UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE CIÊNCIA DA COMPUTAÇÃO

RAFAEL PINHEIRO AMANTEA

A comparison of machine learning approaches for predicting ClaimReview markup attributes from fact-checking websites

Work presented in partial fulfillment
of the requirements for the degree of
Bachelor in Computer Science

Advisor: Prof. Dr. Dante Barone
Coadvisor: Prof. MSc. Eduardo Gabriel Cortes

Porto Alegre
June 2021

ACKNOWLEDGMENT

Thanks to my parents, Marcia and Jarbas; brothers, Artur and Rodrigo; girlfriend, Raíssa; and to all of my friends for all the support provided until this moment. In the most difficult times, you were there to encourage me for not giving up. Without you, none of it would be possible.

Thanks to my advisor and professor Dante Barone; co-advisor Eduardo Gabriel; and my research colleagues Vinicius Woloszyn and Vera Schimdt, who have made this work possible.

ABSTRACT

The spreading of fake news is a reality within modern times. However, in the daily fight against disinformation, the fact-checking agencies are one of the strongest allies. Some techniques have been in place to help in this battle, and one of them is the ClaimReview web markup, which had been introduced to grant access to fact-checking articles meaning by search engines. Despite its importance within this context, barely half of the fact-checkers have adopted it. Therefore, in this work, we provide a starting point for the automatic generation of ClaimReview markup, investigating means to predict ClaimReview's attributes using machine learning models. By experimenting and comparing the baseline approach, Support Vector Machine, with the state-of-the-art (BERT) we have achieved noticeable results, creating a benchmark for upcoming researches in this domain.

Keywords: Machine learning. Fake news. ClaimReview. SVM. BERT.

PUBLICATIONS

As an output of this work's research, we had a paper accepted for publication in the 13th ACM Web Science Conference 2021[1].

[1] https://websci21.webscience.org/

LIST OF FIGURES

Figure 1.1 Concern on disinformation per distinct channels 11
Figure 1.2 Comparison between 2016 and 2020 of fact-checking websites registered in Duke Reporters' Lab per continent. 13

Figure 2.1 Structured representation of a *ClaimReview* markup 17

Figure 3.1 Dataset preprocessing workflow 23
Figure 3.2 Distribution of claims by HTML tag 25
Figure 3.3 Class distribution of *reviewRating* 25
Figure 3.4 Sample of dataset for *claimReviewed* task 25
Figure 3.5 Cross-validation example 28

Figure 4.1 General classification hyperplane representation of SVM algorithm. 30
Figure 4.2 Support Vector Machine 30
Figure 4.3 BERT 33

LIST OF TABLES

Table 3.1 Original Distribution of idioms 22
Table 3.2 Class distribution of *claimReviewed* by HTML Tags 24

Table 5.1 Results for *claimReviewed* Prediction 36
Table 5.2 Results for *reviewRating* Prediction 37

LIST OF ABBREVIATIONS AND ACRONYMS

SVM	Support Vector Machine
BERT	Bidirectional Encoder Representations from Transformers
IFCN	International Fact-Checking Network

CONTENTS

1 INTRODUCTION **10**
1.1 The Fact-Checkers **11**
1.2 This Work **13**
2 BACKGROUND AND RELATED WORK **15**
2.1 Manual Fact-Checking **15**
2.2 ClaimReview Markup **16**
2.3 Datasets **16**
2.4 Automatic Fact-Checking **18**
3 METHODOLOGY **20**
3.1 Automatic Generation of ClaimReview as a Classification Problem **20**
3.2 Training Datasets **21**
3.2.1 Pre-processing 22
3.2.2 *claimReviewed* Dataset 23
3.2.3 *reviewRating* Dataset 24
3.3 Models **24**
3.3.1 Additional variations 26
3.4 Validation **27**
3.5 Metrics **27**
4 IMPLEMENTATION AND EXPERIMENTS **29**
4.1 SVM **29**
4.1.1 Parametrization 32
4.2 BERT **32**
4.2.1 Parametrization 34
5 RESULTS **35**
5.1 *claimReviewed* Classification **35**
5.2 *reviewRating* Classification **36**
6 CONCLUSION **38**
REFERENCES **40**

1 INTRODUCTION

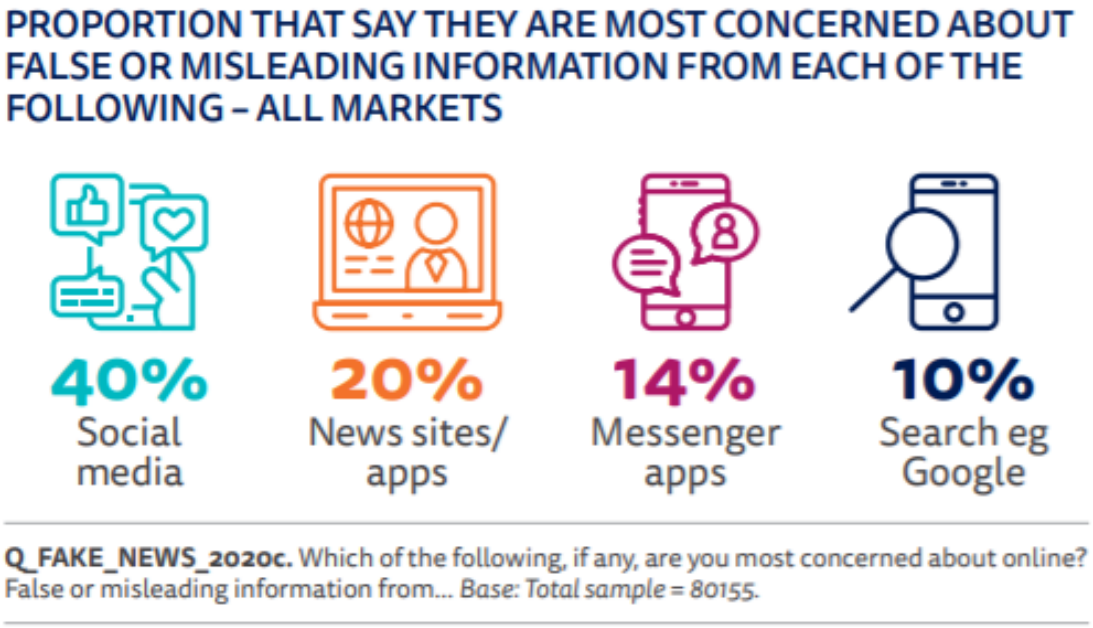
The spreading of fake news throughout the Web has become a critical problem for a democratic society. This practice, which was initially very common among politicians, has also become very popular among content producers within the media, and individuals or organizations interested in making more money by publishing or disseminating disinformation. According to a recent report (BRAUN; EKLUND, 2019), there are substantial amounts of money involved in the context of fake news creation and propagation. For example, a known tactic used by media vehicles is to publish sensationalistic headlines with distorted or partial content, in order to get more accesses and, therefore, profit with advertising. A research (VOSOUGHI; ROY; ARAL, 2018) explains that fake news difuses much more farther, faster, deeper and more broadly when compared to the true. Therefore, this could explain why we are getting more used with it. In the other hand, there is a huge force of journalists, organizations and fact-checkers fighting against this practice worldwide. They have been working unceasingly to identify misinformation and prevent it from further spreading by openly publishing articles exposing and denouncing the falseness.

According to the Oxford dictionary, the *fake news* term is defined as: "*news that conveys or incorporate false, fabricated, or deliberately misleading information, or that is characterized as or accused of doing so*". This entry was added to the dictionary back in 2016, in October's update[1]. It is not a coincidence the fact that in the same year happened the US presidential election campaign, and therefore brought the *fake news* term to the spotlight. More recently, in the year of 2020, due to the beginning of COVID-19 pandemic, fake news became a problem which was directly dealing with human lives. A study (REIHANI et al., 2020) depicted the consequences of not approved treatments, and mentioned cases where the media itself was covering alternative ways to treat COVID-19. However, in spite of this recent popularization of fake news, there are several records of fake news across the history. For example, back in 1475 BC, in Trent, Italy, after a child gone missing a Franciscan preacher claimed that the Jewish community had murdered the child, drained his blood, and drunk it to celebrate Passover.

Fortunately, in the past, misinformation did not spread with the speed it does nowadays. There was no internet, mobile phones, or even printed news papers (depending on the period in time being analyzed). The globalization and digital transformation

[1] https://public.oed.com/updates/new-words-list-october-2019/

Figure 1.1: Concern on disinformation per distinct channels



Source: Reuters Institute Digital News Report 2020

context that we live is the perfect scenario for disseminating fake news. According to the *Reuters Institute Digital News Report 2020*[2], people see social media as the biggest source of concern about misinformation (Figure 1.1), as it is very easy to spread and sometimes hard to lineage the source. No matter what is the context that they are applied (political, social, corporate...), it has become a weapon for mass disinformation.

1.1 The Fact-Checkers

The fact-checking process is the act of validating stories, quotes, posts or news through public records, researches or data. It is a way of verifying the degree of truth in these news. It can be conducted before or after the news is spread. When this task is done before publishing, it works as a way of validating the accuracy of the content, and, in case of misinformation, it can be fixed before spreading. This is a mechanism implemented by several media organizations to avoid publishing fake news. In the other hand, when this task is done after the news has already been published, it is usually executed by third-parties, such as specific media organizations, non-governamental-organizations, etc.

Motivated by the huge amount of fake news being spread through social medias, newspapers and other channels, some organizations are aiming to verify which of these

[2]https://www.digitalnewsreport.org/

viral news are really truth. These are the so called *fact-checkers*. They can be individual non-profitable organizations or agencies, such as *Politifact*[3], from the *Poynter Institute*[4], or *Snopes*[5]; and other times they can be a division belonging to a media organization, such as the brazilian agency *Lupa*[6], from the *Folha de São Paulo*[7] newspaper.

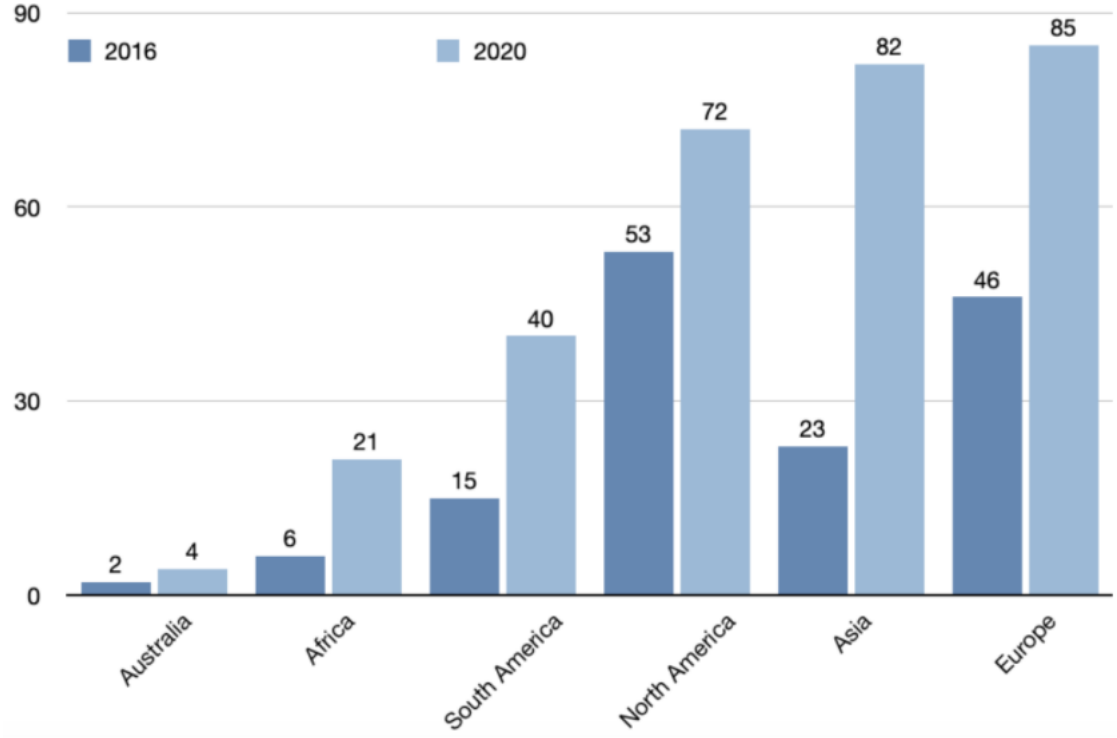
In 2015, a unit of the Poynter Institute was created to bring together fact-checkers worldwide: The International Fact-Checking Network (IFCN)[8]. The goal of this network is to support the increasing fact-checking initiatives by promoting best practices and exchanges in this field. Therefore, for maintaining transparency of the manual fact-checking task and increase trustworthiness in independent fact-checking organizations, the IFCN has developed a code of principles for the fact-checking task, where all signatories have to stick to.

Additionally to this project, Duke University's *Reporters' Lab*[9] is leading a very important initiative for finding and monitoring fact-checkers at work in 84 different countries. The amount of fact-checking websites registered in the database was 304 in October, 2020. Based on their annual census, latest conducted in 2020, 28% of the entities belong to Europe, 27% belong to Asia, 24% belong to North America, 13% belong to South America, 7% belong to Africa, and only 1% belong to Australia, according to the chart presented in Figure 1.2.

As a way of granting access to the meaning of the fact-checking articles by search engines, social medias, and other platforms (such as Ecosia, DuckDuckGo, Google, Facebook...), in 2015 a new web markup called *ClaimReview* was introduced by Google and the Duke Reporters' Lab. This was a good step forward for the fact-checking community and the academic society, since extracting information from fact-checking websites has always been a challenging task, as each one of them would have a different structure, and they are constantly changing their layouts. However, although there are over 300 fact-checkers worldwide, barely half of them have used the *ClaimReview* markup, resulting in low findability of fact-checking articles that debunk fake stories, especially in under-represented countries and languages.

[3]https://www.politifact.com/
[4]https://www.poynter.org/
[5]https://www.snopes.com/
[6]https://piaui.folha.uol.com.br/lupa/
[7]https://www.folha.uol.com.br/
[8]https://www.poynter.org/ifcn/
[9]https://reporterslab.org/

Figure 1.2: Comparison between 2016 and 2020 of fact-checking websites registered in Duke Reporters' Lab per continent.



Source: Duke Reporters' Lab

1.2 This Work

In this work, we are proposing the following research question: *Is it possible to predict the ClaimReview markup?* An automatic solution for parsing fact-checking articles is an important step towards the creation of a large and updated knowledge base of checked facts and for enabling a better interpretation of fact-check articles by search engines. This solution would bring several benefits for journalists, data scientists, and citizens that want to be truly informed about the veracity of publications. We are investigating the viability of a multi-language parser as a solution to make a semantic interpretation of articles from fact-checkers that do not have the *ClaimReview* markup. Our empirical observation shows that the required ClaimReview's attributes by search engines, such as *claimReviewed*, *reviewRating* and *URL* are always present in the chunks of the article, since it is a requirement of the code of principles of the International Fact-Checking Network. Therefore, we hypothesize that the problem of automatic creation of a ClaimReview markup can be reduced to the problem of chunk classification, where the goal is to label correctly the textual chunks that contain the needed information for populating a ClaimReview markup. In our experiments, we test the usage of state-of-art and baseline

text classifiers for predicting *claimReviewed*, *reviewRating* and *URL*.

This work is structured in the following way: in Chapter 2 we present the background of the fact-checking task and related works for the claim extraction using machine learning techniques; in Chapter 3, we introduce the methodology used in our study; in Chapter 4, we present implementation details of SVM and BERT; in Chapter 5 we present and compare the results between the baseline and state-of-the-art for our experiments; in Chapter 6 we conclude the work and discuss future works in the area.

2 BACKGROUND AND RELATED WORK

In this chapter we present the background and research areas associated to this work: the manual investigation of claims through fact-checking agencies; the usage of *ClaimReview* markup to make fact-checking articles machine-readable; the creation of fact-checking datasets to be used with machine learning models; and automatic methods for detection of claims and fake news.

2.1 Manual Fact-Checking

Initially, fact-checking emerged in journalism intending to assess news authenticity by comparing the knowledge extracted from news content, such as claims or statements, with known facts (ZHOU; ZAFARANI, 2020). Traditional fact-checking is the task of manually detecting false information. One of the first studies published in this field was (VLACHOS; RIEDEL, 2014), in which they have defined the fact-checking process as the assignment of a truth value to a claim made in a particular context. By that time, they already recognized that the manual assessment of the truthfulness of a claim was a time-consuming task, and a system that would automate this process could be very helpful. This was their work's main motivation, and according to the study, the automation of the fact-checking process would reduce the time to check the verdict of a claim regardless the persona involved. For example, it could be a journalist which would need to consult several different sources; or could be an ordinary citizen that would need to check the information provided to them.

Hereby, manual fact-checking can be divided into expert-based fact-checking, through agencies or non-profitable organizations, and crowd-sourced fact-checking, through the help of the global community. However, our research only relies on expert fact-checking. Recently, many expert-based fact-checking organizations have emerged to cope with the increasing spread of false information throughout the web, as already mentioned in this study. However, as stated before, manual fact-checking is a time-consuming task done by journalists collecting evidence to check the veracity of claims (WANG; QU, 2017), and the manual verification of news items does not scale with the volume of newly created information (JIANG et al., 2020). Thus, automatic solutions have been developed to tackle the challenge of detecting false information, such as (ALOSHBAN, 2020), (PÉREZ-ROSAS et al., 2017), (VO; LEE, 2020) and (WOLOSZYN; NEJDL, 2018). In

order to improve the automatic detection of false information, the representation of news items as structured information, which algorithms can understand, becomes vital for the task. Unfortunately, in the the availability of fact-checking articles attributes, such as claims or verdicts, as structured information is still limited.

2.2 ClaimReview Markup

Only recently, with the global effort on computational journalism (CASWELL; DÖRR, 2018), structured information has been made available through a schema markup named ClaimReview. It is a global standard that was created by Google and Duke University (Duke Reporter's Lab) through an open process which also involved the global fact-checking community, Bing and Jigsaw. According to the *ClaimReviewProject*[1], the ClaimReview markup is a tagging system embedded to fact-checking articles' HTML that publishers use to flag these articles for search engines, apps and social media platforms. The idea behind it is to identify their key elements through annotations of structured information on web-pages, for example, the person and claim being checked and a conclusion about its accuracy, and then, platforms such as Google, Bing and Facebook can take benefit of these tags' content.

The ClaimReview provides a collection of shared vocabulary which can be used along with Microdata, RDFa, or JSON-LD formats to incorporate information on websites to make them better interpreted by machines. These schemes promote common data formats that allow creating a semantically structured representation of the knowledge available on the web, i.e., Semantic Web. Although ClaimReview defines several properties, the minimum required attributes to be eligible by search engines are: author, claim, date published, URL, and review rating. An example of some ClaimReview's attributes is depicted in Figure 2.1.

2.3 Datasets

Various interdisciplinary research fields from social and natural sciences face the challenge of ground truth data in the form of labeled and structured fake news items. Some attempts have been made to construct a ground truth dataset which can be used for

[1] https://www.claimreviewproject.com/

Figure 2.1: Structured representation of a *ClaimReview* markup

@type	ClaimReview
url	http://danbri.org/2017/TODO
reviewBody	This claim is true. The UK also played a role.
itemReviewed	
@type	Clip
name	President Obama Speech to Muslim World in Cairo
startOffset	350
isPartOf	
@type	VideoObject
name	Clip from President Obama Speech to Muslim World in Cairo
transcript	http://www.nytimes.com/2009/06/04/us/politics/04obama.text.html
url	https://www.youtube.com/watch?v=B_889oBKkNU
endOffset	370
datePublished	2009-06-04
author	
@type	Person
name	Barack Obama
jobTitle	44th President of the United States of America
image	https://upload.wikimedia.org/wikipedia/commons/8/8d/President_Barack_Obama.jpg
sameAs	https://www.wikidata.org/wiki/Q76
sameAs	https://twitter.com/barackobama
datePublished	2017-07-06
claimReviewed	In the middle of the Cold War, the United States played a role in the overthrow of a democratically-elected Iranian government.
author	
@type	Person
sameAs	https://twitter.com/danbri
url	http://danbri.org/

Source: schema.org/ClaimReview

automated fact-checking within a machine learning framework.

In the work by (VLACHOS; RIEDEL, 2014), they introduced a dataset containing statements from two fact-checkers: Channel 4[2] and PolitiFact. They have also noticed some challenges when working with fact-checking websites. These websites would provide heterogeneous and not-binary verdicts, which did not follow a pattern. A claim would not simply be *TRUE* or *FALSE*, but it could be *MOSTLY TRUE* or *HALF FALSE*, for example. This initial dataset had only 221 statements collected from these two websites.

An extension of this work was done in 2016, two years after its publishing, by (FERREIRA; VLACHOS, 2016). In this new study, the author released the Emergent data set, which included 300 labeled rumors collected from PolitiFact by journalists. However, these small datasets could not be leveraged for machine learning purposes. Therefore, (WANG, 2017) introduced a larger data set (LIAR) containing 12.800 human-labeled short statements from PolitiFact to facilitate the development of computational approaches.

2.4 Automatic Fact-Checking

The automatic detection of fake news is a research area which still does not receive enough attention, especially with the extremely favorable context for the dissemination of misinformation today. This area can be subdivided into different tasks, such as claim extraction, for retrieving the statements which should be checked, and review rating detection, for detecting claim's verdict.

Most of previous works within this field have targeted the automation of fake news detection in the context of politics. The first claim detection system was ClaimBuster (HASSAN et al., 2017), which scores sentences using Support Vector Machine (SVM) according to their likelihood of being a politically pertinent statement. However, it does not rate the truth of these statements. Basically, it rates if a sentence is check-worthy on a scale of 0 to 1, where 1 means that it is most worthy.

Another system, ClaimRank (JARADAT et al., 2018), was designed with the purpose of optimizing manual fact-checking efforts by prioritizing the claims that fact-checkers should consider first. It is based on an annotation scheme that is binary, and determines checkable claims rather than check-worthy claims. An interesting aspect of

[2]https://www.channel4.com/

this solution is that it supports both English and Arabic languages, representing an important step for multilingual research within this field. Underneath, ClaimRank's model uses neural networks and receives the claim and its context as inputs. Also, it is based on real claims that popular fact-checking organizations have previously checked (KONSTANTINOVSKIY et al., 2018), and despite being trained with political debates, any kind of text can be applied to the solution.

Additionally, PolitiTax (CARABALLO, 2018) and Full Fact (KONSTANTINOVSKIY et al., 2018) developed their preferred annotation scheme based on their definition of whether or not a sentence is a claim. There is also Logically, which is a model developed by (ADLER; BOSCAINI-GILROY, 2019) that deals with the objective qualities of claims, based on the taxonomies developed by PolitiTax and Full Fact. However, ClaimBuster and Full Fact focuses on live fact-checking of TV debates, where Logically analyzes the bodies of published news' stories.

All the researches mentioned above are investigating deeper the claim extraction task. However, the review rating detection is still a field that has not received much attention. Therefore, in order to foster upcoming studies and researches on automatic conclusion detection from fact-checking websites, in this work we are proposing a starting point for addressing this task, which has several challenges, but has a huge impact for the society.

3 METHODOLOGY

In this chapter, first we introduce the automatic generation of ClaimReview as a classification problem, presenting the two tasks involved in this work. Then, we go through the training dataset used in our experiments and the pre-processing tasks. Later, we conceptually introduce the SVM and BERT models. After, we present the metrics used to evaluate our models' performance. Last but not least, the cross-validation method is explained.

3.1 Automatic Generation of ClaimReview as a Classification Problem

Based on empirical observations made by our research team members, it was noticed that some of the required properties of the ClaimReview markup, such as the *claimReviewed, reviewRating* and URL are available within HTML elements. This observation was the starting point for defining this study, and made us model the *ClaimReview* markup prediction as a classification problem. Within this same context, we have raised an hypothesis that the content of fact-checking pages, such as the nature of its texts, and the structure of the page, like the HTML tags in which the sentences belong to, could be useful in the process of training the classification models.

As mentioned before, the ClaimReview markup is composed of many different properties, and several of them are optional. In this work, we are addressing three mandatory attributes: *claimReviewed, reviewRating* and *URL*. Since the publication's URL is an attribute that can easily be extracted, as it is available by the crawler or by the dataset beforehand, the main focus is over *claimReviwed* and *reviewRatting* attributes. Below we provide a brief description of these properties[1]:

- ***claimReviewed:*** is a short summary of the claim being evaluated.
- ***reviewRating:*** is the assessment of the claim, which supports both numeric and a textual value.

Therefore, for handling the ClaimReview markup classification, we had to divide it into two different tasks: the *claimReviewed* prediction, and the *reviewRating* prediction. The reason behind it is that the *claimReviewed* classification is a binary classification problem, where the positive class represents our target: *claimReviewed*. Basically, a

[1] https://schema.org/ClaimReview

particular sentence can only be a claim (True), or not (False). However, the *reviewRating* classification is more challenging, as it is a multi-label problem, since different fact-checking websites have distinct rating schemes that can have subtle differences, particularly for intermediate values. For example, Snopes have more than 10 rating labels[2], while Lupa has 9[3]. Therefore, (TCHECHMEDJIEV et al., 2019) and (WOLOSZYN et al., 2020) have applied a ratting normalization to enable a comparison among different fact-checker's verdict. In this work, we applied the same rating normalization and employed a multi-class classifier to predict the *reviewRating* of each fact-checking article into the following classes:

- **TRUE**: statements completely accurate.
- **FALSE**: statements completely false.
- **MIXED**: statements partially accurate with some elements of falsity.
- **OTHER**: special articles that do not provide a clear verdict or do not match any other categories.

3.2 Training Datasets

In order to handle two distinct classification tasks, we have generated two distinct training datasets. For creating them, we used the Untrue.news (WOLOSZYN et al., 2020) as the starting point. Untrue.news is a search engine for fact-checkers that provides a multilingual collection of fact-check claims from different fact-checking agencies. Basically, the dataset contains the ClaimReview markup attributes for each fact-check article available, including the properties we want to predict: *claimReviewed, reviewRating* and *URL*. An important detail of the Untrue.news dataset is that each row in the dataset is a distinct fact-checking article's claim.

As detailed in Table 3.1, its corpus consists of 34.383 statements, distributed among 14 different fact-checkers and 4 distinct languages. English instances represent the vast majority of the dataset (77%), followed by Portuguese (18%), German (4.8%) and Spanish (0.2%).

[2]https://www.snopes.com/fact-check-ratings/
[3]https://piaui.folha.uol.com.br/lupa/2015/10/15/entenda-nossos-pinguins/

Table 3.1: Original Distribution of idioms

Fact Checker	Language	Instances
Africacheck	English	559
Aosfatos	Portuguese	849
Channel4	English	1.073
Checkyourfact	English	868
E-farsas	Portuguese	3.486
efe	Spanish	179
Factscan	English	124
G1.globo	Portuguese	185
Healthfeedback	English	90
mimikama	German	1.646
Piaui.folha.uol	Portuguese	1.657
Politifact	English	14.660
Snopes	English	5.330
Truthorfiction	English	3.677
Total		**34.383**

3.2.1 Pre-processing

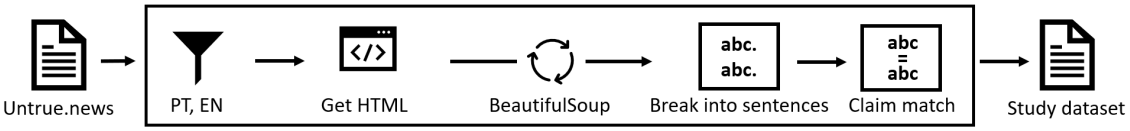
The data available in the Untrue.news dataset is related to claims from distinct fact-checking articles published by different websites. Therefore, despite having information about the claim and its review rating, there was not further details about the article's structure or content, so that we could experiment and evaluate our hypothesis. Due to this reason, we have applied some pre-processing steps over the original dataset to derivate the required information: all sentences within a fact-checking article and the HTML tags in which they are embedded.

In order to narrow down our research, within this study we have only worked with Portuguese and English fact-checkers. Therefore, we started by filtering the Untrue.news dataset by Portuguese and English, since originally it also had data from Spanish and German websites. This approach has reduced the dataset to nearly 32.200 entries across 12 fact-checkers.

One of the most important pre-processing step was the retrieval of the content and structure from the publications. Since we had the *URL* address of each fact-checking article available in the dataset, we used this information to develop a Python crawler that would get the HTML page of each publication. From the HTML, we were able to parse it into plain text, using *BeautifulSoup*[4], and then split the full-text into sentences, also retrieving the HTML tag in which the sentence was embedded. Finally, regarding the

[4]https://www.crummy.com/software/BeautifulSoup/bs4/doc/

Figure 3.1: Dataset preprocessing workflow



annotation process, we used the same process used by Untrue.news to label the HTML chunks that contained the *claimReviewed*. The pre-processing workflow is described in Figure 3.1.

After these pre-processing steps were executed, the granularity of the dataset changed. In the Untrue.news dataset, each record was a distinct claim from a fact-checker's publication, so, a single publication would have only one entry in the set. However, in the new dataset, each record is a distinct sentence from the fact-checker's article, and not necessarily a claim. Therefore, a single fact-checking publication would have several entries in the set (the number of sentences within the article). Due to this reason, the size of the dataset grew from around 32.000 to 4.500.000 entries.

3.2.2 *claimReviewed* Dataset

The dataset used for training the *claimReviewed* classification models was basically the output of the pre-processing steps applied on the Untrue.news dataset, described in previous subsection. A sample of this new dataset is displayed in Figure 3.4, and a description of each column is below:

- *Sentence*: a single sentence, in plain text, from the fact-checking article.
- *Language*: the language of the fact-checker article.
- *claimReviewed*: the class of the sentence. *True* if it is a claim, otherwise, *False*.
- *Fact Checker*: the name of the fact-checking website.
- *HTML Tag*: the HTML tag in which the sentence belongs to.
- *URL*: the URL address of the fact-checking article.
- *reviewRating*: for sentences that are claims (*claimReviewed* = *True*), it indicates the normalized verdict of the claim.

When analyzing this dataset's *claimReviewed* distribution by HTML tags, we could notice that most claims were displaced in the *title* tag, followed by *h1*. However, *h5*

and *h6* did not have any claim. The full analysis is presented in Table 3.2 and Figure 3.2.

Table 3.2: Class distribution of *claimReviewed* by HTML Tags

HTML Tag	# of Sentences	True	False
p	937.905	8.292	929.613
title	67.809	27.038	40.771
li	2.781.709	6.445	2.775.264
h1	34.251	11.421	22.830
h2	160.189	17.042	143.147
h3	259.777	14.751	245.026
h4	185.573	98	185.475
h5	5.617	0	5.617
h6	14.969	0	14.969

3.2.3 *reviewRating* Dataset

In order to train the *reviewRating* classification models, the dataset used in the *claimReviewed* training had to suffer some modifications. Basically, it did not make sense anymore to have sentences which were not claims, as the goal of the task is to predict the *reviewRating* of a claim. Therefore, the granularity of this dataset was changed back to claim, instead of sentences. Also, so that it could be used as a feature in the training, a new column was appended to the dataset, containing the article's list of sentences (publication's content).

- *claimReviewed*: fact-checker's claim.
- *claimReviewed HTML Tag*: the HTML tag in which the sentence belongs to.
- *URL*: the URL address of the fact-checking article.
- *Language*: the language of the fact-checking article.
- *Fact Checker*: the name of the fact-checking website.
- *Sentences*: list of all sentences contained within the respective fact-checking article.

3.3 Models

The methodology of this study was based on the evaluation of two distinct models for both *claimReviewed* and *reviewRating* classification tasks. We have selected Bidirectional Encoder Representations from Transformers (BERT) (DEVLIN et al., 2019) as the

Figure 3.2: Distribution of claims by HTML tag

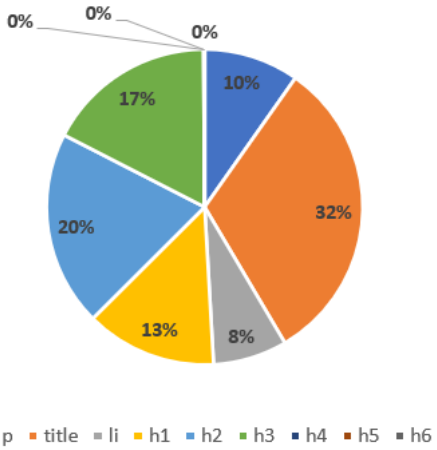


Figure 3.3: Class distribution of *reviewRating*

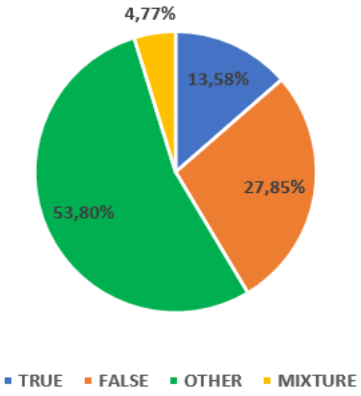


Figure 3.4: Sample of dataset for *claimReviewed* task

	Sentence	Language	claimReviewedClass	Fact Checker	HTML Tag	URL	reviewRatingClass
0	David Cicilline says U.S. has third-lowest minimum wage among developed countries	en	True	politifact	h2	http://www.politifact.com/rhode-island/statements/2014/nov/01/david-cicilline/david-cicilline-says-us-has-third-lowest-minimum-w/	TRUE
1	During a highly charged -- and pretty entertaining -- debate on Channel 10 last Sunday, incumbent Democrat David Cicilline squared off against Cormick Lynch, the Republican who is challenging him for the 1st Congressional District seat.	en	False	politifact	p	http://www.politifact.com/rhode-island/statements/2014/nov/01/david-cicilline/david-cicilline-says-us-has-third-lowest-minimum-w/	-
2	Lynch kept speaking over Cicilline, Cicilline repeatedly patted Lynch on the arm to quiet him, and moderator Bill Rappleye tried to keep everything under control. ("Please, don't touch," he said more than once.)	en	False	politifact	p	http://www.politifact.com/rhode-island/statements/2014/nov/01/david-cicilline/david-cicilline-says-us-has-third-lowest-minimum-w/	-

state-of-the-art. Since this task is a new problem, there is not any natural baseline candidate. Therefore, we have chosen ClaimBuster (HASSAN et al., 2017) because it is related to this problem. ClaimBuster uses Support Vector Machine (CORTES; VAPNIK, 1995) with Bag-of-words and Term Frequency-Inverse Document Frequency (TF-IDF) vectors. Althought BERT typically outperforms SVM models, we would like to understand how better pre-trained models perform in this task.

The Bidirectional Encoder Representations from Transformers, in short, BERT, is a language representation model designed by Google in 2018. It has end-to-end models, pre-trained on a large corpus of data without human labelling, due to an automatic process for generating inputs and labels from the texts. Therefore, it is able to provide a simple and powerful method, obtaining redefined state-of-the-art natural language processing tasks. Its success relies on its ability to model complex and non-linear relationships among the data. In this study, we have used a Portuguese pre-trained model for the articles in Portuguese, and an English pre-trained model for articles in English.

Support Vector Machine (SVM) is a supervised machine learning algorithm for two-group classification problems. It consits on a conventional model that has reliable performance in several text classification tasks (CORTES et al., 2020), (SUENO; GERARDO; MEDINA, 2020), (WANG, 2017). Different from end-to-end models, it receives as input a numerical vector. In this work, we used the bag-of-words representation with Term Frequency-Inverse Document Frequency (TF-IDF) vectors.

3.3.1 Additional variations

Additionally to BERT and SVM approaches mentioned above, some enhancements to both models were also in place. In regards to the *claimReviewed* experiments, we raised an hypothesis that for each fact-checking article, its sentences' HTML tags could be added as a feature to the classifier, in order to produce more accurate results. Therefore, in this work we wanted to make a comparison between these models with the HTML tag as a feature, and without it. The *reviewRating* classification did not consider the usage of HTML tags as feature.

For the BERT experiments, originally we used two distinct pre-trained models, one in English and the other one in Portuguese. However, we also wanted to evaluate the performance of a multilingual pre-trained model applied to the corpus, so that we could compare the results with the language-specifc models for both *claimReviewed* and

reviewRating tasks. In summary, below are described the two variations of our experiments:

- **_HTML_TAGS**: are the models that also use sentences' HTML tags (eg.: <title>, <h1>) as feature.
- **_ML**: are BERT classifiers that employ a multilingual pre-trained model instead of a language-specific model.

3.4 Validation

In order to validate the accuracy of our models and avoid overfitting, we have used the cross-validation technique. In the cross validation method, the dataset is divided into two subsets: the *training* and the *test*. The idea behind this method is to use a subset of data which was left out of the training to test the model.

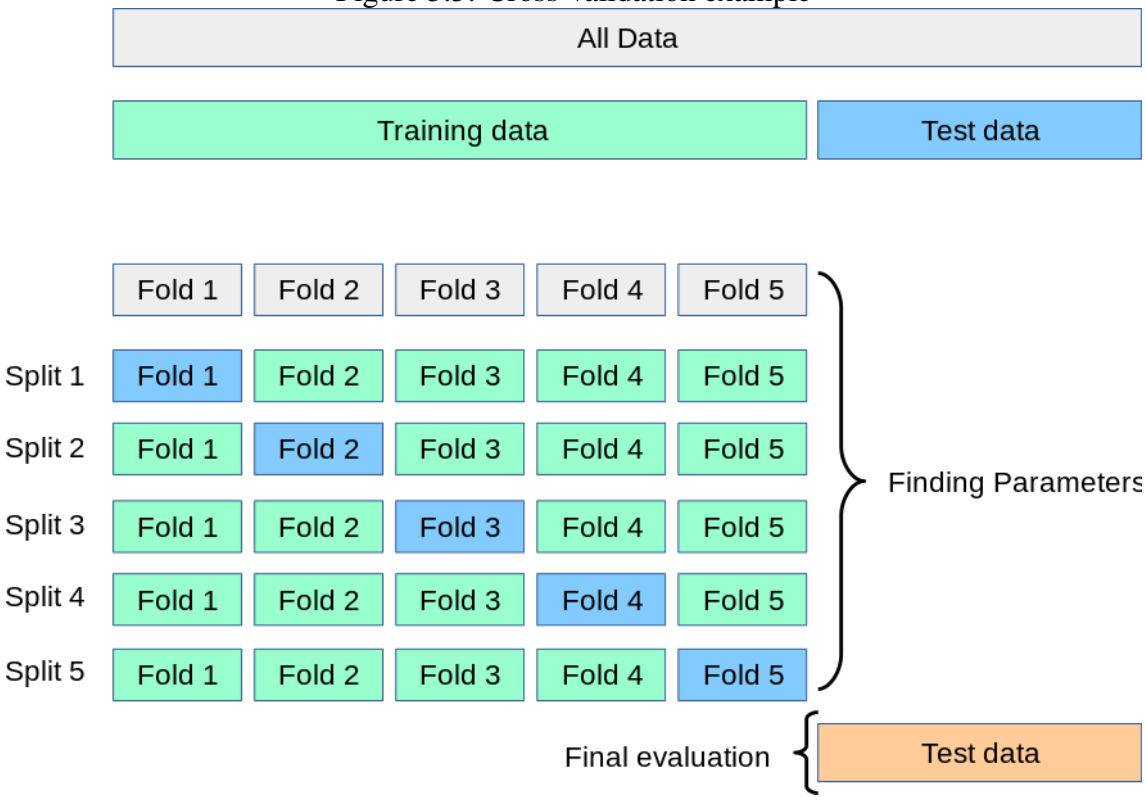
In this study, we adopted the Stratified Group K-Fold technique, which is a particular cross validation implementation. In this approach, the dataset is separated into k equal-sized subsamples, preserving the percentage of samples for each class, and ensuring that the same group will not appear in two different folds. For each fold, a single subsample is retained as the validation data for testing the model, and the remaining k-1 subsamples are used as training data. The cross-validation process is then repeated k times. At the end, all instances are used for both training and validation, and each observation is used for validation exactly once. Finally, the mean measure value was computed using the k results' precision from each fold, producing a single estimation.

Within our research, fact-checking websites were the k-fold groups. Five folds were chosen for English articles. However, for Portuguese articles this amount had to be decreased to four folds, as the number of distinct groups must be at least equal to the number of folds.

3.5 Metrics

In order to evaluate our proposed work as a classification task we adopted the standard information metrics, such as precision, recall and F1-score. The precision aims to identify the proportion of positive identifications which were actually correct. The recall aims to identify the proportion of actual positives that were identified correctly.

Figure 3.5: Cross-validation example



Source: sci-kitlearn.org

And finally, the F1-score is the weighted average of the precision and recall. For the F1 score, the best value is 1 and the worst is 0. For calculating these evaluation metrics, we must consider the confusion matrix of the model:

- True Positives (tp): model correctly predicts the positive class.
- False Positives (fp): model incorrectly predicts the positive class.
- True Negative (tn): model correctly predicts the negative class.
- False Negative (fn): model incorrectly predicts the negative class.

The metrics employed the macro average due to the imbalanced characteristic of the data set and can be briefly described as follows:

- *Precision:* the fraction of the websites classified as fake that are really fake news. $Precision = \frac{tp}{tp+fp}$
- *Recall* is the fraction of the fake websites that were successfully identified. $Recall = \frac{tp}{tp+fn}$
- *F1-score* corresponds to the harmonic mean between precision and recall. $f1 = 2 * \frac{precision*recall}{precision+recall}$

4 IMPLEMENTATION AND EXPERIMENTS

In this chapter we describe the implementation approach for the BERT and SVM experiments. We start discussing the implementation with SVM, which is our baseline candidate, and then present the approach with BERT, the state-of-art.

As a matter of fact, all experiments done within this study were performed through the Google Colaboratory (Colab)[1] platform, which is based on the open source project Jupyter[2], allowing the development and execution of Python code in the web. More technically, it is a hosted Jupyter notebook service that requires no setup to use, while providing free access to computing resources including GPUs. In terms of the resource specifications, Colab has dynamic usage limits that sometimes fluctuate, and it does not provide guaranteed or unlimited resources. This means that overall usage limits as well as idle timeout periods, maximum VM lifetime, GPU types available, and other factors vary over time. As these limits can change quickly, they are not published by Colab.

In our experiments, we have used the GPU runtime in order to optimize our implementations. However, Colab also does not specify the GPU in use for each runtime execution, but the GPUs available often include Nvidia K80s, T4s, P4s and P100s.

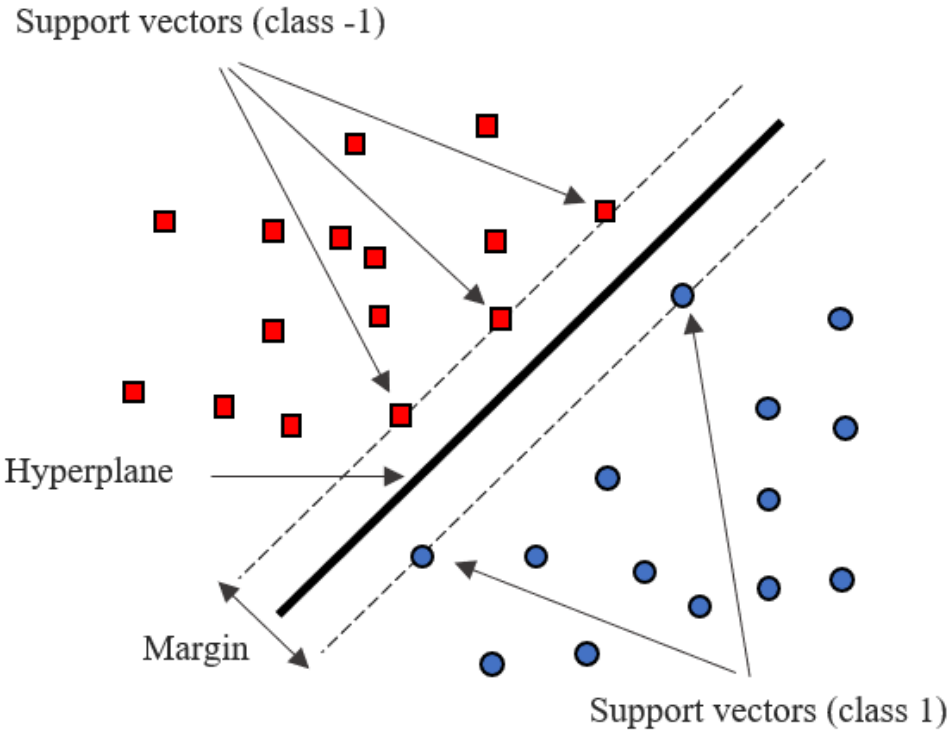
Despite having around 4.500.000 chunks of data within our dataset, due to computational restrictions of Google Colab, we have had to shorten our dataset in order to train the classification models. Therefore, from the 4.500.000 records we initially had, we've randomly selected a subset of 125.000 entries, equally distributed among the fact-checkers, to train the models.

4.1 SVM

Support Vector Machines are supervised learning models, originally designed for bi-classification problems, which works with a simple idea of creating a hyperplane that separates the data into distinct classes. The goal is to maximize the hyperplane's margin between data points of each class. The SVM method is used in the most diverse applications, such as cancer genomic classification (HUANG et al., 2018), fault detection in wireless networks (ZIDI; MOULAHI; ALAYA, 2018) and text classification (CORTES et al., 2020).

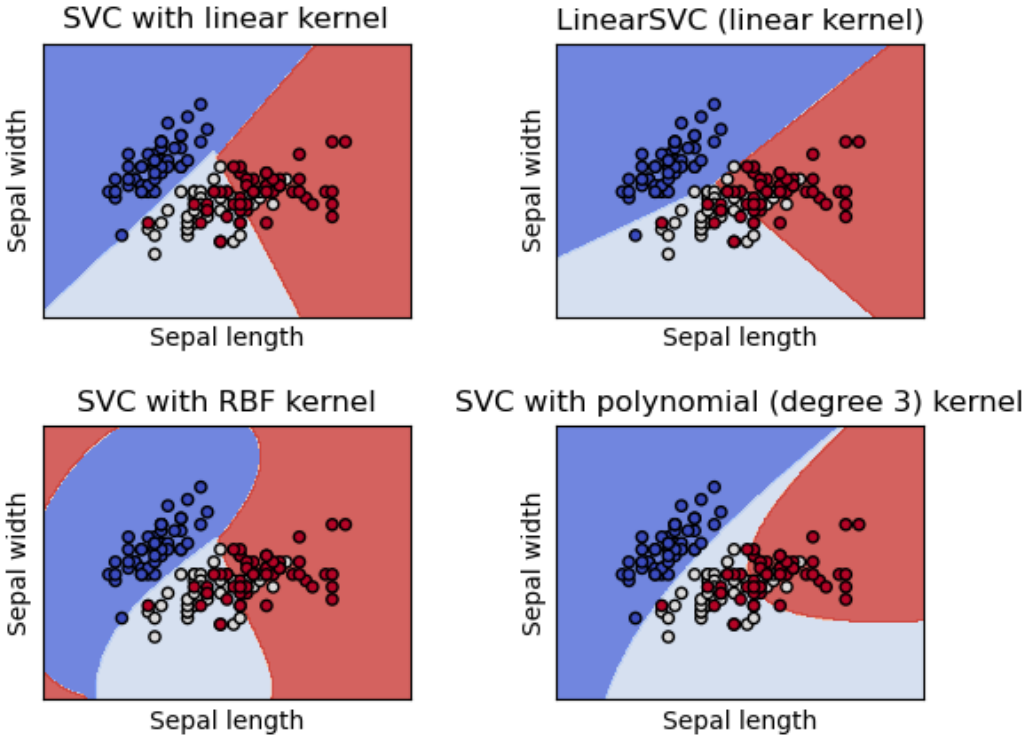
[1] https://research.google.com/colaboratory/faq.html
[2] https://jupyter.org/

Figure 4.1: General classification hyperplane representation of SVM algorithm.



Source: Remasterization of (BAGHAEE et al., 2020)

Figure 4.2: Support Vector Machine



Source: scikit-learn.org

In our experiments, we have used word embedding method, in order to convert our sentences to a numerical vector. Basically, the algorithm's input in our work is a bag-of-words vector representing the words from the text with its respective TF-IDF weight.

A bag-of-words is one of the most popular methods for text representation. Basically it considers each word as feature and counts the frequency of the words within the respective document. However, it does not consider the order and structure of the words. For understanding the *Term Frequency-Inverse Document Frequency* (TF-IDF) concept, it is also important to understand the *Term Frequency* (TF) and *Inverse Document Frequency* (IDF). These concepts are described below:

- **Term-Frequency (TF)**: it is defined as the number of times a word appears within a document. Also, according to (LUHN, 1957), the weight of a term that occurs in a document is simply proportional to its frequency. The term frequency, $tf(t,d)$ can be mathematically defined by $tf(t,d) = \frac{f_{t,d}}{\sum_{t' \in d} f_{t',d}}$, where $f_{t,d}$ is the number of times the term t occurs in document d.
- **Inverse-Document-Frequency (IDF)**: In the TF component, all terms have the same importance, including stop words, which occur much more frequently. Due to this reason, the goal of the IDF is to measure how much information the term provides, by weighting down most frequent terms and scaling up the most rare ones. The IDF, $idf(t,d)$ can be mathematically defined by $idf(t) = \log \frac{N}{1+df_t}$, where N is the number of documents in the corpus and $df(t)$ is the number of documents which contains the term t in the corpus N. This way, the IDF will be very low for stop-words, which are very frequent within documents.
- **Term-Frequency-Inverse-Document-Frequency (TF-IDF)**: the TF-IDF is the ideal measure to evaluate the importance of a word within a document corpus. The $tfidf(t,d)$ can be mathematically defined as $tfidf(t,d) = tf(t,d) * idf(t)$. Usually, this approach allows skipping some pre-processing tasks, like stop words removal, as it performs better generalizations within the available data.

The SVM implementation was held through the Python module Scikit-Learn (PEDREGOSA et al., 2011), which brings several state-of-the-art machine learning algorithms for both supervised and unsupervised problems.

For the *claimReviewed* classification, we have implemented two distinct approaches: using the HTML tag of the sentence as a feature, and not using it. As described in the previous chapter, when we prepared the dataset for this study, we retrieved the HTML tag

which embedded each sentence.

The *reviewRating* classification was more challenging. In order to train the model for predicting the verdict of a claim, we modified a bit our dataset to include article's list of sentences as a feature. Basically, we had to convert the sentences into a bag of words representation. Since most fact-checker articles have the verdict explicited in the page several times, our hypothesis was that using a bag of words would bring good results for this task.

4.1.1 Parametrization

Based on the good performance and results of previous researches on text classification using SVM, such as (CORTES; WOLOSZYN; BARONE, 2018), (CORTES et al., 2020) and (WANG, 2017), a Linear SVM kernel was used in our implementation, for both *claimReviewed* and *reviewRating* predictions. The regularization parameter C was equal to 1.0 and the norm penalty was the default *l2*.

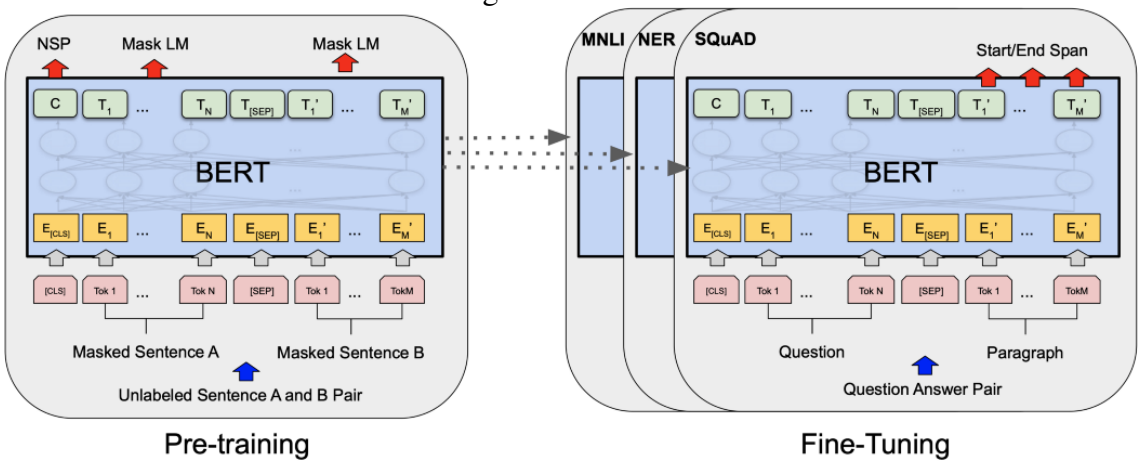
One of the challenges of using the bag-of-words representation was due to the initial amount of records available in the dataset, which was 4.500.000 distinct sentences considering both English and Portuguese articles. This would generate a huge vocabulary, resulting in a vector with mostly zeros (sparse vector). Therefore, besides reducing the using a subset of 125.000 sentences, we have also limited the maximum number of features in the vocabulary to 5.000.

The Stratified Group K-Fold cross validation method had $k = 5$ for English articles. However, for Portuguese the parameter was decreased to $k = 4$.

4.2 BERT

The Bidirectional Encoder Representations from Transformers is a powerful method based on end-to-end models. In (DEVLIN et al., 2019) the BERT implementation is described in details. Basically, there are two steps in the framework: *pre-training* and *fine-tuning*. First, the model is trained on unlabeled data through different tasks, and the parameters are saved. Then, in the second task, the model is initialized with the pre-training parameters, and then fine-tuned using labeled data from downstream tasks (which have separate fine-tuned models).

Figure 4.3: BERT



Source: (DEVLIN et al., 2019)

Underneath its implementation, BERT uses Masked Language Modeling (MLM) and Next Sentence Prediction (NSP). (DEVLIN et al., 2019) explains that the main goal of MLM is to randomly mask some of the input tokens in order to predict the original vocabulary id of the masked word based only on its context. Therefore, the model is able to learn a bidirectional representation of the sentence. Related to the Next Sentence Prediction (NSP), the objective is to pre-train text-pair representations, in order to predict whether both sentences were following each other or not.

Once BERT is an end-to-end model, it does not require a numerical vector as input, like the SVM approach. Therefore, we did not have to use the bag-of-words representation for the sentences within the dataset. Also, the spreading of fake news happens in all countries, and all languages across the globe. Therefore, a very positive thing of the BERT algorithm is the fact that the pre-trained models are available in several distinct languages, and also multilingual, as described in (DEVLIN et al., 2019) and (WANG, 2017). The different models used in our implementation are described in more details within the following subsection.

For the *claimReviewed* classification, we have implemented some combinations of experiments. First, we implemented the English and then the Portuguese pre-trained model, with English and Portuguese subset of data, respectively. Then, we experimented the multilingual pre-trained model trained with both subsets of data: English and Portuguese articles. Also, for all of these combinations we have tested the HTML tag as a feature. The experiments ran on for this task are enumerated below:

1. English pre-trained model, with HTML tag as a feature.
2. Portuguese pre-trained model, with HTML tag as a feature.

3. English pre-trained model, without HTML tag as a feature.
4. Portuguese pre-trained model, without HTML tag as a feature.
5. Multilingual pre-trained model, with HTML tag as a feature.
6. Multilingual pre-trained model, without HTML tag as a feature.

This way, we were able to compare the performance of the HTML tag as a feature for both language-specific and multilingual pre-trained models.

For the *reviewRating* classification, we used the same dataset which was applied to this task through the SVM model, but without the need to convert the list of sentences into a bag of words representation, since BERT is an end-to-end model. The experiments executed were the following:

1. English pre-trained model.
2. Portuguese pre-trained model.
3. Multilingual pre-trained model.

4.2.1 Parametrization

As a wrapper of the Transformer (WOLF et al., 2020) library, the framework SimpleClassifier[3] was used, as it simplifies the implementation and usage of BERT's pre-trained models. Regarding the fine-tuning, we only used one epoch with a learning rate equal to 4e-5. Related to the pre-trained models, we have chosen 3 distinct models in our experiments, which were used in both *claimReviewed* and *reviewRating* classification tasks:

- *bert-base-cased*: pre-trained model with English text.
- *bert-base-portuguese-cased*[4]: also known as *BERTimbau* (SOUZA; NOGUEIRA; LOTUFO, 2020), is a pre-trained BERT model for brazilian portuguese.
- *bert-base-multilingual-cased*: trained on text in the top 102 languages with the largest Wikipedias.

In the same way as the SVM experiments, the Stratified Group K-Fold cross validation method had $k = 5$ for English articles. However, for Portuguese the parameter was decreased to $k = 4$.

[3] https://simpletransformers.ai/
[4] https://github.com/neuralmind-ai/portuguese-bert

5 RESULTS

In this chapter, we discuss the results of the two different approaches described in chapter 4: SVM and BERT. The main goal is to compare the baseline method (SVM) with the state-of-the-art method (BERT) for the tasks of *claimReviewed* and *reviewRating* predictions. First, we will present the results of the *claimReviewed* classification. Then, we will present the results of the *reviewRating* classification.

5.1 *claimReviewed* Classification

The results of the task of *claimReviewed* classification are shown in Table 5.1. According to our experiments, BERT models have generated better results for both Portuguese and English articles, achieving an F1-score of 80.3% for English and 67.4% for Portuguese, in best scenarios. The difference between the best BERT result and the best SVM result for English is 15.7%, while for Portuguese it is 6.2%.

Regarding the addition of the HTML tag as a feature for the English language, it improves the SVM results by 6.3% and 10.5% for BERT. However, for the Portuguese language, the tag's inclusion impaired the performance in 11.9% for the SVM model and 8.3% for BERT. As a matter of fact, the good performance of the BERT models in this task is expected. Deep learning based models have higher predictive power and generalize better on unseen data. Also, the SVM models present some considerable limitations compared with BERT. SVM using a Bag-of-word vector has difficulties to semantically represent the terms of a sentence and does not consider the order of those terms. Moreover, the HTML tags' inclusion improves the classifiers' performance for the English language considerably, while it got worse for Portuguese. The gain in English performance is due to fact-checking websites in English using the same tags to represent the *claimReviewed*, while websites in Portuguese are using different ones. As the number of fact-checking sites employed in this experiment is small, it is not possible to state that the HTML tag should be used for this task. However, we can say that it has a high degree of importance and should be considered in new studies.

When we analyze the difference between BERT multilingual and language-specific model performance, it is possible to observe that the multilingual BERT and the English BERT present a small difference of about 2% of F1-score. Otherwise, the difference between multi-language BERT and the Portuguese BERT is greater with about 6% of

F1-score. These results show that the multilingual model has similar results as the mono-language English model. In contrast, for Portuguese we observe lower performance when compared to the mono-language version.

Table 5.1: Results for *claimReviewed* Prediction

Model	English			Portuguese		
	Precision	**Recall**	**F1-Score**	**Precision**	**Recall**	**F1-Score**
SVM	60.7% ± 6.1	62.2% ± 7.0	58.3% ± 8.4	62.7% ± 9.1	64.1% ± 10.1	61.2% ± 10.1
SVM_HTML_TAGS	67.8% ± 6.5	75.3% ± 5.1	64.6% ± 9.7	52.7% ± 2.67	54.8% ± 3.6	49.3% ± 6.9
BERT	68.0% ± 10.0	70.7% ± 11.8	67.3% ± 9.7	**69.3% ± 13.6**	**70.1% ± 11.8**	**67.4% ± 1.6**
BERT_HTML_TAGS	78.9% ± 12.5	80.9% ± 11.5	78.3% ± 13.0	62.6% ± 5.4	69.2% ± 5.9	59.1% ± 7.2
BERT_ML	70.4% ± 9.3	72.5% ± 10.3	69.8% ± 9.2	63.4% ± 13.3	65.0% ± 13.3	61.2% ± 14.9
BERT_HTML_TAGS_ML	**80.6% ± 11.5**	**81.9% ± 11.0**	**80.3% ± 11.7**	64.2% ± 12.5	65.7% ± 13.8	59.7% ± 16.9

5.2 *reviewRating* Classification

The results regarding the *reviewRating* classification are presented in Table 5.2 and show that the SVM model outperforms BERT in both languages tested, by achieving an F1-Score of 69.7% for English and 63.8% for Portuguese. Thus, the difference between the best SVM result and the best BERT result for English is equal to 16.8%, and for Portuguese, the difference is equal to 2.5%.

Unlike the *claimReviewed* results, the BERT models present the worst results compared to the SVM. Our assumption is that the *reviewRating* classification is a challenging task once the most substantial chunk of information about the claim conclusion is somewhere inside the HTML page. However, the pages' data volume is enormous compared to the chunks used in the *claimReviewed* task, and most of them are considered noise for the problem. Therefore, we believe the SVM got a better performance once it simplifies these large HTML pages into a Bag-of-words vector and some of their elements are relevant keywords, like *"correct", "false", and "wrong"*. On the other hand, BERT should be struggling with the long sequence of words from the HTML page to identify these relevant semantic concepts. Future work should develop methodologies to simplify these large HTML pages in order to maintain relevant information and ignore the noise before submitting it to End-to-End models.

Regarding the BERT model results with the language, we observe that Portuguese results are higher than English ones, with an F1-score 8.4% higher than English. Our hypothesis that justifies this difference is related to the characteristics of the fact-checking articles of each language. In Portuguese, these articles should present structures more friendly to BERT compared to articles in English. Nevertheless, this hypothesis will be verified in future works.

Table 5.2: Results for *reviewRating* Prediction

Model	**English**			**Portuguese**		
	Precision	**Recall**	**F1-Score**	**Precision**	**Recall**	**F1-Score**
SVM	**73.1% ± 8.6**	**74.7% ± 6.1**	**69.7% ± 12.0**	**62.3% ± 2.7**	**70.4% ± 3.9**	**63.8% ± 3.2**
BERT	63.6% ± 1.3	47.6% ± 0.8	52.9% ± 1.1	62.6% ± 1.1	60.1% ± 0.9	61.3% ± 0.9
BERT_ML	62.9% ± 1.2	44.6% ± 0.9	51.4% ± 1.0	57.8% ± 0.7	55.0% ± 1.0	56.2% ± 0.8

6 CONCLUSION

The goal of this work was to design and implement a solution for the following research question: *Is it possible to predict the ClaimReview markup?*. Our experiments proposed a task of automatically predicting attributes of the *ClaimReview* markup from HTML websites, modeled as a classification problem to predict whether HTML elements of fact-checker articles are attributes of *ClaimReview* or not. As discussed in previous chapters, our hypothesis was based on empirical observations which stated that most of *ClaimReview* markup's required properties, such as *claimReviewed, reviewRating and URL* are embedded within fact-checker publications.

Our work has evaluated two machine learning models for this task: BERT, as the state-of-the-art, and SVM, as the baseline. After conducing several experiments to validate the feasibility of this solution, in which the detailed results were presented in chapter 5, we were able to conclude that for *claimReviewed* prediction, overall BERT models achieve significantly better performance when compared to the baseline method SVM. However, for *reviewRating* prediction, the SVM model outperforms BERT in both languages. Also, an interesting thing of the *claimReviewed* prediction is that when the HTML tag was added as a feature, the performance was improved for English articles (for both BERT and SVM models), however, for Portuguese articles the performance has decreased.

Our evaluation showed that our method can generate a significant number of *claimReviewed* and *reviewRating* correctly in two different languages. The results indicate that the attributes from the *ClaimReview* markup can be predicted in order to extract structured information from HTML websites of fact-checkers. The automatic generation of *ClaimReview* markup is an important step towards machine interpretation of fact-checking articles and can be used in two ways:

- to create live knowledge bases that can serve for further training AI tools and;
- to allow a better interpretation of fact-checking articles by search engines.

In the future, we will explore the application of different pre-processing strategies for the HTML documents in order to remove the massive amount of noise and increase the performance of the proposed task. Moreover, we will test the classification task's performance in different languages such as German, Italian, and Spanish. We will also consider Natural Language Generation techniques to create *ClaimReview* automatically.

We hope that our work plays a role in the progressive process of leveraging structured information from fact-checkers to improve the quality of training data to facilitate the detection of false information.

REFERENCES

ADLER, B.; BOSCAINI-GILROY, G. Real-time claim detection from news articles and retrieval of semantically-similar factchecks. **arXiv preprint arXiv:1907.02030**, 2019.

ALOSHBAN, N. Act: Automatic fake news classification through self-attention. In: **12th ACM Conference on Web Science**. [S.l.: s.n.], 2020. p. 115–124.

BAGHAEE, H. R. et al. Support vector machine-based islanding and grid fault detection in active distribution networks. **IEEE Journal of Emerging and Selected Topics in Power Electronics**, v. 8, n. 3, p. 2385–2403, 2020.

BRAUN, J. A.; EKLUND, J. L. Fake news, real money: Ad tech platforms, profit-driven hoaxes, and the business of journalism. **Digital Journalism**, Taylor & Francis, v. 7, n. 1, p. 1–21, 2019.

CARABALLO, J. A taxonomy of political claims. 2018.

CASWELL, D.; DÖRR, K. Automated journalism 2.0: Event-driven narratives: From simple descriptions to real stories. **Journalism practice**, Taylor & Francis, v. 12, n. 4, p. 477–496, 2018.

CORTES, C.; VAPNIK, V. Support vector networks. **Machine Learning**, v. 20, p. 273–297, 1995.

CORTES, E. et al. An empirical comparison of question classification methods for question answering systems. In: **Proceedings of the 12th Language Resources and Evaluation Conference**. Marseille, France: European Language Resources Association, 2020. p. 5408–5416. ISBN 979-10-95546-34-4. Available from Internet: <https://www.aclweb.org/anthology/2020.lrec-1.665>.

CORTES, E. G.; WOLOSZYN, V.; BARONE, D. A. C. When, where, who, what or why? a hybrid model to question answering systems. In: VILLAVICENCIO, A. et al. (Ed.). **Computational Processing of the Portuguese Language**. Cham: Springer International Publishing, 2018. p. 136–146. ISBN 978-3-319-99722-3.

DEVLIN, J. et al. BERT: Pre-training of deep bidirectional transformers for language understanding. In: **Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)**. Minneapolis, Minnesota: Association for Computational Linguistics, 2019. p. 4171–4186. Available from Internet: <https://www.aclweb.org/anthology/N19-1423>.

FERREIRA, W.; VLACHOS, A. Emergent: a novel data-set for stance classification. In: **Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: Human language technologies**. [S.l.: s.n.], 2016. p. 1163–1168.

HASSAN, N. et al. Toward automated fact-checking: Detecting check-worthy factual claims by claimbuster. In: **Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining**. [S.l.: s.n.], 2017. p. 1803–1812.

HUANG, S. et al. Applications of support vector machine (svm) learning in cancer genomics. **Cancer Genomics & Proteomics**, International Institute of Anticancer Research, v. 15, n. 1, p. 41–51, 2018. ISSN 1109-6535. Available from Internet: <https://cgp.iiarjournals.org/content/15/1/41>.

JARADAT, I. et al. Claimrank: Detecting check-worthy claims in arabic and english. **arXiv preprint arXiv:1804.07587**, 2018.

JIANG, S. et al. Factoring fact-checks: Structured information extraction from fact-checking articles. In: **Proceedings of the 2020 Web Conference (WWW 2020)**. [S.l.: s.n.], 2020.

KONSTANTINOVSKIY, L. et al. Towards automated factchecking: Developing an annotation schema and benchmark for consistent automated claim detection. **arXiv preprint arXiv:1809.08193**, 2018.

LUHN, H. P. A statistical approach to mechanized encoding and searching of literary information. **IBM Journal of Research and Development**, v. 1, n. 4, p. 309–317, 1957.

PEDREGOSA, F. et al. Scikit-learn: Machine learning in python. **Journal of Machine Learning Research**, v. 12, n. Oct, p. 2825–2830, 2011.

PÉREZ-ROSAS, V. et al. Automatic detection of fake news. **arXiv preprint arXiv:1708.07104**, 2017.

REIHANI, H. et al. Non-evidenced based treatment: An unintended cause of morbidity and mortality related to covid-19. **The American Journal of Emergency Medicine**, v. 39, 05 2020.

SOUZA, F.; NOGUEIRA, R.; LOTUFO, R. BERTimbau: pretrained BERT models for Brazilian Portuguese. In: **9th Brazilian Conference on Intelligent Systems, BRACIS, Rio Grande do Sul, Brazil, October 20-23 (to appear)**. [S.l.: s.n.], 2020.

SUENO, H.; GERARDO, B.; MEDINA, R. Multi-class document classification using support vector machine (svm) based on improved naïve bayes vectorization technique. **International Journal of Advanced Trends in Computer Science and Engineering**, v. 9, p. 3937, 06 2020.

TCHECHMEDJIEV, A. et al. Claimskg: a knowledge graph of fact-checked claims. In: SPRINGER. **International Semantic Web Conference**. [S.l.], 2019. p. 309–324.

VLACHOS, A.; RIEDEL, S. Fact checking: Task definition and dataset construction. In: . [S.l.: s.n.], 2014. p. 18–22.

VO, N.; LEE, K. Where are the facts? searching for fact-checked information to alleviate the spread of fake news. **arXiv preprint arXiv:2010.03159**, 2020.

VOSOUGHI, S.; ROY, D.; ARAL, S. The spread of true and false news online. **Science**, American Association for the Advancement of Science, v. 359, n. 6380, p. 1146–1151, 2018. ISSN 0036-8075. Available from Internet: <https://science.sciencemag.org/content/359/6380/1146>.

WANG, W. Y. " liar, liar pants on fire": A new benchmark dataset for fake news detection. **arXiv preprint arXiv:1705.00648**, 2017.

WANG, Z.; QU, Z. Research on web text classification algorithm based on improved cnn and svm. In: **2017 IEEE 17th International Conference on Communication Technology (ICCT)**. [S.l.: s.n.], 2017. p. 1958–1961.

WOLF, T. et al. Transformers: State-of-the-art natural language processing. In: **Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations**. Online: Association for Computational Linguistics, 2020. p. 38–45. Available from Internet: <https://www.aclweb.org/anthology/2020.emnlp-demos.6>.

WOLOSZYN, V.; NEJDL, W. Distrustrank: Spotting false news domains. In: **Proceedings of the 10th ACM Conference on Web Science**. [S.l.: s.n.], 2018. p. 221–228.

WOLOSZYN, V. et al. Untrue.news: A new search engine for fake stories. **arXiv preprint arXiv:2002.06585**, 2020.

ZHOU, X.; ZAFARANI, R. A survey of fake news: Fundamental theories, detection methods, and opportunities. **ACM Computing Surveys (CSUR)**, ACM New York, NY, USA, v. 53, n. 5, p. 1–40, 2020.

ZIDI, S.; MOULAHI, T.; ALAYA, B. Fault detection in wireless sensor networks through svm classifier. **IEEE Sensors Journal**, v. 18, n. 1, p. 340–347, 2018.