Identifying Fake News on Social Networks Based on Natural Language Processing: Trends and Challenges

Nicollas R. de Oliveira [1,†] , **Pedro S. Pisa** [2,†] , **Martin Andreoni Lopez** [3,†] , **Dianne Scherly V. de Medeiros** [1,†] and **Diogo M. F. Mattos** [1,*,†]

1   LabGen/MídiaCom–PPGEET/TCE/IC/UFF, Universidade Federal Fluminense (UFF),
    Niterói 24210-240, Brazil; nicollas_rodrigues@id.uff.br (N.R.d.O.); diannescherly@id.uff.br (D.S.V.d.M.)
2   Solvimm, Rio de Janeiro 20090-902, Brazil; pedro.pisa@solvimm.com
3   Technology Innovation Institute (TII), Abu Dhabi 9639, UAE; martinandreoni@gmail.com
*   Correspondence: menezes@midiacom.uff.br
†   These authors contributed equally to this work.

Abstract: The epidemic spread of fake news is a side effect of the expansion of social networks to circulate news, in contrast to traditional mass media such as newspapers, magazines, radio, and television. Human inefficiency to distinguish between true and false facts exposes fake news as a threat to logical truth, democracy, journalism, and credibility in government institutions. In this paper, we survey methods for preprocessing data in natural language, vectorization, dimensionality reduction, machine learning, and quality assessment of information retrieval. We also contextualize the identification of fake news, and we discuss research initiatives and opportunities.

Keywords: fake news; machine learning; natural language processing; social networking

1. Introduction

Veracity of information is an essential part of its integrity. The combat against fake news makes indissoluble the integrity and veracity checking of social networks' information and data consumption in the application layer. The disclosure of fake content implies a waste of processing and network resources. Further, it consists of a serious threat to information integrity and credibility of the provided service [1]. Hence, the sharing of untrue information concerns the Quality of Trust (QoT) applied to the news dissemination [2], referring to how much a user trusts the content of a particular source.

In different countries, it is possible to observe low levels of trust in the mass media, e.g., only 40% in the United States (available at https://news.gallup.com/poll/185927/americans-trust-media-remains-historical-low.aspx), whereas never-read links are highly shared (blindshares), e.g., 59% in the United Kingdom. In 2016, during the United States' presidential elections, American society witnessed an alarming fake news epidemic, which had a multilateral effect. A similar effect also happened in the Brazilian elections in 2018. Due to its potential of dissemination, acceptance, and destruction [3], fake news is currently one of the greatest threats to the concept of logical truth, having a high potential for deteriorating democracy, journalism, justice, and even economy [4,5]. The economy, in particular, had to deal with fluctuations of 130 billion on the stock exchange as a result of a false statement claiming that Barack Obama had been injured in an explosion (available at https://www.forbes.com/sites/kenrapoza/2017/02/26/can-fake-news-impact-the-stock-market/#559102f12fac). In this context, there is a growing joint effort by the academic community to develop approaches that are capable of analyzing, detecting and intervening in the actuation of these misleading contents. Scientific evidence has already revealed the vulnerability of humans to distinguish true from false. On average, human are correct 54% and, thus, our ability to identify fake and legitimate news is almost random [4–7].

In this paper, we aim to present the main algorithms and techniques that assist in linguistic characterization and detection of false news on social networks to guarantee the

information's integrity. The paper characterizes the phenomenon [8,9], investigates the spread on social media, and presents tools and algorithms for detecting fake news. The key factor driving the widespread of fake news is that it is created and published online, more quickly and cheaply than traditional media outlets like newspapers or television. Thus, in this paper, although identifying fake news can be carried out manually by journalism professionals, we focus on automatic identification through computational apparatus. Automatic identification follows different approaches, such as automatic proofing of logical statements through facts already known, analysis of news spread on social networks, analysis of the profile of users who share the news, or natural language processing to extract knowledge stylistic-computational approach [4]. Our methodology considers a well-known natural language processing pipeline [10], and we survey traditional and straightforward algorithms. We review the most prevalent algorithms for performing each step of the information retrieving process while applying a stylistic–computational approach. The paper's scope is limited to the stylistic–computational approach based on natural language processing, as the consumption of data by users on social networks is restricted to information that reaches the end-user using natural language. The user does not access the content dissemination models or the users' reputation models of whom share the consumed content. The paper also presents the quality metrics used in the extraction of knowledge.

The key contributions of this paper are: (i) the definition of fake news in contrast with correlated false-content pieces of information; (ii) the categorization of the traditional processes of fake news identification, eliciting the main dataset and used features to characterize the fake news; (iii) the discussion about the main vectorization schemes for converting natural language data into mathematically operable data; and (iv) the listing of research opportunities and initiatives on fake news detection.

2. Fake News Definition

The fake news term originally refers to false and often sensationalist information disseminated under the guise of relevant news. However, this term's use has evolved and is now considered synonymous with the spread of false information on social media [11]. It is noteworthy that, according to Google Trends, the "fake news" term reached significant popularity in Brazil between the years 2017 and 2018, having its peak of popularity in October 2018, when there was the presidential election in Brazil (available at https://trends. google.com.br/trends/explore?date=all&geo=BR&q=fake%20news).

Fake news is defined as news that is intentionally and demonstrably false [4], or as any information presented as news that is factually incorrect and designed to mislead the news consumer into believing it to be true [12]. Sharma et al. argue that these definitions, however, are restricted by the type of information or the intention of deception and, therefore, do not capture the broad scope of the current use. Thus, Sharma et al. define the term as news or messages published and propagated through the media, containing false information, regardless of the means and reasons behind it [11]. Despite the lack of a clear consensus on the concept of fake news, the most accepted formal definition interprets news as intentionally and verifiably false. Regarding this definition, two aspects stand out: intention and authenticity. The first aspect concerns the dishonest intention of deceiving the reader. The second, on the other hand, relates to the possibility of this false information being verified.

Fake news can be distinguished by the means employed to distort information. The news content can be completely fake, entirely manufactured to deceive the consumer, or it can be tricky content that employs misleading information to address a particular topic. There is also the possibility of imposing content that simulates genuine sources but, in fact, the sources are false. Other fraudulent characteristics of fake news content are the use of manipulated content, such as headlines and images that are not in accordance with the content conveyed, or the contextualization of the fake news with legitimate elements and content but in a false context.

Fake news also has different motives or intentions, such as intentions to harm or discredit people or institutions; profit intentions to generate financial gains by increasing the placement and viewing of online publications; intentions to influence and manipulate public opinion; as well as intentions to promote discord or, simply, for fun are identified as motivations for the creation and dissemination of fake news.

Several concepts compete and overlap with the concept of fake news. A synthesis of these multiple concepts, which are not considered fake news, are listed as follows [4,8,13,14]:

1. **Satires and parodies** have embedded humorous content, using sarcasms and ironies. It is feasible to have its deceptive character identified;
2. **Rumors** that do not originate from news events, but are publicly accepted;
3. **Conspiracy theories**, which are not easily verifiable as true or false;
4. **Spams**, commonly described as unwanted messages, mainly e-mail, spams are any advertising campaign that reaches readers via social media without being wanted;
5. **Scams and hoaxes**, which are motivated just for fun or to trick targeted individuals;
6. **Clickbaits** use miniature images, or sensationalist headlines, in the process of convincing users to access and share dubious content. Clickbait is more like a type of false advertising;
7. **Misinformation**, that is created involuntarily, without a specific origin or intention to mislead the reader;
8. **Disinformation**, which is pieces of information created with the specific intention of confusing the reader.

The characteristics of each of these types of fraudulent content are compared to the fake news in Table 1.

Table 1. Fake news-related terms and concepts.

	Authenticity	**Intention**	**Reported as News**
Satires and Parodies	False	Not Bad	No
Rumors	Unknown	Unknown	Unknown
Conspiracy Theories	Unknown	Unknown	No
Spam	Possibly True	Bad/Advertising	No
Scams and Hoaxes	False	Not Bad	No
Clickbait	Possibly True	Advertising	No
Disinformation	False	Unknown	Unknown
Misinformation	False	Bad	Unknown

2.1. Fake News Characterization

The growth of communications mediated by social media is one of the main factors that encourage the change of characteristics in current fake news [11]. An individual's inability to accurately discern fake news from the legitimate news leads to continued sharing and belief in false information on social media [4–7]. It is difficult for an individual to differentiate between what is true and what is false while being overwhelmed with misleading information that is received over and over again. Furthermore, individuals tend to trust fake news because there is currently public disbelief in relation to traditional communication media. Additionally, the fake news is often shared by friends or confirms prior knowledge, which, for the individual, is more reliable than the discredited mass media. In this context, the identification of fake news is more critical compared to other types of information, since it is usually presented with elements that imbue it with authenticity and objectivity, thus making it relatively easier to obtain the public's trust.

Social media and collaborative information sharing on online platforms also encourage the spread of fake news, an effect called the echo chamber effect [15]. The naive realism, in which individuals tend to believe more easily in information that is aligned with their points of view, the confirmation bias, in which individuals seek and prefer to receive information that confirms their existing points of view, and the theory of normative influence, in which individuals choose to share and consume socially safe options as a preference for acceptance and affirmation in a social group, are important factors in the perception and sharing of fake news that foster the effect of the echo chamber [15]. These concepts imply the need for individuals to seek, consume and share information that is in line with their views and ideologies. As a consequence, individuals tend to form connections with ideologically similar individuals. In a complementary way, social network recommendation algorithms tend to personalize content recommendations that meet the preferences of an individual or group. These behaviors lead to the formation of echo chambers and filter bubbles, in which individuals are less exposed to conflicting points of view and are isolated in their own information bubble [11,16]. The confinement of fake news in echo chambers, or information bubbles, tends to increase the survival and dissemination of such news. This is because the confinement incurs in the phenomenon of social credibility, which suggests that people's perception of the credibility of information increases if others also perceive it as true, since there is a tendency for individuals to consider information to which they are submitted repeatedly as true [7].

The spreading patterns of fake news on social media have often been studied to identify the characteristics of fake news that help discriminate between fake and legitimate news. The problem of identifying fake news can be defined in several ways. The classification can be seen as the execution of a binary classification between false or true, rumor or not, hoax or not. Another way to define the problem is how to perform a classification of several classes, true, almost true, partially true, mainly false or false, or as an unverified rumor, true rumor, false rumor or not rumor [17]. The main difference between the definition of the classification problem is due to the different annotation schemes or application contexts in different datasets. Typically, datasets are collected from annotated statements on fact-checking web sites, such as Politifact (available at https://www.politifact.com/), Full Fact (available at https://fullfact.org/), Volksverpetzer (available at https://www.volksverpetzer.de/) and *Agência Lupa* (available in Portuguese at https://piaui.folha.uol.com.br/lupa/). These sites reflect the labeling scheme used by the specific fact-checking organization.

Sharma et al. identify three characteristics that are relevant to the identification of fake news: the sources, or promoters of the news; the content of the information; and the user's response when receiving the news on social networks [11]. The source, or promoters of the news have a major influence on the rating of the truthfulness of the news. However, Sharma et al. highlight that the lists of possible sources of fake news are not exhaustive, and that the domains used to spread the news can be falsified [11]. In addition, it is important to emphasize that social networks are also populated by bots, which are fake or compromised accounts controlled by humans or programs to present and promote information on social networks. Such bots are responsible for accelerating the speed of propagation of true and false information almost equally, aiming to leverage the credibility and reputation of bot accounts [18] accounts. The second important feature is the content of the spread information. The content is one of the main characteristics to be analyzed to classify the news as true or false. Oliveira et al. identify that fake news and legitimate news dissemination in Brazil behave statistically differently according to the sum of the relative frequency of the words used in the content. Fake news tends to use fewer relevant words than legitimate news [1]. Other textual characteristics include the use of social words, self-references, statements of denial, complaints and generalizing items, and there is a tendency for fake news to have less cognitive complexity, less exclusive words, more negative emotion words and more action words [11]. Finally, user responses on social media provide auxiliary information for detecting fake news. User response is important

for identification because, in addition to propagation patterns, user responses are more difficult to manipulate than the content of the information. In addition, sometimes user responses contain obvious information about the truth [4]. User engagement, in the form of likes, sharing, responses or comments, contains information that is captured in the structure of propagation trees that indicate the path of the information flow. Such information is included in the form of temporal information in timestamps, textual information in user comments and profile information of the user involved in the engagement [11].

The characterization of the information source, propagation and content, and of the user's response allows to define different techniques of fake news identification. For instance, the identification can be based on feedback from the propagation pattern, on the natural language processing applied to the content of messages and application of machine learning mechanisms and, finally, on the user intervention. This paper focuses on solutions based on the analysis of news content.

2.2. Fake News Spreading Process

Several entities, individuals, and organizations interact to disseminate, moderate and consume fake news on social networks. Due to the plurality of actors involved, the problem of identifying and mitigating the spread of fake news becomes even more complicated. The dissemination of fake news heavily relies on social media to the detriment of traditional media, due to the large scale, the reach of social media, and the ability to share content collaboratively. Social media websites have become the most popular form of fake news dissemination due to the increasing ease of access and popularization of computer-mediated communication and Internet access [19]. Concurrently, while in traditional journalism media, the responsibility of creating content remains with the journalist and the writing organization, moderation on social networks varies widely. Each social media is subjected to different moderation rules and content regulation. Information is consumed mainly by the general public or society, which constitutes an increasing number of social media users. The growth in the consumption of information through social media increases the risk of fake news causing widespread damage [11].

Sharma et al. highlight three different actors in the spread of fake news: the adversary, the fact-checker, and the susceptible user [11]. The adversaries are malicious individuals or organizations that often pose as ordinary social network users using bot or real accounts [18]. Adversaries can either act as a source or as a promoter of fake news. These social network accounts also act in groups by propagating sets of fake news. The fact-checker consists of a set of various fact verification organizations, which seek to expose or confirm the news that generates doubts about its veracity. Checking the veracity of the news often relies on fact-checking journalism that depends on human verification. However, there are automated technological solutions that aim to detect fake news for companies and consumers. These solutions assign credit scores to web content using artificial intelligence. Finally, the susceptible user consists of the social network user who receives the questionable content but is not able to distinguish between fake or legitimate news and, thus, ends up propagating the fake news on the user's own social network, even if there is no intention to contribute to the proliferation of fraudulent content.
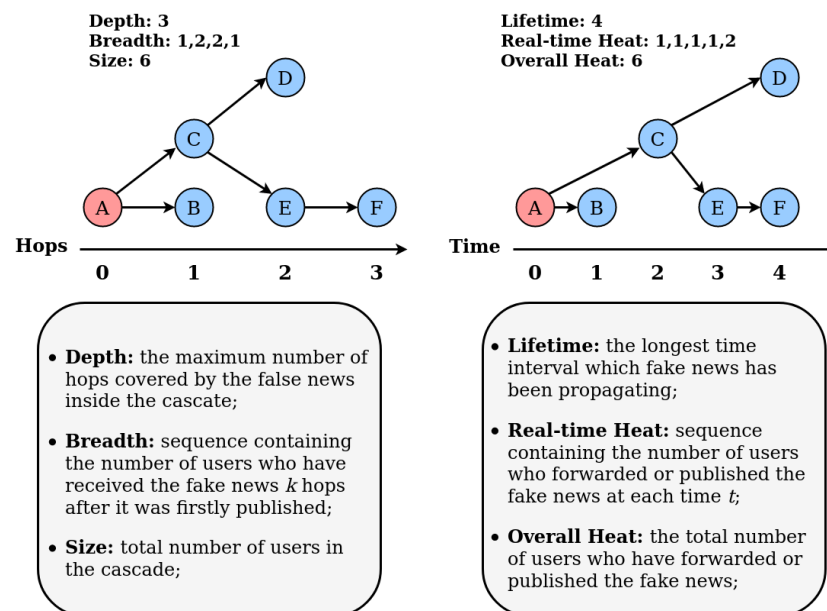
3. Traditional Methods of Detecting Fake News

The identification of fake news can be carried out by manual means, through professionals in journalism. This approach is the most commonly used but it is not compatible with the current volume of content creation and dissemination on social networks. To counteract this scalability problem, automatic methods generally integrate techniques of Information Retrieval, Natural Language Processing (NLP) and Machine Learning in the process of verifying the veracity of news transmitted throughout the Internet.

Automatic methods for fake news detection can be distinguished when discretizing the forms of detection by actuation focus. In the literature, three major analytical theories are envisaged and they are potentially useful in containing the spread of fake news. The

first theory follows an propagation-based analysis, whose focus is on the qualitative, or quantitative mapping of the spread of fake news on social networks, based on empirical patterns or mathematical modeling, respectively. The basis of both mappings is the cascade of fake news, a tree structure that represents the entire process of fake news dissemination. The cascade can be guided by either a hop-based or a time-based perspective, as depicted in Figure 1.

Depth: 3
Breadth: 1,2,2,1
Size: 6

Lifetime: 4
Real-time Heat: 1,1,1,1,2
Overall Heat: 6

- **Depth:** the maximum number of hops covered by the false news inside the cascate;

- **Breadth:** sequence containing the number of users who have received the fake news *k* hops after it was firstly published;

- **Size:** total number of users in the cascade;

- **Lifetime:** the longest time interval which fake news has been propagating;

- **Real-time Heat:** sequence containing the number of users who forwarded or published the fake news at each time *t*;

- **Overall Heat:** the total number of users who have forwarded or published the fake news;

Figure 1. Cascades of fake news in a hop-based (**left**) and time-based (**right**) perspective. The root node *A*, in both perspectives, represents the first user to publish or create the fake news, while the remaining nodes represent users that actively forward or share the fake content. Adapted from [4].

Kwon et al. mapped the propagation pattern of fake news, revealing a tendency for unconfirmed news to exhibit multiple and periodic peaks of discussion throughout the day on Twitter, while confirmed news featured just a prominent peak [20]. In addition, the works of Zhou et al. and Vosoughi et al. warn about the ability of fake news to spread faster, farther and more widely than legitimate news, especially in the political scenario. This conclusion is based on the behavior of the cascade representation of fake news, which achieves a max-breadth, depth and size, more quickly than the cascade representation of legitimate news [3,21].

Although useful, the discovery of empirical patterns of the characteristic propagation of each type of news is a strategy with temporary results due to the high dynamics and behavior variability of fake news. Hence, the joint application of mathematical modeling is convenient. In general, this modeling is based on a regression analysis using classic models, such as the epidemic and the economic.

The mathematical construction of the dissemination of fake news through epidemic modeling aims mainly at the prediction of the number of disseminators (general temperature). This modeling strategy begins with a step that associates each user with one of three states: (i) disseminator; (ii) potential disseminator; and (iii) repentant disseminator. The repentant disseminators are those who delete the post after forwarding or publishing fake news. At this stage, there is also the initial definition of the transition rates between these states. The next step is the construction of the model, which can consider phenomena such as the backfire effect, and the reflection of Semmelweis. The backfire effect is related to the fact that individuals reject more strongly evidence opposing their beliefs. In turn, the reflection of Semmelweis refers to the tendency of individuals to reject new evidence because it contradicts their established norms and beliefs. Hence, these phenomena reveal

individuals' rejection of ideas contrary to their own. The third step is to determine the real transition rates between states [4].

Economic modeling introduces a rational approach to fake news interactions, which attempts to capture and predict the behavior of individuals when exposed to fake news. In this type of modeling, the news generation and consumption cycle are seen as a strategy game between two players, publishers and consumers. For each player, the decision to forward or delete a false news implies pairs of specific and exclusive advantages among themselves. Publishers have the choice between obtaining a short-term advantage (g_p), which maximizes the profit related to the number of consumers reached, or a long-term advantage (b_p), which privileges their reputation, making them an authentic source of news. For consumers, the consequences of this dual decision are divided between an information advantage (g_c), which allows obtaining true and unbiased information, or a psychological advantage (b_c), linked to the confirmatory bias theory that reflects his preference for receiving news that satisfies previous opinions and social needs. In this way, when $g_p > b_p$ and $g_c > b_c$ a chain favorable to the spreading of fake news is built [13].

The user-based analysis considers the role of the user in the dissemination of news, consequently distinguishing a malicious user from those without bad intentions, the naive ones. Whether motivated by monetary or non-monetary benefits, the performance of malicious users on social networks occurs through accounts that hide the real identity of the manager. When analyzing the level of human participation in the management process of these accounts, it can be divided into three categories: social bots, cyborgs and trolls. All of these highly active and partisan malicious accounts have a single purpose of becoming powerful sources of proliferation of fake news. At a low level of human dependency, social bots are accounts controlled by a computer algorithm, the purpose of which is to produce content automatically and interact with humans or other bots. At an intermediate level, cyborgs are accounts that alternate between automated and human activities. Usually, this type of malicious account is registered by a human user, thus providing a camouflage to define automated programs to perform activities on social networks. At the highest level of dependency, trolls are accounts entirely held by real human users that aim to disrupt online communities and provoke an emotional response from consumers [13].

Other works, such as Barreto et al., propose a methodology capable of distinguishing legitimate users and spammers considering the two-neighborhood in Twitter. The proposal is subdivided into three stages, the first of which is the manual pre-selection of possible users. As a criterion for pre-selecting a malicious user, the fact that the user sends messages containing at least one popular topic is used. The second stage includes the collection of data from the network around the pre-selected users. As a final step, the data are analyzed through the evaluation of metrics such as degree distribution, degree centrality, grouping coefficient and PageRank. The authors conclude that spammers have a different behavior of degree distribution, contrary to the expected power law for legitimate users [22].

Even unintentionally, ordinary users are just as likely to become spreaders of fake news as malicious users. In addition to the low ability to detect fake news, normal users are influenced by psychological and social factors. In psychology, these factors are identified as individual vulnerabilities whose one known example is naive realism. This vulnerability formulates a tendency for users to believe that their perceptions of reality are the only points of view, while the others are considered uninformed, irrational or biased. Considering the social field, the dissemination of false news is closely connected to the social dynamics of individuals, being correlated to three theories: (i) prospecting theory, which describes decision making as a process by which individuals make choices based on relative gains and losses compared to its current state; (ii) the social identity theory, which associates the self-concept of individuals, is derived from the perception of belonging to a relevant social group; and (iii) the Normative Influence Theory, in which it emphasizes that acceptance and social affirmation are essential for an individual's identity and self-esteem, making users choose to be "socially safe" [13].

Table 2. Fake news datasets available.

	Content	Quantity	Labeling	Annotator
Buzzface [23]	Social media posts and comments (Facebook)	2263	Categorized on four levels (predominantly true, predominantly false, mix of true and false and no factual content)	Previously checked by news agencies (Buzzfeed)
FAKENEWSNET [15]	Whole articles	23,921	Binary (true or false)	Previously checked by news agencies (PolitiFact and GossipCop)
Fake.Br Corpus [24]	Whole articles	7200	Binary (true or false)	Considers the credibility of the source
LIAR [5]	Short statements (political)	≈12,800	Categorized on six levels (true, predominantly true, half-true, almost true, false, pants-fire)	Previously checked by news agencies (PolitiFact)
Emergent [25]	Related statements and titles	300	Binary (true or false)	Journalistic team
FEVER [26]	Short statements (Wikipedia)	≈185,000	Categorized on three levels (supported, disproved and not enough information)	Trained human annotators
CREDBANK [27]	Social network posts (Twitter)	≈60,000,000	Vector with 30 dimensions containing variable scores at five levels of veracity	Crowd-sourcing
BuzzfeedNews	Social network posts (Facebook)	2282	Categorized on four levels	Journalistic team
BuzzFeed-Webis [28]	Social network posts (Facebook)	1687	Categorized on four levels	Previously checked by news agencies (Buzzfeed)
PHEME [29]	Social media posts (Twitter)	330	Binary (true or false)	Journalistic team and crowd-sourcing

Although the existence of fake news precedes the emergence of social media, its advent has altered and expanded the dynamics of the propagation of fraudulent information, and included new actors in the scenario. Another current factor that facilitates the dissemination of this type of news is the phenomenon of social bubble or echo chamber in which users tend to relate virtually to their like-minders, that is, people who think like them. Two main ideas are present in these social bubbles, the first being known as social credibility. This idea is explained by the fact that people are more likely to consider a source as credible if others also consider it so, especially when there is no way to prove it. The second idea refers to a frequency heuristic, according to which consumers naturally prefer news that is heard more constantly, even if it is false [13].

A third analytical theory refers to the style-based analysis of writing, whose main focus is on the content of the news, that is, the text itself. This analysis starts from the premise that fake news has unique writing profiles, different from their legitimate peers. It is, then, up to the detection methods aligned with this theory to apply techniques for extracting linguistic characteristics.

Among the studies related to the stylistic approach, we highlight the one presented by Rashkin et al. The authors work under the hypothesis that fake news tends to contain a more interesting narrative in order to attract readers [30]. Thus, using a corpus (linguis-

tically, a corpus is a collection of documents on a given topic. A set of corpus is called corpora) composed of news articles of different intentions, sources and discrete degrees of veracity, the method employed uses the extraction of latent lexical features. The analysis of these features allows us to formulate different news profiles depending on their source. Thus, it seems that news from reliable sources usually present some form of concrete basis, such as numerical comparisons and expressions related to money. Conversely, news from less reliable sources had a higher incidence of first and second person pronouns, superlatives, adverbs of mode and words that express hesitation (hedging words).

4. Construction of the Dataset

The characterization of the news identification as a classification problem implies the construction of an adequate dataset. The construction of a dataset with quality and availability is the mainstay of any automatic mechanism for detecting fake news. The importance of the dataset is linked to the need to store the maximum number of contrasting examples, false and legitimate news, to be absorbed by machine learning algorithms [31]. Table 2 contains a compilation of fake news datasets available, both in English and in Portuguese.

In this context, an eventual erroneous data collection has the potential to cause innumerable negative consequences, which vary from the particularization of the analysis, to the obtaining of dissonant results. Therefore, it is prudent to adopt some guidelines as suggested by Rubin et al. for the formation of a corpus of fake news [8]. Rubin et al. argue that any construction of a dataset, corpus, of fake news must adhere to nine important conditions, listed below. (i) Considering both false and true instances allows any predictive methods applied to the dataset to consider patterns characteristic of each type of news. (ii) The information should preferably be in textual format, instead of being presented as media, in audio or video format. Information in these formats must be transcribed, making it manipulable by natural language processing tools. (iii) The homogeneity of the news in terms of size, and (iv) as to the way of writing, there are two other conditions to be considered, avoiding, whenever possible, very different instances. Equally, there is a concern with (v) the form of distribution of the news, since there are suspicions that by knowing how and in what context it was provided, e.g., humorous, or sensational, one can influence readers. (vi) The acquisition of news from the same time interval is a key factor, as the subjects can vary dramatically in a short period. Additionally, (vii) it is advisable to meet some pragmatic aspects, such as copyright costs, availability, ease of obtaining, and privacy of the writers. One should not neglect the (viii) language and (ix) culture to which the collected data belong, as the translation may imply ambiguities or misinterpretations, negatively affecting the efficiency of the detection processes [8,32].

5. Natural Language Processing

Natural Language Processing (NLP), also known as computational linguistics, consolidates itself as a field of research that involves computational models and processes to solve practical problems for understanding and manipulating human languages. Regardless of its form of manifestation, textual or speech, natural language is understood as any form of daily communication between humans. This definition excludes programming languages and mathematical notations, considered to be artificial languages. Natural languages are constantly changing, making it difficult to establish explicit rules for computers [33–35].

Table 3. Characteristics used in each approach to detect fake news based on natural language processing.

Type	Features	Zhou et al. [36]	Fuller et al. [16]	Afroz et al. [37]	Hauch et al. [38]	Monteiro et al. [24]	Rashkin et al. [30]	Rubin et al. [7]
Quantity	Character or token count	x		x		x		
	Word count		x	x	x	x		
	Sentence count	x	x	x	x	x		
	Verb count	x	x		x		x	
	Nominal phrase count [1]	x						
	Noun count					x		
	Stopword count					x		x
	Adjective count					x		x
	Modifier count [2]	x	x	x		x	x	x
Informality	Typographical error ratio	x			x	x		
Complexity	Average of characters per word	x	x	x	x			
	Average of words per sentence	x	x	x	x	x		
	Average of clauses per sentence	x						
	Average of punctuation signs per sentence	x	x	x		x		
Uncertainty	% of modal verbs	x	x	x	x	x	x	
	% of terms that indicate certainty [3]	x	x	x	x		x	
	% of terms that indicate generalization		x			x	x	
	% terms that indicate tendency		x	x	x			
	% of quantifier numbers [4]				x	x	x	
	# of interrogation marks				x			
Non-immediacy	% of passive voice	x	x		x	x		
	Pronouns in the 1st singular person	x	x	x	x	x	x	x
	Pronouns in the 1st plural person	x	x	x	x	x	x	x
	Pronouns in the 2nd or 3rd plural person	x	x	x	x	x		x
Diversity	Lexical diversity: % unique words	x	x	x	x			
	Redundancy: % of function words [5]	x	x	x	x			
	% of content words [6]	x	x		x			
	Random named entities [7]							x
Feelings	% of positive words	x	x	x	x	x		
	% of negative words	x	x	x	x	x		x
	# of exclamation marks				x			
	Humorous / sarcastic content							x

[1] phrases whose cores are nouns. [2] adjectives and adverbs. [3] e.g. "never", "always". [4] intensity adverbs. [5] words with little meaning attached, used to express grammatical relationships between words or to specify the speaker's attitude or mood, e.g., prepositions, pronouns, auxiliary verbs, conjunctions, and articles. [6] words that contain semantic content, e.g., nouns, verbs, adjectives, and most adverbs. [7] presence of proper names, never mentioned in the text before.

In a refined decomposition, NLP can be divided into five primary stages of analysis, which allow the meaning intended by the author to be extracted computationally from a textual document. The five stages are segmentation by tokenization, lexical analysis, syntactic analysis, semantic analysis, and pragmatic analysis. Although it is more consistent with a pre-processing stage, the first stage is the segmentation by tokenization. The tokenization is a mandatory technique since textual documents in natural language are usually composed of long, complicated, and malformed sentences. The next stage is the lexical analysis, which aims to relate the morphological variants to their lemmas, i.e., the primitive form of the words in the dictionary. The third stage is the syntactic analysis, which focuses on the relationship between words, each assuming its structural role in sentences, and how phrases can be part of others, constituting sentences. The semantic analysis constitutes the fourth stage. Linguistically, the semantic analysis attempts to distill the meaning of words, fixed expressions, entire sentences and is thus often applied in resolving ambiguities. Finally, the fifth stage is the pragmatic analysis, which seeks to understand a particular sentence, observing pronominal references and the textual coherence of the structure of the adjacent sentences. Although NLP may introduce other stages of analysis, such as emotion recognition, these five basic stages are sufficient to extract contextualized semantic information from a natural language document [39].

Considering the processing up to the stage of morphological analysis, i.e., up to NLP's first and second stages, it is possible to compose a basic sequence of NLP techniques to ensure the identification, and subsequent removal, of any textual noise that could compromise the extraction and intelligent interpretation of the information contained in each sentence. A sequence of techniques used to perform segmentation and lexical analysis are illustrated in Figure 2. In the sequence, data cleaning and shaping techniques are applied including tokenization, removal of punctuation and special characters, elimination of stopwords, spelling correction, recognition of named entities and stemization or lematization. Guided by this ordering, each sentence of the original text is first subjected to a discretization procedure as shown in Step 1, known as tokenization. In this case, using the space character as a bounding criterion, tokenization transforms each contiguous sentence into a list of tokens, allowing the individual handling of tokens. Basically, each token is seen as an instance of a string. In Step 2, orthographic features such as punctuation, e.g., periods, exclamation and question marks, and special characters, e.g., numbers, dollar sign and asterisk, are removed from each token.

In Step 3, stopwords, or more frequent words, such as connectors, articles, and pronouns, are eliminated. This particular task is based on the principle that the higher the frequency of a word in the corpus, the less relevant information the word has. In Step 4, the spelling is corrected by comparing the token with its closest correspondent in the dictionary. Such a procedure is performed by calculating the Levenshtein distance, i.e., the minimum number of operations required to transform a name in the dataset, into another name contained in a dictionary of names. The recognition of named entities, Step 5, mainly identifies proper names, with subsequent removal of these names. Finally, to reduce unnecessary processing caused by possible redundancies between words, either by inflections or derivations, it is common to adopt Step 6a or 6b, with the following being stemming and lemmatization. In the task of lemmatization, we try to eliminate the possible variants or plurals of the same word, reducing them to the same lemma, known as the dictionary form. On the other hand, in stemization this reduction is made by transforming each word into its radical [40–42].

Expanding textual processing to other linguistic stages, there are NLP techniques that perform the task of syntactic analysis in different degrees of complexity. At a basic level, the tagging Part-Of-Speech (POS) is characterized as a technique that returns only the lowest layer of the analysis tree, i.e., grammatical markup. Thus, each sentence word is assigned a metadata, identifying its grammatical class and conjugation. At an intermediate level, the chunking technique, also called surface analysis, is a technique that analyzes whole sentences, first identifying the constituent parts of the sentences (nouns, verbs, adjectives)

and then linking them to higher-order units with discrete grammatical meaning. Through this technique, it is to select specific syntactic structures as nominal and verbal phrases [10].
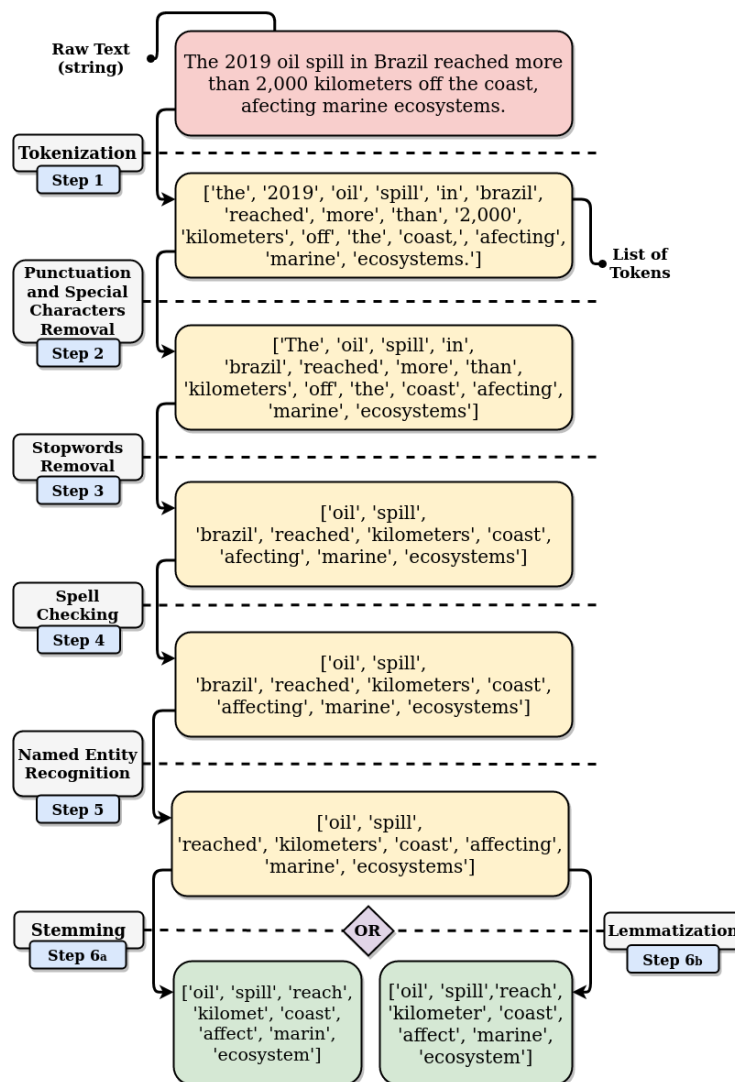
Figure 2. Application of natural language processing in raw text. The tokenization segments contiguous text into a set of tokens. Elements of small semantic relevance are removed, as well as punctuation, special characters, and stopwords. Named entities are identified and removed. Stemization or lemmatization reduces the diversity of tokens.

Sentiment analysis, or opinion mining, inspects the provided text and identifies the dominant attitude or emotion in the text through a degree of polarity, classifying it as positive, negative, or neutral. Another property commonly associated with sentiment analysis is subjectivity, which allows differentiating phrases with a high incidence of opinion, judgment, or emotion from phrases with factual information. Typically, sentence feeling classification works by considering words in isolation, assigning positive points to positive words and negative points to negative words, and then summarizing those points. The simplicity of this logic disregards the order of words, resulting relevant semantic losses [43]. Current online models consider sentence structure and construct the representation of entire sentences. Thus, these models calculate the sentiment based on how the words in the sentence make up the meaning of long phrases.

Currently, among the most powerful tools for extracting knowledge from texts, Stanford CoreNLP (available at https://stanfordnlp.github.io/CoreNLP/.) and NLTK (avail-

able at https://www.nltk.org/) are the best-known tools. Other tools, such as Linguistic Inquiry and Word Count (LIWC) [44] stands out as a textual analysis software capable of analyzing and quantifying emotional, cognitive, and structural components present in the texts. LIWC's ability to reveal latent characteristics of a text is closely dependent on the language of the word dictionary associated with the software. Although originally optimized for the English language, the LIWC dictionary has now been translated into Portuguese [45]. These tools are also useful in extracting features like those seen in Table 3.

6. Vector Representation of Texts

Even if properly standardized, each sentence is not liable to be mathematically operated, since it is still composed of radical words and non-measurable values. It is noteworthy that until this moment, the operations carried out on the data are carried out in character strings. However, for the calculation of machine learning models, data that can be operated mathematically are needed . To obtain a numerical representation, the Vector Space Model is used. This model defines that texts, whether sentences or documents, can be interpreted as a vector space of words, in which each word can be represented in different patterns, such as the binary, Bag-of-Words, Term Frequency-Inverse Document Frequency (TF-IDF). To illustrate the particularities of each vectorization pattern, we consider the corpus in Table 4, which is formed by a collection of four documents, each containing only a single sentence. Due to the uniqueness in the number of sentences adopted in the corpus example, the following descriptions show the possible vector representations at the document level and not at the sentence level, although this is equally feasible.

Table 4. Sample corpus. A toy example for vectorization techniques.

Document 1 (D1)	First sentence of corpus
Document 2 (D2)	The second sentence is short
Document 3 (D3)	The third sentence is short
Document 4 (D4)	The forth sentence is the biggest of corpus

6.1. Binary Vector Space Model

The Binary Vector Space Model consists of the most intuitive vectorization model, in which each word is assigned a value of 1 or 0 according to its presence or absence in the sentence. Although simple, it is possible to see from Table 5 that this pattern of representation is poor from a semantic point of view since it does not provide any information about the importance of a term for the set of texts. However, this representation model is quite useful for techniques that apply filters to data in natural language, as it allows the creation of binary comparison masks. In addition, this representation model requires just a few computational resources for its implementation.

Table 5. Vector representation of the sample corpus shown in Table 4 using the binary model.

Terms	first	forth	the	corpus	short	of	biggest	second	sentence	third	is
D1	1	0	0	1	0	1	0	0	1	0	0
D2	0	0	1	0	1	0	0	1	1	0	1
D3	0	0	1	0	1	0	0	0	0	1	1
D4	0	1	1	1	0	1	1	0	1	0	1

6.2. Vector Space Model of Bag-of-Words

The Bag-of-Words (BoW) model, is characterized as a type of vector model that assigns weights to terms, corresponding to the number of observed occurrences of the terms in the text. Mathematically, the vectors of this representation are expressed according to the equation

$$V_D = [w_1, w_2, ..., w_{n-1}, w_n], \tag{1}$$

where V_D is the weight vector w for each sentence in the document D up to the n-th term.

Table 6 highlights the presence of a weight equal to 2 in the last row of the column referring to the term "a". This is in fact consistent with the number of times that term appears in D4 in the Table 4, however, it does not reflect the semantic importance for the corpus considered.

Table 6. Vector representation of the sample corpus shown in Table 4 using the Bag-of-Words model.

Terms	first	forth	the	corpus	short	of	biggest	second	sentence	third	is
D1	1	0	0	1	0	1	0	0	1	0	0
D2	0	0	1	0	1	0	0	1	1	0	1
D3	0	0	1	0	1	0	0	0	0	1	1
D4	0	1	2	1	0	1	1	0	1	0	1

The BoW representation model, like its predecessor, suffers from the same critical problem, the presumption of equality for the relevance of all terms towards the corpus. Such an assumption can give questionable results since terms with high occurrence in a single document can eventually be overestimated in an evaluation based on the total sum of each term in the corpus [46]. Although this model fails to identify the semantic importance of a term, the computational cost for its implementation is low and allows to identify more prevalent terms both in a document and throughout the corpora via simple operations, the sum of columns, with the weight matrix. It is also noteworthy that the BoW is a first step in the implementation of more complex models.

6.3. Vector Space Model Term Frequency-Inverse Document Frequency

This classic vectorization model computes the TF-IDF value of each word in a document using Equation (2), being defined as the product of two statistical measures, the term frequency (TF) and the inverse document frequency (IDF). The first factor of this multiplication, $tf_{t,d}$, is calculated according to Equation (3) by dividing the number of occurrences $n_{t,d}$ of a term t in the document d, by the total number of terms in document d. The second factor, idf_t, refers to how much that term t is mentioned in other documents. In its formula, expressed in the Equation (4), N is defined as the number of documents in the corpora and df_t considers the number of documents in which the term t appears.

$$TF - IDF = tf_{t,d} \cdot idf_t \tag{2}$$

$$tf_{t,d} = \frac{n_{t,d}}{\sum_{w \in d} n_{w,d}} \tag{3}$$

$$idf_t = log \frac{N}{df_t}. \tag{4}$$

The TF-IDF allows to measure the degree of semantic relevance of a document term, in relation to the entire collection. As expected, Table 7 has the same number of rows and columns as the Bag-of-Words model. A variant of the original TF-IDF, is known as Term

Frequency-Inverse Sentence Frequency (TF-ISF), being widely used in summarizing of texts at sentence level and not at document level like the TF-IDF.

Table 7. Vector representation of the sample corpus shown in Table 4 using the Term Frequency-Inverse Document Frequency (TF-IDF) model.

Terms	first	forth	the	corpus	short	of	biggest	second	sentence	third	is
D1	0.614	0	0	0.484	0	0.484	0	0	0.392	0	0
D2	0	0	0.378	0	0.467	0	0	0.592	0.378	0	0.378
D3	0	0	0.408	0	0.505	0	0	0	0	0.640	0.408
D4	0	0.419	0.535	0.330	0	0.330	0.419	0	0.267	0	0.267

The representation by the TF-IDF model, compared to the others, is the one that carries the greatest correlation between the semantics of the term and its weight in the vector space. This representation is very useful in problems that aim to extract knowledge from the datasets according to the semantics of the documents [40]. However, this representation is sensitive to the use of synonyms of common words. As unusual synonyms have a low frequency of use, even if they refer to common meanings widely represented by other words, the synonym term has a high weight in the TF-IDF representation, although it may not be as significant for the representation of the data. This anomaly is frequently addressed in works that rely on thesaurus dictionaries to normalize the vocabulary of the text [47].

An important point to be clarified is that, regardless of the applied representation, the dimension of the vector is linked to the remaining amount of distinct words contained throughout the dataset, since several of them were removed during the steps described in Section 5. The words kept in the sentence are those that carry meaning and, therefore, are the most important for understanding the central idea of the text. When considering the modeling of machine learning problems based on natural language processing, the remaining words are the characteristics of the dataset on which the learning is to be done.

6.4. Vector Space Model of Feature Hashing

Unlike the previous representations, the representation by Feature Hashing delimits the size of the vector space based on positions in a hash table. This representation uses a hash function to generate the vectors, which maps data of variable size in indexes of a table with fixed size, called hash table, or scatter table. In the context of vectorization, the resulting indexes correspond to the analyzed terms.

Each document can be represented from the N indexes in the table, so that, for a grouping of M documents, their mathematical representation is verified using a $M \times N$ matrix, which identifies the document collection (corpora). The determination of N is arbitrary and may be less than or equal to the total number of terms (tokens). However, the optimum value of positions must be evaluated because, being less than the number of terms observed in the documents, the representation can present inconsistency, since there is a collision of terms in common indexes that can store non-correlated information. For the representation of the corpus example according to the Feature Hashing model, 5 indexes are selected, arbitrarily, considering a vocabulary of 11 different words. Thus, the vectors are checked in Table 8.

Table 8. Vector representation of the sample corpus shown in Table 4 using the feature hashing model. Unlike the other models, only 5 columns are observed for representing the documents, which corresponds to the number of indexes in the hash table.

Hashes	Index 1	Index 2	Index 3	Index 4	Index 5
D1	1	1	1	0	0
D2	0	1	1	1	1
D3	0	1	1	0	1
D4	1	3	1	1	1

The Feature Hashing model provides a compact representation of the data, at the cost of less semantic granularity, since each index in the hash table can contain data that is not semantically correlated.

6.5. Word Embeddings

The choice to treat words as atomic units, that is, without a semantic connection between them, brings simplicity and robustness to the vector space model. Despite allowing an assessment of the similarity between phrases or documents, these models make it impossible to measure by word, making words with close meanings like "sea" and "ocean" invisible to vector modeling. An immediate consequence of this semantic shortage is the difficulty of dealing with synonyms. Another disadvantage is the high dimensionality, a reflection of the sparse character of the vectors generated [48,49]. This vector space model provides a compact representation of the data, at the cost of less semantic granularity, since each index in the hash table can contain data that is not semantically correlated.

As an alternative, Word Embeddings model appears as a form of distributed representation of words, idealized according to the distributional hypothesis. In this hypothesis, each word is characterized by its neighborhood, thus expressing a tendency for words with similar meanings to appear in similar contexts [50]. Such word representations can be obtained by applying predictive models based on neural networks that, when trained with large volumes of textual data, incorporate the semantics of words in small, dense, and fixed-sized vectors. The main advantage of individualized vector representation for each word is the preservation of semantic and syntactic relations between words, thus allowing synonyms or minimally related words to be mapped into similar vectors [51].

The popularization of word embeddings techniques occurred through Word2Vec [49], a tool that computes the vector representation of words using two possible models, the Continuous Bag-of-Words (CBOW) and a Skip-gram. Both models divide the texts into two groups, target word, and context. In particular, the context is interpreted as a limited set of words that surround the target word. The size of this limitation, known as a window, defines the number of words to be considered to the left and right of the target word.

The particularity of the Skip-gram model is its ability to use a target word w_t for predicting the context of words $W_t = [w_{tj}, \cdot, w_{t+j}]$ that surrounds it. As illustrated in Figure 3, the architecture of the Skip-gram model is composed of the input and output layers, interspersed by a projection layer. The size of the input layer, as well as the output layer, is linked to the number of words V existing in the vocabulary used in the training. The size of the projection layer is determined based on an arbitrary N parameter, which expresses the size of the future generated vector of words H (word embeddings). This dimension indicates the number of characteristics used in the numerical representation of each word, being, therefore, less than the dimension of the original vector of each word inserted in the input layer. The connection of the input layer to the projection layer is made through an array of weights W_I of size $V \times N$. Similarly, the connection from the projection layer to the output layer is performed by the matrix W_O of size $N \times V$. As usually done before training neural networks, both weight matrices W_I and W_O are initialized with small random values. The insertion of a target word in the input layer of the neural network

begins with the encoding of that word in its one-hot vector, a column array $N \times 1$ used to distinguish each word in a vocabulary. This vector consists of 0s in all positions, except a single 1 in a position used exclusively to identify the word.

In the training process, two learning algorithms are used for each iteration: forward propagation and back-propagation. Applying the forward propagation algorithm first, the one-hot vector of the input target word is multiplied by the weight matrix W_I to form the H vector of the hidden layer. Then, the H vector is multiplied by W_O thereby generating C identical intermediate vectors, each representing a context word. The model outputs are acquired by applying the softmax function to each intermediate vector:

$$p(w_{t+j}|w_t) = \frac{exp({v'_{w_o}}^\top v_{w_t})}{\sum_{i=1}^{V} exp({v'_{w_i}}^\top v_{w_I})}, \qquad (5)$$

where given the target word w_t, v_{w_i} is its corresponding line in the weight matrix W_I and v_w is its corresponding column in the matrix W_O. This function normalizes the intermediate vector U composed of V floating numbers, transforming it into the probability distribution vector Y. Once the normalized probability vector of each context word has been discovered, the back-propagation algorithm compares them with the one-hot vector of the corresponding word to update the weight matrices W_I and W_O. This update occurs specifically in the corresponding column values of W_O and the corresponding line of W_I.

The reversion of the action of the target word and the context words in the neural network, allows the architecture of the CBOW model to predict a target word from the context of nearby words, as shown in Figure 4. As a consequence of this inversion, the model admits multiple entries, one for each context word. This multiplicity of input vectors incurs the need to calculate the average of the corresponding word vectors, constructed by multiplying the multiple input one-hot vector and the matrix W. A second consequence is the presence of a single softmax function, as opposed to the C existing in the Skip-gram model architecture [52]. The CBOW model converges faster than the Skip-gram. However, the Skip-gram presents better results for infrequent words compared to the former.
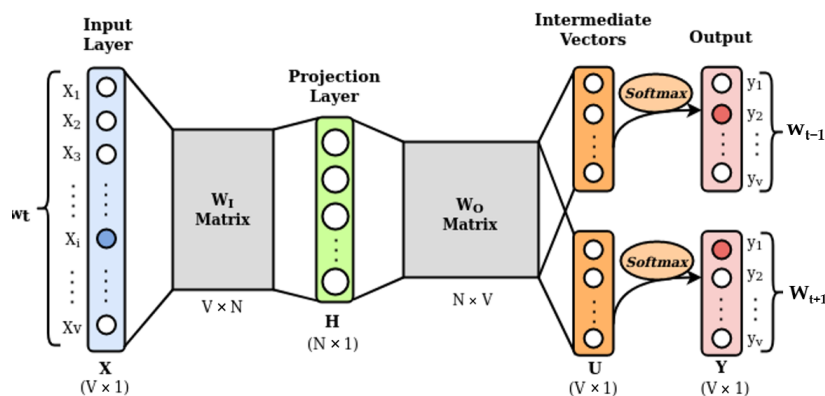
Figure 3. Skip-gram model considering the target word w_t encoded in its one-hot vector X as input. This vector represents the target word as a sequence of 0s V, except for a single value 1 in the position x_i. The C probability distribution vectors are obtained at the output of the model, one for each word in the context. With the model properly trained, it is expected that the highest probabilities of each vector Y, found in the positions y_2 and y_1, will express the context words w_{t-1} and w_{t+1}.
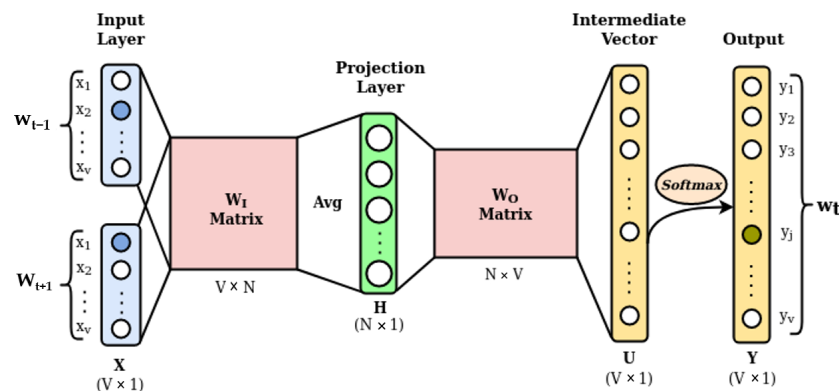
Figure 4. Continuous Bag-of-Words (CBOW) model considering the context words w_{t+1} and w_{t-1} encoded in their one-hot vectors. At the output of the model, a probability distribution vector is obtained. With the model properly trained, it is expected that the highest probability of the Y vector, found in the position y_i, will express the target word w_t.

7. Learning on Natural Language Data from Social Networks

Machine learning is inherently a multidisciplinary field, focused on building computer programs that automatically improve with the experience [53]. Machine learning is related to the extraction of knowledge from raw data. The machine learning algorithms aim to discover how to perform important tasks by generalizing their operations from data examples [54]. Although there are different definitions for machine learning, they all converge on the idea of using algorithms to obtain data, learn from it, and then determine or predict some phenomenon. There are different machine learning algorithms, each indicated for a desired type of output. The supervised learning also called learning with examples, assumes the existence of labeled inputs and outputs, composing a training set, to learn a general rule that maps the inputs to outputs. In contrast, unsupervised learning, is independent of any label on the data, forcing the algorithm to identify patterns in the inputs, so that the inputs that have something in common are grouped into the same category. Reinforcement learning learns as it interacts with a dynamic environment, so that any action that has an impact on the environment provides feedback that guides the algorithm [55].

7.1. Dimension Reduction

When using extensive datasets, especially if they are composed of texts from heterogeneous knowledge domains, it is inevitable to deal with vectors of extremely long characteristics. In addition to increasing computational complexity, the use of vector representations that are too large may not be the most appropriate option. This hypothesis is confirmed in the problem known as the "curse of dimensionality", which expresses the existence of an optimal number of characteristics that can be selected in relation to the sample size to maximize the learning performance [56]. In this scenario, it is convenient to apply some procedure to reduce the dataset, either through the selection of original characteristics or through dimensionality reduction techniques. The latter aims to find less complex vector representations, creating new synthetic characteristics from the original ones.

Dimensionality reduction is the process of deriving a smaller set of degrees of freedom that reproduces the greater variability of a dataset [56,57]. Ideally, the reduced representation should have a dimensionality that corresponds to the intrinsic dimensionality of the data, which is the minimum number of parameters to account for the properties observed in the data. Mathematically, in reducing dimensionality, given the p-dimensional random variable x = $(x_1, x_2, \ldots, x_p)$, a lower-dimensional representation is calculated, s = $(s_1, s_2, \ldots, s_k)$ with $k \leq p$.

Different approaches are proposed to reduce dimensionality, being classified as linear or non-linear. Linear dimensionality reduction is a linear projection of the original data,

in which the *p*-dimensional data is reduced to a *k*-dimensional data using *k* linear combinations of the original *p* characteristics. Two important examples of linear dimension reduction algorithms are Principal Component Analysis (PCA) and Independent Component Analysis (ICA). The objective of the PCA is to find an orthogonal linear transformation that maximizes the variance of the characteristics. The first base vector of the PCA, the main component, describes the direction of greater variability of the data. The second vector is the second-best description and must be orthogonal to the first, and so on in order of importance. Similarly, the ICA's goal is to find a linear transformation, in which the base vectors are statistically independent and non-Gaussian, i.e., the mutual information between two characteristics in the new vector space is equal to zero. Unlike the PCA, the base vectors in the ICA are neither orthogonal nor classified in order. All vectors are equally important. PCA is generally applied to reduce data representation. On the other hand, ICA is normally used to obtain the extraction of characteristics, identifying, and selecting the characteristics that best adapt to the application. Nonlinear methods apply transforms to the data, changing them into a new vector space, in which linear methods can be applied.

Aimed especially at vector representations derived from texts, Latent Semantic Indexing (LSI) is a dimensional reduction technique based on Singular Value Decomposition (SVD). For purposes beyond the area of information retrieval, LSI is also referred to as Latent Semantic Analysis (LSA). The LSI's adaptability to textual data is linked to the sparse nature of the data. LSI proposes to build a "semantic " space in which closely associated terms and documents are placed close to each other.

Assuming *A* as the original matrix *n* × *m*, in which terms and documents are represented in rows and columns respectively, the application of LSI begins by adopting a level of approximation *k*. Hence, *A* can be decomposed as follows:

$$A \approx A_k = U_k D_k V_k^T, \tag{6}$$

where A_k is an approximation of *A*, composed of the product of the term-concept matrix U_k, the singular value matrix D_k and the document–concept matrix V_k. Thus, the A_k matrix expresses the best representation of the semantic structure of the original corpus, omitting all but the largest *k* single values in the decomposition. For this reason, LSI is also known as truncated SVD [58,59]. Regarding the choice of *k*, this is done through empirical tests, evaluating the variance rate of the singular values. The *k* value must be small enough to allow for quick retrieval of information and large enough to properly capture the corpus structure. For textual data, the reduction of dimensionality is preferable to be performed by the LSI technique in comparison to the PCA or ICA because, due to the sparse nature of the data, the PCA and ICA techniques show less significant or flawed results, while the LSI is suitable for sparse data.

The dimensionality reduction techniques lack expressiveness, as the generated characteristics are combinations of other original characteristics. Therefore, the meaning of the new synthetic characteristic is lost. When there is a need to interpret the model, for example, when creating filters based on texts in natural language, it is necessary to use other methods. The feature selection techniques produce a subset of the original features, which are the best representatives of the data. Thus, there is no loss of meaning. There are three types of feature selection techniques [57]: wrapper, filter and embedded.

The wrapper methods, also called closed-loop, use different classifiers, such as Support Vector Machine (SVM), decision tree, among others, to measure the quality of a subset of characteristics without incorporating knowledge about the specific structure classification function. Thus, the method evaluates subsets based on the classifier's accuracy. These methods consider feature selection as a search problem, creating a *NP*-hard problem. An exhaustive search of the complete dataset should be done to assess the relevance of the resource. Wrapper methods tend to be more accurate than filter methods, but have a higher computational cost [57]. A popular wrapper method due to its simplicity is Sequential Forward Selection (SFS). The algorithm starts with an empty set *S* and the complete set of all characteristics *X*. The SFS algorithm searches and gradually adds features, selecting

S by an evaluation function, minimizing the Mean Square Error (MSE). At each iteration, the algorithm selects a feature to be included in *S* from the remaining available features in *X*. The main disadvantage of SFS is that adding a new feature to the *S* set prevents the method from removing any feature that has the slightest error after adding others. The filter methods are computationally lighter than the wrapper methods and avoid overfitting. The filter methods, also called open-loop methods, use heuristics to assess the relevance of the feature in the [60] dataset. The algorithm filters out the characteristic that meets the heuristic criterion. One of the most popular filtering algorithms is Relief. The Relief algorithm associates each feature with a score, which is calculated as the difference between the distance of the example closest to the same class and the example closest to the other class. The main disadvantage of this method is the requirement to label data records in advance. Relief is limited to problems with only two classes, but ReliefF [61] is an improvement on the Relief method that handles multiple classes using the nearest neighboring *k* technique. The built-in methods behave similarly to the wrapper methods, using the precision of a classifier to evaluate the relevance of the characteristic. However, the built-in methods make the selection of characteristics during the learning process and use their properties to guide the evaluation of the characteristic. This modification reduces computational time compared to wrapper methods. The Support Vector Machine Recursive Feature Elimination (SVM-RFE) classifies features according to a classification problem based on training a support vector machine (SVM) with a linear kernel. The element with the lowest classification is removed, according to the criterion *w*, in the form of sequential reverse elimination. The *w* criterion is the value of the decision hyperplane in SVM.

7.2. Similarity and Dissimilarity Metrics

Similarity and dissimilarity measures play a critical role in quantifying the semantic similarity or distance, respectively, between texts. Regardless of the compared textual elements, characters, terms, strings, or corpus, such measures are constantly present in solving pattern analysis problems, whether to summarize, classify, or grouping texts. Assuming a pair of non-null *A* and *B* vectors, composed of the same *n* amount of terms, such that $A = [x_1, x_2, ..., x_n]$ e $B = [y_1, y_2, ..., y_n]$, it is possible to measure the semantic relationship between them in different ways, such as Euclidean Distance, Manhattan Distance and Similarity Cosine.

The dissimilarity metric known as Minkowski distance is given by the equation:

$$Dis(A, B) = \left(\sum_{i=1}^{n} |x_i - y_i|^p \right)^{\frac{1}{p}}. \tag{7}$$

This metric is a generalization of two other equally known ones, Manhattan Distance and Euclidean Distance, for *p* equal to 1 or 2 respectively. Clearly, it is expected that the closer to zero the value of *Dis* is, the more similar *A* and *B* will be.

Among the similarity metrics to compare a set of terms, the Cosine Similarity stands out. This metric uses the concept of internal product, and it is defined between $[-1, 1]$, such that values closer to the upper limit represents greater proximity between the term vectors. Mathematically, the cosine similarity between *A* and *B* is calculated by the equation:

$$Sim(A, B) = \frac{\sum_{i=1}^{n} x_i y_i}{\sqrt{\sum_{i-1}^{n} x_i^2} \sqrt{\sum_{i-1}^{n} y_i^2}}. \tag{8}$$

7.3. Supervised Algorithms

The distinction between supervised algorithms can be made by defining those whose expected results are real value variables, called regression algorithms, and those whose results are categories represented by discrete values, known as classification algorithms. We focus on classification algorithms due to the classification nature of the natural language processing applications covered in this paper.

7.3.1. Support Vector Machine

The Support Vector Machine (SVM), consists of a type of linear classifier algorithm, based on the concept of a decision plan that defines the decision limits. The decision-making process takes place through the generation of an optimal multidimensional hyperplane that separates samples into classes, maximizing the distance between classes or the separation margin. Such a hyperplane is drawn by a subset of samples, called support vectors. The optimal separation is ensured by the definition of a kernel function that minimizes the error function. Although it is essentially a binary classifier, SVM is also adaptable to multiclass problems, where the original problem is divided into binary classification subproblems.

When dealing with a set of non-linear samples, one strategy is to adopt a kernel function, which can find a new dimensional space, mandatory larger than the original, that allows the separation using a hyperplane. Among the most used kernel functions are: Linear, Polynomial, Radial Basis Function (RBF), and Sigmoid. SVM's ability to be less prone to overfitting, i.e., obtaining a separation function with greater complexity than necessary, is closely related to the degree of relevance attributed to samples far from the separation limit. Once the hyperplane is found, most data other than the support vectors are seen as redundant.

The use of supervised algorithms for detecting fake news depends on a large dataset containing both false and legitimate news. However, this imposes the limitation of having a base labeled with false and legitimate news. Although fake news are increasingly numerous and widespread on social media, such news tend to be volatile, as some period after dissemination it loses credibility. A strategy to counter the limitation on the number of fake news to train classifiers is to learn a single class, such as the one based on the One-class Support Vector Machine. One-class SVM is a supervised learning algorithm that derives a decision hyperplane for detecting anomalies. New data is classified as similar or different from the training set. In contrast to typical SVM implementations, the single class takes into account a set of training samples from a single class. Any new sample that does not fit the decision surface defined by the training set is considered an instance of a new class and, therefore, fake news [62,63].

7.3.2. Random Forest

The Random Forest (RF) is a popular classification or regression algorithm, which operates by building multiple decision trees during the training process. During training, the RF allows the application of the bagging method, which allows the algorithm to be repeatedly trained with the same dataset, however, selecting the characteristics randomly. Illustratively, for a training set with input samples $X = x_1, x_2, ..., x_n$ and respective output samples $Y = y_1, y_2, ..., y_n$, the bagging method implies the random and repetitive selection of that dataset K times. Thus, the trees are trained with the same information, so that the final result is formed by the individual predictions m_i of each tree in the set, according to the equation:

$$\hat{m} = \frac{1}{K} \sum_k^{i=1} m_i. \tag{9}$$

A relevant advantage of the RF for the traditional model of decision trees is the fact that the whole dataset is not considered, but only a subset. This implies greater randomness in the model, helping to correct overfitting. In the same sense, by increasing the number of decision trees in the RF, the error rate of the test set converges to a limit, meaning that more populated RFs are less susceptible to overfitting [64].

7.3.3. *k*-Nearest Neighbors

The *k*-Nearest Neighbors algorithm (*k*-NN) depends on the previous choice of a parameter *k*, which determines the number of nearest neighbor samples used in the classification criterion. From a sample not yet classified, the algorithm applies a metric of distance, or similarity, between that sample and all the others already classified, filtering the neigh-

boring k samples that had the shortest distances. The algorithm checks and counts the number of samples included in each class. Finally, the sample is allocated to the majority class of the k-nearest neighbors. This dependence on the value of the initial parameter causes the result of the algorithm to present several classifications if k is too high, or to present noisy samples if k is too small. When being forced to calculate the distance of each new sample with all the others already classified, the algorithm requires a higher computational consumption, being therefore not suitable for very large corpus [65]. It is also worth mentioning the high memory consumption of the algorithm since it is necessary to load the entire dataset in memory for comparison with the new samples.

7.4. Unsupervised Algorithms

Clustering algorithms are the most common form of unsupervised learning. Although they have operational logic, use case, scalability and distinct performances, the generic purpose of using these algorithms is to segregate terms in groups (clusters) according to their semantic characteristics. This procedure of separation into groups is known as clustering.

### 7.4.1. Partitioning-Based Algorithms

This classification is given to the algorithms that are similar in the sense of simultaneously fulfilling two criteria in the data grouping process. The first criterion expresses the obligation to have at least one sample in each group created. The second refers to a membership exclusivity, in which each sample must belong to only one grouping [66,67].

A classic example of this type of algorithm is K-means, a heuristic capable of partitioning data into k clusters by minimizing the sum of squares of distances in each cluster. K-means begins with the random choice of the centroids of each cluster followed by the calculation of the distance between each sample and the centroids, according to one of the metrics of dissimilarity, or similarity, discussed in Section 7.2. Subsequently, each sample is allocated to the cluster whose centroid is the closest. For each new sample allocated to a cluster, the centroid is recalculated, with the possible redistribution of samples to other clusters. The algorithm ends when these changes in the allocation of samples to the clusters cease.

Another example is the k-medoids algorithm, suitable for small data sets. The k-medoids also partitions the data into k clusters adopting the criterion of minimizing the sum of the squares of the distances in each cluster. Although it resembles K-means, K-medoids differs because it effectively chooses one of the input samples as the center of the clusters, unlike K-means, which chooses midpoints. This decision-making characteristic translates into greater robustness to noisy data and outliers, in addition to an ability to handle high dimensionality, which is useful in vector representation of textual data [66,67]. Another advantage of K-medoids over K-means is that the outputs of K-medoids are more easily interpreted, given that the cluster centers are real samples, unlike K-means, which provides a point that can represent an unfeasible sample of data.

Both k-means and K-medoids, as well as other algorithms in this classification, are subject to a unique disadvantage: indeterminacy of the appropriate number of clusters k. To circumvent this indeterminacy, the Elbow and the Silhouette methods are used. The goal is to previously analyze the conformity of the data to different amounts of groups and thus obtain a result appropriate to the data. In particular, the Elbow method measures the compaction of clusters by establishing a relationship between the number of clusters and their influence on the total variation of data within the cluster. Graphically, the best k value is found by identifying the point at which the curve gain decreases dramatically, remaining approximately constant thereafter. Similarly, the Silhouette method measures the quality of a cluster. The ideal number of k clusters is the one that maximizes the average silhouette over a range of possible values for k [68,69]. Figure 5 shows a hypothetical usage example of the Elbow and Silhouette methods. In this hypothetical example, it is seen that for the value $k = 5$ there is a sharp change in the mean square error (SSE) internal to the

clusters observed in the Elbow method and, for $k = 5$, there is also a maximum point of the mean error between the clusters in the Silhouette method, indicating a greater separation between clusters.

(**a**) Elbow Method.

(**b**) Silhouette Method.
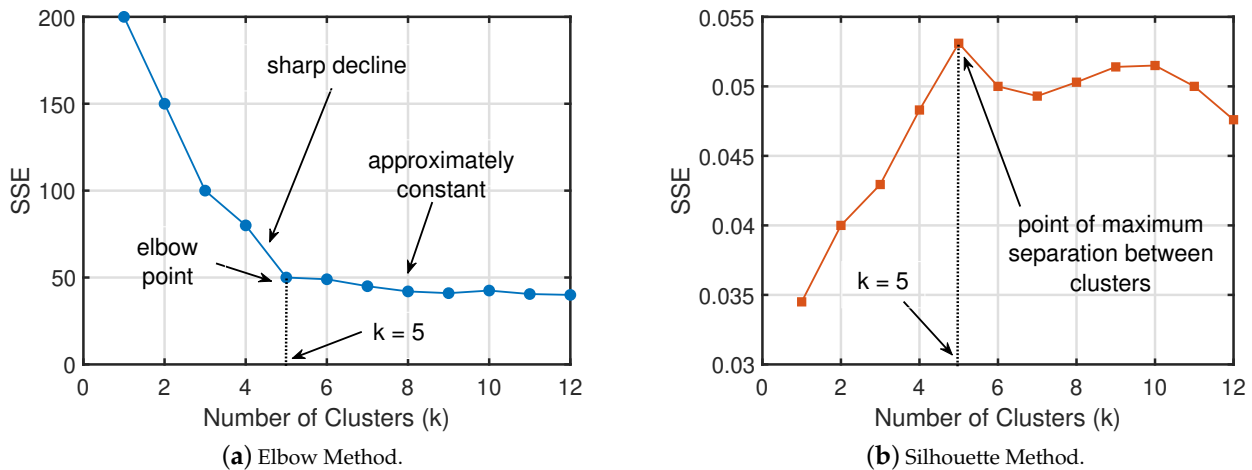
Figure 5. Elbow and Silhouette are complementary methods to determine the optimal number of clusters. Both methods ideally tend to converge to the same k, e.g., $k = 5$ in the example.

It should also be noted that there are variations of the K-means and K-medoids algorithms that consider the degree of relevance of a sample to different groups. In these cases, called fuzzy K-means and fuzzy K-medoids, the center of the clusters is calculated considering the partial relevance of each sample to the clusters.

7.4.2. Density-Based Algorithms

Density-based clustering algorithms share a close relationship with the nearest neighbor approach. In this sense, a cluster, defined as a dense connected component, grows in any direction that density leads. This logic of forming clusters is directly related to the main advantage of these algorithms compared to the partitioning algorithms, which is the possibility of discovering clusters with arbitrary shapes, differently from the typically spherical clusters returned by the K-means algorithm, for example.

Among density-based algorithms, the Density-Based Spatial Clustering of Application with Noise (DBSCAN) algorithm is the most popular. DBSCAN purpose is to find regions that satisfy an established minimum point density and that are separated by regions of lower density. To this end, the algorithm performs a simple estimate of the minimum density level, defining a limit for the number of neighbors, *minPts*, within a radius ϵ. Thus, a sample with more than *minPts* neighbors within that radius is considered a central point. Similarly, a sample is considered to be borderline if, within its neighborhood, there are fewer samples than the defined minimum but the sample still belongs to the neighborhood of any central point. Finally, samples that are not reachable by density from any central point, that is, they are neither central nor border points, are labeled as outliers. A disadvantage of this method is its strongly polynomial complexity, which requires $\Omega\left(n^{\frac{4}{3}}\right)$ time to converge, where n is the size of the dataset [67,70,71].
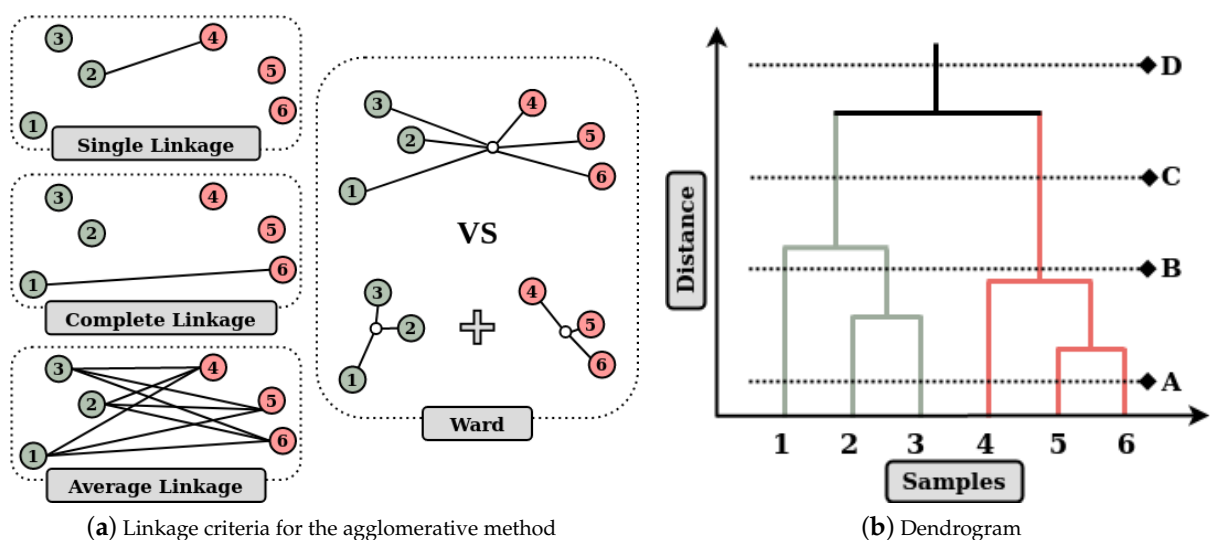
7.4.3. Hierarchical Algorithms

Hierarchical algorithms not only create clusters but consider multi-level logic and calculate a hierarchical representation of the input data. This representation is a particular type of tree, in which the leaf nodes express individual data, and can be constructed using an agglomerative or divisive method. The agglomerative method, also known as the bottom-up approach, begins by considering each sample as a unitary cluster and recursively merging two or more into a new cluster following a chosen link function. Such

functions, when associated with distance or similarity metrics, define unique criteria that elect the merged clusters of each iteration. The single link function, for example, establishes the union considering the distance between the samples closest to each cluster. Conversely, the complete link function considers the distance of the most distant samples to each cluster. At the same time, the average link function averages the distances of all samples in one cluster concerning all samples in another cluster. In particular, Ward's criterion employs Euclidean distance in discovering the pair of clusters that minimize the increase in the total internal variance after the union.

The divisive method, in turn, also known as the top-down approach, starts with a flat structure in which all samples belong to the same cluster, i.e., the same hierarchical level. Therefore, at each iteration, the algorithm divides a parent branch into two smaller subsets, the child branches. The process ends when a stop criterion is reached, often the number k of clusters. At the end of the algorithm, a clustering dendrogram is created, which is a binary tree hierarchy [40,67,72,73]. A possible hierarchical clustering considering the spatial arrangement between samples 1–6 of Figure 6a is illustrated in Figure 6b. Tracing the dotted lines A–D perpendicular to the vertical branches of the dendrogram, it is possible to identify different moments in the clustering process. In *A*, there are 6 unit clusters, that is, each containing samples. In *B*, there are 3 clusters: the unit cluster of sample 1, the cluster of samples 2 and 3, and the cluster formed by samples 4, 5, and 6. In *C*, it is already possible to identify the same pair of groupings depicted in Figure 6a. Finally, in *D* we verify the presence of a single overpopulated cluster, containing all the initial samples.

(**a**) Linkage criteria for the agglomerative method   (**b**) Dendrogram

Figure 6. (**a**) Two-dimensional representation of different connection criteria for the 6 samples already allocated in two clusters. (**b**) Dendrogram resulting from the application of the hierarchical clustering algorithm on samples 1–6. The algorithm employs the agglomerative method using single bond criteria.

7.5. Evaluation Metrics

Regardless of the supervised or unsupervised algorithm, if there is prior knowledge about data labeled based on a ground truth, it is plausible to clearly identify the number of True-Positive (TP), False-Positive (FP), True-Negatives (TN) and False-Negatives (FN). Such classifications make up the calculation of various information retrieval metrics, summarized in Figure 7, such as the following:

- **Accuracy** (A_c) is defined by the ratio of the total of correctly classified samples (TP + TN), by the total number of samples (P + N). For unbalanced data sets, a performance assessment based solely on this metric can generate erroneous conclusions;

- **Precision** (P_r), given a target class, is the ratio between the number of samples correctly classified for the class in question (TP), by the total set of predictions assigned to that class, i.e., correct and incorrect predictions (TP + FP);
- **Sensitivity** (S_s), also known as recall or **true positive rate**, is defined by the ratio of the number of correctly predicted samples (TP) to a positive class and the total of samples that belong to this class, thus including both correct predictions and those that should have indicated this class (TP + FN). The analog for the negative class is called **specificity** or **true negative rate**;
- F_1-**Score** relates precision and sensitivity by a harmonic mean expressed by

$$F_1\text{-}Score = \frac{2}{\frac{1}{P_r} + \frac{1}{S_s}}; \tag{10}$$

Generally, the higher the value of the F_1-Score, the better the classification, reflecting the mutual commitment between precision (P_r) and sensitivity (S_s):

- **Area under the ROC Curve** (AUC) is measured using the Receiver Operation Characteristic (ROC) curve, shown in Figure 7a, which represents the ratio between the true positive rate (TPR) and the false positive rate (FPR), for several cutoff thresholds. This curve graphically describes the performance of a classification model. Briefly, the larger the area under the curve (closer to the unit value), the better the performance of the model, regardless of the cutoff point of the probability of the sample belonging to each class.

(**a**) ROC Curve   (**b**) Information Retrieving Metrics
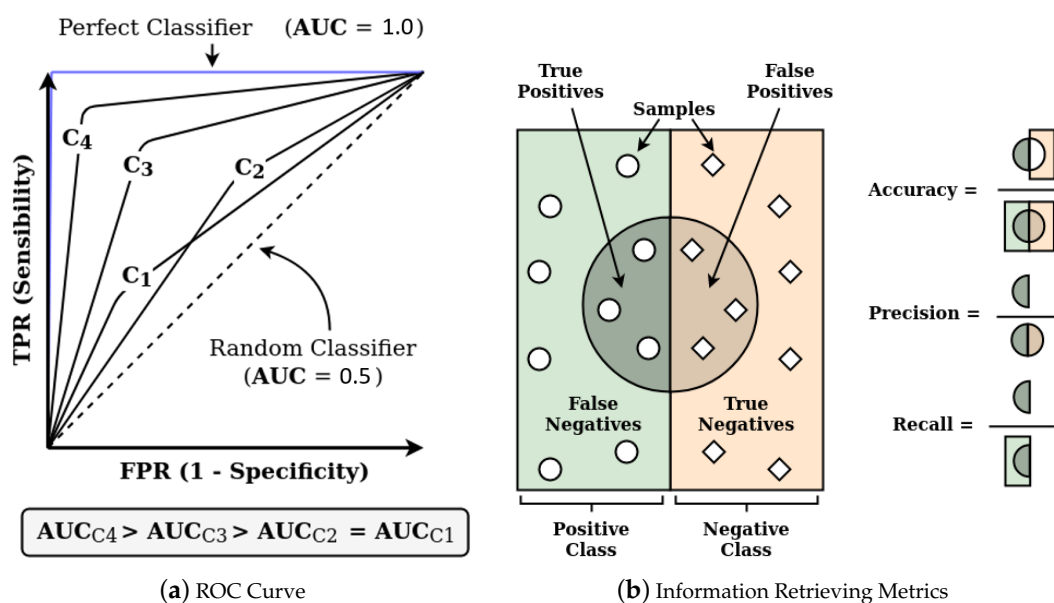
Figure 7. (**a**) Receiver Operation Characteristic (ROC) curve of classifiers and comparison of the Area Under the ROC Curve (AUC). (**b**) Accuracy, precision and sensitivity metrics in a binary classification problem.

8. Research Initiatives

Several research activities exist and seek to characterize and mitigate the challenges caused by fake news. Lazer et al. formalize an initial definition of fake news and discuss the historical background of fake news, starting with defamation in the First World War until the impact of fake news during the United States presidential election in 2016 [74]. Grinberg et al. delve into the impact of fake news during the 2016 elections, analyzing messages from the Twitter social network [75]. The authors collected tweets sent by 16,442 active accounts during the 2016 electoral season, from 1 August to 6 December 2016. The results show that groups of older users, who are between 60 and 80 years old, with right-wing or extreme right political affinity are more likely to distribute and share fake political news.

The recent 2019 Coronavirus Infectious Disease pandemic (COVID-19) is also an event in which a large amount of fake news is disseminated. Recent studies show the correlation between social media usage and misinformation during the pandemic [76,77].

The detection of fake news is studied from several perspectives, such as Machine Learning, Data Mining, and Natural Language Processing. The Bag-of-Words and the frequencies of categories are used to train classifiers such as Support Vector Machines (SVM) and naive Bayesian models [78]. Since the mathematical model is trained from known examples of the two categories, false and legitimate news, it is possible to predict future instances based on numerical clustering and distances. The use of different clustering methods and distance functions is one of the SVM algorithm bases. The naive Bayesian algorithm, in turn, makes classifications based on accumulated evidence of the correlation between a given variable, such as syntax, and the other variables present in the model.

Shu et al. review the detection of fake news on social media from a data mining perspective, including characterization of fake news on psychology and social theories, existing algorithms, evaluation metrics, and representative datasets [13]. Fake News Tracker is a solution for data collection, interactive visualization, and analytical modeling for detecting fake news. The solution uses Natural Language Processing techniques [79]. Other papers present techniques and challenges related to the detection of fake news. Zhou and Zafarani identify and detail fundamental theories related to different disciplines for detecting fake news [4]. Sharma et al. discuss existing methods and techniques that apply to the identification and mitigation of fake news, focusing on the significant advances in each method and their advantages and limitations [11]. Bondielli and Marcelloni survey the literature on the different approaches for automatically detecting fake news and rumors [80]. The authors highlight several approaches taken to collect fake news and rumor data.

Oshikawa et al. presents a comparison of the methods used to detect fake news using Natural Language Processing (NLP) [31]. Similarly, Sharma et al. analyze the literature review on NLP applied to fake news, highlighting the comparison between different machine learning techniques, deep learning, and other techniques [11]. Deepak and Chitturi compare different types of neural networks in detecting fake news [81]. Feng et al. propose a two-level convolutional neural network with a user response generator, in which the neural network captures semantic information from the text, representing it at phrase- and word-level. The user-response generator learns a model of the user's response to the news text [82].

9. Research Challenges and Opportunities

Research into identifying, detecting, and mitigating the spread of fake news is still under development. Nevertheless, it is already possible to identify the main challenges in combating fake news, which are listed following [11].

- **Great interests and the plurality of actors involved.** Due to the volume that the spread of fake news reaches on social networks in a short period, fake news pose a threat to traditional sources of information, such as traditional press. The spread of fake news occurs as a distributed event, and involves multiple entities and technological platforms. Thus, there is an increasing difficulty in studying and designing computational, technological, and business strategies to combat fake news without compromising speed and collaborative access to high-quality information.
- **Opponent's malicious intent.** The fake news content is designed to make it difficult for humans to identify the fake news, exploiting our cognitive skills, emotions, and ideological prejudices. Moreover, it is challenging for computational methods to detect fake news, as the way fake news is presented is similar to true news, and sometimes fake news uses artifices to make it difficult to identify the source or falsify the real source of the news.
- **Susceptibility and lack of public awareness.** The user of social networks is subject to a large amount of information from dubious origins, from information with a humorous nature, such as satires, to information intended to deceive the consumer

of the information posing as legitimate news. However, the user of social networks is not able to differentiate fake news from legitimate news just by content. The user does not have information about the credibility of the source or patterns of spreading of the news on the network. Thus, to increase public awareness, several articles and advertising campaigns are run to provide tips on how to differentiate between false and legitimate news. For example, the University of Portland in the United States provides a guide for identifying misinformation (fake news) (available at https://guides.library.pdx.edu/c.php?g=625347&p=4359724).

- **Propagation dynamics.** The spread of fake news on social media complicates detection and mitigation, as fake information can easily reach and affect large numbers of users in a short time. The information is transmitted quickly and easily, even when its veracity is doubtful [83]. Verification of veracity must be carried out in an agile way, but it must also consider the patterns of propagation of information throughout the network [84].

- **Constant changes in the characteristics of fake news.** Developments in the automated identification of fake news also drive the adaptation of the generation of new disinformation content to avoid being classified as such. The detection of fake news based on writing style, differentiating false and legitimate news by an analysis based on Natural Language Processing, is one of the most-used alternatives due to the unsolved challenges in automatic fact verification from pre-defined knowledge bases. Thus, current approaches to identify fake news based on the content focus on extracting facts directly from the news content and subsequent verification of the facts against knowledge bases [85].

- **Attacks on natural language learning.** Zhou et al. argue that the use of Natural Language Processing to identify fake news is vulnerable to attacks on the machine learning itself [86]. Zhou et al. identify three attacks: the distortion of facts, the exchange between subject and object, and the confusion of causes. The distortion is, in fact, to exaggerate or modify some words. Textual elements, such as characters and time, can be distorted to lead to a false interpretation. The exchange between subject and object aims to confuse the reader between those who practice and those who suffer the reported action. The attack of confusion of cause consists of creating non-existent causal relations between two independent events or cutting parts of a story, leaving only the parts that the attacker wishes to present to the reader [86].

Research opportunities to identify and mitigate fake news focus on rapid or real-time detection of the source, controlling the spread of false information and reducing the impact of fake news on society. Dataset collected in real-time, automatic detection of rumors, and location of the source are challenging research questions [84]. The main opportunities for research and development of solutions to combat fake news are highlighted following.

- **Extracting the most significant features.** Determining the most effective features for detecting fake news from multiple data sources is an open research opportunity. Fundamentally, there are two main data sources: news content and social context [13]. From a news content perspective, techniques based on Natural Language Processing and feature extraction can be used to extract information from the text. Embedding techniques, such as word embedding and deep neural networks are the focus of current researches for the extraction of textual characteristics, and they have the potential to learn better representations for the data. Visual characteristics extracted from the images are also important indicators of fake news. The use of deep neural networks is an opportunity for research in the extraction of visual characteristics for the detection of fake news [11,84].

- **Detection on different platforms and different domains.** Since that users use different social networks, fake news, and rumors spread across different platforms, making it difficult to locate the source of the news or rumor. Tracing the source of false information between different social media platforms is a research opportunity. Therefore, several aspects of the information must be considered. However, most of the existing

approach focuses only on one way of detecting false information: analysis of content, propagation, style, among others. The analysis must then consider different attribute domains, such as topics, web sites, images, and URLs [84].

- **Identification of echo chambers and bridges between chambers.** Social media tends to form echo chambers in communities where users have similar views and ideologies. Users have their views reinforced and are not aware of the opposite beliefs. Therefore, research is needed to identify conflicting echo chambers and connect chambers with opposite positions so that users are faced with different points of view. This bridging also helps in discovering the truth, making users think carefully and rationally in multiple dimensions [84].

- **Development of machine learning models.** There is a need for research in the development of real-time learning models, such as incremental learning and federated learning, capable of learning from manually verified articles and providing real-time detection of new articles with fraudulent information. Another important point is the development of unsupervised models in which the algorithms learn from real data and, then, articles that escape the behavior of real data are classified as false. There is still a dearth of specific datasets for fake news. The lack of publicly available large-scale datasets implies a lack of tests (benchmarks) for comparing the performance of different algorithms [84].

- **Development of data structures capable of handling complex and dynamic network structures.** The complexity and dynamics of social network relationship structures make the task of identifying and tracking posts more complicated. Thus, there is a need to develop complex data structures that reflect the dynamics of relationships in social networks to allow the extraction of knowledge about the spread of false information throughout the network [84].

10. Conclusions

In this paper, definitions, characteristics and the process of disseminating fake news were presented. We also discussed the traditional methods for detecting fake news. The most recent reference databases used in this area of research were compared. The literature shows that Natural Language Processing (NLP) has been used to detect fake news. We discussed how NLP could be used to evaluate information from social networks and compare the different machine learning methods. Unlike previous work, we summarized the key algorithms for processing each step on a Natural Language Processing framework devoted to identifying fake news in social media. We also presented current datasets to train and test fake news discrimination proposals.

Moreover, open questions and challenges are also highlighted to explore potential research opportunities. In this context, additional learning-related approaches and techniques are presented in the work of Palmieri and Giglio [87], as well as an exploratory methodology that allows deepening researches related to online social networks and NLP. Our work helps researchers understand the different components of online digital communication from a social and technical perspective. Dissemination of fake news on multiple multilingual platforms, complex and dynamic network structure, large volumes of real-time unlabeled data, and early detection of rumors are some challenging problems that are yet to be solved and need further research. Finally, we conclude that stylistic-computation approaches for identifying fake news are still a challenging research topic due to the scarcity of available information when just considering the news content. The dissemination of fake news holds complex linguistic constructions that lead to misinformation, as some parts of the news may be correct. Ongoing work and future research focus on correlating stylistic-computational approaches with other features extracted from the dissemination dynamics and from network properties. Therefore, detecting fake content dissemination remains the social media provider's responsibility because only the provider retains information to track the news dissemination, the source-user profile, and the users' feedback. Improving

the reliability and future of the information ecosystem online is a joint responsibility of the scientific community, digital policy makers, management and society.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

1. de Oliveira, N.R.; Medeiros, D.S.V.; Mattos, D.M.F. A Sensitive Stylistic Approach to Identify Fake News on Social Networking. *IEEE Signal Process. Lett.* **2020**, *27*, 1250–1254. [CrossRef]
2. Liu, G.; Wang, Y.; Orgun, M.A. Quality of trust for social trust path selection in complex social networks. *In Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems*; International Foundation for Autonomous Agents and Multiagent Systems, Toronto, ON, Canada, 9–13 May 2010; pp. 1575–1576.
3. Vosoughi, S.; Roy, D.; Aral, S. The spread of true and false news online. *Science* **2018**, *359*, 1146–1151. [CrossRef] [PubMed]
4. Zhou, X.; Zafarani, R. A Survey of Fake News: Fundamental Theories, Detection Methods, and Opportunities. *ACM Comput. Surv.* **2020**, *53*. [CrossRef]
5. Wang, W.Y. "Liar, liar pants on fire": A new benchmark dataset for fake news detection. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Vancouver, BC, Canada, 30 July–4 August 2017; pp. 422–426. [CrossRef]
6. Rubin, V.L. On deception and deception detection: Content analysis of computer-mediated stated beliefs. In *Proceedings of the 73rd ASIS&T Annual Meeting on Navigating Streams in an Information Ecosystem*; American Society for Information Science: Silver Spring, MD, USA, 2010; Volume 47, p. 32.
7. Rubin, V.; Conroy, N.; Chen, Y.; Cornwell, S. Fake news or truth? using satirical cues to detect potentially misleading news. In *Proceedings of the Second Workshop on Computational Approaches to Deception Detection*; Association for Computational Linguistics: San Diego, CA, USA, 17 June 2016; pp. 7–17. [CrossRef]
8. Rubin, V.L.; Chen, Y.; Conroy, N.J. Deception detection for news: Three types of fakes. In *Proceedings of the 78th ASIS&T Annual Meeting: Information Science with Impact: Research in and for the Community*; American Society for Information Science: Silver Spring, MD, USA, 6–10 November 2015; p. 83.
9. Rubin, V.L.; Conroy, N.J.; Chen, Y. Towards news verification: Deception detection methods for news discourse. In Proceedings of the Hawaii International Conference on System Sciences, Kauai, HI, USA, 5–8 January 2015.
10. Manning, C.; Surdeanu, M.; Bauer, J.; Finkel, J.; Bethard, S.; McClosky, D. The Stanford CoreNLP natural language processing toolkit. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Baltimore, MD, USA, 23–25 June 2014; pp. 55–60. [CrossRef]
11. Sharma, K.; Qian, F.; Jiang, H.; Ruchansky, N.; Zhang, M.; Liu, Y. Combating fake news: A survey on identification and mitigation techniques. *ACM Trans. Intell. Syst. Technol. (TIST)* **2019**, *10*, 1–42. [CrossRef]
12. Golbeck, J.; Mauriello, M.; Auxier, B.; Bhanushali, K.H.; Bonk, C.; Bouzaghrane, M.A.; Buntain, C.; Chanduka, R.; Cheakalos, P.; Everett, J.B.; et al. *Fake News vs Satire: A Dataset and Analysis*; WebSci '18; Association for Computing Machinery: New York, NY, USA, 2018; pp. 17–21. [CrossRef]
13. Shu, K.; Sliva, A.; Wang, S.; Tang, J.; Liu, H. Fake news detection on social media: A data mining perspective. *ACM SIGKDD Explor. Newslett.* **2017**, *19*, 22–36. [CrossRef]
14. Chen, Y.; Conroy, N.J.; Rubin, V.L. Misleading online content: Recognizing clickbait as false news. In Proceedings of the 2015 ACM on Workshop on Multimodal Deception Detection, Association for Computing Machiner, New York, NY, USA, 13 November 2015; pp. 15–19. [CrossRef]
15. Shu, K.; Mahudeswaran, D.; Wang, S.; Lee, D.; Liu, H. FakeNewsNet: A Data Repository with News Content, Social Context, and Spatiotemporal Information for Studying Fake News on Social Media. *Big Data* **2020**, *8*, 171–188. [CrossRef]
16. Fuller, C.M.; Biros, D.P.; Wilson, R.L. Decision support for determining veracity via linguistic-based cues. *Decis. Support Syst.* **2009**, *46*, 695–703. [CrossRef]
17. Sharma, S.; Sharma, D.K. Fake News Detection: A long way to go. In Proceedings of the 2019 4th International Conference on Information Systems and Computer Networks (ISCON), IEEE, Mathura, UP, India, 21–22 November 2019; pp. 816–821. [CrossRef]
18. Davis, C.A.; Varol, O.; Ferrara, E.; Flammini, A.; Menczer, F. BotOrNot: A System to Evaluate Social Bots. In *Proceedings of the 25th International Conference Companion on World Wide Web; WWW '16 Companion*; International World Wide Web Conferences Steering Committee: Geneva, Switzerland, 2016; pp. 273–274. [CrossRef]
19. Mattos, D.M.F.; Velloso, P.B.; Duarte, O.C.M.B. An agile and effective network function virtualization infrastructure for the Internet of Things. *J. Internet Serv. Appl.* **2019**, *10*, 6. [CrossRef]
20. Kwon, S.; Cha, M.; Jung, K.; Chen, W.; Wang, Y. Prominent features of rumor propagation in online social media. In *2013 IEEE 13th International Conference on Data Mining*, IEEE, New York, NY, USA, 7–10 December 2013; pp. 1103–1108. [CrossRef]

21. Zhou, X.; Cao, J.; Jin, Z.; Xie, F.; Su, Y.; Chu, D.; Cao, X.; Zhang, J. Real-time news certification system on Sina Weibo. In Proceedings of the 24th International Conference on World Wide Web, Florence, Italy, 18–22 May 2015; pp. 983–988. [CrossRef]

22. Wang, X.; Kang, Q.; An, J.; Zhou, M. Drifted Twitter Spam Classification Using Multiscale Detection Test on K-L Divergence. *IEEE Access* **2019**, *7*, 108384–108394. [CrossRef]

23. Santia, G.C.; Williams, J.R. Buzzface: A news veracity dataset with facebook user commentary and egos. In *Twelfth International AAAI Conference on Web and Social Media*; AAAI: Menlo Park, CA, USA, 2018; pp. 531–540.

24. Monteiro, R.A.; Santos, R.L.; Pardo, T.A.; de Almeida, T.A.; Ruiz, E.E.; Vale, O.A. 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*; Springer: Berlin, Germany, 2018; pp. 324–334. [CrossRef]

25. Ferreira, W.; Vlachos, A. Emergent: A novel data-set for stance classification. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego, CA, USA, 12–17 June 2016; pp. 1163–1168. [CrossRef]

26. Thorne, J.; Vlachos, A.; Christodoulopoulos, C.; Mittal, A. FEVER: A Large-scale Dataset for Fact Extraction and VERification. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, New Orleans, LA, USA, 1–6 June 2018; Volume 1, pp. 809–819. [CrossRef]

27. Mitra, T.; Gilbert, E. Credbank: A large-scale social media corpus with associated credibility annotations. In *ICWSM*; AAAI: Menlo Park, CA, USA, 2015; pp. 258–267.

28. Potthast, M.; Kiesel, J.; Reinartz, K.; Bevendorff, J.; Stein, B. A stylometric inquiry into hyperpartisan and fake news. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*; ACL: Melbourne, Australia, July 2018; Volume 1, pp. 231–240. [CrossRef]

29. Zubiaga, A.; Liakata, M.; Procter, R.; Wong Sak Hoi, G.; Tolmie, P. Analysing how people orient to and spread rumours in social media by looking at conversational threads. *PLoS ONE* **2016**, *11*, e0150989. [CrossRef]

30. Rashkin, H.; Choi, E.; Jang, J.Y.; Volkova, S.; Choi, Y. Truth of varying shades: Analyzing language in fake news and political fact-checking. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, Copenhagen, Denmark, 9–11 September 2017; pp. 2931–2937. [CrossRef]

31. Oshikawa, R.; Qian, J.; Wang, W.Y. A survey on natural language processing for fake news detection. In *Proceedings of the 12th Language Resources and Evaluation Conference (LREC)*; European Language Resources Association, Marseille, France, May 2018; pp. 6086–6093.

32. Rubin, V.L. Pragmatic and cultural considerations for deception detection in Asian Languages. *ACM Trans. Asian Lang. Inf. Process.* **2014**, *13*.10.1145/2605292. [CrossRef]

33. Clark, M.; Kim, Y.; Kruschwitz, U.; Song, D.; Albakour, D.; Dignum, S.; Beresi, U.C.; Fasli, M.; De Roeck, A. Automatically structuring domain knowledge from text: An overview of current research. *Inf. Process. Manag.* **2012**, *48*, 552–568. [CrossRef]

34. Otter, D.W.; Medina, J.R.; Kalita, J.K. A survey of the usages of deep learning for natural language processing. *IEEE Trans. Neural Netw. Learning Syst.* **2020**, 1–21. [CrossRef]

35. Bird, S.; Klein, E.; Loper, E. *Natural Language Processing with Python*, 1st ed.; O'Reilly Media, Inc.: Sebastopol, CA, USA, 2009.

36. Zhou, L.; Burgoon, J.K.; Nunamaker, J.F.; Twitchell, D. Automating linguistics-based cues for detecting deception in text-based asynchronous computer-mediated communications. *Group Decis. Negotiat.* **2004**, *13*, 81–106. [CrossRef]

37. Afroz, S.; Brennan, M.; Greenstadt, R. Detecting hoaxes, frauds, and deception in writing style online. In *2012 IEEE Symposium on Security and Privacy*; IEEE: New York, NY, USA, 2012; pp. 461–475. [CrossRef]

38. Hauch, V.; Blandón-Gitlin, I.; Masip, J.; Sporer, S.L. Are computers effective lie detectors? A meta-analysis of linguistic cues to deception. *Personal. Soc. Psychol. Rev.* **2015**, *19*, 307–342. [CrossRef] [PubMed]

39. Indurkhya, N.; Damerau, F.J. *Handbook of Natural Language Processing*, 2nd ed.; Chapman & Hall/CRC: Boca Raton, FL, USA, 2010.

40. de Oliveira, N.R.; Reis, L.H.; Fernandes, N.C.; Bastos, C.A.M.; de Medeiros, D.S.V.; Mattos, D.M.F. Natural Language Processing Characterization of Recurring Calls in Public Security Services. In Proceedings of the 2020 International Conference on Computing, Networking and Communications (ICNC), Big Island, HI, USA, 17–20 February 2020; pp. 1009–1013. [CrossRef]

41. Navigli, R. Word sense disambiguation: A survey. *ACM Comput. Surv. (CSUR)* **2009**, *41*, 1–69. [CrossRef]

42. Manning, C.; Schutze, H. *Foundations of Statistical Natural Language Processing*; MIT Press: Cambridge, MA, USA, 1999.

43. Socher, R.; Perelygin, A.; Wu, J.; Chuang, J.; Manning, C.D.; Ng, A.Y.; Potts, C. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics: Seattle, WA, USA, 2013; pp. 1631–1642.

44. Pennebaker, J.W.; Francis, M.E.; Booth, R.J. *Inquiry and Word Count: LIWC 2001.* Lawrence Erlbaum Associates: Mahwah, NJ, USA, 2001; Volume 71.

45. Balage Filho, P.; Pardo, T.A.S.; Aluísio, S. An evaluation of the Brazilian Portuguese LIWC dictionary for sentiment analysis. In Proceedings of the 9th Brazilian Symposium in Information and Human Language Technology (STIL). ACL, Fortaleza, Brazil, 21–23 October 2013.

46. Manning, C.; Raghavan, P.; Schütze, H. Introduction to information retrieval. *Natural Lang. Eng.* **2010**, *16*, 100–103.

47. Jarmasz, M.; Szpakowicz, S. Not as Easy as It Seems: Automating the Construction of Lexical Chains Using Roget's Thesaurus. *Advances in Artificial Intelligence*; Springer: Berlin/Heidelberg, Germany, 2003; pp. 544–549.

48. Camacho-Collados, J.; Pilehvar, M.T. From word to sense embeddings: A survey on vector representations of meaning. *J. Artif. Intell. Res.* **2018**, *63*, 743–788. [CrossRef]

49. Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient estimation of word representations in vector space. In Proceedings of the ICLR Workshop Papers, Scottsdale, AZ, USA, 2–4 May 2013.

50. Firth, J.R. A synopsis of linguistic theory, 1930–1955. *Studi. Linguist. Anal.* **1957**, *1*, 168–205.

51. Li, Y.; Xu, L.; Tian, F.; Jiang, L.; Zhong, X.; Chen, E. Word embedding revisited: A new representation learning and explicit matrix factorization perspective. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*; AAAI Press: Buenos Aires, Argentina, 2015; pp. 3650–3656.

52. Hu, B.; Tang, B.; Chen, Q.; Kang, L. A novel word embedding learning model using the dissociation between nouns and verbs. *Neurocomputing* **2016**, *171*, 1108–1117. [CrossRef]

53. Boutaba, R.; Salahuddin, M.A.; Limam, N.; Ayoubi, S.; Shahriar, N.; Estrada-Solano, F.; Caicedo, O.M. A comprehensive survey on machine learning for networking: Evolution, applications and research opportunities. *J. Internet Serv. Appl.* **2018**, *9*, 16. [CrossRef]

54. Domingos, P.M. A few useful things to know about machine learning. *ACM Commun.* **2012**, *55*, 78–87. [CrossRef]

55. Ayodele, T.O. Types of machine learning algorithms. In *New Advances in Machine Learning*; IntechOpen: London, UK, 2010. [CrossRef]

56. Zhai, Y.; Ong, Y.S.; Tsang, I.W. The emerging "big dimensionality". *IEEE Comput. Intell. Mag.* **2014**, *9*, 14–26. [CrossRef]

57. Andreoni Lopez, M.; Mattos, D.M.F.; Duarte, O.C.M.B.; Pujolle, G. A fast unsupervised preprocessing method for network monitoring. *Ann. Telecommun.* **2019**, *74*, 139–155. [CrossRef]

58. Papadimitriou, C.H.; Raghavan, P.; Tamaki, H.; Vempala, S. Latent semantic indexing: A probabilistic analysis. *J. Comput. Syst. Sci.* **2000**, *61*, 217–235. [CrossRef]

59. Deerwester, S.; Dumais, S.T.; Furnas, G.W.; Landauer, T.K.; Harshman, R. Indexing by latent semantic analysis. *J. Am. Soc. Inf. Sci.* **1990**, *41*, 391–407. [CrossRef]

60. Chandrashekar, G.; Sahin, F. A survey on feature selection methods. *Comput. Electr. Eng.* **2014**, *40*, 16–28. [CrossRef]

61. Robnik-Šikonja, M.; Kononenko, I. Theoretical and Empirical Analysis of ReliefF and RReliefF. *Mach. Learn.* **2003**, *53*, 23–69. [CrossRef]

62. Perdisci, R.; Gu, G.; Lee, W. Using an Ensemble of One-Class SVM Classifiers to Harden Payload-based Anomaly Detection Systems. In Proceedings of the Sixth International Conference on Data Mining (ICDM'06), Hong Kong, China, 18–22 December 2006; pp. 488–498. [CrossRef]

63. Gaonkar, S.; Itagi, S.; Chalippatt, R.; Gaonkar, A.; Aswale, S.; Shetgaonkar, P. Detection Of Online Fake News: A Survey. In Proceedings of the 2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN), Vellore, India, 30–31 March 2019; pp. 1–6. [CrossRef]

64. Verikas, A.; Gelzinis, A.; Bacauskiene, M. Mining data with random forests: A survey and results of new tests. *Pattern Recogni.* **2011**, *44*, 330–349. [CrossRef]

65. Kadhim, A.I. Survey on supervised machine learning techniques for automatic text classification. *Artif. Intell. Rev.* **2019**, *52*, 273–292. [CrossRef]

66. Xu, R.; Wunsch, D. Survey of clustering algorithms. *IEEE Trans. Neural Netw.* **2005**, *16*, 645–678. [CrossRef] [PubMed]

67. Fahad, A.; Alshatri, N.; Tari, Z.; Alamri, A.; Khalil, I.; Zomaya, A.Y.; Foufou, S.; Bouras, A. A survey of clustering algorithms for big data: Taxonomy and empirical analysis. *IEEE Trans. Emerg. Top. Comput.* **2014**, *2*, 267–279. [CrossRef]

68. Ketchen, D.J.; Shook, C.L. The application of cluster analysis in strategic management research: an analysis and critique. *Strateg. Manag. J.* **1996**, *17*, 441–458. [CrossRef]

69. Rousseeuw, P.J.; Kaufman, L. *Finding Groups in Data*; Wiley Online Library: Hoboken, NJ, USA, 1990; [CrossRef]

70. Gan, J.; Tao, Y. DBSCAN revisited: Mis-claim, un-fixability, and approximation. In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*; Association for Computing Machinery, Melborne, Australia, May 2015; pp. 519–530. [CrossRef]

71. Schubert, E.; Sander, J.; Ester, M.; Kriegel, H.P.; Xu, X. DBSCAN revisited, revisited: Why and how you should (still) use DBSCAN. *ACM Trans. Database Syst. (TODS)* **2017**, *42*, 1–21. [CrossRef]

72. Benavent, X.; Castellanos, A.; de Ves, E.; Garcia-Serrano, A.; Cigarran, J. FCA-based knowledge representation and local generalized linear models to address relevance and diversity in diverse social images. *Future Gener. Comput. Syst.* **2019**, *100*, 250–265. [CrossRef]

73. Govender, P.; Sivakumar, V. Application of k-means and hierarchical clustering techniques for analysis of air pollution: A review (1980–2019). *Atmos. Pollut. Res.* **2020**, *11*, 40–56. [CrossRef]

74. Lazer, D.M.; Baum, M.A.; Benkler, Y.; Berinsky, A.J.; Greenhill, K.M.; Menczer, F.; Metzger, M.J.; Nyhan, B.; Pennycook, G.; Rothschild, D.; et al. The science of fake news. *Science* **2018**, *359*, 1094–1096. [CrossRef]

75. Grinberg, N.; Joseph, K.; Friedland, L.; Swire-Thompson, B.; Lazer, D. Fake news on Twitter during the 2016 US presidential election. *Science* **2019**, *363*, 374–378. [CrossRef]

76. Pennycook, G.; McPhetres, J.; Zhang, Y.; Lu, J.G.; Rand, D.G. Fighting COVID-19 misinformation on social media: Experimental evidence for a scalable accuracy-nudge intervention. *Psychol. Sci.* **2020**, *31*, 770–780. [CrossRef]

77. Van Bavel, J.J.; Baicker, K.; Boggio, P.S.; Capraro, V.; Cichocka, A.; Cikara, M.; Crockett, M.J.; Crum, A.J.; Douglas, K.M.; Druckman, J.N.; et al. Using social and behavioural science to support COVID-19 pandemic response. *Nat. Hum. Behav.* **2020**, 1–12. [CrossRef]

78. Poddar, K.; Umadevi, K.; Amali, G.B. Comparison of Various Machine Learning Models for Accurate Detection of Fake News. In *Proceedings of the 2019 Innovations in Power and Advanced Computing Technologies (i-PACT)*, IEEE, Vellore, India, 22–23 March 2019; Volume 1, pp. 1–5. [CrossRef]

79. Shu, K.; Mahudeswaran, D.; Liu, H. FakeNewsTracker: A tool for fake news collection, detection, and visualization. *Comput. Math. Organ. Theory* **2019**, *25*, 60–71. [CrossRef]

80. Bondielli, A.; Marcelloni, F. A survey on fake news and rumour detection techniques. *Inf. Sci.* **2019**, *497*, 38–55. [CrossRef]

81. Deepak, S.; Chitturi, B. Deep neural approach to Fake-News identification. *Procedia Comput. Sci.* **2020**, *167*, 2236–2243. [CrossRef]

82. Qian, F.; Gong, C.; Sharma, K.; Liu, Y. Neural User Response Generator: Fake News Detection with Collective User Intelligence. In Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, (IJCAI-18), Stockholm, Sweden, 13–19 July 2018; pp. 3834–3840. [CrossRef]

83. Friggeri, A.; Adamic, L.; Eckles, D.; Cheng, J. Rumor Cascades. In Proceedings of the International AAAI Conference on Web and Social Media, Ann Arbor, MI, USA, 1–4 June 2014; pp. 1–10.

84. Meel, P.; Vishwakarma, D.K. Fake news, rumor, information pollution in social media and web: A contemporary survey of state-of-the-arts, challenges and opportunities. *Expert Syst. Appl.* **2020**, *153*, 112986. [CrossRef]

85. de Oliveira, N.R.; de Medeiros, D.S.V.; Mattos, D.M.F. A Syntactic-Relationship Approach to Construct Well-Informative Knowledge Graphs Representation. In Proceedings of the 4th Cloud and Internet of Things (CIoT'20), Niterói, Brazil, 7–12 October 2020; pp. 75–82. [CrossRef]

86. Zhou, Z.; Guan, H.; Bhat, M.M.; Hsu, J. Fake news detection via NLP is vulnerable to adversarial attacks. In Proceedings of the 11th International Conference on Agents and Artificial Intelligence (ICAART), Prague, Czech Republic, 19–21 February 2019; pp. 794–800. [CrossRef]

87. Giglio, C.; Palmieri, R. Analyzing Informal Learning Patterns in Facebook Communities of International Conferences. *Procedia Soc. Behav. Sci.* **2017**, *237*, 223–229. [CrossRef]