UNIVERSIDADE FEDERAL DO ESTADO DO RIO DE JANEIRO
CENTRO DE CIÉNCIAS EXATAS E TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

Judice Verum, a Methodology for Automatically Classify Fake, Sarcastic and True Portuguese News

Fernando Cardoso Durier da Silva

Orientadores

Ana Cristina Bicharra Garcia

RIO DE JANEIRO, RJ - BRASIL
Dezembro de 2019

Judice Verum, a Methodology for Automatically Classify Fake, Sarcastic and True Portuguese News

Fernando Cardoso Durier da Silva

DISSERTAÇÃO APRESENTADA COMO REQUISITO PARCIAL PARA OBTENÇÃO DO TÍTULO DE MESTRE PELO PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA DA UNIVERSIDADE FEDERAL DO ESTADO DO RIO DE JANEIRO (UNIRIO). APROVADA PELA COMISSÃO EXAMINADORA ABAIXO ASSINADA.

Aprovada por:

Ana Cristina Bicharra Garcia - UNIRIO

Márcio de Oliveira Barros - UNIRIO

Bianca Zadrozny, D.Sc. - IBM Research

RIO DE JANEIRO, RJ - BRASIL

Dezembro de 2019

Cardoso Durier da Silva, Fernando
C268 Judice Verum, a Methodology for Automatically
Classify Fake, Sarcastic and True Portuguese News
, 64f.

Orientador: Ana Cristina Bicharra Garcia
Dissertação (Mestrado em Informática) - Universidade Federal do
Estado do Rio de Janeiro, Rio de Janeiro, 2019.

1. Machine Learning. 2. Fake News. 3. Natural Language Processing.
4. Social Media Analysis. 5. Artificial Intelligence.
I. Bicharra Garcia, Ana Cristina. II. Universidade Federal do Estado do
Rio de Janeiro (2018-). Centro de Ciências Exatas e Tecnologia. Curso de
Mestrado em Informática. Judice Verum, a Methodology for Automatically
Classify Fake, Sarcastic and True Portuguese News .

I would like to dedicate this research to my father and my mother who always supported my choices, and since young age have provided me complete education of quality, they inspired me as a professional and as a human being.

Acknowledgements

I would like to thank my family, my parents and my grandmother, for always supporting, believing on me, and being by my side in all moments of this research.

I would like to thank my teachers for the knowledge, kindness, support and friendship they provided me. In special Ana Cristina Bicharra Garcia, Gleison Santos, Kate Cerqueira Revoredo, Flávia Maria Santoro, Maria del Rosário Girardi, Fernanda Baião, Leonardo Guerreiro Azevedo, Morganna Carmem Diniz, Geiza Maria Hamazaki, Simone Leal Bacellar, Adriana Alvim, Márcio Barros, Pedro Nuno Moura, e Dayanne Prudencio.

I would like to thank the Federal University of the State of Rio de Janeiro UNIRIO as this wonderful institution not only provided the best teachers, the best knowledge, my grounded truth, and my technological basis, but, also offered me friends I will never forget, eternal memories, and comfort.

I would like to thank IBM for providing me flexibility to fulfill my dreams of research and personal development, specifically offered by my direct managers from IBM Global Financing, Rory McClaskey and Flávio Novis.

I would like also to thank Bruno Lima Cardoso my tutor since IBM Research Brazil for providing me advises for my academic and professional career, and always motivating my self improvement and always believing in my capabilities. I would like also to thank my friends that always stood up by my side and sometimes allowed me to pass some of my knowledge to them. Special thanks to Wagner da Silva Maciel Sodré, Jéssica Allonso da Costa, André Francisco Lima, Bianca Gotaski de Melo, Guilherme Correa de Melo, Marcos Moura Pinho, e Yuri Lisboa.

Cardoso Durier da Silva, Fernando. **Judice Verum, a Methodology for Automatically Classify Fake, Sarcastic, and True Portuguese News**. UNIRIO, 2019. 64 páginas. Dissertação de Mestrado. Departamento de Informática Aplicada, UNIRIO.

RESUMO

A ampla adoção da tecnologia da informação facilitou o compartilhamento de informações na Web. Também permitiu o surgimento e a disseminação de notícias falsas. Como um vírus prejudicial, as notícias falsas se espalham mais rápido que a verdade, causando falta de confiabilidade na mídia. Há um esforço mundial para encontrar maneiras de estancá-lo.

No entanto, detectar notícias falsas é um desafio que fica ainda mais difícil nas mídias sociais, porque as pessoas estão acostumadas à comunicação informal que pode incluir sarcasmo e ironia. Sarcasmo e notícias falsas podem ser facilmente confundidas com agentes artificiais. A censura a mensagens sarcásticas viola o direito de expressão dos cidadãos, pelo menos nos EUA, protegido por lei (a Primeira Emenda). Nossa análise revelou as diferentes abordagens para lidar com figuras de fala devido à sua carga de polaridade nas sentenças. Os poucos trabalhos relacionados nessa área apresentam resultados independentes. Esses estudos abordam principalmente a detecção de notícias falsas em textos em inglês usando algoritmos inteligentes artificiais clássicos, como SVM, redes neurais e Gaussian Naive Bayes. Não há estudo sobre a língua portuguesa.

Para nosso experimento, criamos um banco de dados de notícias falsas, sarcásticas e verdadeiras, composto por 11362 notícias, uma vez que não havia um conjunto de dados disponível em português. Esperamos que este conjunto de dados forneça aos pesquisadores uma boa matéria-prima para futuros trabalhos na área. Criamos três raspadores da Web para extrair o conjunto de notícias de fontes conhecidas e populares de notícias falsas, sarcásticas e verdadeiras entre a imprensa eletrônica on-line brasileira, no nosso caso, Folha de São Paulo, E-farsas e Sensacionalista.

Propomos um novo método para detectar notícias falsas chamado Judice Verum, com base em uma fase de pré-processamento de enriquecimento de dados composta por três etapas: primeiro extraia recursos sintáticos, depois extraia recursos semânticos e carga sentimental e, finalmente, faça a incorporação de palavras das notícias.

Esse processo é novo no sentido de que, por nossa engenharia de recursos, somos capazes de fornecer aos nossos modelos de aprendizado de máquina a capacidade de entender a carga sentimental em palavras, frases e variância, e também a semelhança entre documentos por estilometria, e nosso processo está focado em o aspecto linguístico de uma notícia falsa, e não na topologia social, como a maioria dos trabalhos relacionados na área.

Depois disso, nosso modelo usa cinco diferentes técnicas clássicas de aprendizado de máquina: SVM, KNN, Gaussian Naive Bayes, Decision Tree e Random Forest. Concluímos que nosso processo de fusão de dados fez uma enorme diferença no desempenho do modelo em comparação com abordagens independentes, com um aumento comparativo de quase 10 % a mais. Para validar nossos resultados, construímos um conjunto de testes para comparar a evolução da eficiência de nosso modelo ao longo do processo de fusão de dados, extraímos métricas diferentes para comparar como precisão, medida f e área sob a curva de característica operacional do receptor. E até simulou testes-piloto contra um conjunto Gold Standard e um conjunto completamente novo de notícias sarcásticas.

Palavras-chave: Machine Learning; Fake News; Social Analysis; Semantic Web; Natural Language Processing.

ABSTRACT

The widespread adoption of information technology has facilitated sharing information on the Web. It also allowed the rise and spread of fake news. Like a harmful virus, fake news spread faster than the truth, causing unreliability on the media. There is a worldwide effort to find ways to stanch it.

However, detecting fake news is a challenge that gets even harder in social media because people are used to informal communication that might include sarcasm and irony. Sarcasm and fake news can be easily mistaken by artificial agents. Censorship on sarcastic messages infringes citizens freedom of speech right, at least in US, protected by law (the First Amendment). Our analysis revealed the different approaches to deal with figures of speech due to their polarity charge in sentences. The few related works in this area present stand-alone results. These studies mostly address the detection of fake news on English texts using classical artificial intelligent algorithms, such as, SVM, neural networks and Gaussian Naive Bayes. There is no study on Portuguese language.

For our experiment we created a database of fake, sarcastic, and true news composed of 11362 news, since there was no dataset available in Portuguese. We hope this dataset will provide researchers good raw material for future works in the area. We created three web scrapers for extracting the set of news from well known and popular sources of Fake, Sarcastic and True news among the Brazilian online electronic press, in our case Folha de São Paulo, E-farsas, and Sensacionalista.

We propose a new method for detecting fake news called Judice Verum based on a preprocessing phase of data enrichment composed of three steps: First extract syntactic features, then extract semantic features and sentimental charge, and finally do the word embedding of the news. This process is novel in the sense that by our feature engineering we are capable of provide to our machine learning models capability of understanding sentimental charge in words, sentences and variance, and also the similarity in between documents by stylometry, and our process is focused on the linguistic aspect of a fake news, and not on social topology like most of the related works in the area.

After that our model uses five different classical machine learning techniques the SVM, KNN, Gaussian Naive Bayes, Decision Tree, and Random Forest. We concluded that our data fusion process has made a huge difference into the model's performance compared to standalone approaches with a comparative increase of almost 10% more.

In order to validate our results we have built a set of tests to compare the evolution of our model's efficiency along the data fusion process, we extracted different metrics to compare like accuracy, f-measure, and area under the receiver operating characteristic curve. And even simulated pilot tests against a Gold standard set, and a completely unseen set of sarcastic news.

Keywords: Machine Learning; Fake News; Social Analysis; Semantic Web; Natural Language Processing.

Contents

List of Figures

List of Tables

Abbreviations List

NLP	Natural Language Processing
ML	Machine Learning
SVM	Support Vector Machine
Dec.Tree	Decision Tree
R. For.	Random Forests
GNB	Gaussian Naive Bayes
KNN	K Nearest Neighbors

Chapter 1

Introduction

1.1 Motivation

Humanity is now living the Web 3.0, the collaborative web overseen by artificial intelligent engines that through hyperpersonalization can deliver more content and more adequate information to readers, and in contrast the users are eager to give back content to the web and the cyber society through social media posts, shared activity logs from electronic gadgets, online game sessions and social discussion in the web common spaces [4].

Thus we are experimenting life in two worlds physical and virtual. In the physical world, our body and mind interact with the world and other people. We also create experiences in the virtual world through social medias by consuming different kinds of information and producing content to express ourselves in that community [5] [6] [7].

Due this the cyber space we know today, mostly constituted of different online communities, messenger services and leisure online platforms are flooded with information generated by the large number of interactions in between its inhabitants. Those interactions are inherent to the social beings we are, but, ill intended citizens of the cyber space take advantage of that to spread sets of verifiable fake information to cause havoc, discord, and mass manipulation of the cyber society.

Not only the common citizens have been participating into this cyber phenomena of living the cyber space, but, also the politicians and high spheres of government are taking huge advantage of this [4] [8]. Elections can be conducted almost entirely by the discussion in the web, or by eavesdropping upon conversations in between possible

electors users of social media, strategists planning campaigns based almost only on cyber interactions.

One big problem in this space where everyone can tell its version of the story is that in general humans tend to assume truth from what they learn in first instance and only after they really confirm, what makes fake news detection so important and relevant. Humans are fairly ineffective at recognizing deception [1].

It is more common to believe in information got in electronic vehicles than before. We are prone to propagate wrong information, such as fake news and hoaxes, due to a biological stimuli relating to either novelty of the fact, or by its absurdity [9] [10].

Also as we developed rich communication mean through our language we are prone to use figurative language, or other semantic resources to subtly deliver intentional harsh propositions of critique towards given targets, e.g. sarcasm, and irony [11] [12]. But, sometimes it is hard to differentiate what is a joke from what is an information intentionally created to mislead us.

In order to fight misconception and reduce the spread of misinformation many researchers, governmental agencies, and media entities are taking the effort of developing intelligent agents, algorithms and methods to classify information into truth or lie to therefore act accordingly [4] [8] [2] [13]. However, the sarcastic and critic factor is not taken in consideration, as most of our critics tends to be delivered by sarcasm or irony throughout our writings even more in restrictive political scenarios or anonymous impersonal critiques and can be interpreted as fake news, when they really are not.

1.2 Objective

In this research work we want to investigate the state of the art in detecting fake news and discerning them than sarcasm; to understand the current efforts and challenges still open in the literature; to build a fully functional dataset of news of the true, sarcastic, and fake categories; and finally to propose a new method and machine learning model capable of differentiate Portuguese Fake, Sarcastic, and Fake news only by analyzing its own content.

1.3 Research Problems

From the literature that we studied to base this research the current research problems still open in the community are:

- Lack of public, easy to access, available datasets of Portuguese Fake News;
- Most of the works are focused on non-Portuguese language, mostly English;
- Isolated approaches to detect Fake News;
- Lack of satisfactory model performance(Accuracy and F-measure) for the Portuguese scenario;

There are many works in English, Chinese, even Indonesian, and only three in Portuguese, where two of them are manual effort methodologies, meanwhile the other one is much similar to ours as it also tries to handle sarcasm. Also in the literature many researches propose different and isolated approaches that work well for their specific cases, or contexts, but, when applied onto one another contexts, they perform not so great, maybe by combining them, even if not entirely they may boost each other's performance like the ensemble methods. And in the end the models' performance in Portuguese scenario are yet not satisfactory, at least if we compare to the related works for other contexts(languages, topics, etc).

1.4 Proposed Solution

For this research we propose the creation of a methodology to fused sentimental, stylometry and context information into the news instances in order to provide the machine learning models the capabilities of not only better classify a new instance, but classify it by understanding the sentiment imbued into the text, the abrupt imbalance of sentimental charge, the similarities in between documents by its contents, and the stylometry behind all this like which linguistic resources have been used, syntactically and semantically.

1.5 Research Methodology

Our research methodology is a quantitative research, therefore we will run our methodology and models, measuring its performance/metrics in each step, even simulating pilot testing. Then with those metrics in hands we will compare how efficient the model and methodology is for each step, also comparing our results against the ones described in related works.

1.6 Research Scope

In our research we want to classify news into its categories only by analyzing their textual content because we want our models to not be so reliant upon external information sources such as social topology, or links, and to be less computational costly as possible thus, discarding image processing. Another point we want to address is the advantage of classical models in this scenario compared to the neural networks attempts, only briefly mentioning them and not investing much on them. Therefore our research scope is:

- Classes: Fake, Sarcastic, and Fake News;
- Data: News textual only data from sociopolitical context;
- Data Sources: Folha de São Paulo, Sensacionalista, and E-Farsas;
- Models used during the experiment: GNB, KNN, SVM, Decision Tree, and Random Forest;
- Models used for simple comparison purposes: MLP, LSTM:
- Language: Portuguese;
- Metrics: Accuracy, F-Measure, AUC ROC, and Confusion Matrix;

1.7 Research Work Structure

In order to present our work and findings we will be following the structure bellow:

- Introduction
- Theoretical Foundation
- Related Works
- Dataset
- Judice Verum: Methodology for detection fake news based on gradient sentiment polarity, stylometry, and document embedding
- Experiments
- Conclusion
- References

Chapter 2

Theoretical Foundation

In this chapter we want to present the common ground of concepts, definitions, theories and baselines for our comprehension of the research area and theme. This chapter is heavily inspired by our systematic literature review [4], as prescribed in [14] and [15].

2.1 Fake News

There are many definitions of fake news on the literature. In addition, the media has overused the term "fake news" in many different contexts and with distinct intents, which aggravates the problem of understanding what characterizes a given story as fake news. In this section we extend the definitions from [16] to characterize the properties of fake news. Then, we present some papers definition of fake news and its relations with some related concepts, that are commonly wrongly used as interchangeable by the media. Finally basing ourselves upon Shu et al. in [17] for the Fake News definition per say.

2.1.1 Publisher

We define the **Publisher** as the entity that provides the story to a public. For example, the publisher can be an user in a micro-blogging service like Twitter, a journalist in an online newspaper, or an organization in its own website. Note, that the publisher may or may not be the author of the story.

In the case that the publisher is the author of the story, Kumar and Shah [16] clas-

sifies the author based on its **intent** into **misinformation**, if the author has not the intent to *deceive*, or into **disinformation**, if the author has the intent to *deceive*. When the publisher is just spreading the story, ie. republishing content from other story, then we can classify them in bots and normal users.

2.1.2 Content

We define **Content** as the main information provided by the publisher in the story. At the moment of publication, the veracity of this information can be true, false or unknown. If the veracity is unknown, then it can be classified as a **rumor**, according with the definition of rumor from Zubiaga et al [18] as *"an item of circulating information whose veracity status is yet to be verified at the time of posting."*

The information can also be classified as factual, opinion or mixed. Opinion based information have no ground truth, in contrast with factual information, where the facts can be verified against a ground truth. In the case of factual, usually the content is a **claim**, made by the publisher. The veracity of the **claim** is the object of study of *automated fact-checking*, which has a recent report from Nieminen et al [19]

2.1.3 Extra media

In addition to the content, the story may include some media like picture, video, audio. The use of media unrelated to the content, with the objective of increasing the will of the reader to access the content, is considered clickbaiting [20] [21].

In this year of 2019 the fake news fabric has evolved even into producing fake extra media, like tampered videos of allegedly claims from politicians, or audios mimicking the real target of the fake news.

However it is proven already to be very difficult to handle the extra media in parallel to the fake news detection by textual content, due to the heavy preprocessing and data preparation for the machine learning models to be able to comprehend them, as we can see in the suggested workflow by Boididou, C. et al. in [20], figure 2.1

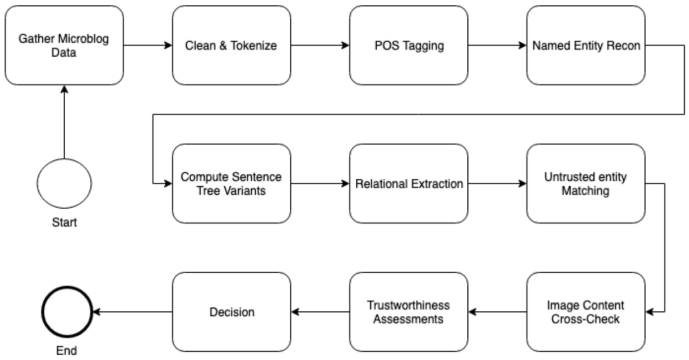

Figure 2.1: Image Processing Pipeline suggested by et Boididou, C. et al.

2.1.4 Fake News Definition and its Impact on Society

The authors use different names to define the same concept that can be observed in our works reviewed. They call it misinformation, rumour, hoax, malicious trend, spam or fake news, but all converge to the same semantic meaning, that is of an information that is unverified, of easy spread throughout the net, with the intention of either block the knowledge construction (by spreading irrelevant or wrong information due to lack of knowledge of the theme) or either manipulate the readers opinion. [22] [23] [24] [25] [26] [27] [3]

The majority of works consider it to be consequence of excessive marketing strategies or political manipulation. It should be observed though, that some authors consider the chance of those stream and spread of misinformation being unintentional sometimes, and happening due to cultural shock (e.g., Nepal Earthquake case described on [25]) or unconscious acts.

In this work we will utilize the definition of fake news from Shu et al. in [17], which is "a news article that is intentionally and verifiable false". Note that this definition shares similarities to our definition of a publisher with the intent to deceive and false factual content. However this definition is simplistic, since it does not cover half truths, opinion based contents, and humorous stories, like satires.

2.2 Sarcasm

The NLP research area takes care of extracting relevant patterns and information from semi-structured data, such as, text, microblogs and book's chapters. One of the challenges of NLP is the understanding of the semantics and the context in which the sen-

tences are written. Our natural language is full of ambiguity, regional variances, linguistic specificity and figurative languages. One of those figurative languages is sarcasm. As defined by [28], sarcasm is "a cutting, often ironic remark intended to express contempt or ridicule". Also Joshi [28] defines sarcasm detection as a NLP sub-task of predicting whether a textual fragment is sarcastic or non-sarcastic.

It is natural to compare and confuse sarcasm and irony [29] [11] [13] [30], but, as Tabacaru [11] states that irony has simply been defined as to say the opposite of what you originally said, but sarcasm is different. Due to a criticism implied within [31], it can be said that sarcasm has a more negative connotation than irony [32]. Different from Irony, Sarcasm is more personal, as its purpose is evident to the participants of the conversation [33]. Furthermore, Averbeck in [34], considers that the main difference between irony and sarcasm is that irony doesn't identify a target, meanwhile sarcasm is more critical and identify a target. In other words, we can say that the sarcasm is more openly critical than irony, since it has a clear and evident trace and target. Sarcasm implies an explicit target of satire for the author of it. On the other hand, it expresses an opposite or incoherent thought in relation to the literal message.

Machines don't understand natural language as well as humans, because machines handle bits, bytes and formal languages, but not ambiguity clarified only if read with a group that shares the same macro and micro culture. So in order to surpass this, A. Joshi et al. [35] elaborated a logic representation of sarcasm in a tuple format like this (S,H,C,u,p,p').

The "S" stands for the speaker of the statement, the "H" stands for the hearer of the statement, the "C" stands for the context in where the statement has been declared, the "u" stands for the utterance, in order words would be the intention behind the declared statement, the "p" stands for the literal proposition of the statement declared, and finally the "p'" stands for the intended proposition, in other words, the true message the speaker uttered to the hearer.

2.2.1 Difference between Fake News and Sarcasm

Given the definitions aforementioned of Fake News and Sarcasm we can see why sometimes they got confused in between their identification process.

The Fake News is an intentionally written content to be shared and deviate the readers and instigate them to spread it throughout their network of contacts, in or-

der to reach even more spreaders until the agenda of the publisher has been fulfilled, like a vile collaborative system.

On the other hand the sarcasm is created to instigate and provoke, but not to spread lies and deviations, instead is to highlight an aspect of critique of an individual, an event, or situation.

Thus, we can conclude that the main difference in between those two kinds of text is their main goal, i.e. to defy or to subtly highlight, and also the sentimental variation along the sentences which composite the text provoking either an imbalance of good adjectives to ridicule, or bad adjectives to flatter in case of sarcasm, or a tone of urgency, fear, sorrow to instigate people's indignation in order to propagate that feeling until a goal is reached.

2.2.2 Bots Role in Fake News Spread

Due to the popularization of artificial intelligence and related areas of cognitive computing, the number of bots has exploded throughout the cyber space and usage of the society in general [36] [37].

Some authors argue that the creation of bots, the cognitive agents, would be more harmful to the information recovery process, due to the fact that they would intensify the propagation of misinformation, hoaxes and spams. [37]

However, we can see in [38] that this is more or less a truthful statement. As they discovered through their experiments that in fact, bots would increase the misinformation propagation indeed, but, they also would increase the true information propagation as well.

Concluding that bots, are not misinformation spreaders but, just information accelerators not favoring one type of it, but accelerating propagation of any kind of information.

2.3 Social media

Through our readings we found that most of the works use the social media and microblogs as their main source of analysis. This is due to the increasing use of social networks by everyone, like Facebook, Twitter and Google+.

Due to cultural habits the related works centered on English language use the Twitter due to its fast writing and sharing mechanics, and also due to the guarantee of delivered message towards its targets, i.e. critiques, offenses, compliments, and marketing tends to be more efficient in that media [1] [39] [40]. For foreign restricted countries such as China, the related works handle similar micro-blogs inspired by Twitter such as Weibo [41] [42] [43].

In addition, the microblogging platforms usually provide an API (Application Programming Interface) to query and consume its data. The APIs usually provides the content of the platform in structured data or plain text, thus reducing the preprocessing step that is commonly used with web crawlers used to filter the information of interest from web pages [44] [36] [45]. We can conclude that the micro-blogs can also be seen as an accelerator of information retrieval and spread.

Another reason for this is that most of newspapers are just too serious and express more a generic political opinion compared to the social networks that express individual opinions of many different users with different beliefs, contexts and cultural backgrounds. Also it is very difficult to find an expressive newspaper that diffuse rumours and fake news, as the assurance of information quality is part of a newspaper's main process.

Nowadays, the politician context is being heavily influenced by the fake news dissemination and existence to the point of some countries being lawfully prepared for such scenario, in Brazil for example, minister Luiz Fux, in a seminary said that if a Brazilian election has been biased by fake news it would be annulled. [46]

Some social media in some countries are beginning to think on new strategies of combating this, by contracting third-party enterprises to help in defining/tagging which information is fake news or not, e.g. For Facebook the strategy was to use checker agencies that monitor news and classify them as fake or not fake, specifically the agency A Lupa uses the following scale: (1) True; (2) True, however, needing more explanation; (3) Too recent to affirm anything; (4) Exaggerated; (5) Contradictory; (6) Unbearable; e (7) False. When this interviews was done, the Facebook team affirmed that this strategy lessened in 80% the Fake News "organically" generated in the US by use of similar agencies there. [47]

In the other hand WhatsApp in Brazil limited the number of messages with the same content that can be shared by the same user, is using a Artificial Intelligence to detect abuses and harass messages and like Facebook using third-party agencies to

check and classify news. Also, the WhatsApp team trained and showed the capabilities of their app to the current president candidates and their communication team in an attempt to avoid possible use of the app for fake news spread. [48]

Another aspect explored by our earlier research at[4] was the mean of spread of such verifiable and intentionally false information. We've found that is more common to find those kind of fakes in social media, because serious newspaper sites follow an investigative process before publishing anything rather than the common user of a social media, unless the news source has the intention to share fake or satirical ones.

Among the social media analyzed, the most common one for the fake news spread was Twitter. Twitter is very accessible as the users can produce information from anywhere whenever they want as long as there is internet connection; the tweets, twitter's microblog posts, are smaller in character limit so a fake news will have the same extent in terms of content as a true news (after all true news consume more characters in order to be compliant to the linguistics rules to write an informative news text), therefore being more convincing; and finally due to its popularity among users to get information about a discussion or opinion about political aspects [49] [50].

In other countries, such as Brazil, there are popular alternative social media to twitter, such as the WhatsApp and more recently Telegram, with limitations in terms of scraping data. Specially with Telegram, the advantages concern data security and privacy given its encryption model. The winner politicians for the mayor positions in the most important cities in Brazil, Sao Paulo, and Rio de Janeiro, chose Whats-App as the main mean to contact the electoral population. They use the media to disseminate the information they want to people. The information would only be viewed by the target users that rarely would care for checking the reliability, either for inertia or for technological ignorance.

For restrictive politics countries such as China, the scraping task is even harder due to the fact that they normally have their own social media, e.g. China has Weibo [4], and the subjects handled by the users from those social media are normally restricted by government also, diminishing the variety of categories to explore and eliminating some interesting ones too, e.g. Politics.

Other interesting aspect observed in the readings was that the advent of social media favored and accelerated the way information has been distributed throughout people. In fact, the similarities among peers within the same social network favors a phenomena called the Echo Chamber: "the likeness of a person spreading information

shared by his/her peers increases when they are all within the same social network bubble, i.e., a set of similar thinking users" [39] [13].

2.4 Natural Language Processing

Computers are used to 0s and 1s, and are very objective upon interpreting codes. So in order to make them understand our writings, our linguistic cues, the sentiment imbued upon words and the hidden meaning behind a set of phrases we need to translate all that first into numbers to them the development engines translate it into low level language of 0s and 1s. The area in charge of such is the Natural Language Processing allied of course with the Linguistics area.

In order to machines perceive sarcasm they need to be able to detect the critique, the sentimental imbalance in the sarcastic text and also to learn what is not sarcastic. To help us we will rely upon the sarcasm linguist theory [11] [51] [40] [1] [52] [35] that imply that sarcasm can be perceived by us humans when positive and funny phrases evolve along the narrative to negative or acidic aspect in relation to a target and vice-versa. Taking advantage of that we will need to tag parts of the discourse and quantify their sentimental charge in order to produce a numerical translation of the text to a machine interpret.

Most of the known methods to detect what is a fake information from a true statement rely upon extra factors to the discourse per say [39]. Most of those methods leverage the social topology for information labeling truthfulness and falsity based upon the information spread network [13], others rely upon social metrics of the spreaders and receivers [5], others rely upon the imagery and extra media imbued inside the information to classify what is true and what is fake, or exaggerated.

The multitask systems aforementioned are interesting and effective methods to classify what is fake from true indeed, but, rely on much data fusion from external sources than to the main object of interest, the text, causing extra computational effort and resource consumption, at least more than only processing text, even taking into consideration the usage of extra resources such as a sentimental score list.

Taking that into consideration we chose to only work with text in order to produce a more independent classifier and capable of being more generalist and easy to deploy upon new scenarios because of its lessen complexity.

2.5 Machine Learning

There are numerous classifiers on the literature. From Random Forests and Naive Bayes to SVMs, since the task of discovering if a text fragment is fake or not is a classical machine learning task of classification. Also, for every knowledge discovery process data is needed, we will need to understand which is the data and the preprocessing involved in this kind work. Therefore this section we will present the different kind of models, and preprocessing techniques used on the literature.

2.5.1 Preprocessing

Most of papers use preprocessing steps in order to either increase the tax of correct rate or to have a faster processing[53] [24] [54].

There are works that focus on automatically detect the starting point of the rumours' stream, by topologic exploration. The authors of [55], proposed an algorithm to do so and obtained good results (compared to the other ones they tested against) finding the origin of the rumour news, furthermore, they discovered key features that tended to appear on those kind of tweets and use them in future works to pre-clusterize the scrapped tweets and agilize the origin tracking process and fasten misinformation classification.

2.5.1.1 NLP Features

Many papers used sentiment analysis to classify the polarity of a news[56] [57] [58] [59] [24]. Some used sentiment lexicons, which demand a lot of human effort to build and maintain, and built a supervised learning based classifier. Some papers which use such approach of sentiment analysis as feature for final classifiers, use chain models like Hidden Markov Models or Artificial Neural Network in order to infer sentiment.

The usage of other techniques based on syntax is relatively low. Papers mainly use parsing, pos-tagging and named entity types. On the other hand, the use of semantics are more common. Many papers used lexicons as external knowledge about words, creating lists of words based on properties of interest. For example, swear words, subjective words, and sentiment lexicons. Commonly used lexicons are WordNet and Linguist Inquiry and Word Count (LIWC)

Another use of semantics on fake news detection is the use of language modelling. Some papers used n-grams as baselines for comparisons with their handcrafted features. Others used n-grams as features to their classifiers [60]. More recent papers [61] [56] used word embeddings for language modelling, mainly the ones that are constructing a classifier using unsupervised learning. Word embeddings is a family of language models, where a vocabulary is mapped to a high dimension vector. These language models assign a real-valued vector to each word in the vocabulary, with the objective that words close in meaning will be close in the vector space. One of most commonly used model is the word2vec from Mikolov et al. [62], which uses a neural network to estimate the vectors.

2.5.1.2 Social Content Features

We grouped the features found in the classifiers in sets based on the source of the feature. The first set groups features based on social media attributes (#likes, #retweets, #friends). The second set has features based on the content of the news (punctuations, word embeddings, sentiment polarity of words).

As we could see in [63], there is a preference for more classical classification algorithms that heavily focus on the linguistic aspects. But also, we can see the increased usage of new methods that aggregate different, yet on the same context, features to give better results and insights, such as Network Topology Analysis Models and Artificial Neural Networks that explore the link between users and other meta information provided by the social media predefined data structure.

Some authors propose to classify the social media entries as fakes by analysing its interaction between users. Based on this, we found interesting the proposed work [64]. Motivated by the collaborative aspect of nowadays web2.0, and by the of swarm intelligence (or collective intelligence), the authors explore how is given the process of forming a collective knowledge from interactions of social networks users, in an event they name as social swarm.

Using a german dataset from an online gamer community, they apply statistics and linguistic analysis to extract text data to pass it through a set of classical machine learning algorithms for classification, those being Näive Bayes, K-Nearest Neighbours (KNN), Decision Tree and Support Vector Machine (SVM). To counterpoint this classical analysis, they try an approach of what they define as ant algorithm.

The Ant Algorithm works much like an ant colony. The news are sprayed with

pheromones, while there is such in the vicinity of the data acquired, the algorithm operates until the pheromone evaporate, increasingly predicting and updating its error ratio, till the thread of total pheromones is totally evaporated. A much interesting and ludicrous approach of such problem. The algorithm only classify the news as Positive or Negative, however for their purposes, it is just what they wanted.

Compared to other classical methods, heuristics and algorithms, this one showed to be the best one with the lesser error rate of all. In our scenario, it could be applied to detect fake news, hoax, rumours or misinformation by modifying its classification function, as most of works that handle fake news detection depends on interaction analysis, and this new algorithm proved to be much more efficient to this task than its classical counterparts, even though its implementation would be more complex.

2.5.2 Most Used Features

For the detection of sarcastic cues in a textual writing, researches take advantage of many different complimentary features, such as, lexical, pragmatic, morphological, syntactical and semantic(emotional and sentimental) aspects, as summarized in Table 3.4.

For the lexical analysis, researchers have been concerned to identify the symbols that compose the textual data. Most of the authors in the literature focus on:

- N-gram, a contiguous sequence of n items; Count of Words; Synonyms,
- Idiomatic expressions,
- Punctuation Count and
- Metaphors.

Unigrams, bigrams, sentence length and capital letters counting have been used to preprocess the input for further input in a given machine learning based classifier [65]. Other works such as [66] [40] [12] [67] bet on counting of punctuation marks as the repeating of such marks are indicative of sarcastic cues. Also [67] include the sequence the words and punctuation, and opposition words. One of the most common strategies used by the literature when handling NLP in general is the Part of Speech Tagging, also known as, POS Tagging; also we have the sentence mapping in order to find the fragments of the sentence so we can map entities such as subject, predicates

For the Morphological Analysis, we considered the word formation and the word categories, such as, suffix, prefix, and stem of the textual data. The morphological features are mostly used as data preparation to reduce the size of the lexicon (dictionary).

For the Pragmatical Analysis, we were concerned with the proposed contextual analyses. The main goal of this task was to understand the context that embraced the sentences of the textual data. For the detection of sarcasm, the works have explored the presence of sentence constructions that includes negations, textual chains, reply threads, rhetorical questions, named entities, overstatement, parallel structure, stylistic placing, false assertions, exaggeration, word rarity, and reported speech [67] [68] [65].

For the Syntactical Analysis, we are concerned with the structure and construction rules of the sentences and its components in a given language. Among the techniques used we have count of proper names, presence of elongations, presence of interjections and ratio of adjective and adverbs. Karoui [67] also handled reflexive pronouns, adjunct adverbs and inversions.

In summary, the most used features were those from the semantic analysis, because all reviewed papers mentioned the use of such features as key indicators of sarcasm or other figurative languages. Among the semantic features we observed sentiments, emotions, presence of laughs, presence of "ToUser," presence of emoticons, and the presence of figurative languages, such as irony, comparison and sarcasm. An interesting observation is that sarcasm is defined as a function of the other figurative languages.

2.6 Machine Learning

Different from what we expected the most used models were not based upon complex neural networks, but, instead were fine-tuned with the aforementioned features simpler models such as KNN, Random Forest, Decision Trees, SVM, and Naive Bayes.

From what we read, there is a common choice of baseline techniques: SVM and Naive Bayes for all comparisons so far. Every new proposed technique, old or new, simpler or complex, is compared against an SVM or a Naive Bayes model in order to guarantee validity and trust.

The fact that there are not much research and most of them are recent is a sign that the area is still new, full of exploratory studies. The lack of robustness on the re-

Features	List
Syntatic	Count of Capital Letters; Count of POS Tags; Stem Word Tokens; Count of Pronouns; Count of Adjectives; POS Tagging; Count of Negations;
Semantic	Word Embeddings(GloVe and Gensin); Sentiment Polarity Score; Emotion Polarity Score; N-grams(N=[1,3]); Sequence of Words; Sentence Subjectivity; Contradiction Marking; Inversions;
Extra-Media	Entity Tagging; Sentiment Polarity Score; RGB Percentages; Background Tagging;

Table 2.1: The most common features used in the literature

sults impaired our analysis to check the improvement or prejudice of sarcasm usage to diminish false positive in fake news detection only by reading the available literature.

2.6.1 Naive Bayes

Based upon the probabilistic theory, this algorithm works by building feature frequency per class, then assuming the independence in between the instance's features, the algorithm tries to classify it by weighting the features against all class' possibilities, the greatest is the chosen class [69].

Computationally less costly than most of the other machine learning algorithms it is sometimes very efficient surpassing even the complex concurrence. And very well known for its success cases in Natural Language Processing feats.

2.6.2 K Nearest Neighbors

K nearest neighbors is one of the simplest machine learning algorithms. Storing the training set in memory it searches among the data points in it finding clusters of the classes passed to it, having a tolerance limit to constitute neighborhood in the K pa-

rameter, i.e. a new observation is considered neighbor of any other if within the radio of k from the existing data point [69]. Our KNN model is implemented by Python's Scikit Learn library and has been configured with the following parameter, a K of 10 neighbors.

2.6.3 Support Vector Machine

The Support Vector Machine is a machine learning algorithm that has the objective of finding a hyperplane capable of separate optimally a set of data points taking in consideration the dimensions, features, that constitute it [69]. Its main objective is to maximize the distance in between the boundaries, and to minimize the distance in between the data points and their respective boundary. Our SVM model is implemented by Python's Scikit Learn[1] library and has been configured with the following parameters, a kernel using linear function, and a decision function of one vs one(ovo).

2.6.4 Decision Tree

The decision trees are amongst the most popular classifiers families. Its principle is quite simple, the model establishes a decision structure like a tree and for each instance the tree is traversed till reaching the terminal node where is the class. Every tree is formed by many nodes. The tree always start at the root node, then its path is split forming branches, branching until the division successfully wraps a single class [69].

The split strategy varies depending on the split algorithm and data. Categorical data may generate binary partitions or even many. Numerical data tends to split by numerical operators, i.e. greater than, less than, equal, etc. Gini entropy and index help on deciding how the partioning is executed based upon the data purity degree, the purer the data, less partitions are required to reach the class.

2.6.5 Random Forest

The random forest as the name implies is an algorithm that relies upon the usage of decision trees as predictors implementing normally the bagging strategy to converge the trees' results to a better unique result [69].

[1] Scikit Learn - https://scikit-learn.org/stable/

Putting it simply, the tree-based predictors try each to better map the training set establishing rules that can divide well the dataset taking in consideration information gain and entropy, then when to applied to new observations the trees each pass that observation throughout their structure, reaching to a conclusion. Then the solutions are voted following an ensemble method voting strategy, normally bagging, then they converge to a single solution to the new observation [70].

2.6.6 Multi Layer Perceptron

The artificial neural networks are computational structures inspired in our nervous system, specifically the neurons. They try to simulate the neuron behaviour in order to perform "simple" tasks such as classification, regression, etc. Different from the human neural network capable of multitasks and wonders[69].

There are many kinds of architectures for these simulation of aeons of biological evolution, although there elements common to all of them: the input layer that is adaptable to the number of features in the dataset, it is expected to have one neuron for each; followed by one or more hidden layers that will process the data coming from the input layer, normally the number of hidden layers can be associated to the number of characteristics to be expected to be found when analyzing the instances, each hidden layer can have as many neurons as needed; and finally the output layer where the net gives out the classification, normally having as many nodes as the possible class values.

The network learns throughout its interaction with the instances of dataset, the net tries to classify the instance in its adequate class then evaluates its decision. Depending on hyperparameters such as learning rate, batch size (number of instances exposed to the net per epoch), optimizer function, and loss function the network update the weights in the neurons(perceptrons) like what our neurons do when stimulating an input socket, more than others to preserve good memories, or traumas. The knowledge in this kind of model is latent and stored in the connections established by the neural network evolution.

2.6.7 Long Short Term Memory Neural Network

Coming from the recurrent neural networks, neural nets capable of remembering arbitrary long-term dependencies across time series domains, this neural net employs the

same strategy, but, with a differential of having a forgetting unit[69].

The main problem concerning the RNN is the vanishing gradient, i.e. gradient weights that are backpropagated until tending to zero, or the exploding gradients, i.e. gradient weights that are backpropagated until they grow to infinity. The LSTM, operate at this problem by tackling the vanishing gradients as the neural network can sometimes let the gradient weights backpropagate unchanged.

Its name comes from its component units the cells, an input gate, an output gate, and a forget gateway. Like its name implies the regulation and forgetfulness of the network is done by the backpropagation and the forget gateway, i.e., the cell will remember information and perpetuate it by an arbitrary amount of time, that is when the forget gateway plays its role, making the cell not propagate the weight thus preserving it, avoiding the vanishing gradient problem.

2.7 Metrics

The metrics are the boundaries to any experiment to be run and give us satisfactory and logical results. For our experiment we chose to use the metrics that come from the confusion matrix as they are universal.

2.7.1 Confusion Matrix

Like a contingency table the confusion matrix plots the frequency of identified classes by the machine learning models. The reading of a confusion matrix is done as: Choose a class to be considered as the positive class at a time; Column and Row of identified class, is considered True Positives; Row of identified class, and Columns of others classes are considered False Negatives; Cells of negative class, and column of positive class, are the False Positives; Cells outside the row and column of positive class, are the True Negatives;

In the Figure 2.2 we can see an instance of the confusion matrix. As we can see the matrix for multiclass problem has all the predicted classes on vertical axis compared against the horizontal axis, where the cells counts the occurrences of a class estimated as j, and predicted as i.

The cells are filled out with T[Letters], where T indicate true attribution, and the

Letters indicate the respective class compared, meanwhile, the E[letter1,letter2] meaning error of class of letter2 when it should be letter1.

Confusion Matrix		Estimate		
		A	B	C
Predicted	A	TA	EBA	ECA
	B	EAB	TB	ECB
	C	EAC	EBC	TC

Figure 2.2: The Confusion Matrix Reading

For each class proper metric extraction, interactively we need to analyze each class against the rest like in what we see in Figure 2.3. In the figure we can see the TP as true positive, FN as false negatives, TN as true negatives, and FP as false positives.

Confusion Matrix (Considering A the Positive Class)		Estimate		
		A	B	C
Predicted	A	TP	FN	FN
	B	FP	TN	TN
	C	FP	TN	TN

Figure 2.3: The Confusion Matrix Reading - Considering A as the positive class.

2.7.2 Accuracy

Accuracy is the Closeness of measurements of a quantity to its expected value, i.e. how much the system gets right [71].

Calculated by the following formula

$$\frac{(TP+TN)}{(TP+TN+FP+FN)}$$

from the values extracted from the confusion matrix in Figure 2.4, considering the class A as the positive one.

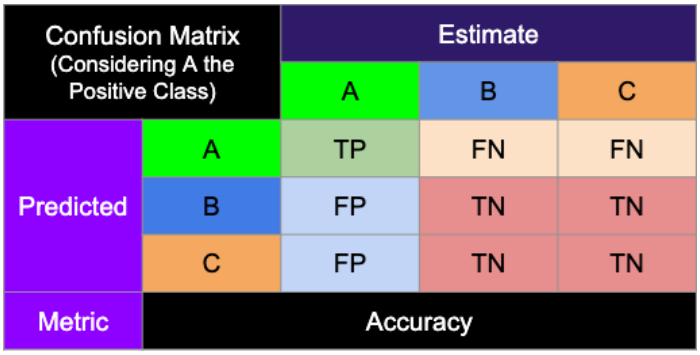

Figure 2.4: How to read accuracy in a Confusion Matrix

2.7.3 F-Meaasure

The f-measure is a classical metric used to compare machine learning models, and it is calculated by the harmonic mean between the Precision and Recall, that is why it is a preferred measurement as it brings two aspects into one combined [72].

The recall is how good the model is in detecting positive events. Meanwhile the precision is when the system is right, how "close" from each other are the right answers [72].

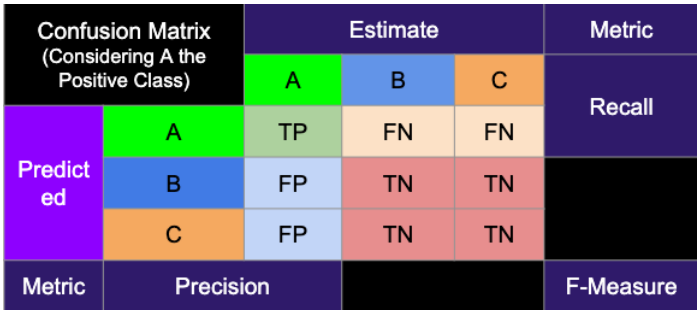

Figure 2.5: How to read f-measure in a Confusion Matrix

Based on Figure 2.5, we can calculate the Precision by

$$\frac{TP}{(TP+FP)}$$

and the Recall by

$$\frac{TP}{(TP+FN)}$$

Thus, f-measure is calculated by the following formula

$$2 * \frac{(Precision * Recall)}{(Precision + Recall)}$$

2.7.4 Sensitivity & Specificity

Sensitivity is how apt the model is to detecting events in the positive class, meanwhile Specificity is how exact the assignment to the positive class is. Both measurements are relevant to check if the machine learning model is not biased by one class in favor of others, like when kids don't learn but instead memorized it [73].

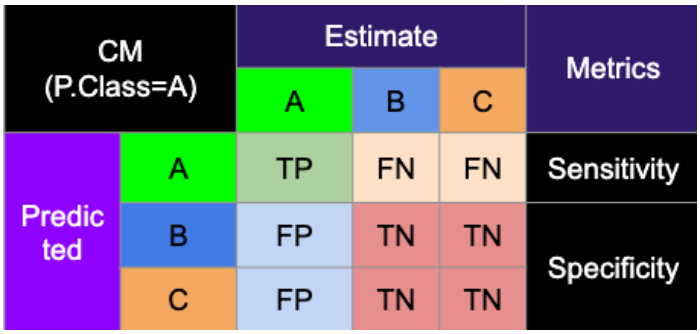

Figure 2.6: How to read specificity and sensitivity in a Confusion Matrix

Based on Figure 2.6 the sensitivity [73] can be calculated by the following formula

$$\frac{TP}{(TP+FN)}$$

, meanwhile the specificity can be calculated by

$$\frac{TN}{(FP+TN)}$$

.

Thus the combination of the two metrics can be plotted and the measurement of the area bellow it shows the coefficient of the capability of the model not only to get right answers, but, to differentiate if it is getting right by learning or by memorizing only. To this plot we attribute the name of ROC Curve, i.e. Receiver Operating Characteristics curve [74].

In order to calculate and plot the ROC Curve we need to calculate the false positive rate(FPR)

$$FPR = (1 - Specificity)$$

The Roc Curve is obtained by plotting the Sensitivity by FPR. The closest to is the area under the curve the best is the model.

Chapter 3

Related Works

We followed the systematic literature review process for software engineering (SLR method), as prescribed in [14] and [15]. We used the Parsifal tool, an online collaborative SLR tool, that allowed us to define a set of keywords, key research questions, query string, inclusion and exclusion criteria, and define the set of search sources.

We defined that our work should cover the aspects of automatic detection of fake news using sarcasm as the key aspect producing false positives. Consequently, we chose microblogs as the population research, classification task as the intervention, sarcasm as the comparison, fake news detection as the outcome, and social medias as the context.

We defined, for the search query, the following keywords and synonyms on Parsifal [1], as we can see on Table 3.1.

Parsifal is already integrated to IEEE, ACM, ScienceDirect and Springer Link digital library sources. This feature facilitates the selection phase of the literature review. Our choices were limited to what Parsifal's SLR automatic tool offered. However, the retrieved papers offered a good material both in terms of quality and quantity of retrieved papers.

The benefits offered by the search automation seem to overcome possible biases that could impair our analysis. Actually, it makes our research easily reproducible. We enriched our review with an extra source of papers coming from a Google Scholar search using the same terms.

The query generated by our chosen keywords was ("Microblog" OR "Facebook" OR

[1] https://parsif.al/

Table 3.1: Keywords used on search.

Keywords	Synonyms	Related To
Classification	Aggregation, Clusterization, Detection, Grouping, Sentiment Analysis	Intervention
Fake News	Hoax, Humor, Miscommunication, Misinformation, Rumour	Outcome
Microblog	Facebook, Reddit, Twitter	Population
Sarcasm	Ambiguity, Joke, Ridicule, Satire	Comparison

"Reddit" OR "Twitter") AND ("Classification" OR "Aggregation" OR "Clusterization" OR "Detection" OR "Grouping" OR "Sentiment Analysis") AND ("Sarcasm" OR "Ambiguity" OR "Joke" OR "Ridicule" OR "Satire") AND ("Fake News" OR "Hoax" OR "Humor" OR "Miscommunication" OR "Misinformation" OR "Rumour"), retrieving a total of 255 articles.

Our first selection criterion was the publication year. Fake news is a recent topic of interest, so we just considered papers published in the last 5 years. Older papers were only selected when they had seminal material important to understanding definitions.

Our second selection criterion concerned wording. We preferred papers that mentioned ambiguity resolution, and sarcasm identification in the context of fake news detection.

We considered three simple exclusion criteria:

- eliminate papers that did not address fake news detection,
- eliminate papers that did not relate to the sarcasm identification or differentiation from fake news, and
- eliminate papers without machine learning or NLP techniques

We have limited our collection to the set of papers to be only the ones published from 2015 to 2019. The resulting set of papers consisted of 24 really relevant papers addressing fake news detection using sarcasm as a key disambiguation tool, as described in Table 3.2.

Table 3.2: Papers by Source.

Source	Accepted	Rejected
ACM	3	8
IEEE	3	1
ScienceDirect	13	159
Springer Link	2	36
Google Scholar	3	27
Total	24	231

We can observe that this research area is recent and less explored than we think or desire. The interest in this research theme has grown as the society has evolved its usage of the web and the consequences of fake news spread has developed, as we can see in figure 3.1

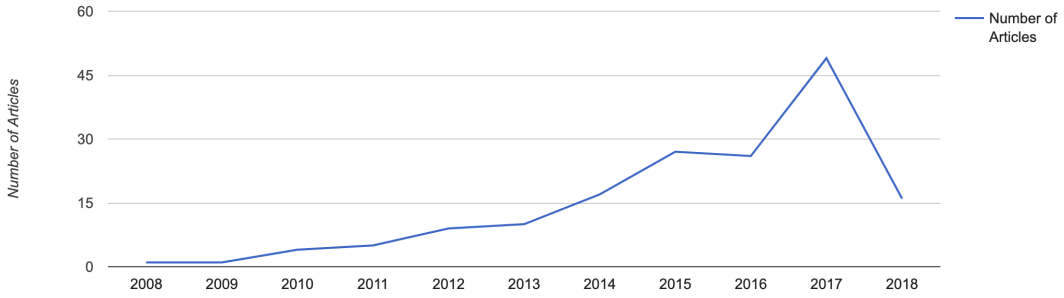

Figure 3.1: Evolution of the research theme

Having selected the papers, in figure 3.2, we analyzed them by first reading the abstract, introduction, theoretical references and conclusions in order to identify the key papers that create the fundamental pillars for the area.

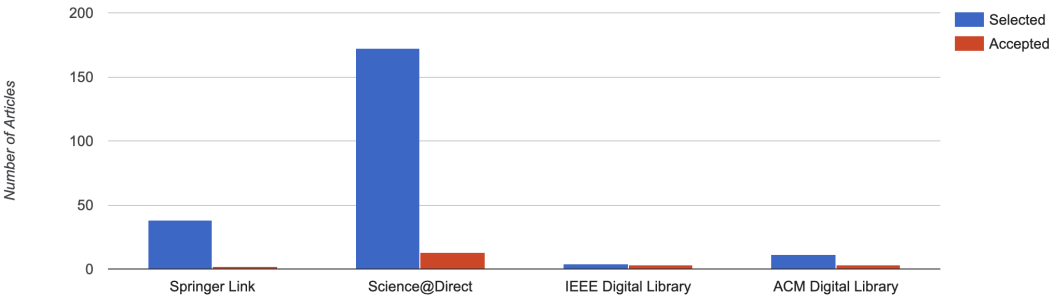

Figure 3.2: Accepted Papers from the literature

Having those most interesting ones identified, we proceed to a second deeper reading over them, in order to review their techniques, definitions, theoretical background, type of study and results.

The relevant aspects we were looking for were:

- the impact of figurative languages, such as, sarcasm, in the production of false positives on fake news detection,
- the challenges to differentiate the sarcasm related figurative languages from actual fake information,
- the methods proposed so far to address these challenges, and
- the identification of the most used machine learning method for handling fake news detection using sarcasm identification.

The definitions of sarcasm or ambiguous figurative language used by each author is important to us, as most of the times people tends to mix those in the same category, although they should not. This kind of scientific work is important to separate in niches what have been done as research in this so recent area.

3.1 Databases Used

Most of the works being in English relied upon web scraped data from known sources of Fake News to their culture in order to fill up their datasets, e.g., The Onion[2].

In the year of 2017, two challenges were proposed by the community, namely the RumorEval (SemEval 2017 Shared Task 8) and the Fake News Challenge. The former had two subtasks, one for stance detection of a reply to a news, and another for classifying the news as true or false. The latter is just a stance detection of a news, which classifies the reply of a news in agrees, disagrees, discussing and unrelated.

There are numerous sites for manual fact checking on the web. Two of the most popular are snopes.com and factcheck.org. In addition, there are specialized sites, for specialized domains like politics, like politifact.com. In contrast, there are also numerous of sites, like theonion.com, that publish news explicitly declared fake. Many of these sites are publishing these news as a satire, humorous, or as a critic. Many papers generated their dataset from these two sources. The fact checking as ground truth to true news and satire online journals as ground truth to false news.

[2]The Onion - https://www.theonion.com/

Wang provided the LIAR dataset [75], composed of statements made by public figures, annotated with its veracity, extracted from the site polifact.com For research focused on rumors, there is the PHEME dataset, by Zubiaga et al. [76]. This dataset groups a number of tweets in rumor threads, and associate them with news events.

The work focused on Potuguese Fake News from Monteiro, R. A. et al in [2] extracted Fake News manually from five different sources of fake news, acquiring 3600 Fake News, and 3600 True News. They extracted the True news automatically from G1, Folha de São Paulo and Estadão. Limiting their scrape by two years gap only due to the heavy costly manual task of labelling the Fake News themselves, somthing unimaginable for us due the short time, and lack of human resources to do so like they did it.

The work focused on Portuguese sarcasm from Carvalho, P. et al. in [77] scraped sarcastic news and comments from a known Portuguese site, although they have never declared which was in their text as source for their sarcastic dataset.

From Portuguese the most similar work to ours is the one from de Morais, J. I. et al. in [78] adopted the same dataset construction as ours, by the year of publication it is possible to have started at the same time or later than ours, they scraped Diário Pernambucano, G1 and Fake News from Lupa. They acquired also an imbalanced dataset of news, but, as far as we investigated not publicly available.

Language	Data sources	Related Works
Non Portuguese (Mostly English)	Scrape News	[5], [6], [1], [3], [39], [50]
	Scrape Microblog	[7], [20], [25], [26], [27], [39], [36], [79], [38], [41], [42], [43], [44], [55], [56], [58], [59], [61], [80]
	Crowdsourcing	[39], [49], [57], [81], [82]
	Image Focused	[21]
Portuguese	Scrape News	[8], [2], [78]
	Scrape Microblog	
	Crowdsourcing	[8], [2]
	Image Focused	

Table 3.3: Dataset Construction Strategy: Scrape News - the works built their dataset by scraping news from known sources. Scrape Microblog - the works built their dataset by scraping social medias, mostly twitter. Crowdsourcing - Works that relied upon challenge, public, datasets, or by human collaborative effort. Image Focused - Works that had image dataset.

As we can observe from table 3.3 most of the works are not only English centered, but also they focus on social media webscraping.

The Portuguese works are scarce and heavily focusing on news scraping, or relying upon heavy human manual effort to build its database, an open challenge yet to be overcome.

Although some of the works affirm they have public datasets available, we couldn't get them, only we were able to mimic their processes and compose them into our own.

In general, we can conclude that the dataset construction for this problem is a very difficult task as it is normally implemented either by manual effort, so gathering people to collaboratively help and also review the labelling, or by automatic web scraping.

The main problems of those approaches are the manual effort, the possible human bias, availability of collaborators, difference from HTML format from site to site, and finally the degrade of current web scraping logic, as the web is evolving in its writing and in short amounts of time the web pages change their HTML resources, functions, or even hide HTML content from the scrapers.

3.2 Preprocessing

The sarcastic element in the texts are relevant for such classification task. Throughout our readings, we also identified a pattern in the surveyed research methods through their experiments and model creations. This pattern was the building block for our proposed process to detect fake news differentiating from sarcasm, a generalization of this pattern can be observed in figure 3.3.

Another relevant aspect we understood from the literature review was the most used features by the research works, table 3.4. We can see that the works value most the syntactic and semantic aspects of the data content, aligned to our focus.

From the syntactic perspective the works focus on establishing numerical summaries to count instances of given part of speech tags into the textual content in order to teach the machine learning which grammatical resources are most inclined to the classes.

Most of the works count the presence of negation, meaning the usage of the particle ñ̈o, or ñ̈onb̈ecause it is pecualiar of English to use them into sarcastic or ironic statements, different from Portuguese where normally we tend to use proper full words to provoke the sentimental charge inversion.

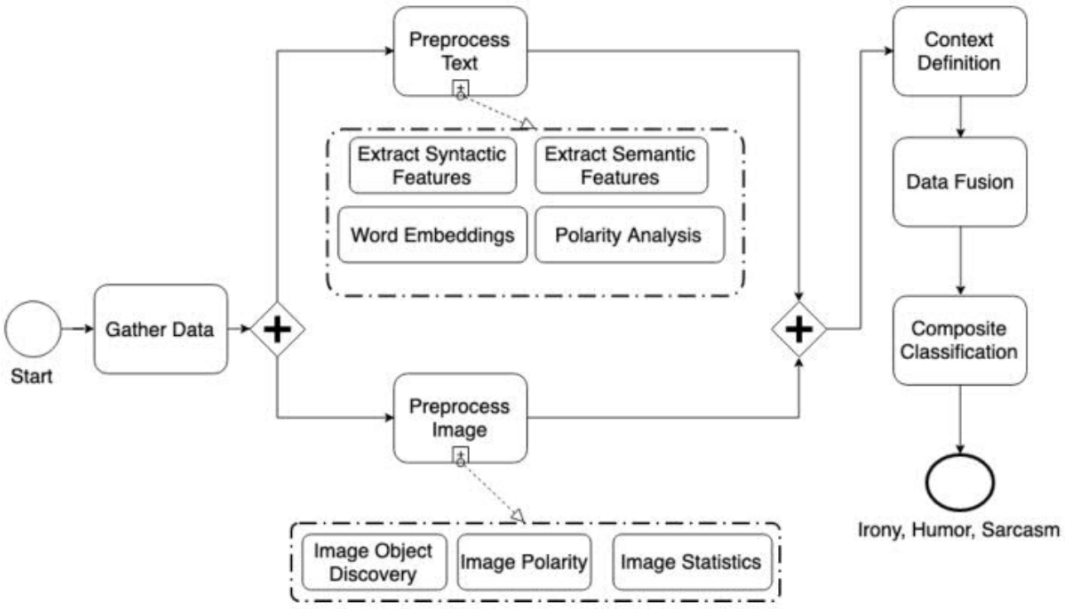

Figure 3.3: High Level, patterns observed from the literature.

The semantic aspects are the most intriguing and interesting because somehow grounded into linguistic theories and mathematics is possible to encode the implicit messages to be delivered inside the textual content to the machine learning models. Given special appraisal to the sentiment analysis and word embedding as the most used features amongst the others.

There is an aspect we don't want to focus on our research but it is worthy mentioning for future works. It is the extra media, basically the extra media are contents embedded inside the textual content of the data that is either image, video, audio or icons [4]. In the related works which use it this aspect is explained by the fact that some fake news leverage upon that kind of resource for clickbaiting(attract users attention by not relevant/related non-textual content) purposes.

3.3 Most Used Detection Techniques

The related works, as seen in 3.5, have a preference for the classical machine learning models, this is due to their higher taxes of accuracy, precision, recall, etc.

Also, we can observe by studying the related works that the models that perform better in general are the ones responsible for multiclass classification task. The cross validation when used has its k fold number set to either 5 or 10, in order to not cause overfit or bias somehow. There is a prevelance for SVM, that is why we can consider it

Features	List
Syntatic	Count of Capital Letters; Count of POS Tags; Stem Word Tokens; Count of Pronouns; Count of Adjectives; POS Tagging; Count of Negations;
Semantic	Word Embeddings(GloVe and Gensin); Sentiment Polarity Score; Emotion Polarity Score; N-grams(N=[1,3]); Sequence of Words; Sentence Subjectivity; Contradiction Marking; Inversions;
Extra-Media	Entity Tagging; Sentiment Polarity Score; RGB Percentages; Background Tagging;

Table 3.4: The most common features used in the literature

one of the main baselines of the state of the art. Another preferred model is the tree based, where we can see many works focusing on them, and there is another extrapolating this concept over the Function Tree that basically is a generic algorithm that allows to modify the data split decision function of the tree.

The most similar work to ours, [78], relied upon three of the five classical models we chose in our approach to detect fake news, and diferentiate them from sarcasm.

From our findings, the main used methods and techniques for classifying fake news in microblogs, social media or newspaper entries are here grouped in table 3.6.

3.4 Evaluation

We can conclude that the fake news research in general is very recent and is still evolving. The most used techniques of the literature rely upon classical machine learning models with NLP preprocessing steps.

One of the main challenges are still the lack of datasets and manual effort in the data acquisition process.

Work	Language	Model	Accuracy	Precision	Recall	F-Measure	Number of Classes
[1]	English	SVM	N.I.	0.88	0.82	0.87	2
[2]	Portuguese	SVM	0.89	N.I.	N.I.	0.89	2
[3]	Indonesian	NB	0.826	N.I.	N.I.	N.I.	2
[8]	Portuguese	Manual	N.A.	N.A.	N.A.	N.A.	N.A.
[27]	English	Func.Tree	0.917	0.916	0.917	0.917	3
[27]	English	NB Tree	0.9	0.9	0.89	0.9	3
[27]	English	R. Forest	0.9	0.89	0.89	0.9	3
[42]	English	SVM	0.91	0.90	0.92	0.91	2
[43]	Chinese	Dec. Tree	0.943	0.931	0.94	N.I.	2
[43]	Chinese	KNN	0.83	0.50	62.5	N.I.	2
[55]	English	Dec. Tree	N.I.	0.87	0.88	0.88	2
[56]	English	LSTM	0.78	N.I.	N.I.	0.78	2
[58]	English	SVM	0.67	N.I.	N.I.	N.I.	2
[59]	English	SVM	0.699	N.I.	N.I.	N.I.	2
[61]	English	LSTM	0.82	N.I.	N.I.	0.482	2
[80]	English	R. Forest	0.995	N.I.	N.I.	N.I.	2
[80]	English	MNB	0.21	N.I.	N.I.	N.I.	2
[81]	English	MLP	0.81	N.I.	N.I.	N.I.	2
[82]	English	NB	N.I.	0.9	0.9	0.9	2
[82]	English	Dec. Tree	N.I.	0.9	0.9	0.9	2
[82]	English	R. Forest	N.I.	0.75	0.74	0.73	2
[82]	English	KNN	N.I.	0.72	0.71	0.71	2
[82]	English	CNN	0.913	N.I.	N.I.	N.I.	2
[82]	English	LSTM	0.973	N.I.	N.I.	N.I.	2
[78]	Portuguese	KNN	0.57	N.I.	N.I.	0.57	4
[78]	Portuguese	SVM	0.58	N.I.	N.I.	0.58	4
[78]	Portuguese	R. Forest	0.72	N.I.	N.I.	0.72	4

Table 3.5: Related Works Machine Learning Approaches and their results. (N.I. = not informed by the authors) [1] uses a Cross validation of 10 k folds. [2] uses a Cross Validation of 5 k folds. [3] as the remainders, adopt the Train Test split of 70:30

Model	Category
Gaussian Naive Bayes	Baseline
K Nearest Neighbors	
Support Vector Machines	
Decision Tree	Tree-based
Random Forest	
Multi Layer Perceptron Networks	Novelty
Long Short-Term Memory Neural Network	

Table 3.6: Most used Machine Learning Model

From the Portuguese works, we have only one automated, meanwhile the other two are manual system aided process, where the intelligence behind this system is on the humans that together decide if a news is fake or not.

Most of the works rely upon social media scraping, then the second most used attempt is to rely upon news scraping.

The neural network works are still falling behind most of the classical models, mainly in the f-measure perspective.

Chapter 4

Dataset Veritas Corpus and the Sentiment Gradient

Due the lack of open datasets in Portuguese to both the fake and sarcastic news, we needed to leverage upon known popular sources of sarcastic and fake news in Brazil, those being E-Farsas' Fake News session[1] and Sensacionalista[2]. And for the True news we chose Folha de São Paulo[3], a source of true news. Also other sources such as Globo and Extra have proven to be hard to scrape upon due to inconsistencies amongst their respective HTML pages.

In order to extract the training set of our models we needed to create a web scraper for each of those sources, as each news source had its own specificities amongst their HTML page tags.

To code our scrapers we relied upon Python programming language and its renowned libraries such as Beautifulsoup(lib for webscraping)[4], NLTK(natural language toolkit)[5] and re(for regular expression)[6].

The code has been built upon Jupyter[7] development environment as a set of notebooks to be used individually in sequence. We preferred this way in order to study, and observe the results of each output of each step of our process.

[1] E-Farsas - http://www.e-farsas.com/secoes/falso-2
[2] Sensacionalista - https://www.sensacionalista.com.br/
[3] Folha de São Paulo - https://www.folha.uol.com.br/
[4] BeautifulSoup - https://www.crummy.com/software/BeautifulSoup/
[5] NLTK - https://www.nltk.org/
[6] Regular Expression - https://docs.python.org/3/library/re.html
[7] Jupyter - https://jupyter.org/

Since our focus is on textual content we extracted from those sources only the news textual content and ignored extra medias embedded into the news, e.g. videos, image and recordings. Given the textual set desired, we extracted directly from it the basic metrics like textual length, average length of sentences, etc.

The syntactic features extracted from the text most used by the state of art are: Part of Speech Tags(POS Tags), word tokenization, sentence tokenization, stemmed words, lemmatized words, punctuation count, and speech person count. Those features may indicate a causal relation of number of syntactic features that indicate presence of truthfulness, critique, or falsity, e.g., the common sense makes us think that a critique contains more adjectives due its nature of attributing characteristics to the critique target, or that truthful information has a balanced number of grammatical objects following some norms of writing.

4.1 Syntactic Features Extraction

As for the second step we performed the syntactic feature extraction, for that we relied much upon NLTK. We tokenized the news, extracted Part of Speech Tags, tokenized the sentences, and performed the counts of it having in the end for that matter: average sentence size, news length, capital words total, capital letters total, punctuation count, and the POS Tag summaries(dictionaries with Part of Speech Tag as key and the counting of its presence in the text as the values).

4.2 Semantic Features Extraction

The semantic features are the implicit information imbued into the text. The ones we chose were the special characters that most of the time are fruit of informal communication, the other special characters we focused on counting also were the presence of social markers such as '#' or ".

And finally the last and most important the sentiment charge of the news, for that we relied upon Polyglot[8] a very well known Python library[83] capable to handle different languages other than the English different from most of the Python's libraries available. Polyglot relies upon a set of lexicons that has a coverage of 95.7% agreement.

[8] Polyglot - https://polyglot.readthedocs.io/en/latest/Sentiment.html

We experimented many approaches for the sentiment analysis, from sentiment analysis on individual words to the entire document. But from literature studies we found that the most adequate approach would be doing the sentiment analysis upon sentences, and outsourcing an array of sentiment evolution, and finally the sentiment analysis on the entire news. In order to maintain metrics and understand possible variances, in the sentiment analysis we extracted also the median and average of the them.

4.3 Sentiment Gradient

One other last thing we wanted our model to be capable of is to perceive the sentimental charge imbalance that occur when we construct sarcasm [11], so we needed to somehow synthesize the array of sentiments into a single feature for our model to understand.

The closest aspect to a sentiment flow of the text is the sentiment charge by sentences we took from the semantic analysis step. Looking thoroughly at what that array of values were, we thought that was the sentiment timeline we were looking for.

Therefore we propose a novel technique of apply derivatives into the array, like what we would do into a time series, then extract the average gradient of them, so we would be able to capture the information we need, i.e. the rise, the fall and the stability of a sentimental gradient as we can see in Figure 4.1 for Fake News, and in Figure 4.2 for Sarcastic News. For this new interpretation and technique we attributed the name

of Sentiment Gradient.

Algorithm 1: Sentiment Gradient Algorithm

Result: Sentiment Gradient of the News

$sentiment\_timeseries$ = empty array;

sentence_array = SentenceTokens(News);

if *Length(sentence_array) > 1* **then**

  for *each sentence in* $sentence\_array$ **do**

    $sentiment\_rate =$

    $sentence[sentiment\_charge]/Length(sentence[tokens])$

    $sentiment\_timeseries$.append($sentiment_rate$)

  end

  return mean(getGradients($sentiment\_timeseries$))

else

  return $sentence\_array[0][sentiment\_charge]$

end

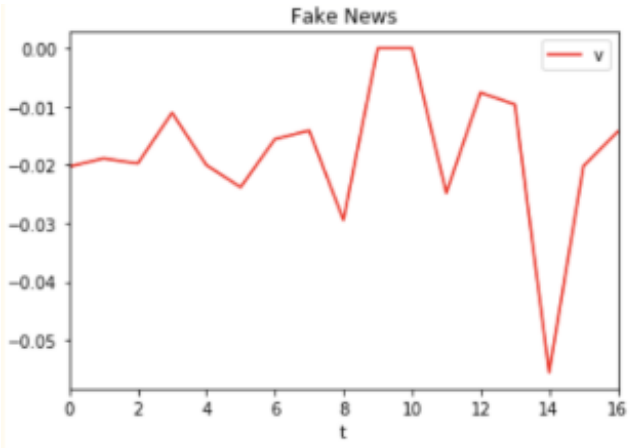

Figure 4.1: Fake News Sentiment Gradient

4.4 Word Embedding

Machine Learning algorithms are keen to numbers, not pure categorical data, therefore we need to transform the textual content of news to mathematical representations such as vectors.

In the literature we have seen many approaches to that [65] [52] [13], in our case we attempt to use the same methods, but, it generated a new vector for each word into the news, so it generated an immense overhead of data, and since we are concerned

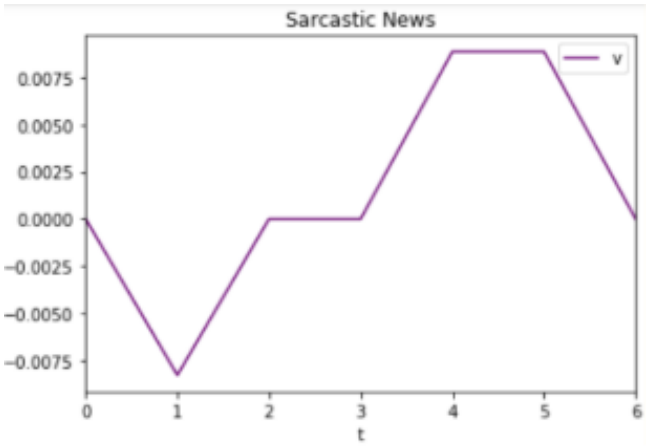

Figure 4.2: sarcastic News Sentiment Gradient

with the news as whole, this level of granulation by word or sentence wouldn't be that meaningful.

So we applied the Doc2Vec technique, similar to the Word2Vec that map each word to a n dimensional vectorial space by similarity in between the words in there and the quantity of them, but aggregated to the document level.

We started the embeddings by 50 dimensions, the default from the major of the works and the usage of word embedding models. Then we tried reducing that in order to prune the dimension of our dataset, using the Python library SelectKBest and iteractively redoing the Doc2Vec by one less dimension.

That way We tried 30 dimensions, till finally we reached 3 dimensions to embed the documents into. Therefore producing the following features into the dataset dv_0, dv_1, and dv_2, as dv stands for document vector component.

Here in figure 4.3 is the vectorial representation of our documents. We can observe that in fact the True news are very different from the Sarcastic and Fake news, and yet the sarcasm and fake news can be confused by one another.

4.5 Imbalanced Dataset

Our dataset is an imbalanced set of news of the fake, sarcastic, and true news categories to be learned by our models. In its entirety the dataset has 11179 news having the following distribution among the classes in Figure 4.4. Thinking on that constraint our data split strategy needs to consider that stratification, so we relied upon Python Scikit

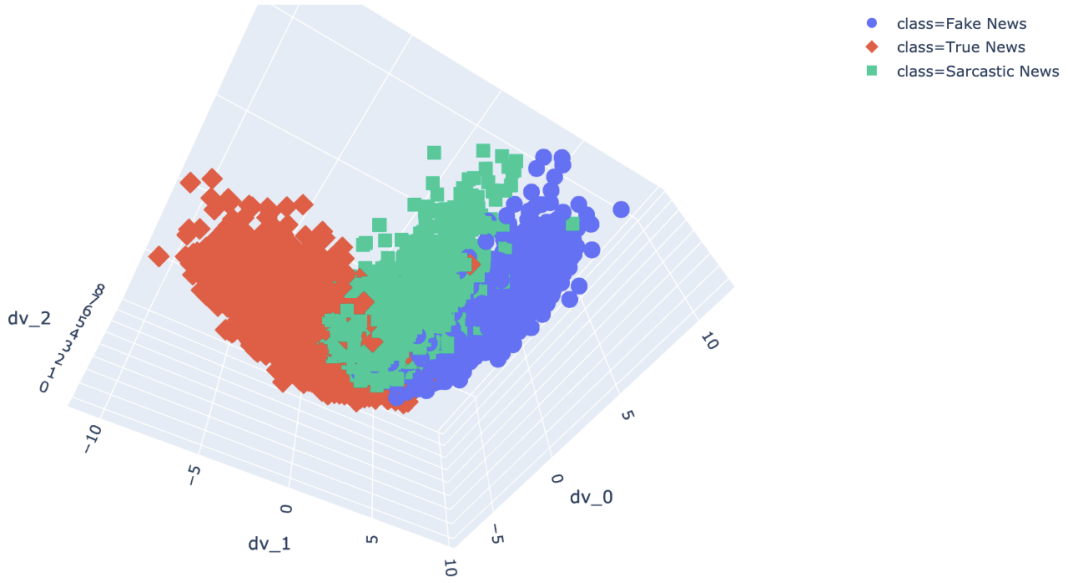

Figure 4.3: Vectorial Representation of our dataset.

Learn library's method of StratifiedShuffleSplit in order to produce the desired set of 70% of the news for training, 15% for test, and 15% for validation. The two 15% sets for test and validation were so chosen to be used as train time tuning and post train validation, respectively.

4.6 Dataset after Preprocessing and Data Fusion

Here in figure 4.5 are the attributes of the dataset after our data fusion process. By the correlation we see that there are relation in between the stylometric attributes and the sentiment analysis related ones, what makes a lot of sense by sarcasm theoretical foundation [11]. And the embedding vectors and the stylometric markers somehow relate to our class, class_codes.

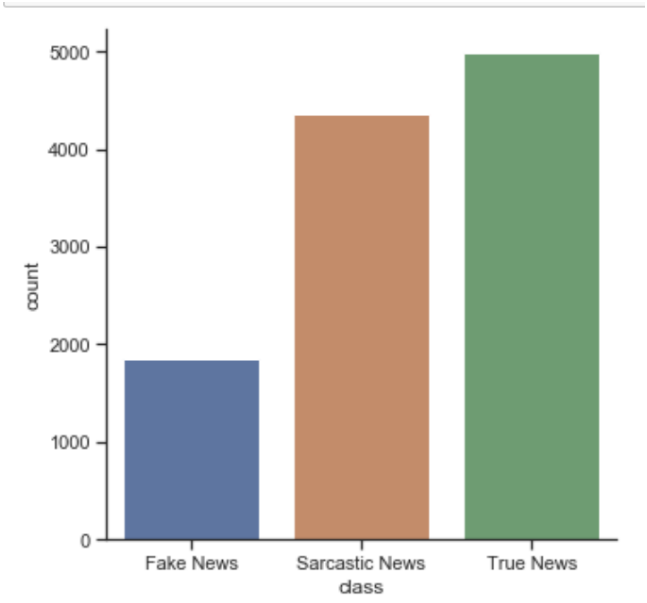

Figure 4.4: News Distribution by Class

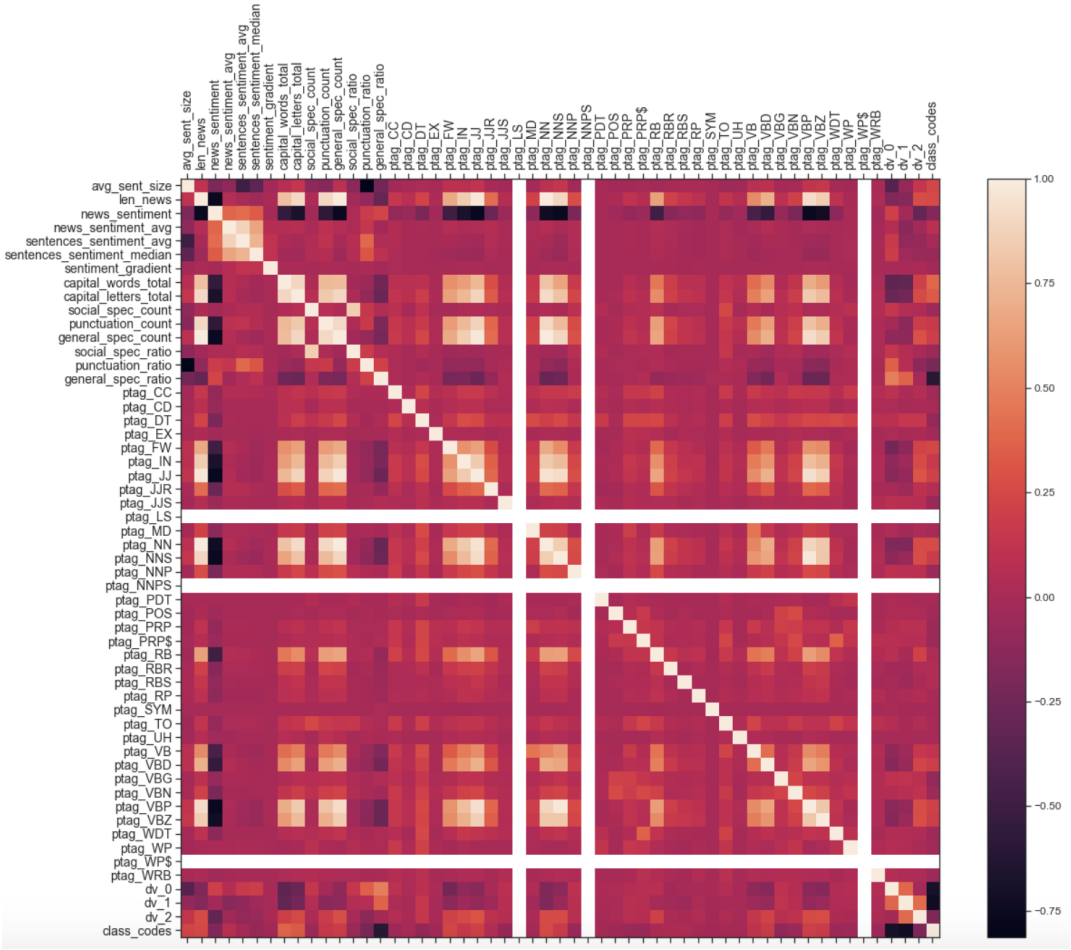

Figure 4.5: Correlation Matrix of our data.

Chapter 5

Judice Verum: Method for Fake News Detection based on polarity grading, stylometry and document embedding

In our work we chose to analyze news that can be interpreted as true, fake and sarcastic that we will differentiate each other by the Fake News definition coined by Shu et al. in [39] that is that Fake News are information that are verifiable and intentionally fake. Therefore based on such definition, the true news are the ones that are intentionally verifiable true and the sarcastic ones being the news that are not intentionally but verifiable fake.

That is the most important part of our research, we want the machine to understand what is the difference between what has been written to defy us disguised as truth, and what has not been written to defy us and is a lie used to deliver other message, basically what differentiate the three classes in our scenario is the final product delivered by the message, i.e., a true information, a true critique delivered in what would be lie if not already known as, and what allegedly true that in fact is false.

Since we are trying to find hidden meanings, subliminal messages and sentimental charge upon text we will have to explore its semantic aspects as well. The most used features by the literature [66] [84] [80] being: Word Sentimental Score, Sentence Sentimental Score, and similarity resources, i.e. word embedding.

In our experiment context what is relevant to us is the news, so the document itself, in order to better model our goal we chose the strategy of document embedding, that is the vectorization of the words of the text then the grouping of it until we reach

document level, that then finally is embedded in relation to other documents in our dataset. We mapped the steps aforementioned into the following process in Figure 5.1.

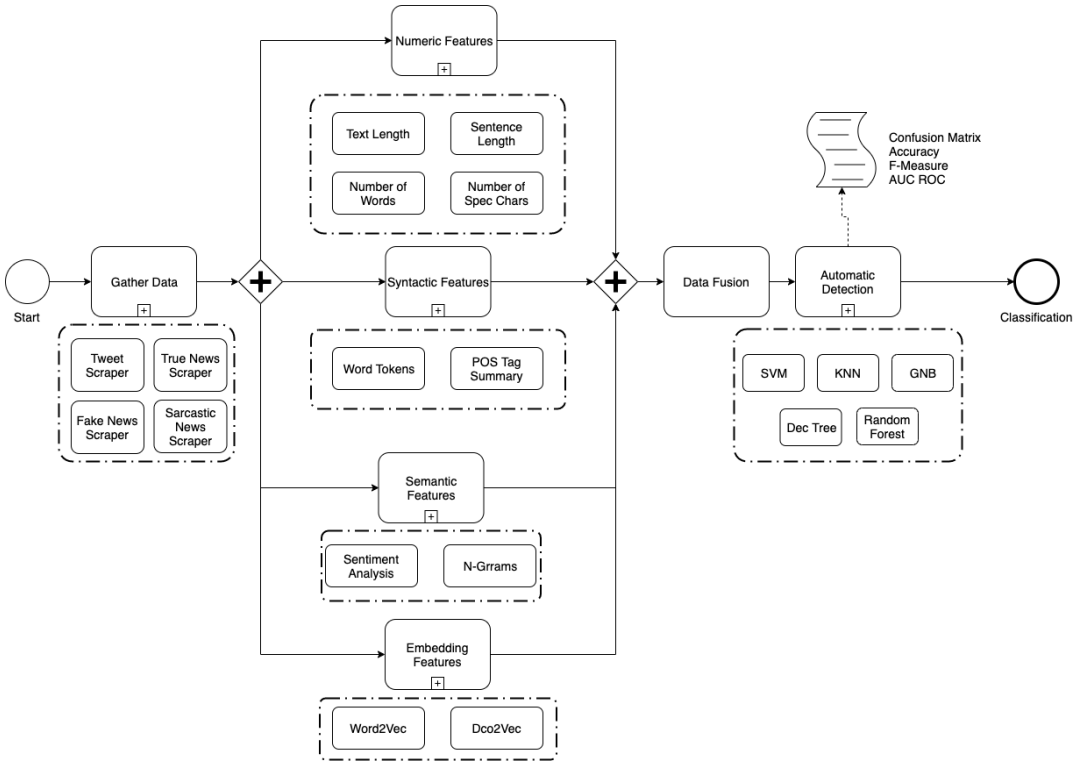

Figure 5.1: News Automatic Classification Process

5.1 Gather Data

The first step of our method is to achieve the data for our model to train upon. In order to do that, we bet on the credibility and ease access sources of the classes we need.

For the Fake news we chose E-farsas a very popular and well known platform for investigating what is false or not, but, manually like most of the current efforts in Brazilian scenario nowadays, there they have a section just for what they've truly labeled as Fake News; for the True News we chose Folha de São Paulo an impartial press known for its credibility and serious news printing in both physical and electronic medias; and Sarcastic News were extracted from a Sensacionalista, a media dedicated to satire, humour, critique to other entities throughout sarcastic writing.

For a web scraper to work we had to understand the patterns of the source HTML writing as to effectively extract the information we need, since if we apply too much

generic scrapping rules we would add more steps into data cleansing and preprocessing.

Done that preparation part we left our scrapers working, looking for news since 2016 till nowadays from the given sources. In order to store the news data and provide a proper database for future works to leverage upon, we sent everything to an online MongoDB server, a NoSQL approach adequate for semi-structured data.

Aligned to our research scope, our news are focused on sociopolitical context, as this context is more prominent to fake news occurrence.

5.2 Pre-Processing and Fusion

The preprocessing of our news data is a combination of sucessfull attempts of literature that are commonly used separated, or combined two on two, and not fully mixed. Each branch in figure 5.1 represents a different approach to preprocessing the news data.

The Numeric Features preprocessing sub-workflow is a set of steps, that can be executed in any order, to extract the basic required info about news that almost every work in literature uses. Those being: Text lenght is the step where the document metrics are taken, i.e., number of characters(text length), average text size, and median text size; Sentence length is the numeric measurement of the principal components of the documents, and where the subliminal messages are delivered[11], here we are measuring sentence length, average sentence size, and sentence median size; the number of words, is concerned by the components of sentences, and taking into consideration the same metrics; number of special characters(Spec. Chars.) is the step concerned about counting the non textual or symbol signs in the text. The literature argues that Fake News tends to abuse or at least use above normality special characters symbols, evoking sentiment and emotions, or maybe for referencing forwarding mechanisms;

The syntactic features preprocessing sub-workflow is the most basic building block in any NLP related machine learning task. Basically in this step we will try to capture the first superficial linguistic resources used to weave the messages delivered to the readers, i.e. words, nouns, adjectives, subject, conjuring agent, verbs, etc. We can also consider this step as almost a condition to the other remaining ones, as even they occur isolated, they would have implemented this step, even discarding it after. Basically the two steps in this sub-workflow are the word tokenization, and the POS Tag Summary. Although not explicit in order to perform a good word tokenization before we will need

to apply lemmatizing and stemming techniques. And for the POS Tag Summary we will extract the Parts of Speech from the texts and count each occurrence of each instance in the news, as recommended by all the related works that rely on POS Tagging.

The semantic features preprocessing sub-workflow is one of the most important sub-workflows, as the sarcasm classifiers deeply rely upon sentiment analysis. And there is also the n-gram extraction, although the n-gram extraction is recommended by most of the works that rely upon LSTMs, since we want our methodology of enriching news info to be generic for the machine learning models to receive the data we kept this step, although it prove not be relevant in feature selection. The most important step of this preprocessing sub-workflow is really the sentiment analysis that we implemented by assesing the sentiment polarity of the entire document, the sentiment of each sentence, and their statistics(average, median, and max, although max has been discarded as well by feature selection).

Finally the last preprocessing sub-workflow, the Embedding. As we progressed through the methodology we were basically enriching the news data with numeric information, that is because machine learning models by themselves don't handle textual input, and rely upon numeric representation of our data, so basically we have to be able to translate all parts of the text and its metainformation into numbers, that is valid for similarities and contextual relations in between the documents and their components(words and sentences). In order to do so the word vectorization is performed, at a high level is a frequency map relation between word tokens and their instance in the documents. Problem is that if we want to really embedding the information we need it is necessary to scale up our observation granularity, from words to sentences, and from sentences to the document itself. Thus, we will perform in this step the Doc2Vec embedding, that will provide the models the capability to be aware of the similarity in between the words and sentences of the document itself, but also how the documents relate to each other.

In the end, after every preprocessing sub-workflow has been completed we join everything together, and eliminate every non-numeric feature to feed the machine learning models in the Automatic Detection sub-workflow only the adequate data.

5.3 Data Mining

Finally the last part of the process is the automatic classification process using the new preprocessed and enriched set. For this automatic classification task we chose five classical techniques: GNB, KNN, SVM, Decision Tree, and Random Forest.

We chose those specific models due to the findings in literature of GNB and SVM being baseline models for testing of new methods, and the remainders being the most used techniques in related works of detecting fake news in English language scenario.

Although the neural network approaches are gaining much popularity in the machine learning field of research, the literature review showed that they are not yet reaching the classical models' metrics, not only that, but we need to first prove our methodology would work for the classical models before testing any newest ones, and finally from a explainable artificial intelligence standpoint, the model chosen would be much easier to explain their rationale than the sub-symbolical ones.

Finally, the output of this methodology and hybrid model configuration, we will outsource not only the classification result, but, also all the desired metrics for testing, and model evolution purposes.

Chapter 6

Experiment and Results

Our experiment consists of gathering data from different sources to constitute our dataset of fake, sarcastic and true news, then splitting it upon train, test, validation, and gold standard set, then teaching machine learning models with such data and check how well they can discern our news into the desired classes(multi-class classification problem).

6.1 Metrics

In the literature there is not a well defined choice on which should be the used metric to measure the models' efficiency for our problem resolution.

Most of the works rely upon accuracy(majorly), and f-score(lesser). But, the accuracy doesn't expose the false positive rate, not even the precision, meanwhile the f-score brings both recall and precision, still lacks the false positive rate we want to check. Another metric most of the works in the classical models approaches lack are the capability of generalization of the models normally measure by the Area Under the Curve of the ROC curve.

In order to cover all the desire aspects and guarantee the efficiency of our method, we chose to extract the confusion matrix and its metrics of the models ran during each moment of our process(each feature engineering milestone).

Also we observed the area under the ROC curve, so we would be able to cover all possible flaws should they appear, e.g., models being good on classifying only a given class not the other, i.e. generalization capability of the models, that therefore would

k	accuracy	f-measure	AUC ROC
1	86.1	80	86
2	84.7	81.5	88.4
3	86.8	84.2	89.1
4	86.8	84.5	89.3
5	87.5	84.4	89.4
10	88.9	84.7	90.78
20	88.3	83.7	89.8
100	83.9	78.6	84.1

Table 6.1: K Nearest Neighbors Parameter Tuning

indicate the generalization of our methodology.

6.2 Hyper Parameters Tuning

In order to provide the best configuration to each of our models we performed a parameter tuning run. For each model we tried different parameter configurations and measured their accuracy, f-measure, and Area Under the Curve of Receiver Operating Characteristic(AUC ROC).

For KNN the parameter tuning is simpler than the rest, we will regulate the number of K neighbors to consider a neighborhood, as we can see in the Table 6.1. The results of this parameter tuning indicates that the neighborhood of 10 elements is more adequate as the data is not so compact and intricate, but, slightly spread, similar to the behavior of the vectorial word embedding space we see in Figure 4.3.

The SVM model is the more time consuming for tuning and very computational costly as well, because its variations are mathematical functions that demand much computer processing power to test. But, the tuning is much important for this model since the wrong choice of a kernel function may cause poor performance in classification as we can see in the Table 6.2. The results of this parameter tuning explain the choice of Linear Kernel for the model of Rubin, V. et al in [1].

For the decision tree we have the max depth as the tunable parameter as we can see in the Table 6.3. It is the maximum length from the root node till the leaf node.

The random forest are much similar to the decision tree, the main differences are in

kernel	decision_function	accuracy	f-measure	AUC ROC
Linear	One vs One	95.5	94.2	96.2
Polynominal	One vs One	72.1	57.7	72.27
RBF	One vs One	77.7	58.5	75.1
Sigmoid	One vs One	57.12	52	62.47
Linear	One vs Rest	95.6	94.26	96.2
Polynominal	One vs Rest	72.1	57.8	72.2
RBF	One vs Rest	77.6	58.5	75
Sigmoid	One vs Rest	57	51.9	72

Table 6.2: Support Vector Machine Parameter Tuning

max_depth	accuracy	f-measure	AUC ROC
3	88.9	87.9	90.4
7	95	93.9	95
10	95	93.5	96

Table 6.3: Decision Tree Parameter Tuning

the number of predictors, the number of trees in the forest, and the number of features to be considered randomly by each tree. Results in Table 6.4.

6.3 Results

The results of our experiment were very satisfactory in general, and helped us on deciding which would be the best classical model to classify news into fake, sarcastic, and true categories.

As we can see in the table 6.5, our proposed process of news data fusion in fact was efficient for accuracy of our models.

We attribute this increase upon each step to the addition of the implicit information awareness acquired by the model throughout our data fusion process, i.e., by the sentiment analysis, word embedding, and stylometric mapping (through POS Tag

max_depth	features	predictors	accuracy	f-measure	AUC ROC
3	10	300	90,7	88.3	91.4
7	10	300	95	94.5	96
10	10	300	96	95.4	96.62
3	None	300	90	87.6	90.6
7	None	300	95.28	97	95.53
10	None	300	96	94.88	96.2

Table 6.4: Random Forest Parameter Tuning

Accuracy	Step1	Step2	Step3
GNB	71.37	71.61	78.41
KNN	89.68	89.98	95.58
SVM	92	91.59	96.30
Dec. Tree	88.31	88.67	88.96
R. For.	89.02	89.38	95.46

Table 6.5: Accuracy evolution - Step1: Raw Numeric Dataset; Step2: Fused Syntactic and Semantic Information; Step3: Fused all with Doc Embeddings.

Count) the models acquired the capability of understanding what were the implicit messages delivered through sarcasm, and the sentimental charge of Fake News.

Even though the good results, we want to guarantee that the other metrics are also as good and aligned in order to prove that our model is not overfit.

For this purpose we will rely upon the confusion matrix and the metrics that can be extracted from it, i.e. Precision, Recall, F-Measure, and AUC ROC.

6.3.1 Primary Results

The first metric we need to observe is the confusion matrixes extracted from our models, scored against the validation set for each milestone of our process.

The Naive Bayes in table 6.6 classified well the Sarcastic news, but had some problems on discerning the Fake ones and True too. It seems that it understood the subliminal messages delivered by sarcasm, but, struggled to differentiate the Fake news from the True news, in fact classifying many fake news as true ones.

GNB	Fake	Sarcastic	True
Fake	100	71	106
Sarcastic	19	618	16
True	32	125	590

Table 6.6: GNB Confusion Matrix - tested against 1677 observations from the validation set

The KNN in table 6.7 very well for True news, better than the Naive Bayes, but, overall it was able to classify the news well.

The Support Vector Machine in table 6.8 as expected after the hyperparameter tuning was one of the best performing models able to discern very well the sarcastic news.

The Decision Tree in table 6.9 performed greatly too, less than the SVM one, but,

KNN	Fake	Sarcastic	True
Fake	205	42	30
Sarcastic	35	598	20
True	41	41	665

Table 6.7: KNN Confusion Matrix - tested against 1677 observations from the validation set

SVM	Fake	Sarcastic	True
Fake	254	22	1
Sarcastic	24	617	12
True	3	19	725

Table 6.8: SVM Confusion Matrix - tested against 1677 observations from the validation set

was able to reach good results as well.

Dec.Tree	Fake	Sarcastic	True
Fake	250	21	6
Sarcastic	23	611	19
True	4	20	723

Table 6.9: Decision Tree Confusion Matrix - tested against 1677 observations from the validation set

The random forest in table 6.10 by far had the best performance, we assume that this can be explained due to the fact of its ensemble strategy of combining 300 different decision tress that trained upon 10 features each in order to reach a consensus of classifying the news.

Also we analyzed the evolution of the f-measure of our models along the milestones of our process, available in Figure 6.1, and can check the increase in the F-measure an harmonious mean in between Precision and Recall. The last point of observation in Figure 6.1 is the Gold Standard. As we can see, although the F-measure has fallen for some models, it fell little fluctuating along the expected measurement.

Then we can observe the AUC of ROC curves of our models in order to assure the efficiency of them, eliminating any doubt about model overfitting and non-generalization, available at table 6.11.

And in Figures 6.2, 6.3, 6.4, 6.5, and 6.6 we see the plot of the ROC curves of our models against the validation set, and as we can see the models perform well.

R. For.	Fake	Sarcastic	True
Fake	249	22	6
Sarcastic	11	635	7
True	0	18	729

Table 6.10: Random Forest Confusion Matrix - tested against 1677 observations from the validation set

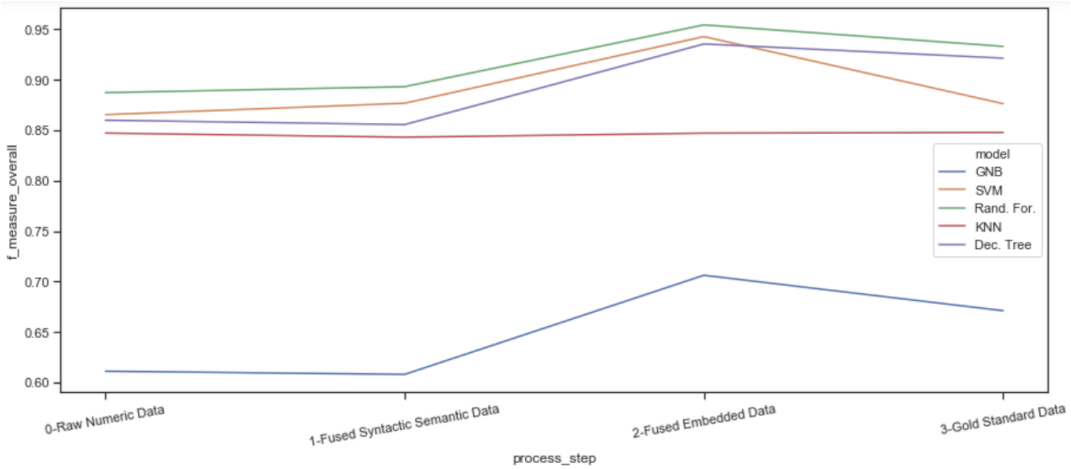

Figure 6.1: Methodology Evolution in F-Measure.

6.4 Cross Validation

Also in order to again test our method against any possibility of overfit or bias, we conducted the cross validation, assuring our models are well-trained, we performed a cross-validation test using 10 k folds(also seen in Rubin, V. et al. in [1]), table 6.12.

The results corroborated to our expectations and served well the purpose of it, assuring our models' efficiency, available in figure 6.7. We can perceive minor variance among the results and close similarity to the main run of our models.

AUC ROC	Step1	Step2	Step3
GNB	72.57	72.78	79.50
KNN	90.0	90.48	90.78
SVM	90.49	90.69	96.21
Dec. Tree	90.70	90.84	96.02
R. For.	92.62	92.18	96.62

Table 6.11: AUC ROC evolution - Step1: Raw Numeric Dataset; Step2: Fused Syntactic and Semantic Information; Step3: Fused all with Doc Embeddings.

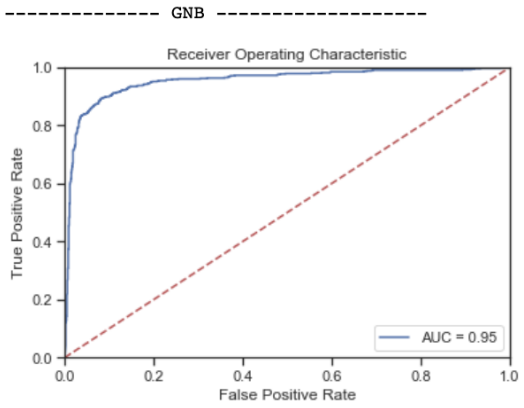

Figure 6.2: ROC Curve GNB using the Validation Set.

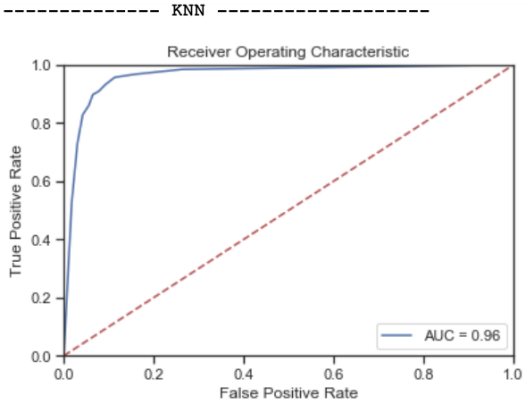

Figure 6.3: ROC Curve KNN using the Validation Set.

6.5 Gold Standard

And to simulate a pilot test and totally assure our models' efficiency we performed a test against a gold standard set, the 30 most recently published news from each class and presenting them to our models to classify, results available at tables 6.13, 6.14, 6.15, 6.16, and 6.17.

Model	K Folds	Cross Val. Score	Std. Dev.
Gaussian Naive Bayes	10	0.79	0.01
K Nearest Neighbors	10	0.88	0.009
Support Vector Machines	10	0.95	0.003
Decision Tree	10	0.95	0.006
Random Forest	10	0.96	0.005

Table 6.12: Cross-Validation Results

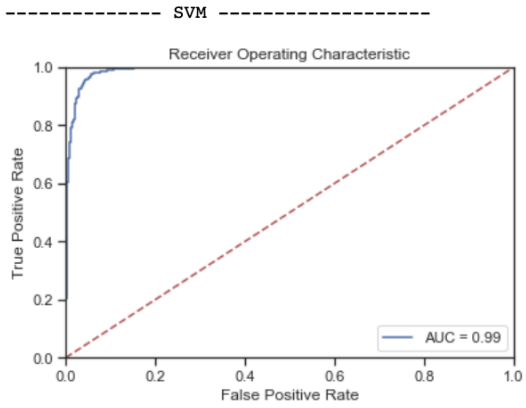

Figure 6.4: ROC Curve SVM using the Validation Set.

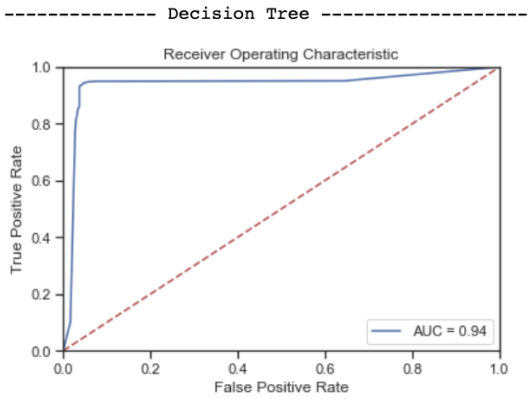

Figure 6.5: ROC Curve Decision Tree using the Validation Set.

6.6 Pilot Simulation

Sarcasm may sometimes be heavily dependent of the stylometric way the author wrote his/her text [11] [85] [86], would be interesting to test our model against another source of sarcasm, just to check. So we scraped news from Não Salvo[1], another source well known for sarcastic and satirical texts, and to our relief and joy, the models not only were able to handle the new input, but, also to clearly classify them as they should be

[1] Não Salvo - https://www.naosalvo.com.br/

GNB(Gold)	Fake	Sarcastic	True
Fake	10	8	12
Sarcastic	1	27	2
True	0	4	26

Table 6.13: Gold Standard GNB Confusion Matrix - tested against 90 observations never seen before

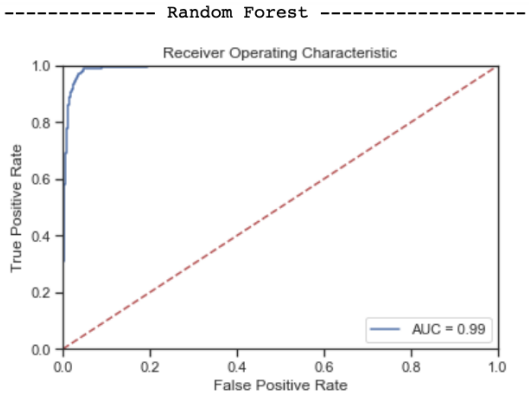

Figure 6.6: ROC Curve Random Forest using the Validation Set.

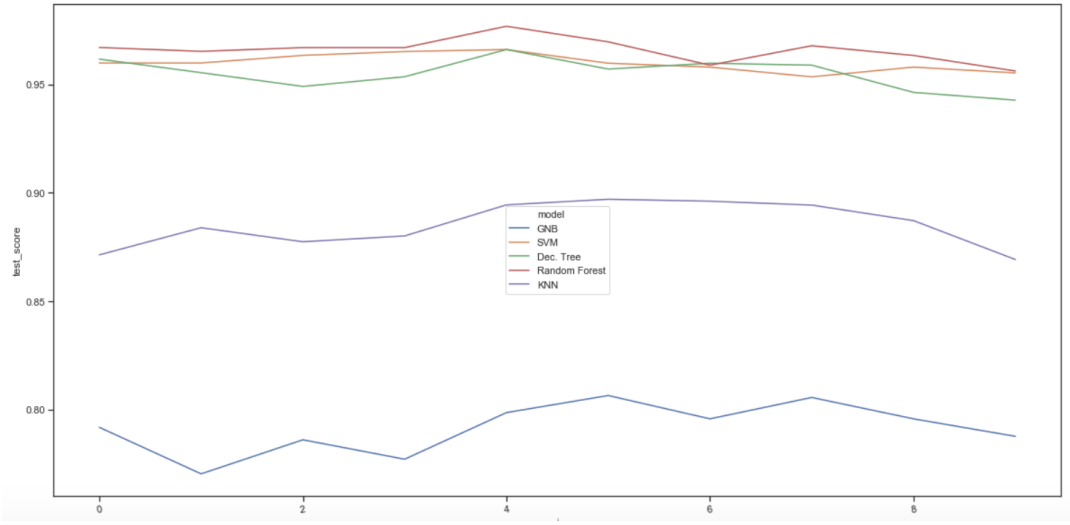

Figure 6.7: Model cross validation of 10 k folds.

sarcastic, with an accuracy of 98.9% for our best model Random Forest, and for the other models an average of 90%.

6.7 Neural Networks

The MLP has a simple structure of four fully connected hidden layers of size equal to the number of features, the first being ruled by a tanh activation function, and the remainder by relu. The final layer, or the output has a softmax activation function. The hyperparameters selected were the batch size of 100, an Adam optimizer with learning rate of 0.001 and decay of 0.000006. And Sparse Categorical Entropy Loss function.

The LSTM has a simple structure as well, of three LSTM layers of input size of num-

KNN(Gold)	Fake	Sarcastic	True
Fake	19	5	6
Sarcastic	2	28	0
True	0	0	30

Table 6.14: Gold Standard K Nearest Neighbors Confusion Matrix - tested against 90 observations never seen before

SVM(Gold)	Fake	Sarcastic	True
Fake	25	5	0
Sarcastic	4	24	2
True	0	4	26

Table 6.15: Gold Standard SVM Confusion Matrix - tested against 90 observations never seen before

ber of features(54), and fully connected in between them, and fully connected to the output softmax layer. The hyperparameters selected were the batch size of 100, an Adam optimizer with learning rate of 0.001 and decay of 0.000006. And Sparse Categorical Entropy Loss function.

Each neural network has been ran through 100 epochs, and got us the following results in table 6.18.

6.8 Hypothesis Test

With all the results above, comparing to the related works metrics reported by the authors themselves we are able to affirm that our methodology and hybrid model configuration went very well against.

For our hypothesis test we ran 1000 rounds of our models training and scoring through cross validation and built a set of distribution for each of our process' milestones much like what we did for the f-measure comparison, thus the to be compared distributions of a hypothesis test, as we want to know if there was an improvement present in the execution of our methodology.

Dec.Tree(Gold)	Fake	Sarcastic	True
Fake	25	4	1
Sarcastic	2	28	0
True	0	0	30

Table 6.16: Gold Standard Decision Tree Confusion Matrix - tested against 90 observations never seen before

R.For.(Gold)	Fake	Sarcastic	True
Fake	25	5	0
Sarcastic	1	29	0
True	0	0	30

Table 6.17: Gold Standard Random Forest Confusion Matrix - tested against 90 observations never seen before

Model	Accr.	F-Measure	Loss	Gold Accr.	Gold F-Measure	Gold Loss
MLP	93.86	92	0.1912	77.78	88.7	0.5416
LSTM	83.12	92.64	0.4297	75.56	88.77	0.6546

Table 6.18: Neural Networks Metrics

As the null hypothesis we have: The methodology accuracy scores come from the same distribution from the isolated approaches', therefore having not improved our results. And for the alternative hypothesis is that our methodology accuracy scores don't come from the same distribution from the isolated approaches', therefore having made some difference in the detection.

For step1 we consider distribution of accuracy from the first milestone, i.e. the fusion of only numeric features, the step 2 we consider the distribution of accuracy from the second milestone, i.e. the fusion of syntactic and semantic features altogether with numeric ones, and finally step 3 that is the final step where we fuse everything with the embedding vectors. Thus, our distributions come from the same population, we are comparing the distributions of metrics from step1 to step2, then from step2 to step3, to assert the method benefits. Given that we will need to apply Wilcoxon Test, an statistical hypothesis test for distributions with characteristics like ours, and to measure how an intervention upon a population has been effective applied and provided improvement(increase of mean). We chose to use an confidence interval of 90%.

As we can observe by the results from table 6.19, the results were favorable, so indeed the methodology (data fusion of sentimental, semantic, syntactic, and contextual information into the news data) is beneficial.

6.9 Experiment Implementation

In order to do our experiments we used Jupyter notebooks. The proposed organization was of four notebooks for the scrapers, one for each source, and one that would be generic, then one notebook to aggregate and preprocess the data retrieved from the scrapers. Then another one for the experiments, that basically contained the models'

Wilcoxon Test	(step2,step1)			(step3,step2)			(step3,step1)		
Model	P-value	Stats	Result	P-Value	Stats	Result	P-Value	Stats	Result
GNB	0.361	25.5	H0	0.002	55	H1	0.002	55	H1
KNN	0.547	21.5	H0	0.005	36	H1	0.020	47.5	H1
SVM	0.037	45	H1	0.002	55	H1	0.002	55	H1
Dec. Tree	0.858	17	H0	0.002	55	H1	0.002	55	H1
R. Forest	0.676	23	H0	0.002	55	H1	0.002	55	H1

Table 6.19: Statistical Tests to discard null hypothesis. Result=H0=accepts null hypothesis; H1=reject null hypothesis;

definition, the dataset preparation steps like the dataset split, the gold standard establishment and the metrics extraction functions.

The notebooks ran in specific order to gather data, preprocess, and experiment so we would be able to generate our observations and get the aforementioned results.

The only external artifact to this repository where the jupyter notebooks were is the database, a MongoDB Instance on Cloud that is fed by the preprocessing notebook's result.

The project is available at github, a version management system online well known for the community, at this link https://github.com/FernandoDurier/news-scrapers.

Chapter 7

Conclusion

7.1 Contributions

We wanted to experiment in this work the capability of machines to be able to identify what is a false information created for defying purposes(Fake News), what is not necessarily true information produced to deliver a subliminal critique(Sarcastic News), and a true factual information produced to inform(True News). By the linguistic literature we studied [11] [87] [34] we understood how the satire and sarcastic critique in fact could be confused as fake news, and that for its identification indeed there are sentimental charges imbued that could be used altogether with stylometry to better classify what is sarcastic, or what is fake.

7.1.1 Experiment Results

We attribute the success of our classifications to that capability of computing stylometry(presence of certain grammatical structures in the text), and sentimental gradient imbued into our model configurations.

As we could see by the execution and evolution of our experiment, our hybrid model is capable of discerning true, fake and sarcastic news, even passing the sanity checks of gold standard set, and the application of the model to a completely new set of sarcastic news proving that our model is capable of generalization, achieving whopping accuracy of 98.5% for Random Forest and average 90% for the remaining models.

We want to highlight the importance of parameter tuning tests since it can be a

differential factor for model performance, for example in the usage of Support Vector Machine algorithm, the wrong parameter setting may lead to abysmal results like when we tested SVM with kernels different from the linear one.

By establishing the gold standard, cross validations and stratified split not only we prevent overfitting scenario, but also, prevent data leakage in between the train, test, and validation sets.

Also by enforcing the usage of almost all statistical metrics available we established a triple way of verifying the performance of automatic fake news classifiers, as the accuracy showed how much the model was guessing correct, the f-measure showed us how much the model is precise and reliable, and the AUC ROC were there to monitor how much generalist our models were during the methodology evolution, wrapping up an important set of constraints to model evaluation for future works to leverage upon.

Above all confirmations we have observed from our reassurances, we also rejected the null hypothesis we rose in the beginning of this work with an statistical significance of 0.0056 in average, observable in table 6.19, what means that yes our methodology could improve the models' accuracy by fusing contextual, sentiment, syntactic, and semantic information into news data.

7.1.2 Comparison against Related Works

Compared to the related works our proposed model's metrics are superior in scenarios of same problem of discerning true, fake and sarcasm [78] [1], is superior or equal to related works in English domain that classify only True and Fake news [84] [88], and also superior or equal to in scenarios of Portuguese fake news and true news detection [78].

Also, different from the related works, we implemented diverse measurements to assure our models' efficiency in stratification split, gold standard set, and cross validation sanity check. And from a contribution standpoint, we see as contributions the hybrid model, the process proposed and the dataset provided for future works to the community to leverage upon.

7.1.3 Classical Models vs Neural Networks

We observed in our studies that in the literature the neural networks approaches are not necessarily better than the classical models such as SVM, Random Forest, etc. in fact, we perceived that they were sometimes outperforming, contrary to the common sense [89] [90] [91] [82].

In order to check the real problems with neural networks in this research area, we implemented a Multi Layer Perceptron(MLP) and a Long Short-Term Memory(LSTM), the most used Neural Networks in the literature.

And we experienced the aforementioned problem in our neural network tests against the classical models. Having a difference of 2 to 4% in between the accuracy metrics of the best classical models and MLP, and 10 to 15% of accuracy in between the LSTM and the best classical models, from the f-measure, in table 6.18.

Getting even worse when we apply the neural networks to the gold standard set, getting the terrible performance of worst 8% than best models' from MLP, and whopping 13% worst than the best models' from LSTM, in table 6.18.

Another relevant observation we do is that the related works that use such methods, don't rely upon the precision, recall, nor f-measure, only accuracy or ROC Curve, as we can observe in the chapter 3 on table 3.5.

7.1.4 Theoretical Standpoint

Although many researchers argue that the social media and such information obtained from its metrics, is a key-feature for election prediction, others argue that this approach is too simplistic due to the lack of certainty over the real goal of political discussion on such social medias, as many tend to be satirical and not really serious, or the lack of an algorithmic and logic formalism preliminary definitions and even arguing that the good performance/scoring of the election winners on social networks per say would not be enough to establish a causality relationship to the urn victory. [63] Also, there is a work [56] which creates an attention based ANN with textual, social and image information sources and applied it on twitter and Weibo datasets, achieving 75% accuracy.

On the social information propagation used as preprocessing step, we come to conclude that it is a very favorable approach, since it helps on identifying key-features to

be used as enrichment on classifying process, helps on finding the starting point of spread and pretag it as a rumour spreader (which proved to decrease the propagation rate from that point forward) and helps on mapping the external contextual elements from the microblogging entries.

As we reviewed, the preferred methods of handling the problem of fake news, rumours, misinformation detection is the machine learning approach, mainly, involving composite classifiers that are in fact neural networks composed by classical classification algorithms that heavily focus on lexical analysis of the entries as main features for prediction, and the usage of external contextual information (e.g. topologic distribution of microblogging entries, users profiles, social media metrics, etc.) to improve classification results as a preliminary process step of such models.

The natural language processing approaches are used on the literature more as a preliminary step than a solution per say. We are not saying that it is not relevant, we are arguing that it is more a part of the final machine learning solutions than what we expected.

About the usage of bots, we can conclude that they can be viewed as catalysts of information propagation, either for good purposes or bad ones. They don't favor a type of entry, but instead help propagating it faster due to its computational capabilities that surpass those of a human being, and due to its popularity that turned them to be easier to manufacture and easier to use and being adopted by users. Of course, there are many ways to improve their information validation characteristics in future works, but, it would demand a lot of preprocessing of those external contextual elements we saw on topologic analysis of entries.

Different from many surveys we read [92] [18], we came to conclude that the current state of art of automatic detection of fake news is of using composite network analysis approaches on the machine learning techniques choices, we came to conclude that a new more generic concept of fake news could be defined so it would ease future metamodelling of the entry object and enable better generalistic misinformation detecting agents to be manufactured.

Our research tries to reunite every possible effort invested in this area of both detecting fake news and sarcasm in order to find the best methodology of each task and elaborate an state-of-art process of using the sarcasm detection to diminish the false positive fake news. With this we hope not only to help ourselves in our future works, but, also provide some kind of help to other researchers in the area and fasten their

research process.

Through the reading of the related works we gathered from our SLR process we can see that these two research areas are too recent and full of opportunities [68] [39] [13] [4], having most of the works being established by the last 3 to 4 years.

7.2 Challenges and Limitations

Along the research some other interesting aspects appear that would be nice to be implemented, but, since they would leak the scope of our research and weren't consider also in our planning we will need to think on them as future works.

Our Work is limited to the Portuguese news, the context of mostly social/political related news, and the news/article format.

Most of the works provide superficial and not much profound definition over the figurative languages, so we had to go after linguistic and psychological references [77] [93] [11] in order to satisfy our curiosity over this subject and get better definitions.

From the systematic literature review standpoint we relied upon the sources and papers the automatic tool we used and the chosen research query provided us.

7.3 Future Works

The most interesting aspects the related works approach that we intentionally try to avoid are the extra content fused to data to provide better insights. Like most of the works leveraged upon social network topology to determine the sources of fake news and the spread path, others relied upon image processing such as analyzing the contents of images, and its metadata to check for alterations and tampering.

Another interesting also aspect also would be the "explainability" of the detection models, a theme that has no mention whatsoever in the literature at least not within the terms of Explainable Artificial Intelligence. A very needed and interesting from data protection laws, ethics, and privacy standpoints.

Novel methods such as the neural networks are prone to be used in almost every research we read in the area, but, from comparisons between related works results we checked that the neural networks are still lacking in metrics to surpass the classical

models, or classical model based systems. So would be interesting for future works to compare those approaches against one another.

And finally, would be interesting to apply our model to other contexts, like news from other categories, e.g. comics, science, economy, real state, etc., to other textual formats different from news, like microblog posts, forums, free writing, classical literature to check the expansion capability of our proposed method.

Bibliography

[1] V. Rubin, N. Conroy, Y. Chen, and S. Cornwell, "Fake news or truth? using satirical cues to detect potentially misleading news," in *Proceedings of the second workshop on computational approaches to deception detection*, pp. 7–17, 2016.

[2] R. A. Monteiro, R. L. Santos, T. A. Pardo, T. A. de Almeida, E. E. Ruiz, and O. A. Vale, "Contributions to the study of fake news in portuguese: New corpus and automatic detection results," in *International Conference on Computational Processing of the Portuguese Language*, pp. 324–334, Springer, 2018.

[3] I. Y. R. Pratiwi, R. A. Asmara, and F. Rahutomo, "Study of hoax news detection using naïve bayes classifier in Indonesian language," pp. 73–78, IEEE, Oct. 2017.

[4] F. Cardoso Durier da Silva, R. Vieira, and A. C. Garcia, "Can machines learn to detect fake news? a survey focused on social media," in *Proceedings of the 52nd Hawaii International Conference on System Sciences*, 2019.

[5] M. Maasberg, E. Ayaburi, C. Liu, and Y. Au, "Exploring the propagation of fake cyber news: An experimental approach," 2018.

[6] B. Osatuyi and J. Hughes, "A tale of two internet news platforms-real vs. fake: An elaboration likelihood model perspective," in *Proceedings of the 51st Hawaii International Conference on System Sciences*, 2018.

[7] R. Torres, N. Gerhart, and A. Negahban, "Combating fake news: An investigation of information verification behaviors on social networking sites," in *Proceedings of the 51st Hawaii International Conference on System Sciences*, 2018.

[8] M. R. Pinto, Y. O. de Lima, C. E. Barbosa, and J. M. de Souza, "Towards fact-checking through crowdsourcing," in *2019 IEEE 23rd International Conference on Computer Supported Cooperative Work in Design (CSCWD)*, pp. 494–499, IEEE, 2019.

[9] B. A. Strange, A. Duggins, W. Penny, R. J. Dolan, and K. J. Friston, "Information theory, novelty and hippocampal responses: unpredicted or unpredictable?," *Neural Networks*, vol. 18, no. 3, pp. 225–230, 2005.

[10] A. H. Tran, T. Uwano, T. Kimura, E. Hori, M. Katsuki, H. Nishijo, and T. Ono, "Dopamine d1 receptor modulates hippocampal representation plasticity to spatial novelty," *Journal of Neuroscience*, vol. 28, no. 50, pp. 13390–13400, 2008.

[11] S. Tabacaru, "Uma visão geral das teorias do humor: aplicação da incongruência e da superioridade ao sarcasmo," *Revista Eletrônica de Estudos Integrados em Discurso e Argumentação*, pp. 115–136, 2015.

[12] S. Bharti, B. Vachha, R. Pradhan, K. Babu, and S. Jena, "Sarcastic sentiment detection in tweets streamed in real time: a big data approach," *Digital Communications and Networks*, vol. 2, no. 3, pp. 108–121, 2016.

[13] M. Flintham, C. Karner, K. Bachour, H. Creswick, N. Gupta, and S. Moran, "Falling for fake news: investigating the consumption of news via social media," in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, p. 376, ACM, 2018.

[14] P. Brereton, B. A. Kitchenham, D. Budgen, M. Turner, and M. Khalil, "Lessons from applying the systematic literature review process within the software engineering domain," *Journal of systems and software*, vol. 80, no. 4, pp. 571–583, 2007.

[15] B. Kitchenham and P. Brereton, "A systematic review of systematic review process research in software engineering," *Information and software technology*, vol. 55, no. 12, pp. 2049–2075, 2013.

[16] S. Kumar and N. Shah, "False information on web and social media: A survey," *CoRR*, vol. abs/1804.08559, 2018.

[17] K. Shu, A. Sliva, S. Wang, J. Tang, and H. Liu, "Fake news detection on social media: A data mining perspective," *SIGKDD Explorations*, vol. 19, no. 1, pp. 22–36, 2017.

[18] A. Zubiaga, A. Aker, K. Bontcheva, M. Liakata, and R. Procter, "Detection and Resolution of Rumours in Social Media: A Survey," *ACM Computing Surveys*, vol. 51, pp. 1–36, Feb. 2018.

[19] S. Nieminen and L. Rapeli, "Fighting misperceptions and doubting journalists objectivity: A review of fact-checking literature," *Political Studies Review*, p. 1478929918786852, 2018.

[20] C. Boididou, S. E. Middleton, Z. Jin, S. Papadopoulos, D.-T. Dang-Nguyen, G. Boato, and Y. Kompatsiaris, "Verifying information with multimedia content on twitter," *Multimedia Tools and Applications*, vol. 77, no. 12, pp. 15545–15571, 2018.

[21] M. Huh, A. Liu, A. Owens, and A. A. Efros, "Fighting fake news: Image splice detection via learned self-consistency," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 101–117, 2018.

[22] E. C. Tandoc Jr, Z. W. Lim, and R. Ling, "Defining fake news a typology of scholarly definitions," *Digital Journalism*, pp. 1–17, 2017.

[23] L. Zheng and C. W. Tan, "A probabilistic characterization of the rumor graph boundary in rumor source detection," pp. 765–769, IEEE, July 2015.

[24] J. A. Ceron-Guzman and E. Leon-Guzman, "A Sentiment Analysis System of Spanish Tweets and Its Application in Colombia 2014 Presidential Election," pp. 250–257, IEEE, Oct. 2016.

[25] J. Radianti, S. R. Hiltz, and L. Labaka, "An Overview of Public Concerns During the Recovery Period after a Major Earthquake: Nepal Twitter Analysis," pp. 136–145, IEEE, Jan. 2016.

[26] S. Ahmed, R. Monzur, and R. Palit, "Development of a Rumor and Spam Reporting and Removal Tool for Social Media," pp. 157–163, IEEE, Dec. 2016.

[27] M. Rajdev and K. Le, "Fake and Spam Messages: Detecting Misinformation During Natural Disasters on Social Media," pp. 17–20, IEEE, Dec. 2015.

[28] A. Joshi, V. Sharma, and P. Bhattacharyya, "Harnessing context incongruity for sarcasm detection," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, vol. 2, pp. 757–762, 2015.

[29] E. Sulis, D. I. H. Farías, P. Rosso, V. Patti, and G. Ruffo, "Figurative messages and affect in twitter: Differences between# irony,# sarcasm and# not," *Knowledge-Based Systems*, vol. 108, pp. 132–143, 2016.

[30] E. Kapogianni, "The ironic operation: Revisiting the components of ironic meaning," *Journal of Pragmatics*, vol. 91, pp. 16–28, 2016.

[31] J. Mesing, D. Williams, and D. Blasko, "Sarcasm in relationships: hurtful or humorous?," *International Journal of Psychology*, vol. 47, p. 724, 2012.

[32] C. J. Lee and A. N. Katz, "The differential role of ridicule in sarcasm and irony," *Metaphor and symbol*, vol. 13, no. 1, pp. 1–15, 1998.

[33] K. Barbe, *Irony in context*, vol. 34. John Benjamins Publishing, 1995.

[34] J. M. Averbeck, "Comparisons of ironic and sarcastic arguments in terms of appropriateness and effectiveness in personal relationships," *Argumentation and advocacy*, vol. 50, no. 1, pp. 47–57, 2013.

[35] A. Joshi, P. Bhattacharyya, and M. J. Carman, "Automatic sarcasm detection: A survey," *ACM Computing Surveys (CSUR)*, vol. 50, no. 5, p. 73, 2017.

[36] V. Subrahmanian, A. Azaria, S. Durst, V. Kagan, A. Galstyan, K. Lerman, L. Zhu, E. Ferrara, A. Flammini, and F. Menczer, "The DARPA Twitter Bot Challenge," *Computer*, vol. 49, pp. 38–46, June 2016.

[37] C. Shao, G. L. Ciampaglia, O. Varol, A. Flammini, and F. Menczer, "The spread of fake news by social bots," *arXiv preprint arXiv:1707.07592*, 2017.

[38] S. Vosoughi, D. Roy, and S. Aral, "The spread of true and false news online," *Science*, vol. 359, no. 6380, pp. 1146–1151, 2018.

[39] K. Shu, A. Sliva, S. Wang, J. Tang, and H. Liu, "Fake news detection on social media: A data mining perspective," *ACM SIGKDD Explorations Newsletter*, vol. 19, no. 1, pp. 22–36, 2017.

[40] A. Kumar, S. R. Sangwan, A. Arora, A. Nayyar, M. Abdel-Basset, *et al.*, "Sarcasm detection using soft attention-based bidirectional long short-term memory model with convolution network," *IEEE Access*, vol. 7, pp. 23319–23328, 2019.

[41] Q. Wang, T. Lu, X. Ding, and N. Gu, "Think twice before reposting it: Promoting accountable behavior on Sina Weibo," pp. 463–468, IEEE, May 2014.

[42] K. Wu, S. Yang, and K. Q. Zhu, "False rumors detection on Sina Weibo by propagation structures," pp. 651–662, IEEE, Apr. 2015.

[43] G. Liang, J. Yang, and C. Xu, "Automatic rumors identification on Sina Weibo," pp. 1523–1531, IEEE, Aug. 2016.

[44] C. Buntain and J. Golbeck, "Automatically Identifying Fake News in Popular Twitter Threads," pp. 208–215, IEEE, Nov. 2017.

[45] M. D. Conover, B. Goncalves, J. Ratkiewicz, A. Flammini, and F. Menczer, "Predicting the Political Alignment of Twitter Users," pp. 192–199, IEEE, Oct. 2011.

[46] P. Felipe, "Justiça eleitoral é desafiada por fake news," *Agência Brasil*.

[47] J. Valente, "Redes sociais adotam medidas para combater fake news nas eleições," *Agência Brasil*.

[48] B. Capelas, "Whatsapp anuncia planos para tentar combater 'fake news' no brasil," *Estadão*.

[49] A. Olivieri, S. Shabani, M. Sokhn, and P. Cudré-Mauroux, "Creating task-generic features for fake news detection," in *Proceedings of the 52nd Hawaii International Conference on System Sciences*, 2019.

[50] D. Murungi, D. Yates, S. Purao, J. Yu, and R. Zhan, "Factual or believable? negotiating the boundaries of confirmation bias in online news stories," in *Proceedings of the 52nd Hawaii International Conference on System Sciences*, 2019.

[51] D. Sperber and D. Wilson, "Irony and the use-mention distinction," *Philosophy*, vol. 3, pp. 143–184, 1981.

[52] R. Schifanella, P. de Juan, J. Tetreault, and L. Cao, "Detecting sarcasm in multimodal social platforms," in *Proceedings of the 24th ACM international conference on Multimedia*, pp. 1136–1145, ACM, 2016.

[53] A. Boutet, D. Frey, R. Guerraoui, A. Jegou, and A.-M. Kermarrec, "WHATSUP: A Decentralized Instant News Recommender," pp. 741–752, IEEE, May 2013.

[54] S. Fong, S. Deb, I.-W. Chan, and P. Vijayakumar, "An event driven neural network system for evaluating public moods from online users' comments," pp. 239–243, IEEE, Feb. 2014.

[55] Sahana V P, A. R. Pias, R. Shastri, and S. Mandloi, "Automatic detection of rumoured tweets and finding its origin," pp. 607–612, IEEE, Dec. 2015.

[56] Z. Jin, J. Cao, H. Guo, Y. Zhang, and J. Luo, "Multimodal Fusion with Recurrent Neural Networks for Rumor Detection on Microblogs," pp. 795–816, ACM Press, 2017.

[57] N. Hassan, F. Arslan, C. Li, and M. Tremayne, "Toward Automated Fact-Checking: Detecting Check-worthy Factual Claims by ClaimBuster," pp. 1803–1812, ACM Press, 2017.

[58] S. Vosoughi, M. Mohsenvand, and D. Roy, "Rumor Gauge: Predicting the Veracity of Rumors on Twitter," *ACM Transactions on Knowledge Discovery from Data*, vol. 11, pp. 1–36, July 2017.

[59] J. Ross and K. Thirunarayan, "Features for Ranking Tweets Based on Credibility and Newsworthiness," pp. 18–25, IEEE, Oct. 2016.

[60] P. Dewan, S. Bagroy, and P. Kumaraguru, "Hiding in plain sight: Characterizing and detecting malicious Facebook pages," pp. 193–196, IEEE, Aug. 2016.

[61] A. P. B. Veyseh, J. Ebrahimi, D. Dou, and D. Lowd, "A Temporal Attentional Model for Rumor Stance Classification," pp. 2335–2338, ACM Press, 2017.

[62] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, pp. 3111–3119, 2013.

[63] N. J. Conroy, V. L. Rubin, and Y. Chen, "Automatic deception detection: Methods for finding fake news," in *Proceedings of the 78th ASIS&T Annual Meeting: Information Science with Impact: Research in and for the Community*, ASIST '15, (Silver Springs, MD, USA), pp. 82:1–82:4, American Society for Information Science, 2015.

[64] C. Kaiser, A. Piazza, J. Krockel, and F. Bodendorf, "Are Humans Like Ants? – Analyzing Collective Opinion Formation in Online Discussions," pp. 266–273, IEEE, Oct. 2011.

[65] P. Manjusha and C. Raseek, "Convolutional neural network based simile classification system," in *2018 International Conference on Emerging Trends and Innovations In Engineering And Technological Research (ICETIETR)*, pp. 1–5, IEEE, 2018.

[66] B. Charalampakis, D. Spathis, E. Kouslis, and K. Kermanidis, "A comparison between semi-supervised and supervised text mining techniques on detecting irony in greek political tweets," *Engineering Applications of Artificial Intelligence*, vol. 51, pp. 50–57, 2016.

[67] J. Karoui, F. B. Zitoune, and V. Moriceau, "Soukhria: Towards an irony detection system for arabic in social media," *Procedia Computer Science*, vol. 117, pp. 161–168, 2017.

[68] S. Kannangara, "Mining twitter for fine-grained political opinion polarity classification, ideology detection and sarcasm detection," in *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, pp. 751–752, ACM, 2018.

[69] F. Amaral, *Aprenda mineração de dados: teoria e prática*, vol. 1. Alta Books Editora, 2016.

[70] F. Zhang, H. Fleyeh, X. Wang, and M. Lu, "Construction site accident analysis using text mining and natural language processing techniques," *Automation in Construction*, vol. 99, pp. 238–248, 2019.

[71] K. R. McCloy, *Resource management information systems: Remote sensing, GIS and modelling*. CRC Press, 2005.

[72] L. Torgo and R. Ribeiro, "Precision and recall for regression," vol. 5808, pp. 332–346, 10 2009.

[73] H.-L. Chen, B. Yang, G. Wang, S.-J. Wang, J. Liu, and D.-Y. Liu, "Support vector machine based diagnostic system for breast cancer using swarm intelligence," *Journal of medical systems*, vol. 36, no. 4, pp. 2505–2519, 2012.

[74] J. Davis and M. Goadrich, "The relationship between precision-recall and roc curves," in *Proceedings of the 23rd international conference on Machine learning*, pp. 233–240, 2006.

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

[76] A. Zubiaga, M. Liakata, R. Procter, G. W. S. Hoi, and P. Tolmie, "Analysing how people orient to and spread rumours in social media by looking at conversational threads," *PloS one*, vol. 11, no. 3, p. e0150989, 2016.

[77] P. Carvalho, L. Sarmento, M. J. Silva, and E. De Oliveira, "Clues for detecting irony in user-generated contents: oh...!! it's so easy;-," in *Proceedings of the 1st international CIKM workshop on Topic-sentiment analysis for mass opinion*, pp. 53–56, ACM, 2009.

[78] J. I. de Morais, H. Q. Abonizio, G. M. Tavares, A. A. da Fonseca, and S. Barbon Jr, "Deciding among fake, satirical, objective and legitimate news: A multi-label classification system," in *Proceedings of the XV Brazilian Symposium on Information Systems*, p. 22, ACM, 2019.

[79] F. Menczer, "The Spread of Misinformation in Social Media," pp. 717–717, ACM Press, 2016.

[80] B. Bhutani, N. Rastogi, P. Sehgal, and A. Purwar, "Fake news detection using sentiment analysis," in *2019 Twelfth International Conference on Contemporary Computing (IC3)*, pp. 1–5, IEEE, 2019.

[81] S. Ghosh and C. Shah, "Towards automatic fake news classification," *Proceedings of the Association for Information Science and Technology*, vol. 55, no. 1, pp. 805–807, 2018.

[82] R. K. Kaliyar, "Fake news detection using a deep neural network," in *2018 4th International Conference on Computing Communication and Automation (ICCCA)*, pp. 1–7, IEEE, 2018.

[83] Y. Chen and S. Skiena, "Building sentiment lexicons for all major languages," in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Short Papers)*, pp. 383–389, 2014.

[84] D. Katsaros, G. Stavropoulos, and D. Papakostas, "Which machine learning paradigm for fake news detection?," in *IEEE/WIC/ACM International Conference on Web Intelligence*, pp. 383–387, ACM, 2019.

[85] A. Reyes, P. Rosso, and D. Buscaldi, "From humor recognition to irony detection: The figurative language of social media," *Data & Knowledge Engineering*, vol. 74, pp. 1–12, 2012.

[86] J. D. Campbell and A. N. Katz, "Are there necessary conditions for inducing a sense of sarcastic irony?," *Discourse Processes*, vol. 49, no. 6, pp. 459–480, 2012.

[87] R. W. Gibbs, "Irony in talk among friends," *Metaphor and symbol*, vol. 15, no. 1-2, pp. 5–27, 2000.

[88] P. Liu, W. Chen, G. Ou, T. Wang, D. Yang, and K. Lei, "Sarcasm detection in social media based on imbalanced classification," in *International Conference on Web-Age Information Management*, pp. 459–471, Springer, 2014.

[89] G. Gravanis, A. Vakali, K. Diamantaras, and P. Karadais, "Behind the cues: A benchmarking study for fake news detection," *Expert Systems with Applications*, vol. 128, pp. 201–213, 2019.

[90] A. Bondielli and F. Marcelloni, "A survey on fake news and rumour detection techniques," *Information Sciences*, vol. 497, pp. 38–55, 2019.

[91] A. Dubey, L. Kumar, A. Somani, A. Joshi, and P. Bhattacharyya, "when numbers matter!: Detecting sarcasm in numerical portions of text," in *Proceedings of the Tenth Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pp. 72–80, 2019.

[92] S. Kumar, M. Jiang, T. Jung, R. J. Luo, and J. Leskovec, "MIS2: Misinformation and Misbehavior Mining on the Web," pp. 799–800, ACM Press, 2018.

[93] H. H. Clark and R. J. Gerrig, "On the pretense theory of irony.," 1984.