ahead.

Although the FireLoc project already evaluates a fire report using the supplied images, an image cannot capture some details. Therefore, a text box is provided to offer additional information in Portuguese, to facilitate the transfer of details to the responsible authorities. This is where this thesis project will focus on with the analysis of the text provided, in an initial phase through social media contributions and from the FireLoc reports, it is possible to obtain insightful information, like the sentiment behind it and how the event changes over time. All this information will make the report of the occurrence, more robust for the authorities to act accordingly.

1.2 Objectives

One technique to detect potentially catastrophic incidents is through FireLoc reports and social media, which is a good source of crowdsourced texts that can be useful as reports. The information about the occurrences can be presented in different ways, by different people. The main objective of this thesis is to develop an analysis of the text given by users. This will allow to recognize the sentiment behind the writing and how it changes across time, and also capture events to be able to make a timeline of the occurrences, helping to understand the evolution of the forest fire, all resorting to event extraction. Since the main source of information for this thesis is reports from citizens captured by both sources of data defined before, it is possible that valuable information can be retrieved, helping the responsible authorities.

According to what was defined previously, the main goals established regarding the thesis are:

- Study of natural language processing techniques from which the Fireloc project may benefit from;
- Improve the analysis of user contributions, by implementing a set of functionalities that enable the extraction of additional knowledge, such as sentiment and event information from text.
- Enhance the capabilities of the text, implementing a set of functionalities that can bring value, like visualization dashboards to enable get deeper insight and understanding of the data.
- Report on the conclusions reached.

1.3 Contributions

While participating in the development of the FireLoc project, the following contributions from this work were made:

- Documentation of the current state of the art regarding some NLP areas that fall within the scope of the thesis;
- Gather data from public sources related to real fires;
- Integration of sentiment analysis into the FireLoc system;
- Development of event extraction, as well as a visualization dashboard through visualization tool, and grouping events as a suggested feature;

1.4 Document Structure

Firstly, this first chapter gives context and motivation, as well as the work's aims and contributions. In the second chapter, it is possible to understand the most important definitions and fundamentals of the key themes in this thesis. The third chapter provides an analysis of the prevailing methods and strategies defining the most suitable routes for the thesis project. The fourth chapter presents the problem being solved and the approach for the development of the project. The fifth chapter discusses the experiments and outcomes, where tests were done on the various models and performance was evaluated. Finally, the last chapter will present a conclusion that summarizes the work conducted the results achieved, and also future work.

Chapter 2

Background

This chapter presents the different concepts that are used in this thesis. The following sections cover research on data acquisition methods, natural language processing, sentiment analysis, event extraction, and model development. Data acquisition is an important step since it is the method to retrieve valuable information and get insights from it. Natural Language Processing (NLP) plays a major role in this thesis since it is the source of the various fields in textual analysis. One example is sentiment analysis, which provides a general idea of what users are experiencing. Another example is event extraction, which is an important area to identify and extract events in the text.

2.1 Data Extraction

Nowadays, the Internet is a powerful tool that holds much information. That information can be regarding any theme, including from events taking place in the region to simpler things like the result of a football match. This is why data extraction can be valuable, because it allows to extract data from a variety of different sources, such as databases, documents, and also from websites. There are many ways to extract the data that is available online, it all depends on the purpose defined.

Once the project FireLoc works from information provided by many users, it is possible to conclude that it operates from crowdsourcing data. Crowdsourced data is data collected and provided by a large group of people through online platforms or mobile apps. So, relying on several websites, extracting information that replicates crowdsource data is possible. Most of them are on social network websites.

Social networks are applications where people with similar interests interact to create relationships and enhance communication. The most used platforms for this kind of data are:

- Facebook [1];
- Twitter [2];
- Instagram [3].

Each has different purposes and types of usage. Facebook and Instagram are multi-purpose social networks that post opinions through texts, images, and many other functionalities. Twitter in particular is built around posting short 280-character messages, also called "tweets." Since Twitter allows users to post text, images, and videos and is often used to describe events as they happen, it is a popular platform for real-time communication. Making it a reliable application to use.

Therefore, to extract data from these platforms, it is required either to use an Application Programming Interface (API) or scrape the data directly from the social media application. API is a mechanism that allows communication between two different components of the software.

By combining the power of data extraction techniques with crowdsourced data from social media platforms it will be possible to extract, analyze, and interpret event-related information. This fusion of technology and collective human input enables the acquisition of global insights into occurrences, promoting improved catastrophe monitoring, emergency response, and general knowledge discovery.

2.2 Natural Language Processing

The ability of computers and their surrounding technologies to understand humans through text has been developed in the field of NLP. Natural Language Processing is the technology that enables the understanding of human language in the format of text, and it has evolved significantly over the years.

So, NLP is the area of Artificial Intelligence (AI) (Castanha et al., 2022) with numerous applications to understand and interact with humans, allowing accessible communication. This technology is constantly evolving and used more often in the market. For example, it can bring insights about a business, feedback, and sentiment, and may help to add worth.

Nowadays it is possible to acknowledge several areas of NLP (Shivahare et al., 2022), some being Sentiment Analysis, Text Classification, and Urgency Detection.

One of the ways to explain an NLP system is through an approach to the levels of language used by (Jurafsky and Martin (2009)). A system must go through all these levels to be robust and complex, and those are composed by:

[1] http://facebook.com/

[2] http://twitter.com/

[3] http://instagram.com/

- **Phonetics and Phonology** - Knowledge about linguistic sounds;
- **Morphology** - Knowledge of the meaningful components of words;
- **Syntax** - Knowledge of the structural relationships between words;
- **Semantics** - Knowledge of meaning;
- **Pragmatics** - Knowledge of the relationship of meaning to the goals and intentions of the speaker;
- **Discourse** - Knowledge about linguistic units larger than a single utterance.

Upon passing through the various levels of text manipulation like the ones mentioned before, the subsequent step involves the implementation of a series of pre-processing procedures on the text. The pre-processing step is fundamental because it makes the text more accessible for the machine to understand. By passing through this process it will likely achieve better results when applied to machine learning algorithms.

Pre-processing is a step that involves preparing the data for modeling and is commonly used in all areas of machine learning. However, there are exceptions when it comes to the usefulness of pre-processing steps, such as in the case of deep learning, where some models are able to handle raw data and do not require the appliance of this technique. Additionally, in some cases, pre-processing may not improve the performance of the model and could even degrade it, like removing certain words from the text that have important meaning or context.

Some steps of pre-processing are:

- Sentence Segmentation;
- Tokenization;
- Part-of-speech tagging (PoS);
- Entity Recognition;
- Stemming/Lemmatization.

Each step is essential, although they have different purposes, in the particular case of sentence segmentation, its principal concept is to distinguish different sentences. That means the capability to indicate where a sentence starts and ends.

Tokenization is the ability to split each sentence into different structural components giving a chance to easily turn a sentence into smaller parts and assign meaning to each of those parts.

PoS is the grammatical role that a word plays in a sentence, such as whether it is a noun, verb, adjective, adverb, or pronoun, and is typically annotated in text data for natural language processing tasks.

Entity recognition is the procedure of characterizing a specific token or a group of tokens of each phrase based on some categories. For example, it is possible to identify different entities, like people, organizations, or locations.

When it comes to Stemming, it is the process of reducing inflected (or sometimes derived) words to their base form, or stem, in order to improve the performance of NLP and Information Retrieval (IR) models. Stemming is used to reduce the dimensionality of the data by reducing the number of unique words in the dataset and to group related words together.

Lemmatization is the process of reducing words to their base form, or lemma, in order to improve the performance of natural language processing and information retrieval models. It is similar to stemming, which also involves reducing words to their base form. However, unlike stemming, which is based on heuristics and often results in words that are not actual words, lemmatization is based on the morphological analysis of the words and results in base forms that are actual words.

The above procedures use text-processing algorithms, which are essential for applying an NLP pipeline, some tools automate this process like:

- **NLTK**[4];
- **SpaCy**[5];
- **TextBlob**[6];

Natural Language Toolkit (NLTK) is a library in Python that provides tools for working with human language data (text). It is a popular library for NLP. This library includes a wide range of tools for pre-processing, analyzing, and modeling text data, including tools for tokenization, stemming, lemmatization, part-of-speech tagging, parsing, and semantic analysis.

SpaCy emerges as a dynamic, open-source gem in the realm of natural language processing, all penned in the Python language. Its architecture is known for its speed, efficiency, user-friendliness, and utility. This toolkit offers a plenitude of instruments to navigate through the text. It includes tools for tokenization, stemming, lemmatization, part-of-speech tagging, dependency parsing, and named entity recognition, as well as tools for building and training machine learning models for NLP tasks.

Parallel with the last two mentioned, TextBlob is a Python library for NLP. It is built on top of the popular NLTK library, it provides a simple, and intuitive interface for working with human language data. TextBlob also includes a wide range of NLP tools but is mostly known for pre-processing.

These tools are useful, not only for the pre-processing technique but for a wide variety of other tasks.

[4] `https://www.nltk.org/`

[5] `https://spacy.io/`

[6] `https://textblob.readthedocs.io/en/dev/`

NLP has the advantage of allowing the extraction of information from text, which can be very useful. Text Mining is a field that is directly related to NLP, as both fields use techniques to extract meaningful information from text.

Text Mining is a technique that extracts valuable information from unstructured text. This discovery is attainable through the analysis of a large quantity of data. In the realm of commerce, text mining stands as a promising task since it is efficient and unlocks profound value.

The tasks in the context of text mining according to (ShrihariR and Desai, 2015), (Truyens and Van Eecke, 2014) are:

- **Summarization** - Summarize the data without changing the meaning of the content;
- **Visualization** - Simplicity to discover the information;
- **Text Classification** - Assigning texts to one or more pre-defined categories;
- **Text Clustering** - Grouping similar texts together;
- **Entity Extraction** - Finding the subject of discussion;
- **Sentiment Analysis** - Finding the sentiment of a text.

2.3 Sentiment Analysis

Sentiment analysis is the area of NLP responsible for the process of extracting information from text data and determining the sentiment or attitude of the person writing the text. Through the text, it can be done an analysis to determine which sentiment is captured, the possible sentiments are positive, negative, or neutral. Sentiment analysis is a widely used technique in NLP and has a variety of applications, including customer feedback analysis and social media analysis, among many others.

There are several different approaches to sentiment analysis, including rule-based approaches, machine learning approaches that can be split into supervised learning and unsupervised learning, and hybrid approaches. Machine learning approaches, in particular, have become increasingly popular in recent years, with the use of transformers and subsequently of pre-trained transformers which is an evolution.

There are three main approaches that NLP uses to capture the sentiment: one based on rule-based, another based on machine learning, and the other based hybrid approach. The first one presents defects in the execution from (Kaur et al., 2017).

2.3.1 Rule-based Approach

Rule-based approaches to sentiment analysis involve using a set of pre-defined rules or heuristics to classify the sentiment of a piece of text. These approaches typically rely on the use of a lexicon, which is a pre-defined list of words that are associated with specific sentiments. These pre-defined lists can be found through libraries which are the most common way the acknowledge the sentiment. Lexicon dictionaries can be created manually by annotating a set of words with their part of speech and meaning, or they can be generated automatically using techniques such as word embeddings or sentiment lexicons. There are several lexicon dictionaries that are specifically designed for use in sentiment analysis, which is a type of natural language processing (NLP) task that involves classifying the sentiment of a piece of text. Some examples of the most known lexicon dictionaries with the purpose of sentiment analysis are:

- WordNet[7];
- SentiWordNet[8].

WordNet stands as a comprehensive lexical database in the English language, a valuable resource commonly harnessed for various NLP endeavors, notably sentiment analysis. It offers a vast reservoir of words, accompanied by intricate nuances of meaning and interconnections. This depth enables the capture of word context, proving particularly advantageous for NLP undertakings. Detecting sentiment within a text becomes viable by discerning words affiliated with positive or negative emotions. WordNet's prominence in NLP applications is far-reaching and well-established.

Like WordNet, Sentiment Lexicon based on WordNet (SentiWordNet) stands as a lexical repository, adorned with sentiment annotations for English words. It's a valuable asset for NLP, specifically for sentiment analysis. The mechanism involves gauging sentiment within a text by identifying words tagged with positive or negative sentiment scores. SentiWordNet includes a list of words along with their sentiment scores, with a range of gradients of sentiment intensity.

Though primarily conceived for English, these lexical gems can also shine in other linguistic realms, extending their utility to languages like Portuguese.

In the realm of the rule-based approach, sentiment interpretation hinges on the application of predefined rules or heuristics to words nestled within the text. In an illustrative instance, words like "love" or "happy," radiant with positive connotations, would be pegged as uplifting. Conversely, words laden with negativity, such as "hate" or "sad," would garner the opposite sentiment.

The simplicity of implementing rule-based systems is an evident advantage, a quick gateway to sentiment classification. Yet, a pitfall emerges as it sidesteps the contextual dance words performed and the intricate relationships they weave.

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

[8] https://github.com/aesuli/SentiWordNet

An instance, "The football game was horrible, but in the end was all worth it," underscores the shortcoming.

Overall, the rule-based approach is a widely used method for sentiment analysis, but it is often used in combination with other techniques, such as machine learning, in order to improve the accuracy of sentiment classification.

2.3.2 Machine Learning Approach

Machine learning is highly used for sentiment analysis. This approach is excellent since it can work autonomously and improve from experience. Machine learning goes through some steps, like learning from given data. This step is called training, the test is the step that is added to the algorithm to get the predicted result.

This approach can be split into two main categories, like how it is possible to gauge from (Srivastava et al., 2020).

Those are:

- **Supervised Learning** is one of the categories because of the human factor, which plays a significant role in using this category. The humans help label the data that will attach to the correct answers. After the data has been marked, it will be provided to the technique algorithm, and an accurate result will be expected;
- **Unsupervised Learning** which does not use labeled data for the training. Allowing the algorithm to work on the given information without any human interaction, the machine will then try to predict a result.

Regardless of the chosen category, it is required to select some machine learning techniques for training. Some of the most used are Naive Bayes (NB), Support Vector Machines (SVM), Decision Tree (DT), and K-Means.

These algorithms are some of the most used for the classification of sentiment analysis. One cannot, however, limit oneself to the aforementioned algorithms, since nowadays neural networks are used more frequently for this type of task. It will be referred to in a more in-depth manner in section 2.5.

Some of the algorithms for classification specified in the paper (Akpan and Starkey, 2021) are:

- **Naive Bayes** is a supervised learning algorithm developed based on the Bayes theorem. It is widely used because it classifies data into different classes. This algorithm works by calculating the probabilities for every class and selecting the outcome which has the highest probability.

  A primary assumption of the NB is that the effect of the value of a variable on a given type is independent of the importance of other variables.

  The equation on which it is based is:

$$P(\mathbf{X}|\mathbf{Y}) = \frac{P(\mathbf{Y}|\mathbf{X})P(\mathbf{X})}{P(\mathbf{Y})} \tag{2.1}$$

The part given by P(X | Y) is the probability of X, given that Y occurs. The P(X) is the probability of X happening, and P(Y | X) is the probability of Y, given X, and finally, P(Y) is the probability of Y.

- **SVM** is a popular machine-learning method since it works for both regression and classification. The SVM aims to find the optimal hyperplane that splits multiple groups.

  These hyperplanes can be either linear or nonlinear. An SVM classifier tries to identify a hyperplane with the ability to maximize the distance between the closest points to the hyperplane. When presented with labeled data, SVM tries to infer an optimal line or lines separating data sets into classes.

  Generally, and since it is mainly defined in a 2D space, the hyperplane splits a plane into two sections, each with a different category. The decision surface is usually a line that links the data points that lie closest to the boundary of the class.

  These data points are called support vectors.

- **Decision Tree** is a supervised learning technique that creates a model that tries to assume a target's value depending on the input inserted. In the decision trees, it's possible to see the roots or nodes that represent the test and attributes, the branches are the results of the tests, and the leaves show the class distributions.

  Each path from the root of a decision tree to one of its leaves can be converted into a rule by simply combining the tests along the path to form the antecedent part, with the leaf's class prediction taken as the class value.

  DTs have the strengths of being self-explanatory, easy to understand and can be converted to a set of rules easily.

K-means is a clustering algorithm that is commonly used in NLP tasks, including sentiment analysis. It is an unsupervised learning algorithm that is used to group a set of data points into clusters based on their similarity.

In the context of sentiment analysis, k-means can be used to classify text into different sentiments by clustering words that are associated with similar sentiments together. For example, k-means could be used to cluster words that are associated with positive sentiments, such as "happy" and "excited," together, and words that are associated with negative sentiments, such as "sad" and "angry," together.

Machine learning approaches involve training a machine learning model in order to classify the sentiment behind the text.

2.3.3 Hybrid Approach

The hybrid approach for sentiment analysis combines elements from both rule-based and machine-learning-based approaches. In this approach, a combination of lexicons, dictionaries, and machine learning algorithms are used to analyze the sentiment of a piece of text.

One of the main advantages of the hybrid approach is that it can benefit from the strengths of both rule-based and machine-learning approaches. For example, dictionaries and lexicons can initially provide the sentiment of the text, while machine learning methods can be trained on datasets and can learn to identify more complex sentiments. However, through machine learning, if some words are outside the vocabulary of the text on which the model was trained, it may not perform well in its classification.

By combining both types of approaches, the hybrid approach can achieve better performance than either approach alone.

The hybrid approach is a powerful approach for sentiment analysis that combines the strengths of both rule-based and machine-learning approaches in order to achieve better performance.

2.4 Model Evaluation

To evaluate the performance achieved by the classification algorithms, it is required to do a model evaluation. Several metrics allow the understanding of the performance achieved by the algorithm developed. The metrics will represent whether the model was good or bad.

While performing classification, there are four types of possible outcomes, those are:

- **True Positive** number of cases classified correct as positives;
- **False Positive** number of cases classified incorrectly as positives;
- **True Negative** number of cases classified correctly as negatives;
- **False Negative** number of cases classified incorrectly as negatives.

In the context of sentiment analysis, model evaluation involves using a set of pre-labeled test examples to evaluate the accuracy, precision, recall, and F1 score of the model on the task of predicting the sentiment of a given piece of text.

Accuracy is the percentage of correct predictions for the test data. It is calculated by dividing the number of correct predictions by the number of total predictions.

$$\textbf{Accuracy} = \frac{\textbf{True Positives} + \textbf{True Negatives}}{\textbf{True Positives} + \textbf{True Negatives} + \textbf{False Positives} + \textbf{False Negatives}} \tag{2.2}$$

Precision is the fraction of true positives among all the ones predicted in a certain class.

$$\textbf{Precision} = \frac{\textbf{True Positives}}{\textbf{True Positives} + \textbf{False Positives}} \tag{2.3}$$

Recall is the fraction of examples that belong to a class correctly classified, within all those that should've been classified as belonging to a specific class.

$$\textbf{Recall} = \frac{\textbf{True Positives}}{\textbf{True Positives} + \textbf{False Negatives}} \tag{2.4}$$

F1 Score considers both recall and precision and determines the value. The closest to 1 better the value of the prediction.

$$\textbf{Recall} = 2 * \frac{\textbf{Precision} * \textbf{Recall}}{\textbf{Precision} + \textbf{Recall}} \tag{2.5}$$

Although these metrics give a good idea, of whether the model is good or not, some of these metrics might have some problems. Accuracy for instance is a simple metric but misleading if the classes in the dataset are imbalanced. In this case, the model may achieve high accuracy by predicting the majority class all the time, even if its predictions for the minority class are not very accurate.

The proportion of positive cases properly predicted by the model is measured by the recall, but it does not account for the number of false positives, indicating that even if a model has a high recall, it may still have a low precision if it makes a lot of false positive predictions.

When it comes to the metric Recall it measures the proportion of positive instances that were correctly predicted by the model, not taking into account the number of false positives, meaning that a model with high recall may still have low precision if it is making a lot of incorrect positive predictions.

Finally, while the F1-score is a good compromise between precision and recall, it may not be the optimal statistic to use if the cost of false positives and false negatives is not equal. For example, if the cost of false negatives is much higher than the cost of false positives, then a model with a high F1 score may not be the best choice.

2.5 Language Models

The development of language models has played a crucial role in the evolution of sentiment analysis, a field of natural language processing that aims to automatically identify and extract emotions from the text. In the initial stages of sentiment analysis, rule-based approaches were the method that was most used. Rules and lexicons were used to classify text as positive, negative, or neutral. However, these approaches had big limitations when it comes to classifying big datasets.

Machine learning-based approaches were introduced to correct the limitations imposed by the rule-based approach, allowing models to be trained on large datasets and learn to classify text based on patterns in the data. These approaches represented a significant improvement over rule-based approaches, but they still had limitations in terms of their ability to capture complex linguistic patterns.

Nowadays, the development of deep learning approaches, such as Recurrent Neural Networks and Transformers, has revolutionized the field of sentiment analysis. These models are able to learn complex patterns in the data making them highly effective for sentiment analysis tasks.

The following sub-sections will reference the main models that have been through Recurrent Neural Networks to Transformer models. The information is based on (Ganai and Khursheed, 2019) and (Vaswani et al., 2017).

2.5.1 Recurrent Neural Networks

Recurrent Neural Networks (RNN), are one of the most used models for most language applications. It is well-suited for language models thanks to its recurrence. This model is ideal for solving problems where a sequence is more important than the individual item.

A Recurrent Neural Network is basically a fully connected neural network that contains the refactoring of some of its layers into a loop. The loop is typically an iteration over the addition or concatenation of two inputs. This is why the RNN performs well in NLP tasks (Hu et al., 2020).

So, the principal principle of RNNs is to save the output of a particular layer and then feed this back to the input. This will allow the prediction of the output layer, making it easier to predict the next word in a sentence.

Like any other model, disadvantages are inherent, RNN suffers from vanishing gradient and exploding gradient. The recurrence of these networks means that, by nature, deep networks with several points contain an operation among the incoming data and the neuron's weight.

By calculating the network error and using it to update the network weights, it is necessary to walk back through the network update weights one by one.

In the case of the update, if the gradient is small, multiplication with a small number is done, meaning the network either takes a very long time to train or doesn't work.

If the recurring weight is high, the problem is other, facing an exploding gradient.

2.5.2 Long Short Term Memory

Long Short Term Memory (LSTM), is the approach that came after RNN. This model came to improve the Recurrent Neural Networks since the solution for the vanishing gradients problem.

LSTM introduced one additional stream of information inside the time-states with transformations controlled by gates.

This way, the LSTM is divided into two types:

- Short-Term Memory;
- Long-Term Memory.

When it comes to short-term memory, it only operates when information is acquired. This information is only retained for a few seconds, and then it might be kept or discarded.

In contrast, there is long-term memory, which retains the information allowing be accessed later. All the data is kept in long-term memory.

A limitation of the model is that it encodes the sequence to a fixed length. This way, it imposes limits and results in worse performance for very long sequences.

Since all the words are captured into a vector, if the output of a word depends on a specific input, then attention is not given to a simple LSTM-based Encoder-Decoder model.

2.5.3 Transformers

To overcome the problems mentioned in subsection 2.5.2, the Attention Based Model was created.

Transformers models are attention-based models. According to (Vaswani et al., 2017), at each decoding step, the decoder looks at any particular encoder state. Bidirectional input is used where the input sequences provide forward and backward, then concatenated into a vector before going to the decoder. Instead of encoding the input into a single fixed vector, the attention models generate a vector explicitly filtered for each output time step. The vector changes based on the input time steps, taken by alpha representing attention weights.

The usage of the attention models results in higher performance over traditional encoder-decoder models, even on long sentences.

Transformers networks are built based on self-attention mechanisms without recurring to traditional approaches.

Although the RNN and LSTM methods are still widely used, the transformers bring improvements and assert themselves in the most recent projects.

2.5.4 Pre-Trained Transformers

Pre-trained transformer models are based on the transformer architecture, which is a type of neural network that is particularly well-suited for NLP tasks.

Pre-trained transformer models can help with sentiment analysis by providing a representation of the text that captures these contextual and semantic features (Ramprasath et al., 2022). For example, a pre-trained transformer model might be trained to predict the next word in a sentence given the previous words, and in doing so, it will learn to capture the context and relationships between words. This representation can then be fine-tuned for the specific task of sentiment analysis by adding a classification layer on top of the pre-trained model and training it on a labeled dataset of texts with their corresponding sentiments.

Nowadays, pre-trained transformers are currently the most used methods for any NLP project.

2.6 Event Extraction

Event Extraction is a field of NLP that focuses on the identification and extraction of events and event-related information from text. An event is an occurrence or action that takes place at a specific time and location, indicating useful information to keep track of. One of the key challenges in event extraction is the need to handle the wide variety of ways in which events can be expressed in natural language. For example, the same event might be described in different ways depending on the context.

It is an important task in NLP because it allows to extract structured information from unstructured text and use it for a variety of purposes.

By capturing events, it is possible to answer a lot of questions regarding the time of events, and how they evolve over time and gain more insight into the occurrence.

Event extraction can be separated into two parts (Xiang and Wang, 2019):

- **Open-domain event extraction** - is the task of extracting events from any text that is not restricted to a specific domain or topic. For example, extraction events from articles, posts from social media, or from a specific text, although never restricted to a particular subject;

- **Closed-domain event extraction** - is the task of extracting events from text that is restricted to a certain topic or subject. For example, extract data about events from medical journal articles.

Both of the strategies are helpful according to the purpose of the project that is being developed. Regardless of which method is best for the selected task, it is part of the extraction of events to understand whether they are actually related to the topic to be analyzed or have simply been misattributed. Thus, it was necessary to apply classification methods of which some are better known and others less. Those classification models are:

- **Fine-Tuned Classification;**
- **Few-Shot Learning Classification;**
- **Zero-Shot Learning Classification.**

Fine-tuning in NLP is the process of training a pre-trained model usually Bidirectional Encoder Representations from Transformers (BERT) on a new task using task-specific data. The fine-tuning process works by adjusting the parameters of the pre-trained model for a specific classification task using labeled data, allowing a more accurate prediction for specific categories. This method uses the knowledge in the pre-trained model optimizing it to a given task making it effective for NLP (Mohammadi and Chapon, 2020). It can be also used, in different tasks such as sentiment analysis, topic detection, and question answering.

Few-shot learning consists of training models that effectively learn through limited methods on limited text data. Sometimes the acquisition of annotated datasets in a complete and rigorous way is not easy to find and given the case study of the project in concrete even more difficult, there is also the possibility of creating an annotated dataset but it is quite time-consuming. In order to overcome these challenges Few-Shot Learning allows the model to learn through a few examples per class or category, making it possible to adapt to new tasks with minimal labeled data. This method is not limited to the area of NLP, working in several areas of machine learning some of them computer vision, recommended systems, and speech recognition.

However, within the NLP area, it works for several tasks such as text classification, sentiment analysis, question answering, and many others.

Zero-shot learning although the name is similar to Few-Shot Learning, it differentiates from because this one focuses on scenarios where some of the classes are new and or not available during the training phase. Making the goal utilize transfer learning and external information to make predictions on unseen classes.

Both zero-shot and few-shot are recent methods, but with a lot of potential, since they don't need a large amount of data to classify them or even no data, they will be methods that will be addressed in many of future projects.

These methods are extremely important because they are necessary for identifying events, which can later be combined with some of the NLP tasks and can provide a lot of additional information. For example, if an event is captured via a social network or public data source, this capability can be combined with the capture of sentiment analysis to obtain a public opinion.

2.7 Visualization tools

Visualization tools are very important because no matter how important the analysis and transformations performed on the data are, the conclusions would not be so clear if there was no visual support. Therefore, visualization tools are quite relevant, supporting decision-making. Given that sometimes the data density is high, and ends up not allowing a better understanding of them, it makes these kinds of tools an asset.

Through visualizations, it becomes easier to identify patterns, trends, and relationships between data. According to the visualization needs and the task, there is a range of graphs, charts, maps, and many other visual elements, allowing to demonstrate the data in different formats.

In addition, the visualizations can have two types of formats, dynamic that allows the alteration of parameters changing accordingly to the visualization to be analyzed, static representing only the data for which the visualization was created not changing over time.

2.8 Clustering Events

Clustering is a fundamental unsupervised machine learning technique that allows grouping event data, this method is directly related to event extraction as it allows events to aggregate similar events together in the same cluster based on characteristics or attributes defined previously. The objective of this is to divide the events into clusters so that within the same clusters the events are more similar than the events in other clusters. Thus allowing the discovery of patterns and structures in the data without predefining labels or categories.

There are multiple types of clustering algorithms, each one having its characteristics and approach, however, they can be generalized into two types (Xu and Wunsch, 2005):

- **Partition-Based Clustering;**
- **Hierarchical Clustering.**

The first item mentioned in the previous list refers to a type of clustering in which the data is divided into a specific cluster number usually referred to as (K), where this value is defined by the user, with the main objective being to assign data points to the cluster in a way to maximize the similarity within each cluster and minimize the similarity between clusters, some of them are (Kutbay, 2018):

- **K-Means;**
- **K-medoids;**

- **Fuzzy C-means.**

Hierarchical Clustering is the second on the list of the two types of generalization which was seen previously, group's similar data points, creating a hierarchical representation of the data structure. Unlike Partition-Based Clustering, this one doesn't require the user to specify the number of clusters in advance. Usually this method represents the merging or the splitting methods at different levels through a dendrogram, enabling the users to choose the best number of clusters based on their needs. Can also be divided into some algorithms like:

- **Agglomerative Hierarchical Clustering;**
- **Divisive Hierarchical Clustering.**

Each of the types seen is part of the definition of clustering, and its use is adapted depending on the project to be carried out and according to what the user wants, however, the fundamentals of clustering have been exposed in order to obtain a better understanding about the subject.

2.9 Synthesis

Over the last few decades, the field of NLP has changed significantly, allowing text-related tasks to be performed in a simpler way, thus enabling greater extraction of relevant information at a lower computational cost. In this way, it is possible to take advantage of text to capture useful information through various tasks such as sentiment analysis, event extraction, and text mining, among many others.

This evolution has occurred mainly due to the development of machine learning areas, namely with the rise of pre-trained transformer models These models have achieved remarkable performance across NLP tasks, gaining popularity in various investigations.

Despite all this, the field of NLP is still largely unknown and has many challenges to solve, but its ability to deal with text is very promising.

This chapter introduced different concepts and background information in depth, which will be useful for a better understanding of the following areas, namely data extraction, sentiment analysis, and event extraction. With the presence of this chapter, it is possible to understand how each of the different areas studied works and what it is based on, facilitating the tasks of executing these themes.

The next chapters will link the information acquired in this chapter in order to make it easier to understand it in subsequent topics. In addition, some of the aspects mentioned will be visualized and applied from a practical perspective.

Chapter 3

Related Work

In the following sub-sections, a study will be carried out of approaches and methodologies that have been used in other projects in the area being explored. Understanding the optimal steps for the success of a specific project allows to gain insights into the best potential execution and the benefits or drawbacks. It will be able to examine what improvements have been achieved in the present state of the art of the projects to be carried out in these areas as a result of the exploration.

3.1 Data Extraction

Data extraction is an important process because it enables people and organizations to collect large amounts of data from several sources. With the data collected, it is possible to make better decisions regarding the theme of research and obtain better insights.

As it was said in the section 2.1, there are several sources to extract data, like databases, documents, and also from websites.

The purpose of the project and thesis is based on contributions from citizens, it is required to extract data from social platforms.

According to (Gonçalo Oliveira et al., 2015), it is possible to acknowledge that the usage of social networks has increased in popularity, being a part of everyone's life. There are two that stand out from the wide variety that exists, Twitter and Facebook. Both networks are highly accessible and have hundreds of millions of users all over the world, who read, post, and share real-time messages, at a fast pace. In this specific case, the author uses the social network Twitter, as it is an excellent source of text data provided by its users.

The choice of the social network Twitter makes sense since it has approximately 237.8M active users registered in the second quarter of 2022 [1] using it.

Inspired by the high numbers of monthly usage, a lot of works mine informa-

[1] Numbers according to `https://www.statista.com/statistics/970920/monetizable-daily-active-twitter-users-worldwide/`

tion from Twitter, making it possible to exploit the usage of this social network for our benefit.

Some of the projects that use Twitter as a data source are (Chiorrini et al., 2022), (Pota et al., 2020), (Souza and Vieira, 2012), (Zahoor and Rohilla, 2020).

3.2 Pre-Processing

It is known that pre-processing is vital for tasks to obtain good performance. The pre-processing element may vary according to the project and the data source. The usual process of pre-processing is mentioned before in subsection 2.2.

This step is not necessarily required to be applied to the text, as previously stated, the use of transformers and pre-trained models may deliver some extra information if pre-processing is not used.

One of the advantages that pre-trained models, i.e., BERT, provide is that they do not need some pre-processing steps because they use all the available information and learn from it. When applying some of the steps, the accuracy might be lower.

Since the data source for this thesis is Twitter, some care regarding the text is necessary.

Twitter, usually in "tweets" contains mentions, URLs, and Retweets. So accordingly to (Chiorrini et al., 2022), the following approach is taken.

- Removing mentions for the text pre-processing phase is necessary because users often cite other Twitter usernames with the character '@' to direct their messages.
- URLs are very common in tweets, i.e., links to other web pages, videos, images, or news. That is required to be removed.
- Retweets are also removed because they are usually marked with the prefix "RT" and are easily identifiable.

On the other hand, the paper (Pota et al., 2020) makes a different approach, instead of removing the most known noisy entities of the present tweets, like emails, URLs, currency, dates, and others, the normalization of these values was done. Normalizing the information makes it simpler for the machine to understand.

Some examples of normalization are validating the data format by ranging the days from 01 to 31, the months from 01 to 12, and the years of any sequence of four digits. Also, for example, the normalization of time by making hours range from 00 to 23, while the minutes vary from 00 to 59.

These are some examples of normalization that were applied, but also some transformation was used to emoticons and emojis.

Emoticons are short sequences of symbols, letters, or numbers representing a particular facial expression. Nowadays, this way of expressing is widely used to transmit moods, firmly bound to the overall sentiment. These elements can be translated into a word that expresses the same mood, like in Table 3.1.

Table 3.1: Emoticons Transformation

Symbol	**English**
:) :-) 8-) :-] :-))	Happy
:-(:(:-	Sad
:-P x-p	Joking
<3 < 3 :*	Love

Emojis, are elements of a standard set of small pictorial glyphs depicting different items, from smiling faces to international flags. Emojis have been increasingly used in social media to replace certain words and sentiments. Like emoticons, these elements can also be converted into words, as seen in Table 3.2.

Table 3.2: Emojis Transformation

Symbol	**English**
1F600	Grinning Face
1F62D	Crying Face
1F620	Angry Face

With both approaches presented, at least two solutions exist, choosing the best decision according to the project and what information may be helpful.

3.3 Sentiment Analysis

As mentioned in section 2.6, sentiment analysis can be split into two main approaches, the rule-based approach, and the machine-learning approach. Hybrid approaches can be achieved through the main other approaches.

In the following subsections, it will be exhibited the state of art of each approach.

3.3.1 Rule-Based approaches

Rule-based approaches are one of the most used, which is why many projects rely on them. The problem is that the Portuguese language has few dictionaries. That is why in the paper (Tavares et al., 2021), the initial strategy proposed by authors was to identify the sentiment by appealing to the Portuguese language tools. Since it was a low-resource language, and it was hard to find a sentiment analysis tool that could be directly applied to detect the sentiment of a sentence in Portuguese, the strategy was dropped. So the alternative method chosen was to translate the Portuguese language into the English language and then apply the tools.

Since the Portuguese language is not widely used in all projects, many papers resort to dictionaries that are available. In the paper (Souza and Vieira, 2012), they propose the usage of two dictionaries. OpLexicon[2], since this lexicon is built with around 15000 polarized words, classified by their morphological category, annotated with polarities, from positive, negative, and neutral. Also, they used SentiLex [3] to annotate the adjectives not present in OpLexicon.

By comparing both of them, OpLexicon obtained higher rates than SentiLex. According to the authors, it was unsurprising since SentiLex was built for specific domain analysis and was used to enrich the OpLexicon.

The paper (Pereira, 2021) presents a survey of sentiment analysis in the Portuguese language, analyzing the several tools, lexicons, datasets, and other resources available.

Regarding sentiment lexicons, the paper suggests some dictionaries for Brazilian Portuguese, although only two work for European Portuguese.

Those are:

- **SentiLex**;
- **Onto.PT**[4].

By reading these papers, it is possible to understand that the approaches regarding the Portuguese language effectively are not plentiful, yet there are some solutions.

3.3.2 Machine Learning approaches

The machine learning approach is another strategy that is widely utilized.

This way and according to the paper (Carvalho et al., 2017) a comparative analysis of machine learning methods on the subject of sentiment analysis of Brazilian election news was carried out, the three machine learning algorithms used were Support Vector Machine, Naive Bayes, and Maximum Entropy. However, before applying the machine learning algorithms, three different feature extraction methods, namely Chi-Square, Categorical Proportional Difference (CPD) and Categorical Probability Proportional Difference (CPPD). This paper was based on a dataset that has already been labeled, which mentioned news about the 2014 elections.

By examining the results in Table 3.3, the Naive Bayes obtained better results.

[2] https://search.r-project.org/CRAN/refmans/lexiconPT/html/oplexicon_v3.0.html

[3] https://b2find.eudat.eu/dataset/b6bd16c2-a8ab-598f-be41-1e7aeecd60d3

[4] http://ontopt.dei.uc.pt/

Table 3.3: Chi-Square accuracy using 5000 features from the different algorithms

Algorithm	**Accuracy**
Support Vector Machine	69.33%
Naive Bayes	67.88%
Maximum Entropy	66.39%

Although the paper analyzed before was for a specific project, in the next paper (Zahoor and Rohilla, 2020), a different approach was made.

The focus of the paper was on different case studies of tweet sources, like:

- **Haryana Assembly Polls** - Polls were one of the important events that occurred last year, and a number of tweets were recorded with this hashtag. Around 16,000 tweets were collected in raw form.
- **The Sky is Pink** - A movie released on 11th October 2019 and was starring Priyanka Chopra, Zaira Waseem, and Farhan Akhtar. Recording several tweets about it.
- **UNGA** - Session of an assembly that discusses several themes. 5000 tweets were collected.
- **Howdy Modi** - Occured in Houston and was led by the Prime Minister of India Narendra Modi and the President of America Donald Trump

For all the case studies, the following machine learning algorithms were applied Naïve Bayes, Random Forest, and SVM.

Table 3.4: Algorithms accuracy in different datasets

Algorithms	Haryana	Sky	UNGA	Howdy
Support Vector Machine	96%	98.29%	94.19%	96.8%
Naive Bayes	96.8%	90%	98%	98%
Random Forest	95.94%	98.2%	94.66%	98.24%

Looking at Table 3.4, it is possible to conclude that Naive Bayes if the values are averaged, overall got better results.

By analyzing the results of the previous tables, it can be seen that all the algorithms used in the different projects had very similar results, which leads to the conclusion that the results of the algorithms will depend a lot on how the data is organized.

3.3.3 Pre-Trained Models

Like it was mentioned before in 2.5.4, without a doubt, transformers are the way to go when it comes to NLP projects. So with the development of transformers, pre-trained transformers were created.

These pre-trained models have been specifically designed for the purpose of fine-tuning for a specific task. The concept of these pre-trained models is to train the model on a large set of data and various datasets to learn patterns in the data.

There are several pre-trained transformers language models, according to the paper (Casola et al., 2022), and some of the most known are:

- **BERT** - Bidirectional Encoder Representations from Transformers;
- **Robustly-optimized BERT approach (RoBERTa)**;
- **eXtreme MultiLabel Text Classification Neural Network (XLNet)**;
- **A Lite BERT (ALBERT)**;
- **Google Electra**.

Each one of them has different properties that distinguish it from the others. According to some of the papers on similar sentiment analysis projects, an analysis of the performance was realized.

Following the paper (Kumawat et al., 2021), three different models were analyzed, BERT, RoBERTa, and Google Electra. These models were selected for experimentation on the "Twitter US Airline Sentiment" dataset, which is an annotated dataset made available by Crowd-Flower for the supervised training of the model. This dataset contains information about travelers who have expressed their feelings about the airline. Of those three, a very similar test accuracy was performed.

Table 3.5: Twitter US Airline Sentiment Accuracy Results

Model Name	**Test Accuracy**
BERT	0.812%
RoBERTa	0.808%
Google Electra	0.798%

As it is possible to see in Table 3.5, BERT had a better result than the others. Although the difference was minimal in the order of 0.01.

Another paper (Pipalia et al., 2020) analyzed XLNet, RoBERTa, BERT, and DistilBERT. In order to run sentiment analysis, we took into account the data set of reviews from the IMDb [5] platform, which is used for binary classification of sentiment. It also has 25,000 film reviews for training and another 25,000 for testing This way, the models analyzed had similar accuracy values.

In the last paper (Zhang et al., 2020), several pre-trained models were compared to six different public datasets with annotated sentiment polarities. Some of the datasets analyzed are API reviews which contain sentences from posts in Stackoverflow regarding the performance of the API, Stackoverflow dataset which has sentences from threads tagged with JAVA. Another dataset is from

[5] `https://www.imdb.com/`

Table 3.6: Models accuracy results from IMDb reviews

Model Name	Test Accuracy
BERT	0.936%
RoBERTa	0.942%
XLNet	0.962%
DiltilBERT	0,924

mobile app reviews that have been collected from the author. Some of the models analyzed are BERT, XLNet, RoBERTa, and ALBERT. The results were close to each other once more.

Table 3.7: Macro F1 score from the dataset API

Model Name	Macro F1 score
BERT	0.81%
RoBERTa	0.81%
XLNet	0.76%
ALBERT	0.82

Looking at the final Table 3.7, it is clear that the Macro F1 score values were not significantly different for any of the methods evaluated, with only the XLNet having lower values.

Evaluating the Tables 3.5, 3.6 and 3.7, it is possible to conclude that, generally, the accuracy was close and did not vary much.

In each paper, it is possible to find one thing in common, clearly, of all the possibilities of pre-trained models, the ones selected were always the same.

Overall there is not one good option because all the pre-trained models obtained good values, but the ones that stood out the most were BERT and RoBERTa. So, looking at the most similar projects that exist, some of them being mentioned in the next sub-sections, it is possible to conclude that most of them resort to BERT.

3.4 Event Extraction

Event extraction as mentioned in Section 2.6 is a sub-field of NLP, that has the purpose of identifying and extracting events or event-related information from text.

This can be a hard task since extracting useful structured representations of events from a disorganized corpus of noisy text is a challenge.

The paper (Ritter et al., 2012) had the purpose of extracting events mentions and resolving temporal expressions. To that, the annotation of the corpus was realized, which was then used to train sequence models to extract events. The technique applied in the project is an established approach for sequence-labeling tasks in noisy text from Twitter. It applied event phrases consisting of many different parts of speech, such as verbs, nouns, and adjectives, and provided important context for downstream tasks, like event categorization. From which, 1.000 tweets were annotated.

Also, in this project, as mentioned before the temporal expressions were resolved by making use of a tool called TempEX, because there are many ways users can refer to the same calendar date, like "next Friday", or "tomorrow". TempEx takes as input a reference date, some text, and parts of speech. The classifier obtained a Precision of 85% a Recall of 55% and an F1 Score of 67%.

On the other hand, the paper (Sakaki et al., 2010) focuses on the detection of a specific event, such as "hurricanes" and "typhoons". They split the tweets captured into positive or negative categories using SVM as a classifier, by preparing positive and negative examples as a training set, making it possible to classify tweets automatically into positive and negative categories. The positive category refers to the true tweet referring to the actual event. In this research, a temporal model for event detection from Twitter was proposed. The method is based on the examination of actual data, which shows that the number of tweets for a target event follows an exponential distribution. The probability density function of the exponential distribution is used to model the time between tweets. The performance of classification got an average Recall of 87.50%, a Precision of 63.64%, and an F1 Score of 73.69%.

There are other several papers like (Lin et al., 2011) or (Ling and Weld, 2010) and (Gonçalo Oliveira et al., 2015) that present other different approaches.

Once the methods for the extraction of events have been mentioned above, it is necessary to find methods to classify whether the events actually belong to the event topic for this purpose two known methods are used:

- **Few-Shot Learning Classification;**
- **Zero-Shot Learning Classification.**

One of the papers that mention Few-Shot Learning Classification is the (Muthukumar, 2021) the process is exemplified using a pre-trained BART model from HuggingFace. The procedure used was based on an analysis that was divided by 3 labeling criteria (ratings from n stars upwards where stars upwards, are considered positive, where n = 2, n = 4, and n = 5), with each node tuned into 30 different examples, 10 per labeling criteria (5 positive and 5 negative). Basically in this paper, the procedure was created to evaluate the model without any previous training, having a performance very close to the fine-tuned model. Obtaining an accuracy of 85.6% while the fine-tuned version averaged an accuracy of 85.9%, making the difference between the two minimal. In this way, it is proved that the technique of few-shot learning has a lot of potential for use.

The paper that references Zero-Shot Learning Classification is (Chong et al., 2022), which compares the technique of Zero-Shot to human annotation in text classification. The testing performed in this paper is based on a pre-trained model and facebook/bartlarge-mnli. Based on the labels found through topic modeling related to cyberbullying behaviors, these served as input for the zero-shot model. The labels used were:

- **Flamming;**
- **Harassement;**
- **Racism;**
- **Sexism;**
- **Discrimination;**

Once the labels have been collected, they are passed on as input to the technique in question, each of which is transformed into a hypothesis. Based on the analysis of the raw text the zero-shot assigns a label that is supposed to be the most closed-fitting, thus attributing the cyberbullying behavior to the text.

In the end, the author when comparing the performance of the zero-shot technique with human annotation obtained different accuracies for each behavior, namely flamming obtained an accuracy of 71.67% being on par with human annotation. However, the zero-shot model is not so efficient in identifying discrimination since it only had 11.67% accuracy, racism obtained 45% accuracy rate, harassment obtained 36.6% accuracy, and sexism 20%. In the end, the conclusion drawn from the paper is that in terms of flamming behavior, the zero-shot model had better accuracy, but it performed worse in the remaining models, having to be improved the model for better recognition in these same models.

Another paper that explores a different theme but with a very similar procedure is the (Patadia et al., 2021). Also using the same pre-trained model and obtaining an accuracy of 72.34%. Because a zero-shot model does not need to be trained, the model does not need to be trained all over again when new labels are introduced, saving a significant amount of time and computing costs.

3.5 Visualization Tools

As mentioned in the previous chapter, visualization tools are an excellent way to display data in order to draw the best conclusions. The paper (Ali et al., 2016) presents several tools that are used nowadays to express high-density data, which are:

- **Tableau[6];**
- **Microsoft Power BI[7];**
- **Ploty[8];**

[6] https://www.tableau.com

[7] https://powerbi.microsoft.com/en-us/

[8] https://plotly.com

- **Gephi[9].**

Tableau is a visualization tool focused on business intelligence, providing a wide range of options to create custom visualizations. It is fast and flexible, supporting most of the data formats and connections to several servers that provide the data. Has an intuitive interface, with several charts to benefit from, also it's simple to create visualizations without having to create any code, although more complex visualization might require some programming.

The second item on the list is Microsoft Power BI, which is a powerful cloud-based business analytics service, and it counts with more than 60 types of source integration allowing to create visualizations in minutes. Since it is a Microsoft tool, can be integrated into other well-known tools like Office, SharePoint, and SQL Server. This tool has a feature that distinguishes it from others, that is, it can use NLP to query data.

Plotly is also a tool for visualization that is built around a Python and Django framework. It has a free version, although with limited features. In order to unlock them, it is required to buy a professional membership. Like the previous tools presented, this one also allows the creation of dashboards and charts, it has a wide variety of charts making a more complete visualization according to the needs required.

The final tool in the list is Gephi, which is written in Java and OpenGL. It is used to handle large datasets making it a valuable tool for social network analysis and link analysis, which uses graphs composed of nodes connected by edges. Like the ones presented before, this one doesn't require programming skills but needs a good knowledge of graphs.

After a brief introduction of each tool mentioned, through the paper (Shakeel et al., 2022), it is possible to see a collection of seventy articles to identify, classify, and investigate the various aspects, and theories of data visualization. Also, this paper presents the strengths and weaknesses of each tool mentioned before and many more. Table 3.8 shows the strengths and weaknesses of the tools mentioned in the previous paper so that is possible to understand which of them might be suitable for the different projects.

Neither paper mentions which of the visualization tools is better, but since the best-known tools have been exposed, the choice should be according to the need of the project and what the user wants to demonstrate in the visualization.

3.6 Clustering Events

As mentioned in the previous 2.8, clustering techniques can be generalized into two types:

- **Partition-Based Clustering;**

[9] https://gephi.org

Table 3.8: Comparison of Visualization Tools

Tool	Strengths	Weaknesses
Tableau	• Multiple visualization imports • Available visualization mapping • Free version is public • Learning material	• Paid version high-priced • Free version lacks privacy
Power BI	• Availability and cloud service • Dashboard single view • Integration with Microsoft tools • Learning tutorials	• Not easy to learn • Challenging with large data
Plotly	• Convenient functions • Customization options • 3D visualizations	• Slow for complex visualizations • Limited customization and interactivity
Gephi	• Specialized in network data • Open source • Node and Edge Customization	• Supports specific data formats • Lacks advanced analytics

- **Hierarchical Clustering.**

Thus, it was necessary to find papers that analyzed the techniques. Thus in order to analyze Partition-Based Clustering, a paper will be referenced (Hai et al., 2012).In this paper, the authors analyze the different performances of clustering algorithms on two well-known datasets. The Iris dataset is one of them and is composed of samples of 3 species of iris flowers (iris setosa, iris versicolor, and iris virginica) in which for each of the samples 4 features are evaluated. The other dataset is Wine, which is widely used for testing various algorithms including clustering, describes the different characteristics of different wines, and aims to classify them based on their chemical properties. The dataset is composed of thirteen input features.

For both datasets, three algorithms belonging to partition-based clustering techniques were tested, namely K-Means, K-Dmeans, and PPDK-Means. The algorithm K-Dmeans refers to a version of K-Means that is based on distributed clustering, the case of PPDK-Means differs from the previous ones because it is a distributed clustering algorithm based on privacy protection. After testing for different values of k, the algorithm that stands out is K-Means with a higher accuracy and lower time of execution than the others.

Although the last paper presents algorithms focused on the topic of distributed clustering, it is clear from the conclusion that K-means is superior. The same conclusion is also drawn from another paper (Dharmarajan and Velmurugan, 2013) that analyzes different algorithms.

In this last referenced paper the algorithms analyzed are K-Means, K-Medoids, and Fuzzy C-Means, in which K-Means is clearly the most consistent algorithm, and presents a clear superiority.

When it comes to hierarchical clustering the paper (Tang et al., 2022) references the capability of this technique, by comparing it with the well-known K-Means algorithm, which is a partition-based clustering technique.

In this paper, a dataset of energy consumption is analyzed for the time period between January 1, 2022, and July 27, 2022, in which the highest temperature, the lowest temperature, and the energy consumption during this time period are present. The results of an initial analysis indicate that the hottest day was on July 27th with 38 ºC in which the electricity consumed was 14.41 kWh, however, one of the hottest days had only a consumption of 31.29 kWh, which may be related to the need for cooling. On February 23, the lowest temperature recorded was -2ºC, and yet it registered 4.78 kWh, on another day that recorded the lowest temperatures the consumption was 23.64 kWh. These results indicate that there is no relationship between the need for the consumption of electricity.

Although that is no relationship in electricity consumption, the objective is to find the range between high and lower temperatures that require a demand for electricity use for both cooling and heating and for this purpose a study was conducted based on different values of k for K-Means, resulting in a value of 3 clusters, which showed a reasonable result for the use of electricity for heating but not for cooling. In the case of hierarchical clustering, the number of clusters was 5 representing a reasonable result in all clusters for electricity consumption, being according to the authors a more credible option than K-Means.

In addition, this paper (Rani and Rohil, 2013) complements the knowledge regarding the hierarchical clustering method, demonstrating some of the algorithms of this method, defining them, and presenting the positive and negative aspects. It is a good reference paper for this method, because it is possible to explore the algorithms, implying that there is not one better than another, and being in charge of the user is the most appropriate choice for the development of the project.

3.7 Final Remarks

In conclusion, the research conducted in this chapter of the thesis has shown that there are a variety of approaches and techniques that can be used for sentiment extraction, event extraction, visualization tools, and clustering events. However, none of the approaches studied are directly comparable to the specific purpose of this thesis since it is in Portuguese, and it is related to a specific detection of an event. Despite this, the papers studied provided valuable insights and useful ways to approach the task.

The results of this research have been used to develop this thesis project, which aims to improve or bring something new related to sentiment analysis, event extraction, and clustering events in small texts in the Portuguese language.

Overall, the literature review and analysis of the mentioned papers played an important role in shaping the direction of this thesis. Some of the aspects to be used for the development include the use of pre-trained transformers, specifically BERT, as it is very well documented and achieves very favorable results, the application of data pre-processing on the text, which is an added value, and the best tool to be used for displaying the data, which is Tableau. The choice of these methods will certainly contribute to the development of the thesis as it can provide valuable insights.

Chapter 4

Problem and Approach

This chapter presents the problem to be solved in more depth, which will serve as the main focus of the master's thesis, in addition, it will demonstrate the approach designed to contribute to solving the problem.

By presenting the problem and approaching it, a well-structured workflow pipeline will be introduced that aims to contribute to mitigating the problem.

4.1 Problem

During catastrophic events such as earthquakes, tsunamis, fires, or floods, authorities face the challenge of dealing with a significant volume of information via phone calls, or even the lack of them. Nowadays most of the information comes from that source, making it hard the analyze what's going on, the location, how it happened, and what it involves, impacting their ability to respond. Thus, the paradigm needs to be changed and opt for written contributions that can come from multiple data sources such as from the FireLoc [1] project, or even from messages present in social networks. This creates another problem.

The problem consists of collecting reliable contributions with relevant information from society, particularly citizens who are in the presence of a catastrophic event such as those mentioned above. With these written contributions it is possible to collect more valuable information that allows the competent authorities to respond accordingly. However, with a significant volume of contributions, it becomes difficult to identify the most useful and relevant information, which may affect how resources for the event are allocated, affecting the response capacity, and the resolution capacity.

Thus, the problem in this thesis is that currently, the method of communication between citizens and the authorities responsible for transmitting a report of an occurrence may not transmit all the information necessary for an adequate response. Or even, due to the large flow of communications, diverse information is obtained, many of which do not correspond to what is actually happening. The

[1] https://fireloc.org

following sub-section proposes a contribution to solving this problem.

4.2 Approach

Once the problem is specified, this will go into more detail on each of the steps of the pipeline which are part of the approach to contribute to solving the problem. In each of the next sub-sections, it will be specified all the decisions taken in order to achieve the best execution for the project's purpose.

In this project, the catastrophic event taken into consideration is related to wildfires, although it can work with any other event. So adapting to the problem related to this thesis the following pipeline is shown.

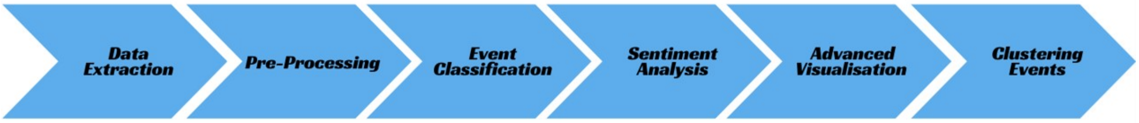


Figure 4.1: Workflow of the approach

The first phase can be specified by extracting data from Twitter since it is a great source of information and is highly used by citizens who usually comment on what they are visualizing and living which can be very useful.

Since the capture of data is done from the first phase, it is time to pass to the next phase which is pre-processing, inside this phase, some other tasks need to be taken. One of them is the cleanup of the information, this task references the need to remove some details that can affect the performance of the machine learning algorithm. The next task, which is inside the pre-processing phase is the classification. This is one of the most important steps since it evaluates all the retrieved texts and selects the ones that are related to the event. It will be taken into consideration three classification methods. The last responsibility in the second phase is to assign the geographical location, which takes into account the text from the previous stage, analyzes it, and allocates a location based on the words.

The third phase is sentiment attribution, for which was employed the pre-trained transformer model RoberTa, which will identify the corresponding sentiment through the text.

The second-to-last phase is to use the data from the previous steps to draw conclusions based on the visualizations. This allows a better understanding of the data through different types of graphs.

Finally, the last step is the aggregation of the captured events. This step is relevant because with it it is possible to understand if there is any relationship between them. It may allow us to understand the patterns or if there is any trend between events.

By following this pipeline, results can be obtained that can provide value both to the responsible authorities and to the FireLoc project. For the authorities, this will contribute to a better reaction as well as a better organization of the resources given for this purpose. As for the FireLoc project, implementing this will enable the system to become more robust and complete, making the project a reference for all citizens and a basic system for use by the competent authorities.

4.2.1 Data Extraction

This sub-section will be referred to why the social network Twitter was used, and it will be explained in detail all the procedures taken into consideration in the data extraction.

Twitter has been used since it allows posting short 280-character messages, also called "tweets". Since Twitter allows users to post text, images, and videos and is often used to describe events as they happen, it is a popular platform for real-time communication, making it a reliable application to use for this thesis.

Therefore, to extract data from the Twitter platform, it is required either to use an API or scrape the data directly from the social media application.

As tested in preliminary tests, where an API was used to pull data, the conclusion was that this is a method that limits its potential. The only way around these limitations is if we have superior exclusive access. Some of the restrictions that this method has are the fact that it can only access the last seven days, and not being able to access a different time window. Another limitation is the restriction on the number of tweets that can be captured limiting the exploration of this data source. Finally, the restriction that was noted was the fact that the use of the query is quite weak because it is not possible to perform advanced searches when using the API of Twitter.

In order to overcome these limitations, it was taken into consideration to use a Twitter scrapper that is called sntwitter, which works in a similar way to Twitter API. So, in order to get the right tweets that suit the problem, it is necessary to choose which words will be used in the query to filter the most suitable texts. Therefore the words selected were "incêndio" or "fogo". The reason for the usage of both words is that in Portuguese it is possible to reference a wildfire in these ways. Also, in order to get more precise results, more data is required in the query, therefore it was essential to give the time frame to acquire the texts between one beginning date and one ending date. Furthermore, it is necessary to specify which language is going to be explored and what kind of filter to apply in order to only get certain results.

4.2.2 Pre-Processing

This sub-section will deal with all the content present in the text in order to remove some of the details that may affect subsequent steps in the pipeline, and also add some complementary data for easier manipulation.

In this way, the text is subjected to the treatment of some data sets such as:

- **Remove URLs**;
- **Transform emoji into text**;
- **Transform emoticons into text**;
- **Remove RT**;
- **Remove @ and # characters**;
- **Remove special characters**;
- **Single character removal**;
- **Remove multiple spaces**;

Given the above list, an explanation of each of the components will be given. Considering the case of lowercase, this is a procedure that is a text normalization method in which, in the case of sentiment analysis, capitalization makes no difference, for example, "I Love You" carries the same sentiment as "I Love You." Another factor that justifies the use of lowercase in texts is the reduction of entries, This means in the example case "Love" counts as one and the word "LoVe" counts as another, while simply converting to lowercase only gets the entry "love" always preserving its meaning.

URL removal is also a step to take into consideration in pre-processing. These are removed for the reason that they do not add useful information for the purpose of what is being done, and in the case of sentiment analysis again, their removal allows the focus to be on words that actually carry meaning. This can also be considered noise as again they contain special characters, numbers, and random strings adding no value.

Now regarding the conversion of emojis into texts, it is a step that is also applied since, of course, emojis cannot be read, and they are quite important since people use emojis to reinforce the sentiment they are feeling at the time of writing. How this procedure works can be explained using the following example:

As can be seen from Figure 4.2 each emoji has a meaning associated with a sentiment.

Just as in the procedure for converting emojis to text, the same procedure is necessary for emoticons. Emoticons are sets of characters that are intended to represent a facial expression. Just as emojis reinforce a feeling, emoticons work in the same way, as it is possible to acknowledge in the next list.

- **:)** = "happy"
- **:(** = "sad"
- **:'(** = "crying"

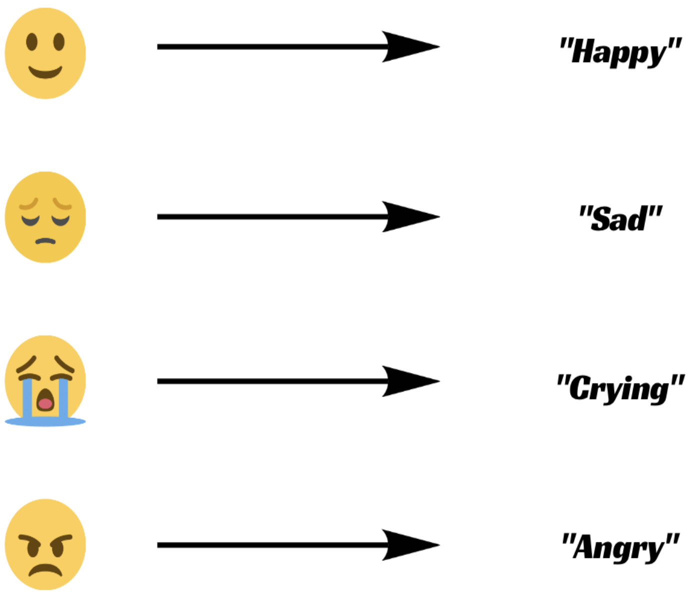


Figure 4.2: Example of emojis and their meaning in text

- >:(= "angry"

This way it is also necessary to remove the RT from the text as it indicates that the text refers to a retweet (repost from someone else post). This indication is present at the beginning of the text and can be considered noise since it does not add any value to the application of NLP algorithms. Just like the RT the @ and # need to be removed.

The pre-processing procedure includes many removals and special characters, single characters, and multiple spaces are also some of those that need to be removed. When talking about special characters, it is necessary to remove punctuation, symbols, and non-alphanumeric values, thus allowing noise reduction and facilitating the tokenization procedure.

Another task that was carried out in the pre-processing was to analyze the similarity of the sentences in order to reduce the amount of text present in the dataset. In this way, the Levenshtein distance [2] was used, which consists of calculating the difference between two strings. It quantifies the minimum number of single-character edits required to transform one string into another.

To calculate the distance there is a Python library that makes the intended comparisons, called FuzzyWuzzy. FuzzyWuzzy calculates the percentage of similarity between two sequences returning the similarity percentage.

To prevent duplicate occurrences, the dataset's data is organized by day, and when the Levenshtein distance computation is performed, if two events on the same day are similar one of them is deleted.

[2] https://blog.paperspace.com/implementing-levenshtein-distance-word-autocomplete-autocorrect

4.2.3 Exploratory Data Analysis

Given the data obtained earlier, it was convenient to make a visualization of the data collected so far to see what relevant information they could convey. In this way, the visualizations obtained were divided into three main groups.

- **Months**;
- **Days**;
- **Hours**.

In the exploratory data analysis (EDA), the analysis of datetime data, with a particular emphasis on months, days, and hours, captivates interest. This captivation arises from the inherent value of deriving insights from comprehending temporal patterns and trends embedded within the dataset. Through the lens of examining data segmented into months, days, and hours, it is possible to find patterns and correlations that might otherwise remain obscured when adopting a holistic view of the data. This way the libraries that will be used to perform these visualizations are seaborn and matplotlib.

4.2.4 Fire Related Classification

After some visualizations of the data collected through the EDA, it is time to classify the data if it is effectively related to an event that represents a fire or not.

This is a very important step because it is through this step that it will be possible to determine which data is useful. This way, only the contributions that refer to a fire will result, avoiding the contributions corresponding to noise that, according to the classifier, have nothing to do with a fire report.

With this being said it was necessary to apply methodologies that handle the task and for that three were taken into account, and those are:

- **Fine-Tuned Classification**;
- **Few-Shot Learning Classification**;
- **Zero-Shot Learning Classification**.

All of the list's components will be used to comprehend the fundamental differences between them, as well as a performance comparison.

4.2.5 Fine-Tuned Classification

One of the approaches taken into account is a fine-tuned classifier. This consists in using a model that was refined for a determined NLP task, and the task for this project is to get the ability to distinguish between true and false reports effectively. The way this is done is through a classification of the text to see if it actually belongs to the event label. Based on what had been determined in section 3.3.3 a pre-trained BERT model is the best choice for this specific task.

Once it was already decided that the base model to be used would be BERT, it was necessary to find a model already pre-trained for the Portuguese language. After some research, the Hugging Face BERTimbau Large also known as the bert-large-portuguese-cased model emerged. This model is trained to deal with some NLP-focused tasks, and also as the name implies it works for the Portuguese language.

Through this pre-trained model and already having a labeled dataset, the process is to train the model based on the dataset, and from then on, all the input data will be based on the prediction made by the fine-tuned model.

4.2.6 Few-Shot Learning Classification

In a way to test through different emerging methodologies, Few-Shot Learning was used in this case.

This paradigm consists of training models that effectively learn through limited methods on limited text data.

Sometimes the acquisition of annotated datasets in a complete and rigorous way is not easy to find and given the case study of the project in concrete even more difficult. There is also the possibility of creating an annotated dataset but it is quite time-consuming. In order to overcome these challenges Few-Shot Learning allows the model to learn through a few examples per class or category, making it possible to adapt to new tasks with minimal labeled data.

However, within the NLP area, it works for several tasks such as text classification, sentiment analysis, question answering, and many others.

In the context of the project, this method will be used for the task of text classification in the area of NLP.

For the procedure of this technique, some sample sentences will be supplied that will have the associated labels that will act as a comparison with the dataset data and assign the correct label using similarity metrics, and if they effectively correlate to the event the data is preserved.

4.2.7 Zero-Shot Learning Classification

The last method used is Zero-Shot Learning. This method allows NLP models to recognize and classify text instances into classes or categories that were not seen during the training phase, without the need to train the data explicitly for these classes.

What differentiates this method from Few-Shot Learning, is that this one focus is on scenarios where some of the classes are new and or not available during the training phase. The goal is to leverage transfer learning and external information to make predictions on unseen classes.

The way this method will be applied is by defining the set of labels that we want to be associated with the dataset phrases. The labels chosen were selected based on a preliminary analysis of the existing content, so the labels are "incêndio", "clima", "animais", "comida", "festividades", "sexual", "temperatura", "noticias".These labels were chosen because they are related to the keyword "incêndio" or "fogo".

Once the labels were determined, a zero-shot classifier was selected using the pre-trained BERT-based model for Portuguese text, which is then used to iterate over the dataframe in which if the classifier associates the label "incêndio" with the contribution it was considered an event. All other labels that are associated with the text are discarded, having as output only a dataset with events that only report fires.

4.2.8 Assign Location

The final phase of pre-processing is one of the most important since it is here that the coordinates for the event are allocated, allowing it to be located geographically. Therefore, in this way, it was necessary to find a way to identify the location through the text, and given some research, some possibilities were found such as:

- **Google Maps Geocoding API**;
- **Bing Maps REST Services**;
- **OpenCage Data API**;
- **Nominatim (OpenStreetMap)**.

When analyzing in more detail all the hypotheses collected, some considerations must be taken into account, namely whether the use of the tool is free and works as intended. In this way, both the Google Maps Geocoding API and Bing Maps REST Services are paid services, so for this purpose, they are not the most suitable. In this scenario, both OpenCage Data API and Nominatim (OpenStreetMap) are already free tools, but the first is restricted to daily usage of free consultations, which is no longer the case with the second, which is entirely free. Thus, the Nominatim tool was selected for its simplicity and the fact that it is free.

So the approach is to extract the text from the dataset and then use a Named Entity Recognition (NER) to extract the location, which is then sent to an API, which returns the coordinates that will be utilized for geographic visualizations.

4.2.9 Sentiment Analysis

After the pre-processing phase was carried out, we moved on to one of the most fundamental phases of the pipeline, which is the sentiment analysis part.

Through it, it will be possible to classify the text and assign a sentiment, allowing to understand what the citizen wants to transmit and its intensity. In order to support all the sentiment assignments, some visualizations were made, which will help to understand some of the variations of the data.

Thus, the first step for the sentiment analysis task is to select the best-suited model since the contributions were in Portuguese. Therefore, after deeper research, the model "twitter-xlm-roberta-base-sentiment" from cardiffnlp [3] emerged, which is trained for multiple languages, and it was also trained with 198M tweets and fine-tuned for the sentiment analysis task, especially for eight languages such as Arabic, English, French, German, Hindi, Italian, Spanish and Portuguese.

Once the model has already been selected and in order to simplify the execution of the task, the pipeline has been determined, which is a function provided by the "transformers" library. This will allow the loading of the previously defined model and the tokenizer for the specified task. Thus, by giving the dataset, the sentiment is assigned for each of the contributions.

4.2.10 Visualization tool

One of the most important ways of exposing the information obtained, which was gathered through an extensive pipeline such as the one carried out so far, would be through a dynamic visualization platform. In this way, the reasons that led to this decision-making are that the views obtained through the individual execution of each sub-section would not be sufficient to draw concrete conclusions from the events that were taking place. In this case, if a layperson wanted to analyze the individual visualizations provided by each program, would have difficulties, due to the lack of contextualization of what is being analyzed and it requires a previous execution of a set of programs with the corresponding libraries installed. Therefore, given what was mentioned, it was necessary to resort to one of the most well-known data visualization tools, which is Tableau.

What motivated the choice of this tool is the fact that it does not need to be on just one device and can be centralized on their online platform, thus facilitating access to the visualization. Another factor was the fact that its interface is relatively simple to understand and deploy new visualizations, and also because it can create very interactive visualizations allowing a more detailed exploration of what is being observed.

Once the platform was selected, it was necessary to decide how the visualization would be carried out and what data would be in close-up. Since it is a problem in which the identification and location of the event are the fundamental points to be visualized, a map would be necessary, and also some complementary

[3] https://huggingface.co/cardiffnlp/twitter-xlm-roberta-base-sentiment

graphics that provided some more useful information. In addition, several sliders have been added, allowing you to analyze different events over different time periods.

4.2.11 Clustering Events

One of the last intended goals would be to find a way to group the data to try to relate events that occurred, respecting some predefined conditions. Therefore, it would have to be an unsupervised method, and given the conditions of the problem, the best technique to follow would be a clustering one. This technique is good because clustering groups particularities based on points and their similarity without having predefined class labels.

Within the clustering techniques, there are several methods such as:

- **Hierarchical Clustering**;
- **Partition-Based Clustering Methods**.

The main difference between the two is that while the hierarchical method finds the number of clusters through the dendrogram without prior knowledge, the partition-based clustering method does not allow this and has to define the number of clusters in advance. Since hierarchical clustering is the best choice, the choice of algorithm becomes simpler and HDBSCAN becomes the best since its purpose is to group data according to their density, allowing better interpretability. It is a wise choice as it groups data points into clusters based on density and easily identifies outliers as being noisy points.

Once the technique and algorithm were defined, it was necessary to proceed to execution. In this manner, it was necessary to find the best number of clusters for the problem but the chosen algorithm already chooses the best number. However, through a dendrogram (visual representation of the merging process and the number of clusters through a specific cut-off level or through a threshold), more information can be retrieved like potential patterns or relationships within the data. To apply the algorithm it is necessary to pre-process the data for clustering, and according to the problem, it was necessary to group based on Latitude, Longitude, and Datetime.

After grouping the data, a final visualization was created to show how the events linked to each other across time, allowing inferences to be reached about location and sentiment.

4.3 Final Remarks

In summary, this chapter has taken into account the implementation strategies from data extraction to event clustering. Data acquisition will be meticulously collected using a library that facilitates the extraction of "tweets" from the Twitter social network, followed by some pre-processing strategies aimed at improving the content of the text and making it more interpretable for algorithms that may be implemented in the future. Next, the classification of events will be carried out in order to see if a certain event is related to the event being analyzed. Three different methodologies were used for this, fine-tuning a classifier, few-shot classification, and zero-shot, all of which use a pre-trained BERT model.

Furthermore, a pre-trained model called twitter-xlm-roberta-base-sentiment was used for sentiment analysis. This choice was due to the fact that it was trained with content that uses the Portuguese language, thus allowing a better prediction of which sentiment the text refers to.

Since there is a need to explore the data in a visual way, an analysis of the best tools that fit the context of the thesis was selected, and it was concluded that Tableau would be the most suitable tool since it allows advanced and public visualizations to be shared with others.

Finally, an analysis of which type of clustering algorithm suits the purpose of the thesis project is hierarchical clustering, since the number of clusters is not explicitly known, and the algorithm selected was HDBSCAN.

The choice of these approaches was based on research and all the development took into consideration the topics examined in this chapter.

Chapter 5

Results

In this chapter, the results obtained in this work are analyzed. For each subsection 4.2, there is a section in this chapter where the data collected and the visualizations created are examined, in order to gather valuable insights to contribute to the objectives defined in the thesis and furthermore also to the FireLoc project. By presenting and interpreting these results, this chapter aims to provide an understanding of the study carried out.

5.1 Data Extraction

To start the procedure of extracting data from Twitter through snstwitter, it was necessary to specify some details in order to obtain the correct data related to the event. The capture of this data is done through a query, where in this specific case and in the context of this thesis, it is necessary to indicate the language, the time period, and the keywords allowing to get certain results.

So in the end the query would look like this:

- **query** = ′fogo OR incêndio -filter:replies lang:pt since:2017-06-01 until:2017-11-01′

This query selected all the Portuguese tweets between one of June and one of November, that have the keyword "fogo" or "incêndio" in them, and only tweets that are posted and originated from a user of this platform are considered, other types of text like replies are discarded. This time period was selected because it was a period when fires ravaged the Portuguese territory and the number of contributions would be higher.

This way, the output of this query will collect a set of tweets that contain the date, including the hour they were published, the id of the tweet, the text that is the most important, and also the username of the person or entity who made the tweet.

Thus, to demonstrate the result of the query applied, the following tables were obtained.

Table 5.1: Retrieved Tweets

Tweets
Novo incêndio em Bragança Macedo De Cavaleiros Vinhas;
Incêndio ocorreu agora pouco na fronteira com o Vaticano sede da Igreja Católica;
Bombeiros no local com dois caminhões alto bomba Causas do incêndio ainda desconhecidas;
Novo incêndio em Lisboa Cascais São Domingos De Rana FogosPT;
Bugueiro de Canoa Quebrada sofre queimaduras de segundo grau ao apagar incêndio em seu veículo Foto postado por;

Table 5.2: First dataset from Twitter scrapper

Datetime	Tweet Id	Text	Username
2017-06-01 22:08:53+00:00	870404177725460480	Tweet 1	uainoticiaspro
2017-06-15 22:38:48+00:00	875481081209798656	Tweet 2	JornalNoticias
2017-06-17 18:23:04+00:00	876143168688926721	Tweet 3	observadorpt
2017-08-14 14:38:15+00:00	897105090259144705	Tweet 4	FogosPT
2017-09-27 12:01:04+00:00	913010598161653761	Tweet 5	exame

When looking at the table 5.1 it is possible to acknowledge some of the texts retrieved and used to complement the table 5.2.

Once all the Twitter data extraction is done, the final dataset will look like the table 5.2 and this will be the starting point for all the following steps. The dataset collected from the keyword "incêndio" from the period mentioned before got 64,795 lines, and the data gathered from the same time frame of the word "fogo" corresponds to 38,6138 entries.

In a first analysis looking at the data, it is possible to verify that the dataset with the keyword "incêndio" in a general way effectively corresponds to contributions from an event in which a forest fire is occurring. Regarding the other keyword, since it has more than five times of entries, it is possible to conclude that looking at the data that has a lot of lines that can't be considered as reports. The high number of data retrieved with the keyword "fogo" is due to the fact that this word has a double meaning, effectively referring to a forest fire or expressing a feeling of pain, displeasure, or even indignation in the Portuguese language.

Given the conclusion drawn above it is necessary to perform some kind of filtering to obtain only the contributions that refer to the occurrence of an fire event.

5.2 Pre-Processing

As mentioned in the 4.2, in the pre-processing phase, the data needs to go through a series of steps to make it easier to read and remove all the noise from the data. Basically, the procedure for deleting all forms of data involves reading through the text and identifying the set of data that corresponds to the one being examined and removing it, such as locating the '@' in the text and removing it.

However, it is not just removing data, but also filtering it to add useful information that can facilitate data manipulation. Therefore, filtering was applied to the column that refers to the "Datetime" in order to be able to extract as much information as possible in order to make it more organized and easily accessible.

In this way, a single column resulted in four more, which are the year, month, day, and hour. So the day 2017-06-01 23:49:25+00:00 becomes the following, year: 2017, month:6, day:1, and hour:23.

Organizing the data in this way will allow more concise conclusions to be drawn through visualizations.

Also, another task that was carried out in the pre-processing phase was the similarity between texts, using the library FuzzyWuzzy, allowing to reduce the number of contributions of the same event on the same day. The procedure is done in a cyclical way. The first entry in the dataset is compared to the second entry. If the result of similarity is more than 80%, the text is considered similar, a counter is added to the first entry, and the second item is excluded, forwarding to the next entry. If the similarity of the entry is less than 80%, the contribution is maintained and not discarded.

In the case of the dataset captured with the keyword "incêndio" at the beginning, there were 64,796 entries, but after the implementation of Levenshtein distance 22095 entries were removed. Leaving the dataset with 42701 contributions.

Based on all that has been said in this sub-section, the captured text was subjected to a treatment resulting in a sample which is presented in Table 5.3.

Table 5.3: First preprocessing dataset

Tweets
novo incêndio em porto gondomar foz do sousa covelo; novo incêndio em santarém ferreira do zêzere igreja nova do sobral; ainda nem cheguei coimbra já caem cinzas do incêndio da mealhada já cheira queimado céu está assim; novo incêndio em santarém abrantes mouriscas fogospt; Bugueiro de Canoa Quebrada sofre queimaduras de segundo grau ao apagar incêndio em seu veículo novo incêndio em leiria caldas da rainha landal fogospt;

Table 5.4: Second dataset from preprocessing

Datetime	Year	Month	Day	Hour	Text	Similar
2017-06-03	2017	6	3	22	Tweet 1	2
2017-07-13	2017	7	13	4	Tweet 2	1
2017-08-10	2017	8	10	18	Tweet 3	0
2017-09-03	2017	9	3	21	Tweet 4	0
2017-10-15	2017	10	15	21	Tweet 5	1

As it is possible to verify through the table 5.4, there were some changes made, namely the removal of some columns that are not necessary like the username and the tweet id, which were not kept for privacy reasons. However, from the table 5.4, you can already see that the data is more organized and cleaner, allowing you to have more accessible data for the next phases.

5.2.1 Exploratory Data Analysis

As mentioned before, the initial analysis will focus on the time period, namely months, days, and hours.

All the initial visualizations follow the same logic with the main objective of understanding the data trends. The first visualization is composed of several plots where the first one indicates the data count, the second one considers the density over the count, the third one only visualizes the density and the fourth one is a visualization of the density but in a more smooth way, making it more generalized and less sensitive to variations in the data. The second visualization is composed of a plot that shows the count of data in order to obtain a visualization of easy understanding. The first figure was taken into account the months and this first visualization obtained the following figure 5.1.

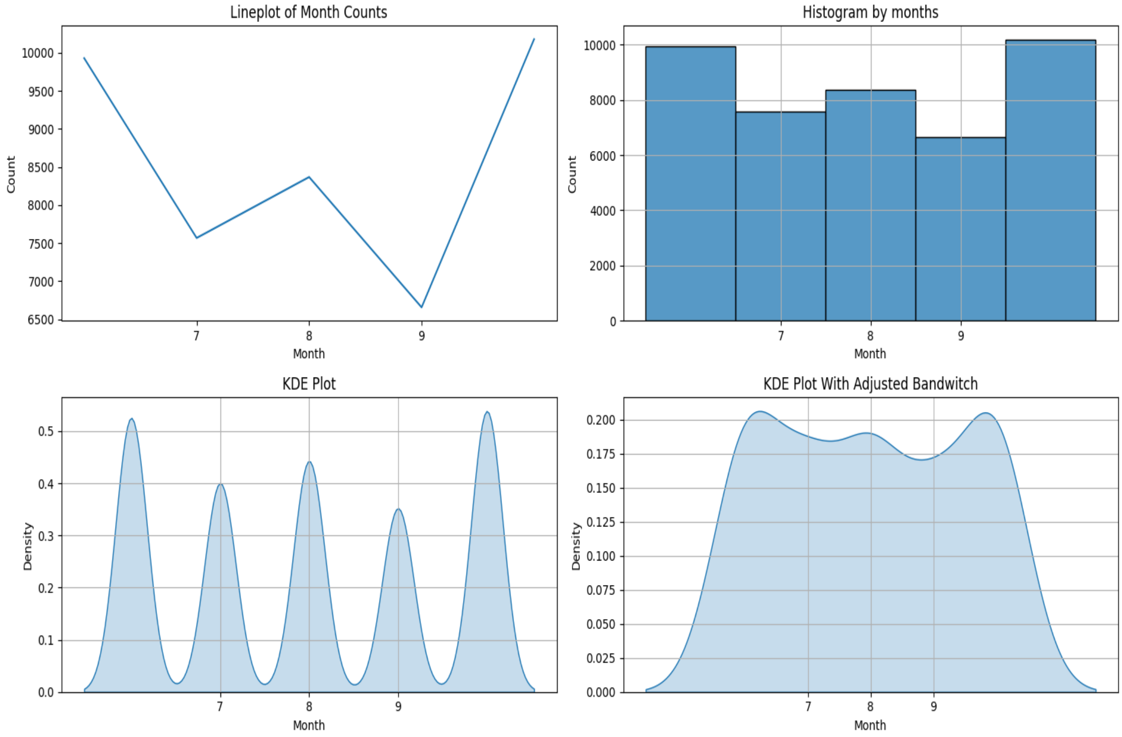


Figure 5.1: Number and density of tweets filtered per month, from June to October 2017

As can be seen in figure 5.1, it can be concluded that 2017 was an atypical year, because unlike what usually happens in the months of July, August, and September, in the year of visualization these months obtained a lower number of contributions and this may be a consequence of the absence of forest fires. This conclusion will be reflected in the same way in the densities since a lower density is obtained in the months mentioned above, unlike what happens in the months of June and October.

In this way, the next table is displayed in order to obtain more details on the counting of contributions per month.

Once the months were analyzed, the group to be analyzed is the days.

Table 5.5: Count of reports by month

Month	Count
June	9929
July	7568
August	8368
September	6657
October	10178

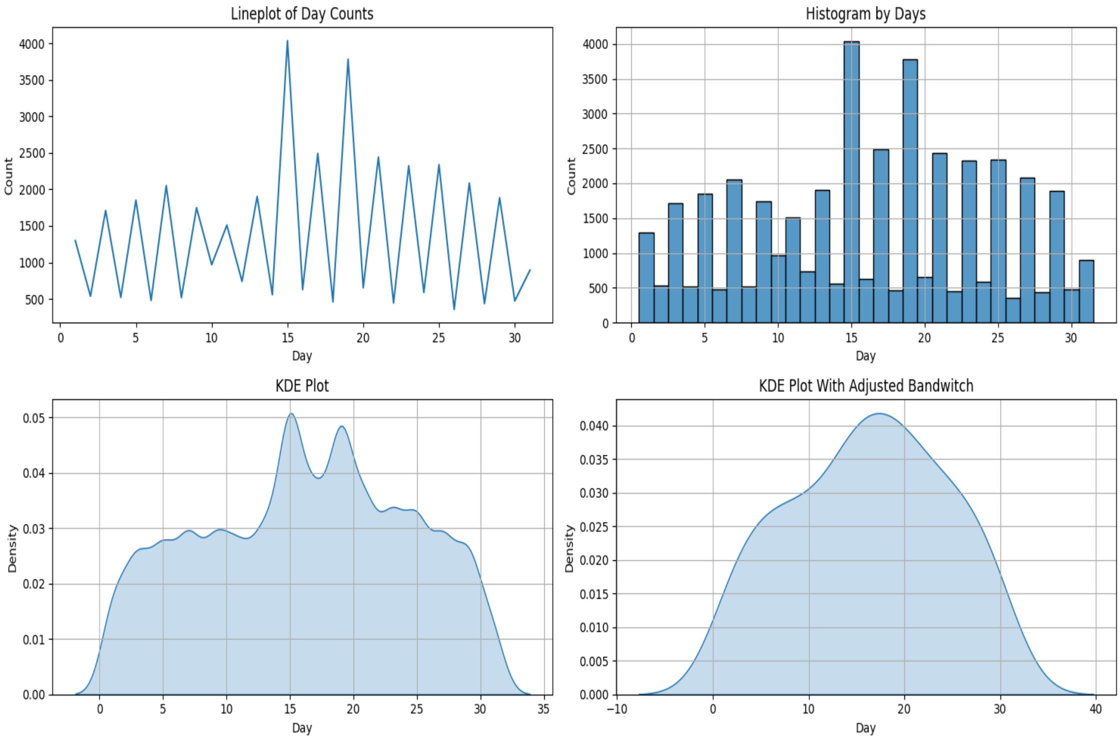


Figure 5.2: Number and density of tweets filtered per day, in the year 2017

In the figure 5.2 it is possible to see that, unlike the previous group, this one has a larger range of values which can make it difficult to visualize. However, in a first analysis, as it is possible to acknowledge, the day that obtained the highest number of contributions is undoubtedly the 15th of all months, and between the period from the 15th to the 19th, it is denoted that the density is higher. One of the conclusions that can also be drawn is that the number of contributions captured is higher in the second half of the month.

Looking at the figure 5.3, it is possible to see that the highest number of contributions is 4035 and the lowest number is 358.

Once the values that refer to the days were explored, it was time to move on to more concrete data that refer to the group of hours. With the analysis of these data, it is possible to understand several factors about the citizens and about the events to be visualized.

It is possible to verify through figure 5.4 that from 7 a.m. to 22 p.m. an exponential increase in contributions has occurred. This can be explained by the transition from night to day and also because it is the period in which citizens

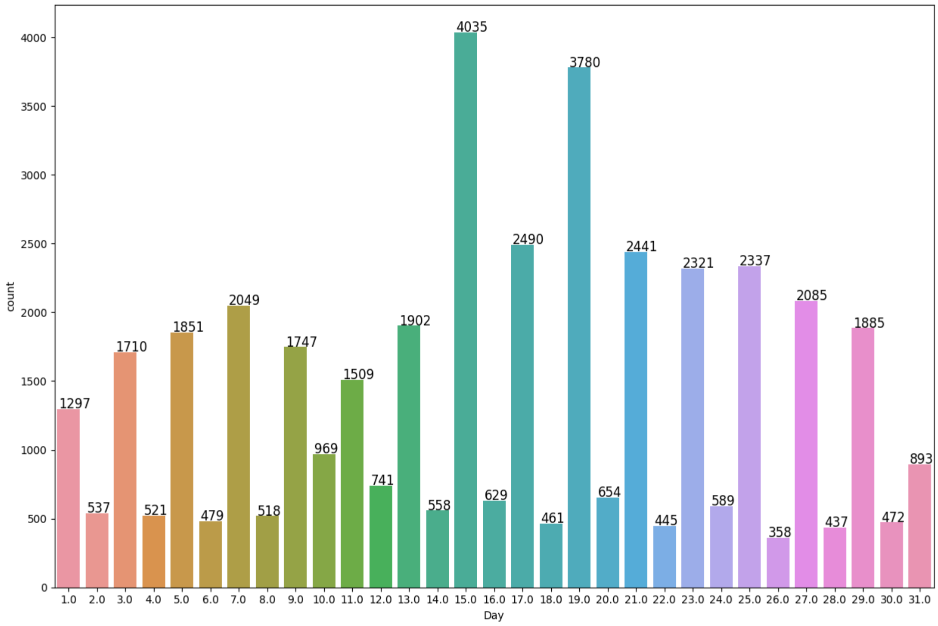


Figure 5.3: Count of tweets per day, in the year 2017

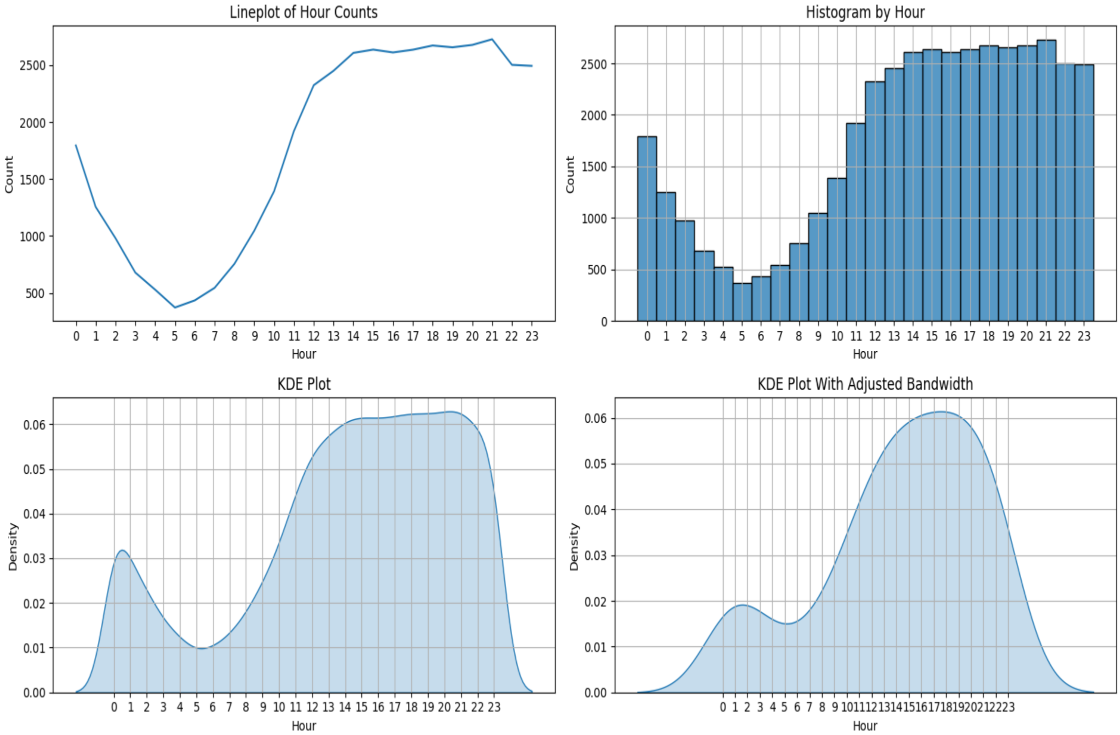


Figure 5.4: Number and density of tweets filtered per hour, in the year 2017

usually wake up. It can also be concluded that the number of contributions is almost constant in the period from 2 p.m. to 9 p.m., as it is possible to conclude from the count and density, with the maximum reaching 9 p.m. The high number of these reports may be due to the fact that during this period the heat starts to get higher, and also because they are more favorable times for the use of social networks.

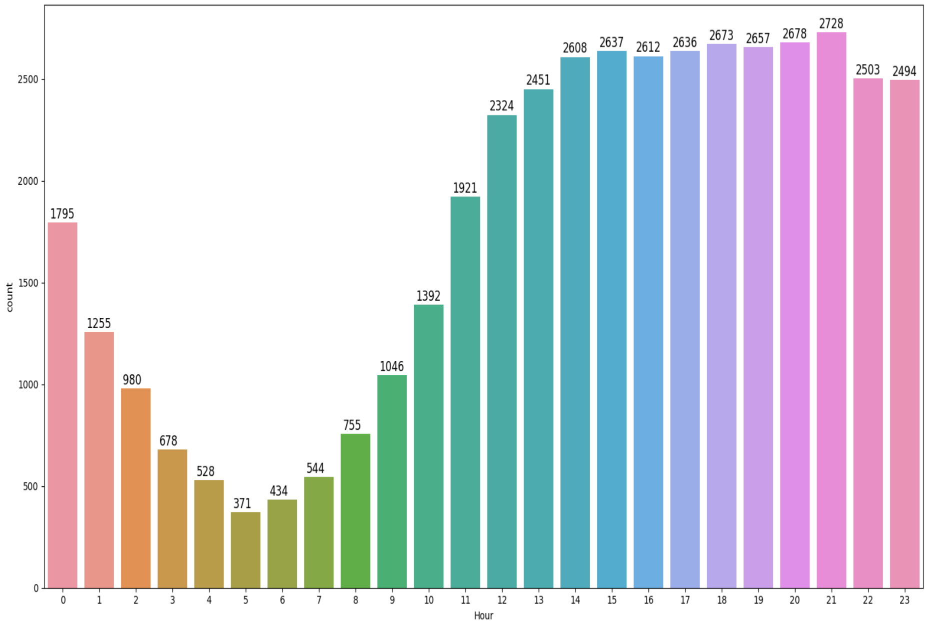


Figure 5.5: Count of tweets per hour, in the year 2017

To complement the set of plots and to reinforce the idea of values by the hour, in Table 5.5, it can be shown that the lowest number of contributions was at five in the morning with a value of 371, and the highest value was obtained at 21 with a value of 2728 reports, in addition, it can be seen that between the period with the highest number of contributions, it ranges from 2608 to 2678.

The contributions seen in the exploratory data analysis all refer to the dataset that is composed of the keyword "incêndio".

5.2.2 Fine-Tuned Classification

Based on what was said previously, the first approach taken into account is a fine-tuned classifier, this will make it possible to classify whether the event reported belongs to an occurrence.

Everything starts with a dataset that has news headlines. This dataset is related to the fire news data, which is labeled, allowing for an increase in usable information that might be relevant for its usage. This dataset consists of a column called "text" which is composed of the contributions collected from various data sources, and another column called "label", which identifies whether the report actually refers to a fire, this identification is given through two values: 1 (a contribution that actually deals with a fire) or 0 (a contribution that has nothing to do with a fire). This dataset will be used to fine-tune our BERT model.

Once it was already decided that the model to be used would be BERT, it was necessary to find a model already pre-trained for the Portuguese language. After some research, the BERTimbau Large also known as the bert-large-portuguese-cased model emerged. This model is trained to deal with some NLP-focused tasks.

Since the fundamental bases for running a fine-tuning program for the model had been determined, it was necessary to initialize the execution of the program itself. Therefore, in the beginning, it was necessary to import all the data mentioned so far. This data will pass to the training phase, which is done through the transformer's library, and thereafter the evaluation phase. In this way, 30 train-test splits were generated and for each of them, the classifier was executed and saved the results. Then the best training arguments determined from the file, where the saves are, will be used to determine the best possible result, with the aim of testing whether the result is effectively favorable. This whole procedure occurred with support in 2 major classes. The first is 'TrainingArguments' in which various configurations are specified to train the model, receiving the file where the training data is stored, and in addition, receiving the number of epochs. The epochs represent the times that the model will pass through the entire training dataset. The second class is the 'Trainer' class, which also comes from the transformers library and is used to facilitate the training process, using several parameters such as the training model, the training arguments, and finally the data to be trained. A confusion matrix is then created to obtain data for the main metrics, such as accuracy, precision, recall, and F1 score.

Table 5.6: Results obtained by the first 10 epochs of Fine-Tuned Classification

Accuracy	Precision	Recall	F1
98.2%	88.2%	88.2%	88.2%
98.6%	93.7%	96.7%	95.2%
96%	81.8%	90%	85.7%
93.8%	64.2%	81.8%	72%
96.9%	87.5%	84%	85.7%
96%	85.1%	82.1%	83.6%
96%	82.7%	85.7%	84.2%
96.4%	78.5%	68.7%	73.3%
97.3%	75%	93.7%	83.3%
96.9%	84.6%	88%	86.2%

After the execution of this program, it is possible to conclude from Table 5.6, obtained a performance overall where the accuracy was superior to 96% and in precision, the results were also superior to 81%, and again the recall was with results greater than 83% while the F1 score obtained results close to 82%.

5.2.3 Few-Shot Learning

According to what was said previously, it is known that Few-Shot Learning is a recent technique and that to work it does not need to train the model based on an entire dataset, working only through a few examples. This way, ten examples of each of the desired labels were created, one that identifies as "incêndio" and the other as "outros". These examples are composed of a sentence and the respective label, respectively.

Some of the examples used are:

- "text": "Incêndio destrói parte de fábrica na Zona Industrial de Leiria", "label": "incêndio";
- "text": "novo incendio em braganca mogadouro urros", "label": "incêndio";
- "text": "Fogo em Sintra mobiliza mais de 150 bombeiros e três meios aéreos", "label": "incêndio"
- "text": "Fogo de artifício ilumina o céu na festa de encerramento do festival", "label": "outros"
- "text": "fogo ninguem quer vir comigo amanha setubal", "label": "outros"
- "text": "guarda municipal preso por porte ilegal de arma de fogo", "label": "outros"

As it is possible to acknowledge from the list above, the sentences that actually mention an event that reports an occurrence have the label "incêndio" and those that do not correspond have "outros". When looking at the sentences that correspond to the label "outros", they all have the keyword "fogo" because as mentioned before it has a double meaning, and a lot of the contributions captured aren't related to a fire.

Once the examples provided have been defined, the procedure pipeline is to calculate the embeddings of the phrases that come from the dataset. This happens by preprocessing through a tokenizer, which then passes through a transformer model and calculates embeddings, by taking the average of the last hidden state, thus returning a PyTorch tensor. The same calculation is performed for the example phrases.

Considering that all sentences have their corresponding embeddings, the cosine similarity between sentences from the dataset and sentences given as examples is calculated, thus allowing to determine which is the example sentence with the greatest similarity and assigning the corresponding label. In the end, sentences labeled "outros" are discarded in order to have only contributions that actually correspond to a fire event.

The final dataset will be only with the reports that effectively correspond to a fire event. For testing purposes and in order to use the evaluation metrics to understand the performance of Few-Shot Learning, the annotated dataset "news.xlsx" is used, which contains several news headlines and indicates whether it is related to an event. So to achieve the intended purpose, the program was executed over the entire dataset and the label was compared with the "fire-related" column that contains the annotation given by humans. This will allow to understand the performance based on the examples that were provided.

Table 5.7: Results obtained regarding Few-Shot Learning

Accuracy	Precision	Recall	F1
30.2%	13.6%	96.4%	23.8%

Looking at the findings in the last table 5.7, the accuracy attained was roughly 30.28%, indicating the percentage of the total occurrences in the dataset that were properly predicted. Regardless of accuracy, it obtained 13.61% of positive predictions, and the model has a pretty high false positive rate. In terms of recall, the model correctly detects around 96.48% of the real positive instances in the dataset; a high recall is typically preferable because it accurately captures the bulk of positive events. Finally, the F1-score was 24%, which accounts for both false positives and false negatives. The low proportion shows that the trade-off between accuracy and recall is not particularly high in this situation.

5.2.4 Zero-Shot Learning

As mentioned in sub-section 4.2.7, Zero-Shot Learning is a method that works where classes are new or not available during the training phase. In this way, a set of labels were identified so that with the support of the transformers library and pipeline class it was possible to associate the labels with the text. The approach involves using the pipeline code line to the method that is being utilized in this case is "zero-shot classification" and the model that, as previously said, belongs to a Bert pre-trained model. Later and cyclically traversing the dataset this pipeline will be called, which will serve as a classifier, giving the labels that are present in a sample set of pre-defined phrases. Those phrases will serve as the context for the classifier, which returns to which labels it belongs, and in the case of assigning a label, it is kept, otherwise, it is discarded. In order to evaluate the effectiveness of Zero-Shot Learning, once again the dataset labeled with news titles was used to see if the predictions actually corresponded to fire-related events.

Table 5.8: Results obtained regarding Zero-Shot Learning

Accuracy	Precision	Recall	F1
43.7%	12.3%	64.8%	20.6%

Looking at the table 5.8, it can be concluded that the recall metric is relatively high compared to the other metrics, which means that the model identifies positive instances well but contains a considerable number of false positives. The fact that a low F1-score is obtained suggests that there may be more room to improve the balance between precision and recall.

5.2.5 Assign Location

As seen, the location assignment API that best suits the project, due to its characteristics, would be Nominatim. So, the procedure is to receive the text from the dataset and through a NER (Named Entity Recognition) which is provided by spaCy find in the text an argument that is equal to the tag "LOC". This allows to get the locations present in a sentence, later allowing to send this information to the API that in return gives the coordinates in the format of latitude and longitude that will be used for geographic visualizations. Since the data captured in the first section 5.1 captures data related to the Portuguese language, it is important to note that there are other countries that write in this same language. So it was necessary to apply a restriction. After obtaining the conversion of the text into latitude and longitude, there is the possibility of realizing what the country code is (unique code that identifies a country or a geographic area) and if it is different from 'pt' this code is discarded, as it is not relevant for the purpose of the thesis or the project

Taking into account the text example "Novo incendio em Leiria Caldas Da Rainha Nadadouro FogosPT ", when running a NER model that is provided by spaCy the output of the location result will be Caldas da Rainha. Then the result is sent to the Nominatim API which will return the coordinates of Longitude: 39.4071857 and Latitude: -9.1346004 which refers exactly to the captured location.

5.3 Sentiment Analysis

For the execution of sentiment analysis, it will be necessary to have defined some data, and also a pre-trained model to be used for this specific task which needs to be compatible with the Portuguese language. So the model found was twitter-xlm-roberta-base- sentiment from cardiffnlp, and as support, the pipeline class that is part of the transformers library will be used. The procedure involves defining the first attribute of the pipeline that corresponds to the task to be executed, which in this case is "sentiment-analysis", the model, and after the tokenizer. Later and as done before, it will be necessary to iterate over the entire dataset returning the sentiment of each contribution. In the end, the output of the execution of this task over the dataset will have a sentiment, either positive, negative, or neutral.

As can be seen from the figure 5.6 and as expected, most of the contributions are negative, leading one to believe that it is, in fact, the identification of a catastrophic event that is occurring. However, it is important to mention that although the largest percentage belong to negative, there is still a small percentage of neutral feeling and positive feeling. In this way, a pie chart was created showing how the distribution of sentiment is expressed in the contributions.

Some visualizations can reveal some useful information to see the sentiment of the citizens over time. So in this way, when analyzing the figure 5.7 in more detail, there are some conclusions that can be drawn. When the sentiment is analyzed over time, it is clear that the negative sentiment is unquestionably the one that remains more stable, while the neutral has some contributions that show some failures of continuity over time. Finally, and not least, the positive feeling appears very sporadically over the analyzed time. Another detail that can be examined through the graphic is that the y-axis represents the score achieved for each sentiment. This score defines the value from 0 to 1, with the closer to the

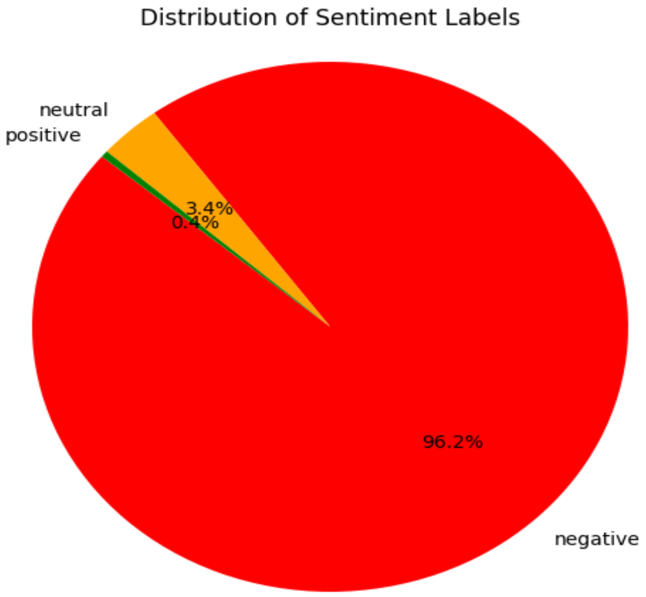


Figure 5.6: Sentiment of the data regarding fire events

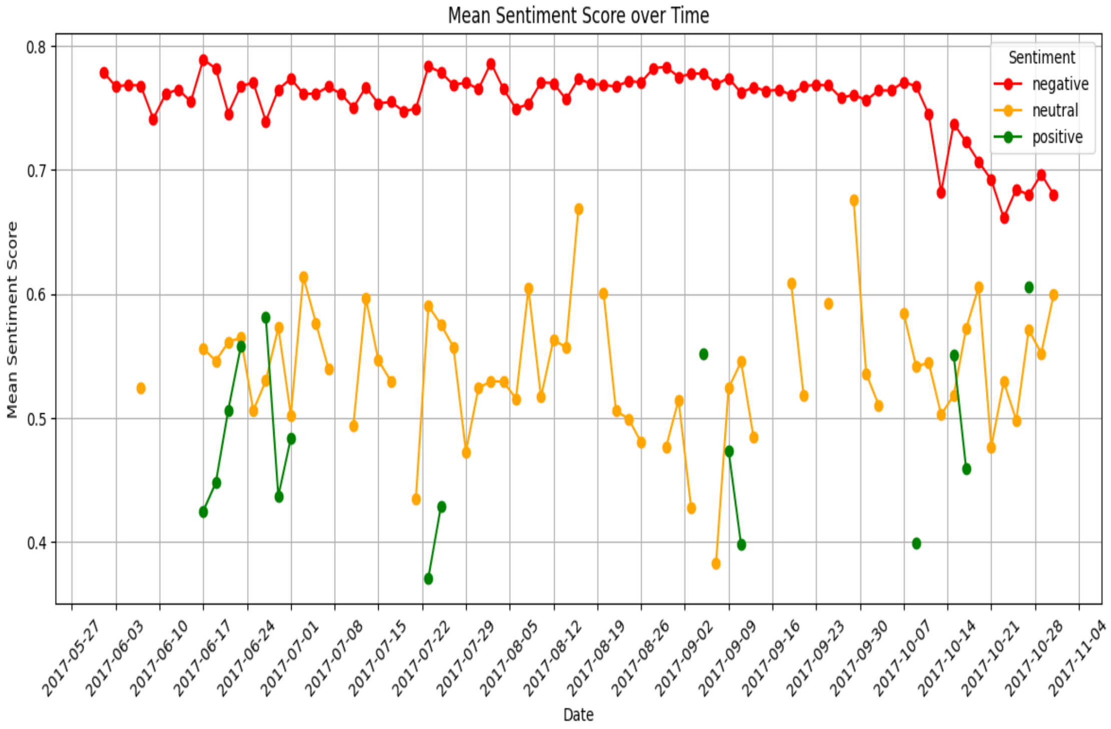


Figure 5.7: Mean Sentiment score over the time period analyzed

value one, the stronger the confidence connected with this feeling. Thus, when analyzing in greater detail, the negative sentiment is largely comprised between the range of 0.7 to 0.8, indicating that the captured contributions are highly associated with the sentiment. However, in the case of the neutral sentiment, a large part is found between 0.5 and 0.6, giving it a lower confidence level. Moreover, among these feelings, the one that is lower is the positive sentiment, which has a very variable confidence level and is lower relative to the neutral sentiment.

In order to support the conclusions drawn from figure 5.7, a graph was also made that allows to visualize the number of contributions over time for each sentiment this graph is exposed in figure 5.8.

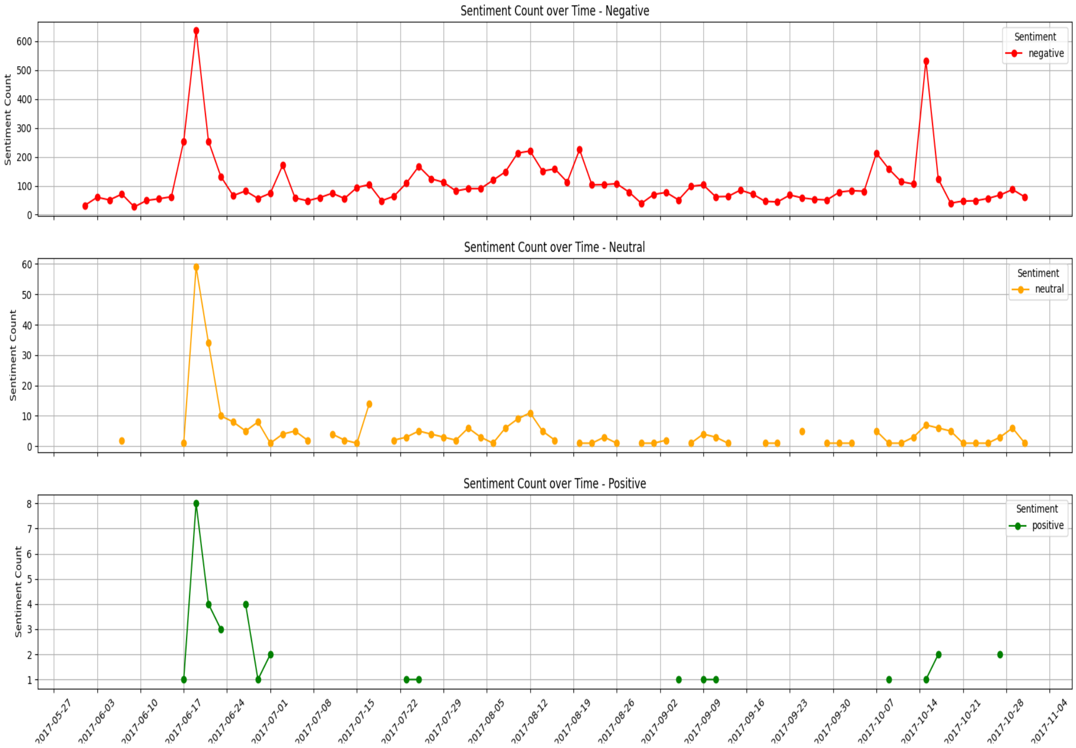


Figure 5.8: Sentiment count over time

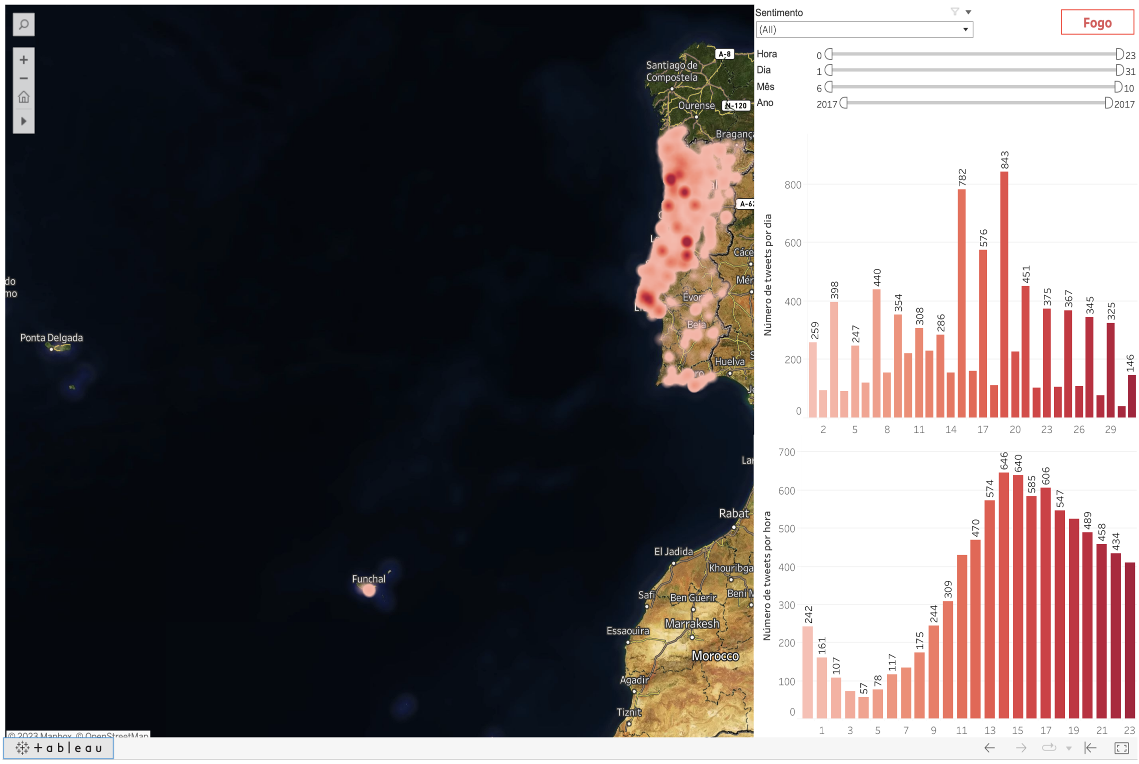
As can be seen, the scales for each type of graph are different. This is due to the high number of negative contributions that there are compared to the other sentiments. In addition, it is clear the continuity of negative sentiment over time which again does not happen with both positive and neutral sentiment.

5.4 Visualization

As mentioned in the previous chapter section the choice of the most suitable tool for this project is Tableau, and in this way, the creation of a visualization proceeded.

Thus, using the aforementioned tool and its capabilities, the resulting visualization was created.

By looking at the figure 5.9, it is possible to see that the visualization is simple, yet very concrete, as it is not a visualization that requires a very deep understand-



Sentimento
(All)

Hora 0 – 23
Dia 1 – 31
Mês 6 – 10
Ano 2017 – 2017

Figure 5.9: Visualization of the fire reports in the year 2017

ing of what is being visualized only a little bit of contextualization, leaving aside the complexity. On the left side, there is a density map. This type of map allows the visualization of the distribution of points in the geographical area, being often used to demonstrate the density of events or occurrences along the map. This uses color gradients or heatmaps to indicate areas with high or low density, where lighter colors represent areas with low density and darker colors represent areas with high density. The color gradient used for this map is red, and the presence of more stronger color is evident in the center and north zone, suggesting that throughout the time span to be viewed, it was where a higher number of fire events happened. In addition, the map can be enlarged to see in greater detail where the event is taking place, even allowing the collection of what type of terrain or area it is affecting, can be in a residential area, or even a forest. In addition to visual data, as the cursor is passed over each point, there is a feature that provides further textual detail about each event, such as what time it happened, how many comparable occurrences were reported on that same day, and the collected text that describes the incident. The number of events reported per day will affect the size of the marked geographic point, and the more events there are on that day, the larger the point will be.

On the right side, there are buttons where it is possible to change the parameters to see what events have occurred. In this way, it has a drop-down menu to change the sentiment, and a little below it is possible to see the limit bars that define in greater detail the period of time to be analyzed such as the hour, day, month, and year, and depending on the change done, the maps and graphics that are found below the buttons change all simultaneously. Below the buttons are two bar graphs. The first defines the number of tweets captured per day, where the y-axis is the number of tweets and the x-axis is the days, while the second graph shows the number of tweets captured per hour, where the y-axis, like the previous graph, is the number of tweets and the x-axis are the hours. These graphs are informative, giving the user information on how the distribution of data on the analyzed conditions occurs.

With this visualization, it enables easy and autonomous usage by individuals who do not have a thorough understanding of the subject, as well as retrieval of information about occurrences.

Another feature is that this visualization was made available to everyone associated with the FireLoc project, giving them complete access and analytical power by hosting the visualization online with one of Tableau's functions[1].

5.5 Clustering Events

For the clustering of events, the data was grouped according to date, latitude, and longitude, allowing to collect events that occurred in the same area in the same period of time. In order to try to get some extra information about how the clusters group together, a dendrogram was performed using the scipy.cluster.hierarchy library.

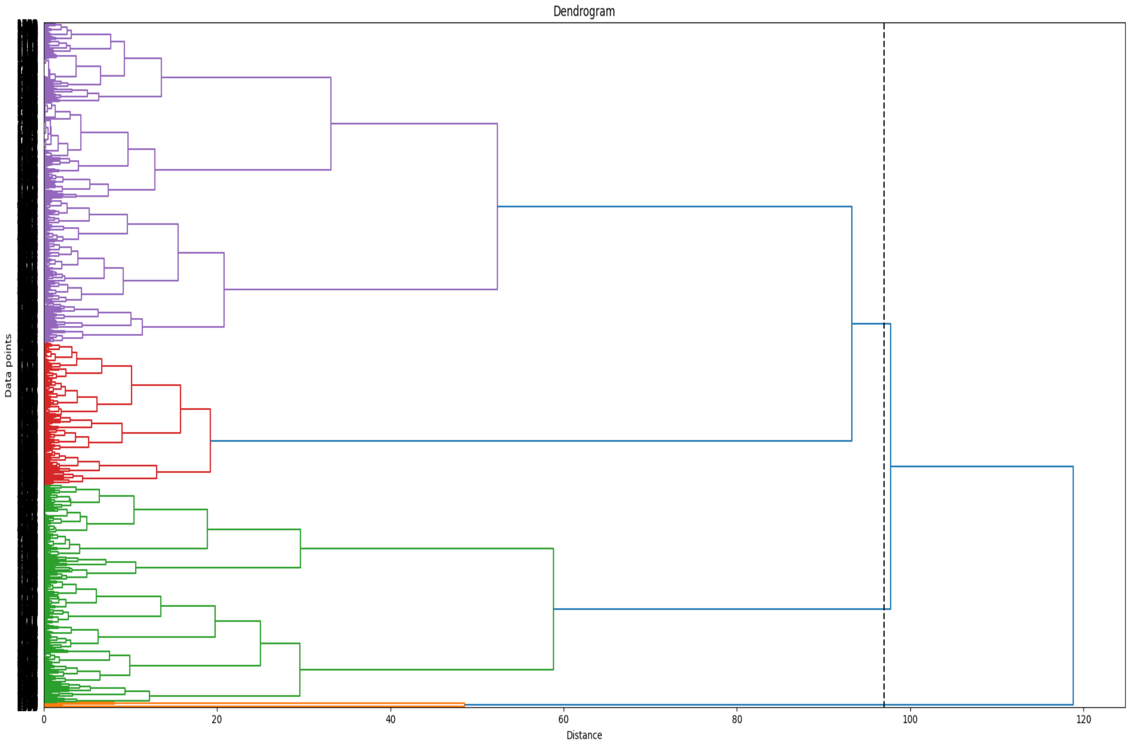


Figure 5.10: Dendrogram of the fire events data

Looking at the figure 5.10, it is possible to see that the number of clusters is

[1] https://public.tableau.com/app/profile/tiago.concei.o3198/viz/FireLoc/Painel_Incendio#1

high at the root level. If it was to cut distance around the value 97 it would catch all levels of linkage, generalizing the data not meeting the objective. It is possible to retrieve more information at the root level, however, if it was decided to cut the distance that best fits the number of clusters defined by the dendrogram, a lot of information would be lost.

Because the number of clusters could not be determined immediately using the dendrogram, due to high density on the root level, one of the benefits mentioned previously of HDBSCAN, an algorithm given by the library sklearn.clusters, is that it can determine the best number of clusters based on the density of the data. The number of clusters that the method determined to be most appropriate is 334 clusters. Each of these clusters is made up of at least 5 events, if a cluster does not obtain that required number of entries, the clusters are eventually discarded as well as the events. In case any event cannot be grouped into any cluster it is considered a noisy point.

The data was therefore grouped by cluster, based on location and date, as can be seen in the table below.

Table 5.9: Example of Cluster 166 content

Datetime	Content
2017-08-10	novo incêndio em viseu tondela santiago de besteiros
2017-08-10	novo incêndio em aveiro oliveira de azemeis vila de cucujaes
2017-08-10	1420 abrantes incêndio ativo com 2 frentes combatidas por 505 operacionais 162 meios terrestres 7 meios aéreos foto ruben mini
2017-08-12	incêndio com duas frentes lavra com intensidade em ferreira do zêzere
2017-08-12	incêndio em miranda do corvo uma visão dantesca

As can be seen from the table 5.9, you can draw the conclusion that the data was grouped based on the date and not on its location, because according to the text, you can see that it covers different areas of Portugal.

Once all of the clusters were formed, like in the example of the table 5.9, the portion of viewing the data was constructed, and in order to gain a better picture of how the data is categorized, the boundaries of Portugal were obtained through a shape file, allowing a geographical view. This visualization also includes text that supports the points on the map. Given that this visualization has many values to show due to the high number of clusters, an animation had to be created to display the data slowly allowing better interpretation.

In the specific situation of figure 5.11, it is clear that there is a large number of events that occurred in that period of time to be analyzed, more precisely 15 located in the center zone of the map. This is due to the fact that many events occurred in that location from 2017-08-10 to 2017-08-18, as can be seen in the text box.

In order to support the figure 5.11 that groups the events and to be able to analyze the sentiment, another visualization was created. This one is exposed in figure 5.12, regarding the same set of events. As can be seen from the figure, all 15 contributions are considered of negative sentiment, starting on day 2017-08-10 with an intensity of 0.75, having a slight rise, and then, on day 2017-08-14, a large

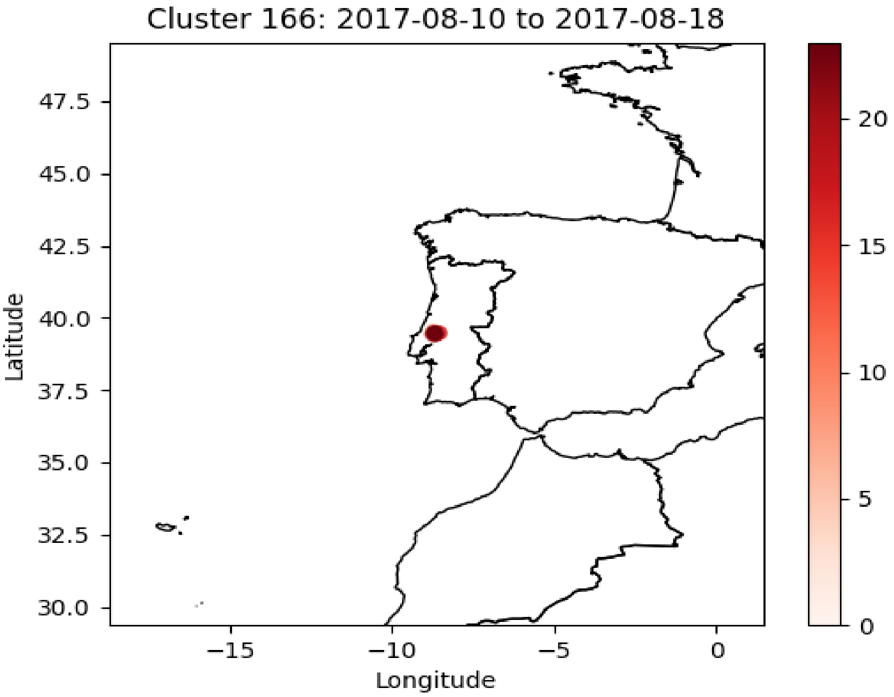


Figure 5.11: Cluster Example between 2017-08-10 to 2017-08-18

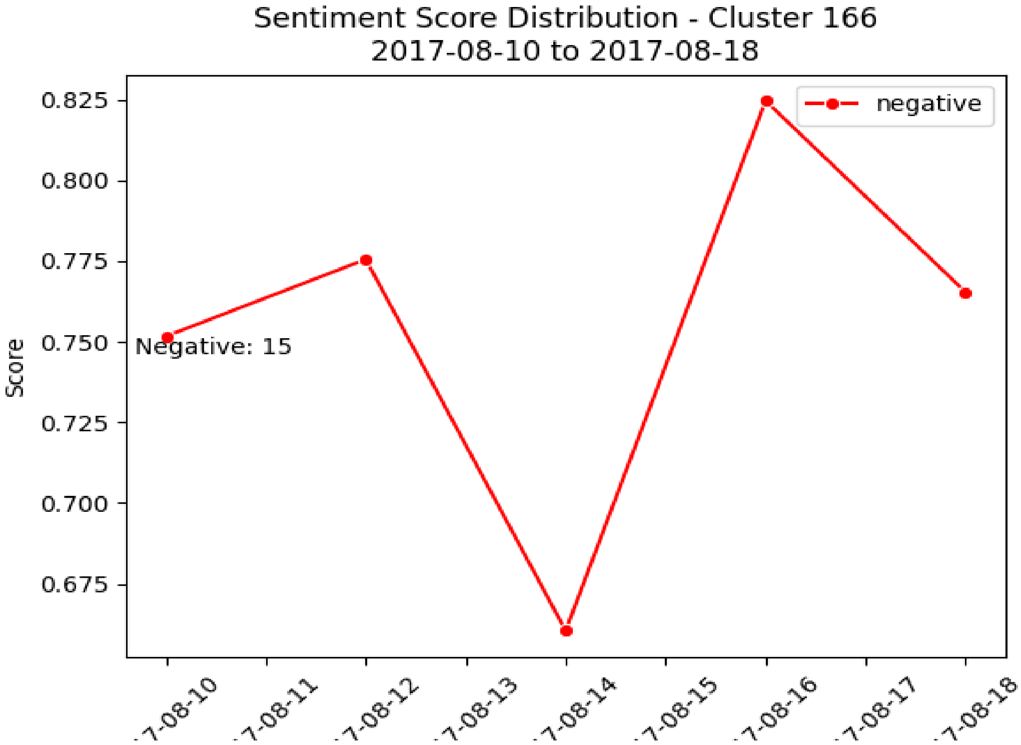


Figure 5.12: Sentiment change from clustered event

drop with an intensity below the value of 0.675. This value indicates that on that day the reported event could indicate a fire that was not as devastating as the ones that are included in the same location and in the same period of time, or it could mention the same event indicating that the fire has lost strength. However, the following days also show an increase in intensity. A particularity of this visualization is that, for each day, if there is more than one contribution of an event they are grouped and averaged by the intensity of the sentiment, allowing to simplify the visualization and the understanding of the data.

5.6 Final Remarks

In summary, and taking into account all the results obtained in the previous sections, some conclusions can be reached, in the part of data extraction, namely in the selected data source. Twitter is a social network with a lot of potential since it has thousands of daily contributions, and in the case of a catastrophic event, there will certainly be people referring to it. However according to the results obtained, most of the contributions come directly from official sources, such as newspapers, and only a small part corresponds effectively to citizens, making it difficult to collect extra information.

Regarding the pre-processing phase in data exploration, it is noted that the months with the highest number of contributions occur in June and October of 2017, which may indicate that fire-related events, contrary to expectations, occur in months when the temperature is not supposed to be so high. Regarding the days, there is a higher number on the days between 12 and 19, and regarding the hours there is an exponential growth from 9 a.m onwards, which may indicate that from that time onwards the heat begins to be felt. In this same phase, the classification is made to see if the contribution is really related to the catastrophic fire event, and in this way, three types of classification were taken into account. The first one is made through fine-tuning using labeled data in which it obtained quite interesting results in which the average accuracy is 96%, average precision is 81%, recall with 83%, and an F1 score of 82%. In addition, two recent classification methods were tested, namely Few-Shot Learning and Zero-Shot Learning, which, although they did not have very favorable results, demonstrate potential since they require very little or no data to be trained. The last step taken at this stage was the location assignment. The tool used for the purpose was Nominatim, which despite assigning the correct coordinates many times there were incorrect assignments that were later revealed through the visualizations, creating discrepancies between the report location with the coordinates.

As for the sentiment analysis phase, as is expected, most of the contributions are negative, since they report contributions of the occurrence of a fire event. It has also been seen that the presence of contributions connoted as neutral are sporadic, and those of positive sentiment are rare. At this stage, the attribution of sentiment comes with a score that is an indicator of confidence or strength that comes through the words associated with each sentiment. In the reported contribution, if it is close to 1, the more certain it is to be associated with that sentiment. Some conclusions are drawn as the continuity over the period of time analyzed. The negative sentiment has a score practically always higher than 0.7, demonstrating that it is very likely that it is effectively a negative sentiment. Regarding the neutral sentiment, it is seen that it is between 0.5 and 0.6 and the confidence is lower than the negative, leading to the idea that those that are closer to 0.5 may be misclassified. Finally regarding the positive sentiment, the score is all spread but the vast majority have a confidence level below 0.5, which may mean that they may also be wrong classified.

In the second-to-last phase, which is the visualization, the tool chosen was Tableau for its ability to create dynamic and interactive dashboards. In this way a visualization was created that was composed of a map and additional graphics, thus allowing any user to obtain the maximum information about the events that occurred. However, as mentioned earlier the failure of the Nominatim API creates irregularities in this type of visualizations.

Regarding the last phase, which refers to the grouping of events, hierarchical clustering algorithms were used so that in this way the events were grouped by their location (latitude, longitude) and the date. This allows to understand the variation of sentiment over time regarding the captured events, which can give information regarding the development of the event or events that occurred according to the contributions.

Finally, and from a general perspective, all phases were fundamental for the integral success of this thesis since they allow to extract of useful information, being an added value both for the FireLoc project and for the communication of these data to the competent authorities.

Chapter 6

Conclusion and Future Work

Effective reporting mechanisms are essential because they facilitate the response of the appropriate authorities. Crowdsourcing is one way to gather reports, but in order to develop a workable information platform for communication between the citizens and responsible authorities, it is required to have an efficient system in place. Fortunately, such a system already exists and is known as FireLoc, which works exclusively for forest fires. It takes a lot of data to increase the effectiveness of this kind of technology in real-world settings. However, it is essential to have a system that verifies the text of each report to avoid misinformation and allow the collection of more information.

In the context of this thesis, a pipeline of tasks was developed in order to be able to collect the largest number of data from a text, to be integrated into the FireLoc project, a tool that analyzes the contributions made by a user regarding a fire event. The results of the developed work, the challenges, and the upcoming work are all examined in this last chapter.

The conclusion that can be drawn is that Twitter is a very useful social network for providing data, it can have some problems such as the veracity of that data, or even if the data is related to what is intended. In addition, through exploratory visualizations, it is possible to conclude that the month with the most contributions in 2017 was October, indicating that heat and the risk of fire prevail in months that are no longer considered risky. The days with the highest number of reports were the 15th and 19th. With regard to the time of day, it can also be concluded that from seven in the morning, there was an exponential growth in contributions, which may indicate that it is from this time that it begins to be felt and that it coincides with the waking hours of many citizens, which reflects an increase in contributions. Another conclusion was based on tests performed to find the best methodology to understand whether or not the event is related to fire, three different methods were tested and two major conclusions were reached. If there is an annotated dataset to serve as training, the best alternative will be a fine-tuned classifier based on the BERT model. However, if there is no annotated data according to performance, the best thing is to resort to Few-Shot Learning in which some examples must be given to classify what is intended. However, another conclusion that could be drawn was the fact that when you wanted to assign a location based on the text, Nominatim has some attribution errors which became misleading in some of the contributions collected, and this was also reflected in the visualizations created. With regard to the sentiment analysis, the conclusion is that, as expected, the contributions are mostly negative, since the capture of contributions with the keyword "fogo" are rarely connoted as positive. Furthermore, the use of Tableau has proved to be very advantageous, since it is possible to create complete visualizations with a higher level of complexity, being able to represent the data studied and analyzed in the best possible way. Finally, the last conclusion that can be drawn is that HDBSCAN was used as a clustering algorithm, which was very positive since it was able to effectively group the data, making it possible to see the variation in sentiment over the time the cluster was grouped.

However, there were some difficulties along the way. For example, after the data was extracted, a large portion of the contributions belonged to official organizations like newspapers and websites that track these events, and very few of these reports actually matched with citizens. Even in the data collection phase, when looking for the keyword "fogo," very little information actually reflected a report because the word has two different meanings. One of the biggest difficulties encountered was the fact that the allocation of location through Nominatim was sometimes not the most accurate, noting discrepancies in the texts according to the coordinates. These location flaws were reflected in the visualizations produced by the tableau and the views produced by the clustering method.

With that said, the contributions of the work developed with the proposed objectives were the following:

- Documentation of the state-of-art regarding NLP, which can be used for the development of a better system for the analysis of citizen's contributions, presented in Chapter 3;
- Implementation of various resources such as data retrieval, event grouping, event classification, sentiment analysis, and event extraction, among others that allow an analysis of user contributions;
- The proposal for a visualization tool that facilitates the understanding of data in a pragmatic and more intuitive way.

Regarding future work, the system has room for improvement, starting with the quality of the data to be collected. This step can be achieved by a different choice of data source or a greater restriction for capturing this data.Another aspect to be improved is the choice of a better API for collecting the location since the choice used, despite being completely free, does not provide the best accuracy of the event. Another aspect that would make sense to apply in the future, would be that through the contributions made by citizens, and with support in the capabilities of NER, manage to capture in the text some facilities such as houses, buildings, or even gas stations, which represent some danger. As the captured facility applies a danger scale, which would allow transmission to the responsible entities the events that demonstrate the most danger as a priority.

References

Akpan, U. I. and Starkey, A. (2021). Review of classification algorithms with changing inter-class distances. *Machine Learning with Applications*, 4:100031.

Ali, S. M., Gupta, N., Nayak, G. K., and Lenka, R. K. (2016). Big data visualization: Tools and challenges. In *2016 2nd International Conference on Contemporary Computing and Informatics (IC3I)*, pages 656–660.

Carvalho, C. M. A., Nagano, H., and Barros, A. K. (2017). A comparative study for sentiment analysis on election Brazilian news. In *Proceedings of the 11th Brazilian Symposium in Information and Human Language Technology*, pages 103–111, Uberlândia, Brazil. Sociedade Brasileira de Computação.

Casola, S., Lauriola, I., and Lavelli, A. (2022). Pre-trained transformers: an empirical comparison. *Machine Learning with Applications*, 9:100334.

Castanha, J., Indrawati, Pillai, S. K., Ramantoko, G., and Widarmanti, T. (2022). A systematic literature review on natural language processing (nlp). In *2022 International Conference on Advanced Creative Networks and Intelligent Systems (ICACNIS)*, pages 1–6.

Chiorrini, A., Diamantini, C., Mircoli, A., Potena, D., and Storti, E. (2022). Emotionalberto: Emotion recognition of italian social media texts through bert. In *2022 26th International Conference on Pattern Recognition (ICPR)*, pages 1706–1711.

Chong, W. J., Chua, H. N., and Gan, M. F. (2022). Comparing zero-shot text classification and rule-based matching in identifying cyberbullying behaviors on social media. In *2022 IEEE International Conference on Artificial Intelligence in Engineering and Technology (IICAIET)*, pages 1–5.

Dharmarajan, A. and Velmurugan, T. (2013). Applications of partition based clustering algorithms: A survey. In *2013 IEEE International Conference on Computational Intelligence and Computing Research*, pages 1–5.

Ganai, A. F. and Khursheed, F. (2019). Predicting next word using rnn and lstm cells: Stastical language modeling. In *2019 Fifth International Conference on Image Information Processing (ICIIP)*, pages 469–474.

Gonçalo Oliveira, H., Marques, J., and Cortesão, L. (2015). Exploiting twitter for the semantic enrichment of telecommunication alarms.

Hai, M., Zhang, S., Zhu, L., and Wang, Y. (2012). A survey of distributed clustering algorithms. In *2012 International Conference on Industrial Control and Electronics Engineering*, pages 1142–1145.

Hu, H., Liao, M., Zhang, C., and Jing, Y. (2020). Text classification based recurrent neural network. In *2020 IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC)*, pages 652–655.

Jurafsky, D. and Martin, J. H. (2009). *Speech and language processing*. Pearson Prentice Hall.

Kaur, H., Mangat, V., and Nidhi (2017). A survey of sentiment analysis techniques. In *2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, pages 921–925.

Kumawat, S., Yadav, I., Pahal, N., and Goel, D. (2021). Sentiment analysis using language models: A study. In *2021 11th International Conference on Cloud Computing, Data Science Engineering (Confluence)*, pages 984–988.

Kutbay, U. (2018). Partitional clustering. In Pirim, H., editor, *Recent Applications in Data Clustering*, chapter 2. IntechOpen, Rijeka.

Lin, J. J., Snow, R., and Morgan, W. (2011). Smoothing techniques for adaptive online language models: topic tracking in tweet streams. In *Knowledge Discovery and Data Mining*.

Ling, X. and Weld, D. S. (2010). Temporal information extraction. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence*, AAAI'10, page 1385–1390. AAAI Press.

Mohammadi, S. and Chapon, M. (2020). Investigating the performance of fine-tuned text classification models based-on bert. In *2020 IEEE 22nd International Conference on High Performance Computing and Communications; IEEE 18th International Conference on Smart City; IEEE 6th International Conference on Data Science and Systems (HPCC/SmartCity/DSS)*, pages 1252–1257.

Muthukumar, N. (2021). Few-shot learning text classification in federated environments. In *2021 Smart Technologies, Communication and Robotics (STCR)*, pages 1–3.

Patadia, D., Kejriwal, S., Mehta, P., and Joshi, A. R. (2021). Zero-shot approach for news and scholarly article classification. In *2021 International Conference on Advances in Computing, Communication, and Control (ICAC3)*, pages 1–5.

Pereira, D. A. (2021). A survey of sentiment analysis in the portuguese language. *Artif. Intell. Rev.*, 54(2):1087–1115.

Pipalia, K., Bhadja, R., and Shukla, M. (2020). Comparative analysis of different transformer based architectures used in sentiment analysis. In *2020 9th International Conference System Modeling and Advancement in Research Trends (SMART)*, pages 411–415.

Pota, M., Ventura, M., Catelli, R., and Esposito, M. (2020). An effective bert-based pipeline for twitter sentiment analysis: A case study in italian. *Sensors*, 21:133.

Ramprasath, M., Dhanasekaran, K., Karthick, T., Velumani, R., and Sudhakaran, P. (2022). An extensive study on pretrained models for natural language processing based on transformers. In *2022 International Conference on Electronics and Renewable Systems (ICEARS)*, pages 382–389.

Rani, Y. and Rohil, H. (2013). A study of hierarchical clustering algorithm. *International Journal of Information and Computation Technology*, 3:1115–1122.

Ritter, A., Mausam, Etzioni, O., and Clark, S. (2012). Open domain event extraction from twitter. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '12, page 1104–1112, New York, NY, USA. Association for Computing Machinery.

Sakaki, T., Okazaki, M., and Matsuo, Y. (2010). Earthquake shakes twitter users: Real-time event detection by social sensors. pages 851–860.

Shakeel, H. M., Iram, S., Al-Aqrabi, H., Alsboui, T., and Hill, R. (2022). A comprehensive state-of-the-art survey on data visualization tools: Research developments, challenges and future domain specific visualization framework. *IEEE Access*, 10:96581–96601.

Shivahare, B. D., Singh, A. K., Uppal, N., Rizwan, A., Vaathsav, V. S., and Suman, S. (2022). Survey paper: Study of natural language processing and its recent applications. In *2022 2nd International Conference on Innovative Sustainable Computational Technologies (CISCT)*, pages 1–5.

ShrihariR, C. and Desai, A. (2015). A review on knowledge discovery using text classification techniques in text mining. *International Journal of Computer Applications*, 111:12–15.

Souza, M. and Vieira, R. (2012). Sentiment analysis on twitter data for portuguese language. pages 241–247.

Srivastava, S., Nagpal, A., and Bagwari, A. (2020). Various approaches in sentiment analysis. In *2020 12th International Conference on Computational Intelligence and Communication Networks (CICN)*, pages 92–96.

Tang, Z., Cui, Y., Xing, J., Huang, J., Jin, J., Tang, Y., and Wu, F. (2022). Comparison of k-means and hierarchical clustering used in power consumption analysis. In *2022 IEEE 6th Conference on Energy Internet and Energy System Integration (EI2)*, pages 1696–1700.

Tavares, C., Ribeiro, R., and Batista, F. (2021). Repositório do iscte – instituto universitário de lisboa.

Truyens, M. and Van Eecke, P. (2014). Legal aspects of text mining. *Computer Law Security Review*, 30(2):153–170.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. (2017). Attention is all you need. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Xiang, W. and Wang, B. (2019). A survey of event extraction from text. *IEEE Access*, 7:173111–173137.

Xu, R. and Wunsch, D. (2005). Survey of clustering algorithms. *Neural Networks, IEEE Transactions on*, 16:645 – 678.

Zahoor, S. and Rohilla, R. (2020). Twitter sentiment analysis using machine learning algorithms: A case study. In *2020 International Conference on Advances in Computing, Communication Materials (ICACCM)*, pages 194–199.

Zhang, T., Xu, B., Thung, F., Haryono, S. A., Lo, D., and Jiang, L. (2020). Sentiment analysis for software engineering: How far can pre-trained transformer models go? In *2020 IEEE International Conference on Software Maintenance and Evolution (ICSME)*, pages 70–80.