Fake News Websites

Diogo Miguel Ferreira Pinheiro

Thesis to obtain the Master of Science Degree in

Information Systems and Computer Engineering

Supervisors: Prof. Maria Luísa Torres Ribeiro Marques da Silva Coheur
Prof. João Paulo Baptista de Carvalho

Examination Committee

Chairperson: Prof. Manuel Fernando Cabido Peres Lopes
Supervisor: Prof. Maria Luísa Torres Ribeiro Marques da Silva Coheur
Member of the Committee: Prof. Bruno Emanuel Da Graça Martins

October 2021

Acknowledgments

I would like to thank my parents for believing in me, for giving me hope and encouragement.

A special thanks for my two supervisors, João Paulo Carvalho e Luísa Coheur, who helped me in this project, even though I was not good at communicating, when I needed help they were there to help.

I also would like to thank Paulo Pena for giving guidance on the start of this project.

Last, but not least, I want to give a huge thank to all my friends that were there for me and did not let me give up. Always reassuring that everything would be fine.

Thank you.

Abstract

Fake News have been around us for a long time. With the sudden rise of social media and influx of information from various sources most people can not differentiate a truth from a lie. In this thesis, we approach this problem, focusing in Portuguese news websites. We try to find key elements that distinguish real news websites from fake websites. In this thesis, we proposed a system capable of distinguish trustworthy websites from websites that spread fake, non verified news. The system starts by receiving an url. Then an http request is made and the system tries to find the sub-page with the web news site information. That page is processed and information is extracted. The system also finds other features, like the country localization and the provider. After gathering the features, the evaluator calculates the website score by giving a weight to each feature, marking the website as fake or trustworthy.

We created a data set with fake and trustworthy websites to test our system.

We evaluated our system with different machine learning algorithms and concluded that the most important features are the country that they are hosted and the security protocol.

Keywords

Fake News · Web Scraper · Information Retrieval · Fake Websites

Resumo

Notícias falsas já existem há muito tempo. Com o aumento repentino das mídias sociais e a abundância de informações de várias fontes, a maioria das pessoas não consegue diferenciar a verdade de uma mentira. Nesta tese, abordamos este problema, sendo o foco websites em Portugal. Tentamos encontrar os elementos-chave que distinguem os sites de notícias reais dos falsos. Nesta tese, propomos um sistema capaz de distinguir websites confiáveis de websites que partilham notíciais não verificadas e falsas. O sistema começa por receber um url, que depois faz um pedido http. Quando esse pedido é terminado o sistema procura a sub-página que tem as informaçoes do jornal online. Essa página é processada e informação é extraída. O nosso sistema encontra também outras propriedades como, o país onde o jornal se encontra e o fornecedor de internet. Depois de extrair estas propriedades o avaliador calcula uma pontuação atribuindo a cada propriedade um peso, marcando os websites como falso ou não. Tínhamos um conjunto de dados com sites falsos e confiáveis para testar nosso sistema. Avalíamos o nosso sistema com diferentes algoritmos de aprendizagem automática e concluímos que as propriedades mais importantes dos websites são o país onde estão localizados e o seu protocolo de segurança.

Palavras Chave

Notícias Falsas; Coleção de dados web; Recolhimento de informação; Sítio eletrónico Falsos;

Contents

List of Figures

Listings

Acronyms

ERC Entidade Reguladora de Comunicação

URL Uniform Resource Locator

TP True Positives

TN True Negatives

FP False Positives

FN False Negatives

GA Genetic Algorithm

1

Introduction

Contents

1.1 Motivation

Nowadays with the increase of information, people are prone to stumble on fake information. Having a trustworthy source to get information is a necessary need, however it has been getting more difficult to find.

With the expansion of social media and the ease to create, publish and access information, the amount of posts and articles with news that have emerged makes it impossible to verify the credibility of everything we consume during our day to day life.

There are several websites that publish news and are not allowed to do it. In Portugal in order to being able to publish news you need to have a certification from an entity that regulates media in Portugal.

In other countries this rule may not apply. Each country has its own laws, an example is USA, there is no law that requires certification, however the state of New York requires a certification in order to publish news [1].

1.2 Problem

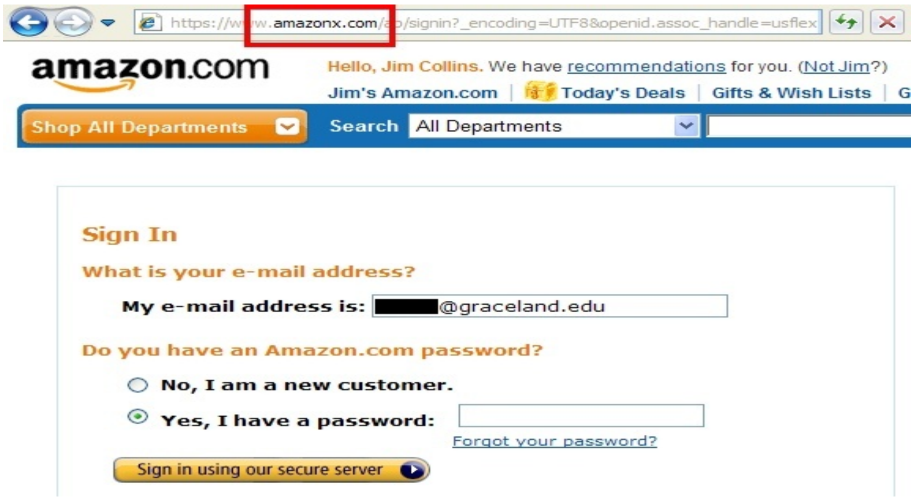
It has become extremely difficult to filter every news post that we come by on our day to day life.

While we scroll in Facebook or Twitter news feeds, we are overwhelm with information and a great part of them are fake. A study [2] conducted in 2016 found that Facebook referred to untrustworthy websites over 15% of the time, in contrast it would only refer to trustworthy websites 6% of the time. Those news serve to catch the user attention and make him click the story, giving the post more and more visibility.

Identifying Fake news is not an exact science where we can always be right, sometimes news we call fake are exaggerating facts and making the information seem drastic. A news story has a lot of nuances and its very difficult to know for sure that the article we are reading is 100% right. Even the most trustworthy journal can use dubious sources or even sources that are wrong. The best defense against Fake News is our judgement and doing research when we read a news story. There are several features and identifiers that we can use to know if the news we are reading is worthy of our attention, specially for our use case in Portugal. A trustworthy source of news has some characteristics that can be used to distinguish from malicious and deceitful news sources. In Portugal exists an identifier that every news outlet has to have in order to publish news. Untrustworthy news websites will not have this identifier.

This identifier is not the only feature that can be used to determine which news source is deem of our attention. We can also use the location of their website, security protocol and provider.

[1] https://dos.ny.gov/certificate-publication-domestic-limited-liability-company-0



https://www.amazonx.com/ap/signin?_encoding=UTF8&openid.assoc_handle=usflex

amazon.com

Hello, Jim Collins. We have recommendations for you. (Not Jim?)

Jim's Amazon.com | Today's Deals | Gifts & Wish Lists | G

Shop All Departments | Search | All Departments

Sign In

What is your e-mail address?

My e-mail address is: [redacted]@graceland.edu

Do you have an Amazon.com password?

○ No, I am a new customer.

● Yes, I have a password:

Forgot your password?

Sign in using our secure server

Figure 1.1: Example of a fake amazon

As we can see in the figure 1.1[2], at the first glance this news website seems legitimate, however there are some sings that we can be on the look out for like, http, banner with the information of the website.

1.3 Goals and Accomplishments

This thesis objective is to research and implement a system that can classify a news website as fake or trustworthy.

The system will be compose of an interface to insert the website to be evaluated, a web scraper that finds important information from the website such as the news group, a feature extractor that retrieves websites features, like the host country and the evaluator that scores the website to decide if the site is fake or not.

This thesis is manly focused in news websites from Portugal, therefore the goal is to distinguished between fake news websites that target the Portuguese readers.

The system was able to get a test accuracy of 81% against a dataset with 51 websites. The system was able to retrieve the technical features successfully, however the context based features like ERC and news group failed on some case websites.

1.4 Outline

In Chapter 2 we discuss previous related work that could address our problems. In Section 2.1 we talk about how we can extract data. In Section 2.2 we identify techniques and features to distinguish between

[2] https://my.graceland.edu/ICS/Resources/Information_Technology/Dont_Get_Scammed.jnz

fake and legitimate websites. In Section 2.3 we discuss how we could search webpages. In Section 2.4 we explore websites blacklists in order to identify quickly already processed websites.

In Chapter 3 we present our system and describe how we built it. In Section 3.1 we show an overview of our system with a picture of the modules. In Section 3.2 we talk about how we would receive the websites to evaluate. In Section 3.3 we describe how our scraper was built. In Section 3.4 and Section 3.5 we talk about how our User Interface and database are build. In Section 3.6 and Section 3.7 the focus is on the websites features that we decide to use and how we find the Entidade Reguladora de Comunicação (ERC) of a news website. Finally in Section 3.8 and Section 3.9 we discuss how our evaluator works and how we did our web deployment.

In Chapter 4 we explain how we manage our evaluation process.

Finally, we wrote our conclusions and future work in Chapter 5.

1.5 Contributions

We were fortunate to be offered a list of fake news websites in Portugal by Prof. Bruno Martins, with that list we created a dataset with fake and legitimate websites. That dataset is composed of twenty nine fake websites and twenty two legitimate websites.

We also created and deployed an website that displays our system, in that website we can insert an url and verify the legitimacy of an website as we explain in our thesis. This system is compose of five modules, first one is the entry point, where is given an url for a website to be evaluated, the second module is the web scraper where the system looks for information in the website pages, the feature extractor is our third module, is in this module where we extract the website features, our fourth module scores the website giving it a value between 0-100, the closer an website is to 100 the more likely to be fake. Lastly there is a database where we save all our results.

2

Related Work

Contents

The detection of Fake News is the subject of many different types of research and articles. Whether using natural language processing methods or using knowledge-based systems. In our case we want to identify Portuguese websites that spread fake news.

As previously mentioned, in order to solve this problem we had to divide it into smaller ones. Therefore we divided this section in four parts.

In Section 2.1, we discuss how we can obtain website links to be evaluated.

In Section 2.2, we talk about different ways to test if a website is trustworthy. We also address various approaches to verify if a news article is fake.

In Section 2.3, we present and compare different types of Web Crawlers and how we can built one. It is given an example of a Web Crawler that was used to gather reliable and efficiently health information.

In the last Section, 2.4, we discuss how to create a list with fake websites.

2.1 Gathering Data

Fake news media is everywhere, and is especially dominant in today's social media where most people tend to get their news. To achieve their goal fake news need to be shared, and Social Media is the easiest way to spread fake news and has the potential to cause serious negative impacts [3]. It is also important to refer that many of the accounts that publish fake news are made by bots. These automated accounts are active in spreading fake news, and target users that are most likely to influence other people [4].

To gather these websites the most efficient is to collect tweets and Facebook posts created by users. Due to the enormous amount of data that this two platforms provide. More than 500 million tweets and 400 million Facebook posts are made every day [5] [1].

Articles on the subject of Fake News on social media collect their data with annotations in different ways. They use expert journalists, fact-checking websites and crowd-sourced workers. Using these methods, in order to get a sizable dataset to work on, is hard, time consuming and costly.

To overcome these problems, datasets publicly available are used. Such as CREDBANK, which has over 60 million tweets. BuzzFeedNews is another dataset, with news posts from facebook [3].

It is also possible to obtain our data. To gather tweets, we can use a Twitter API. With the Twitter API, it is possible to obtain live tweets and tweets already made [6]. The Facebook Graph API is also used. It is a HTTP based API that applications can query data, which helps to get Facebook posts [2].

[1] http://www.internetlivestats.com/twitter-statistics~

[2] https://developers.facebook.com/docs/graph-api/

2.2 Legitimate vs Fake website

The hardest and most crucial thing to do is to identify fake news Websites. On the surface we can have an apparently legitimate website, with well formed news and a pleasant design to make users believe that the website is trustworthy. Where do we start to know if a news website is fake or not?

There are many different ways to identify the veracity of a news and their websites, as explained in [7] and [8]:

- **Linguistic** where the data (news) is extracted and analyzed for linguistic patterns. Natural language is often used for this task.
- **Network** where the social media information is used. User behavior is analyzed. The source of the rumor is also identified, if possible.
- **Structural** analyze the structure of the websites that publish the news and analyze the structure of the news.

Within the linguistic approach, as referred in [7], data representation can be "bag of words". With this, a single word or a set of sequential words called "n-grams" can help to determine if the article is fake or not. It is possible to improve this method with "shallow syntax"[3]. "Shallow Syntax" in natural language process is a method of analyzing syntactic structure of the text. This approach is simple, however, as it uses only single word or n-grams. Its analyzes are not very accurate nor has useful context information. It is better when paired with other methods.

Still on Linguistic, exists Deep Syntax. This method goes further than the one mentioned before. It utilizes grammar to predict deception, and is implemented through a Probability Context Free Grammar. Sentences are transformed in a set of rules, and are placed in a parse tree(representing grammar), in the Figure 2.1 [4] we can see an example of a parse tree. These rules will have a probability assigned to catch fake information.

Semantic Analysis is other approach under linguist category. This method takes in to consideration contradictions and lack of facts in the story. If the news article contradicts it self or does not have enough facts, it is marked as fake news. These tests prove the lack of experience and deep understanding of the subject from the writer. This is possible by comparing other texts of similar topics, a common example is reviews from restaurants and stores. A real review will have similar facts with other reviews, and a fake will have nothing in common to other reviews. Some problems with this approach is the possibility of multiple fake reviews with congruent facts which will temper with this solution. One common example is reviews on a website, if a great number of people makes a bad, or good, review writing the same untruthful story, will make this approach fail.

[3] http://nlpprogress.com/english/shallow_syntax.html

[4] https://www.wikiwand.com/en/Parse_tree

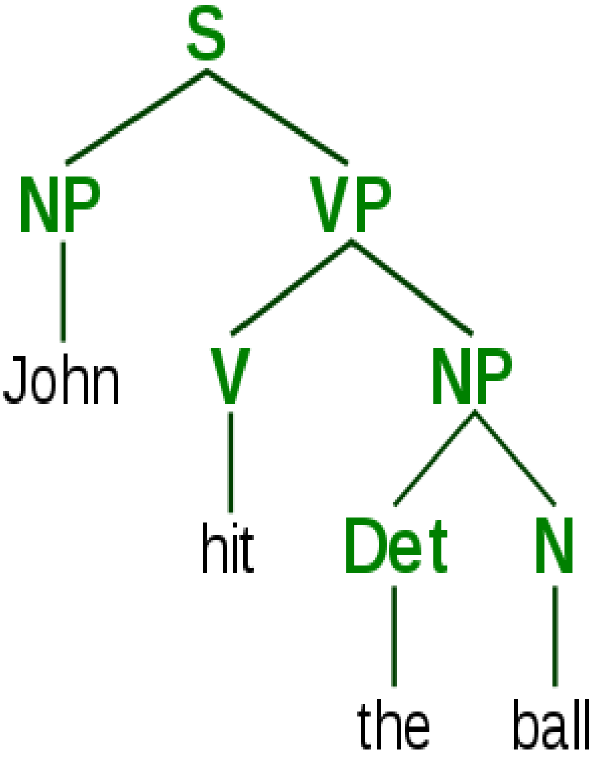


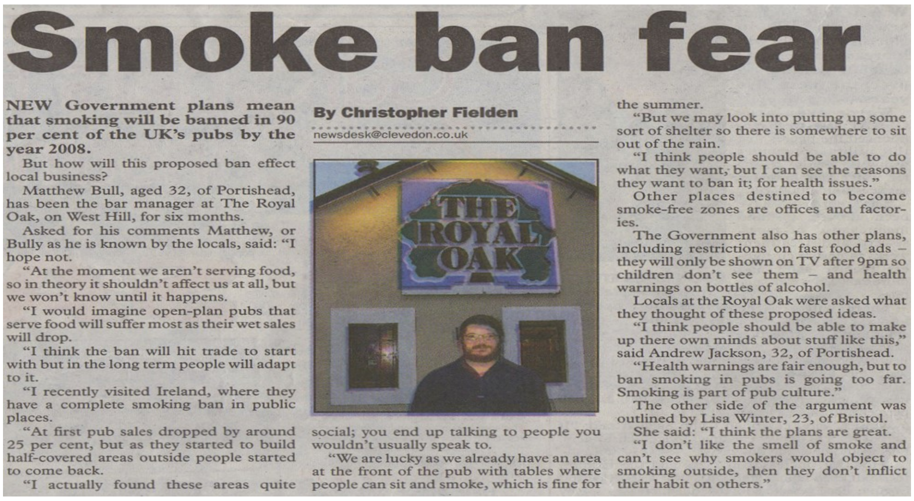
Figure 2.1: Parse Tree

In Network Approaches it is possible to find methods like the use of Linked Data, as mentioned in [7]. The Linked Data method uses human knowledge already collected to evaluate a new statement of their veracity. There are several database of knowledge such as DBpedia ontology[5], as referred in [7]. The Linked Data fact checking analysis is a network that finds the shortest path to the fact being tested. The nodes from the path are then evaluated. If the path has high-degree nodes, representing generic information statement, it is classified as false.

Other method under the Network approaches is the Social Network behaviour as mentioned in [7]. A considerable amount of information is published through blogs with sensationalists headliners that are spread and shared in Social Media like Facebook and Twitter. This method uses the metadata and behaviour of questionable sources to dismiss fake news. This system can be effective in influencing political perceptions. Using hyperlinks and associated metadata that can be resorted to check if the information is truthful.

An example exposed by the article [9] there was a story reported during Hurricane Harvey about looters targeting Trump supporters. The story was not fact checked and since then has been debunked. It is emotionally charged (message characteristics), the tag line of this outlet: “airing out America’s dirty laundry”, reveals the source is not a mainstream news organization, and the sources within the story

[5] https://wiki.dbpedia.org/services-resources/ontology



Smoke ban fear

NEW Government plans mean that smoking will be banned in 90 per cent of the UK's pubs by the year 2008.

But how will this proposed ban effect local business?

Matthew Bull, aged 32, of Portishead, has been the bar manager at The Royal Oak, on West Hill, for six months.

Asked for his comments Matthew, or Bully as he is known by the locals, said: "I hope not.

"At the moment we aren't serving food, so in theory it shouldn't affect us at all, but we won't know until it happens.

"I would imagine open-plan pubs that serve food will suffer most as their wet sales will drop.

"I think the ban will hit trade to start with but in the long term people will adapt to it.

"I recently visited Ireland, where they have a complete smoking ban in public places.

"At first pub sales dropped by around 25 per cent, but as they started to build half-covered areas outside people started to come back.

"I actually found these areas quite social; you end up talking to people you wouldn't usually speak to.

"We are lucky as we already have an area at the front of the pub with tables where people can sit and smoke, which is fine for the summer.

"But we may look into putting up some sort of shelter so there is somewhere to sit out of the rain.

"I think people should be able to do what they want, but I can see the reasons they want to ban it; for health issues."

Other places destined to become smoke-free zones are offices and factories.

The Government also has other plans, including restrictions on fast food ads – they will only be shown on TV after 9pm so children don't see them – and health warnings on bottles of alcohol.

Locals at the Royal Oak were asked what they thought of these proposed ideas.

"I think people should be able to make up there own minds about stuff like this," said Andrew Jackson, 32, of Portishead.

"Health warnings are fair enough, but to ban smoking in pubs is going too far. Smoking is part of pub culture."

The other side of the argument was outlined by Lisa Winter, 23, of Bristol.

She said: "I think the plans are great.

"I don't like the smell of smoke and can't see why smokers would object to smoking outside, then they don't inflict their habit on others."

By Christopher Fielden

newsdesk@clevedon.co.uk

Figure 2.2: Example of a news article

include Twitter posts, many later identified as belonging to accounts with a history of disseminating misinformation and deleted since (structural and sources characteristics).

However as mentioned in [7], these methods have better results when not used separately but in combination with others.

The structure of a news article can be in some cases a telltale of a false news. That is because a news article includes a predefined composition. Without that structure we can assume with some confidence that the information being read is not from a trusted source. A well formed article can be represented with this attributes [6] [6]:

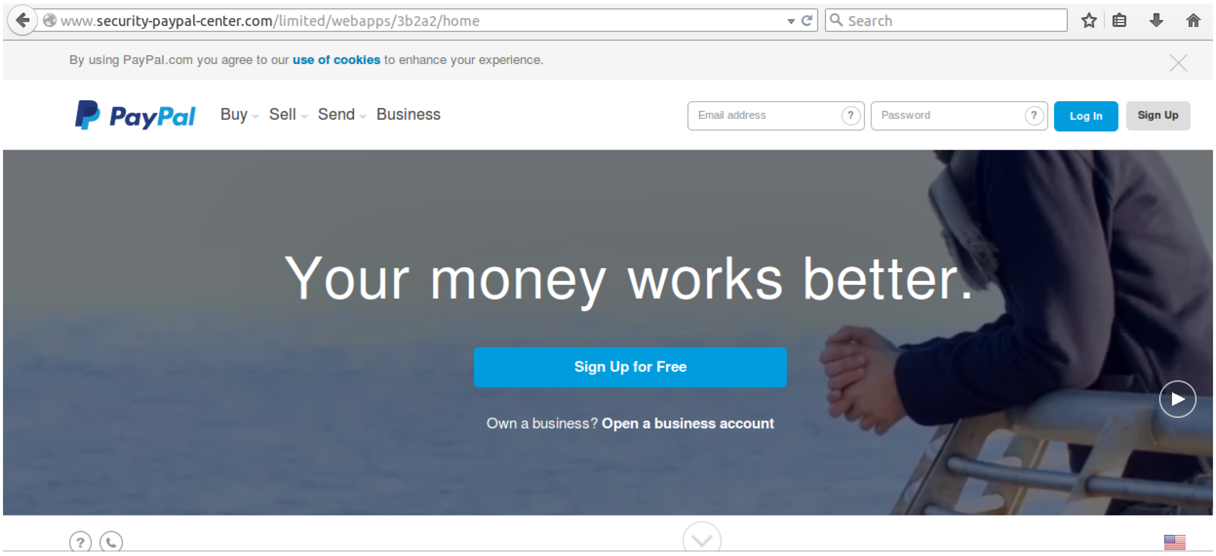
- **Source**: Author of the publication
- **Headline**: A title that captivates the reader attention
- **Lead**: Introductory statement
- **Body**: The story of the news, In this part lies the main text that should have facts
- **Visual Content**: Sometimes the body of the publication has a video or image to attract the reader or make a point more evident

As we can see in Figure 2.2 [7] these five characteristics are almost always present in a news article either online or in a classic newspaper.

With this it is possible to filter the websites in which the news they publish do not satisfy this structure. However this is not a good way to identify websites that spread Fake News. Since a news article can be structured in this way and it would still be used to pass false information.

[6] https://sites.google.com/a/wjps.org/the-blazer---newspaper-class/news-writing-resources/news-structure

[7] https://www.christopherfielden.com/short-story-tips-and-writing-advice/newspaper-articles.php



www.security-paypal-center.com/limited/webapps/3b2a2/home

By using PayPal.com you agree to our use of cookies to enhance your experience.

Figure 2.3: Fake Paypal website

The media is not the only area to be affected with misinformation. Fake sites exists not only to spread fake articles and information but also to create Web Spam sites, concocted sites and spoof sites. Web Spam [8] tries to manipulate search engines rankings for specific keywords, so their sites become the top search. Their goal is to sell the website, since has more visibility than the other ones. Concocted Websites try to look legitimate, like a fake bank or online shop. Their objective is to steal money and information from users. Spoof Websites try to copy a legitimate website, like PayPal, and scam users into thinking that they are in the correct website.

In the Figure 2.3[9] we see a spoof PayPal website, nearly identical to the real Paypal website, however it is possible to identify that is fake because of the URL link.

As previously said not only news are affected with fake websites but also the health department. They often deal with inaccurate and misleading information and advice. In some of these websites they also sell counterfeit drugs, as mentioned in [10].

There are multiple tools that tackle this problem as mentioned in [8], either with *Lookup* techniques that require users reports to create blacklists websites. Or *Classifiers Systems* that use simplistic metric features and classification heuristics. In [8] we have a comparison of several of these tools with different techniques, that we will discuss next. We also discuss a tool created by their study, AZProtect.

Lookup Systems such as the *Microsoft IE Phishing Filter*, *Mozilla Firefox FireFish* use this approach where there is a server side blacklist with URLs with the fake websites. Usually this blacklist is populated with help of online communities and users of the systems.

The advantage of this kind of systems is that they have high precision, i.e., they do not mark a

[8] https://www.crazyegg.com/blog/glossary/web-spam/
[9] https://umbrella.cisco.com/blog/2015/02/11/paypal-phishing-sophistication-growing/

legitimate website as a fake one. And they are also easier to implement than a *classifier system*, since checking a list for an URL to confirm is authenticity is relatively easy. However the blacklists have older websites rather than new ones. Because they are not always being updated, and fake websites can appear before being introduce in the blacklist.

Classifiers Systems are client-side tools applying rules or heuristics to websites content or domain registration. Several systems were made, like *Netcraft* [10] that uses domain registration information (for instance, the country where it is hosted, registration date and host name). Another example is *SpoofGuard*, as mentioned in [11], that uses image hashes, password encryption verification and URL similarities. *SpoofGuard* compares image hashes with other images hashes taken from honest websites. For example, if the Amazon logo appears in a web page that do not belong to Amazon, that web page will be consider suspicious. *SpoofGuard* also verifies the URL by searching for the character "@". Because, a "@" is in a URL the string to the left is ignored and the user goes to the adress that is after the "@". For example, if we have www.google.com"@"www.facebook.com when clicking that link we browse to Facebook.

Unlike the *Lookup Systems*, classifiers Systems are not dependent of Blacklists therefore they can proactively verify if a website is fake or not. Nevertheless, they are usually slower to verify the web page than the *Lookup Systems* and have fake positives results.

There are also Hybrid Systems that use both approaches mention above. The Hybrid System uses a blacklist to block URIs, and a classifier system to identify other fake websites.

AZProtect is a tool mentioned in [10] based on the preceding systems that uses several attributes, such as:

- **Inlinks**: Legitimate websites have other websites with links pointing at them.
- **Outlinks**: Fake Websites tend to have fewer web pages and therefore less links.
- **HTTPS**: Fake Websites do not use Secure Sockets Layers Protocol.
- **Language**: Usually a fake website has multiple versions in different languages.
- **Hosting**: Fake Websites are normally hosted on free platforms.

Each of these attributes has a weight value between 0-1. AZProtect has a classification model with a SVM composed of a page-level classifier and a site-level classifier. The page-level classifier compares pages features vectors, against training pages. Computes the average and maximum similarity for pattern and duplicate detection. Having a web page *a* compares against all web pages belonging to *b* (*b* is a website belonging to a data set with fake and real websites this page). *a* is then given a score between 0-1 where 1 is identical and 0 completely different. Each *a* page results in a vector of similarity

[10] https://www.netcraft.com/

scores of length k, and there is one vector for every b. For each one of this vectors the average and maximum similarity score is calculated. Resulting on a page-site similarity vector for each web page.

The site-level classifier receives as input the total number of pages classified, the number of pages classified as fake and the percentage of pages classified as fake. With this information the website is then labeled either fake or legitimate.

As our goal is to find Portuguese websites that spread fake news. Because of that we talk with a journalist from *Diário de Notícias*, Paulo Pena, that explained his process to identify fake news. He also suggested news articles about fake news.

In [12], an article published by Portuguese newspaper *Diário de Notícias*, we can see, that in Portugal, in order to publish news you need to be registered in the *Entidade Reguladora para a Comunicação Social*, ERC for short. ERC regulates all the media, Television, Radio and the Press.

A Portuguese News Website is obliged to be registered in the ERC authority entity and in the web page *Ficha Técnica* is the journal ERC number or their editor name, that has to be registered in the ERC database. The ERC number is an important and trustworthy indicator that a News Website from Portugal is legitimate and do not have second intentions besides stating real news.

The article [12] also states that the referred website shares the same IP address to other known Fake News Websites that were previously discovered.

Is also very common for the fake Portuguese websites to be hosted in other countries rather than Portugal [13]. An example given is the website *Direita Política* that is hosted in Canada by the company iWeb[11]. This fact can also be use as an indicator that the website is suspicious.

In these news articles they are not using an automatic method to retrieve the host of the websites, they are searching for the host manually.

2.3 Searching Web Pages

On the previous section we reviewed the features that differentiate a fake website from a legitimate one. Now we need to obtain the website information which implies, a need to navigate through their web pages. In [14] the authors discuss techniques and tools for information retrieval and refer to Web Crawlers. [12] Web Crawlers are computers programs that navigate through a website until all the web pages that exist on that website are indexed, therefore collecting all the hyperlinks.

In [15] was proposed a web Crawler that searches the web for patient-driven solutions. Multiple types of crawlers were discussed, and a web crawler architecture was defined.

As seen in the Figure 2.4, taken from [1], a web crawler is composed by:

[11] https://iweb.com/

[12] https://www.techopedia.com/definition/10008/web-crawler

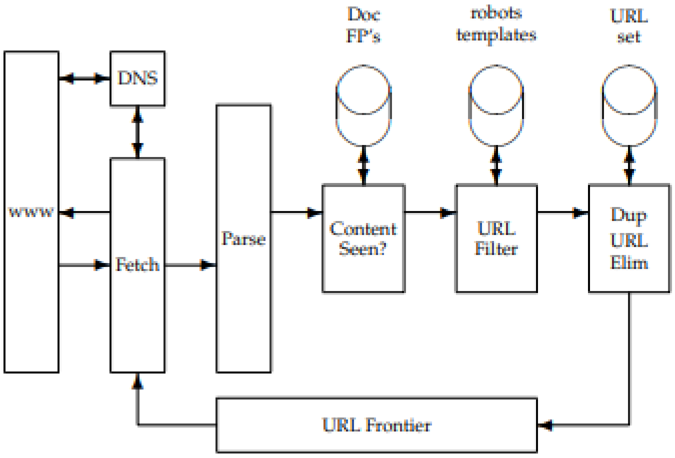


Figure 2.4: Architecture of a web crawler [1]

- **Uniform Resource Locator (URL) Frontier**: Stores and orders the URLs to be processed.
- **Fetcher**: Gets the URLs given by the frontier following the communication protocols HTTP, HTTPS and FTP
- **DNS Resolver**: Interacts with the fetcher for the web resource name resolution
- **Parser**: Extracts information on the web resources pointed by the fetched URLs and extracts URLs inside that resources
- **Duplicate Content Detection**: Detects repeating content by interacting with a storing system that contains signatures of already processed documents
- **URL Filter**: Filters URLs from that website.
- **URL Duplicate**: Stops URLs from being added to the frontier if they were already crawled.

Due to the necessity for gathering efficiently different types of information, several types of Web Crawlers exists [16].

- **Universal Crawler**: not limited to web pages of a particular website, this crawler keeps following links and gets all web pages they encounter.
- **Focused Crawler**: the user enters a condition or a topic that guides the crawler, getting web pages relevant with the topic chosen.
- **Hidden Web Crawler**: there is a part of the web that is not indexed by traditional crawlers, deep web, because there is no hyperlinks to access. Usually they build meaningful queries to access that information through queries interfaces.

- **Mobile Web Crawler**: the crawling is done in the server side where the information resides reducing the network load caused by traditional crawlers.
- **Continuous Crawler**: information is always changing and being updated and for that the continuous web crawler maintains the index database updated. This increases the consumption of resources.

Our type of crawler falls on the Focused Crawler, because we want to crawl a specific set of web pages on a specific web site.

A Focused Crawler, as quoted in [17], uses an algorithm that loads a web page and gets all the links. Then it rates the links and decides where to go next. A Focused Crawler is categorized by their guiding component and can be divided into three main categories [17]:

- **Classic Focused Crawlers**: give priority to the links based on similarity between the topic and the page containing the links.
- **Semantic Crawlers**: are a variation of the classic focused crawlers, that apply semantic similarity criteria to find the page relevance.
- **Learning Crawlers**: are given a training set with relevant and not relevant web pages to train the Learning Crawler.

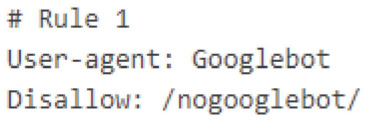
In this project our objective is to crawl the pages of a news web site and find features, like the ERC number, so we can identify the veracity of the site. The Classic Focused Crawler is ideal for finding the ERC number because we know that the news Web sites have that information in the web page *Ficha Técnica*(Datasheet) or *Sobre* (About) and we can use that information to navigate and find the ERC number faster and easier.

Although Web Crawlers undoubtedly useful they have some problems, as mentioned in [15]. Using a web crawler without precautions can originate an accidental Denial of Service Attack since the load that the crawlers puts on the host can be quite high. To solve this problem a maximum frequency of requests to the host must be set. The Web Crawlers should follow the rules of the *robots.txt* file that every website has. This file has the web pages that the host does not want to be crawled.

In Figure 2.5[13] the crawler *Googlebot* is not allowed to go to the folder *nogooglebot* or subdirectories of the website being crawled. Using * instead of the name of a specific crawler infers that no crawlers is allowed on that folder.

Web crawlers need to have protection against traps that the website hosts may have for them. In [18] we see that websites can mechanically generate content having almost infinite URLs to crawl, their

[13] https://support.google.com/webmasters/answer/6062596?



```
# Rule 1
User-agent: Googlebot
Disallow: /nogooglebot/
```

Figure 2.5: robots.txt file

objective is to insert a large amount of their content into a search engine. Having their sites show up in the first results of a web search.

As mentioned in [15], there are open-source crawlers that facilitate our task and avoid having to design and create a web crawler from scratch.

Some open-source crawlers are [14]:

- **Apache Nutch**: is very extendable due to his flexible plugin system. However it has bad documentation and is difficult to setup and configure. It uses Java as a programming language. This was the crawler chosen in [15].
- **Heritrix** also uses java as a programming language. It has good documentation, good performance and a easy setup. Does not support continuous crawling and is not dynamically scalable.
- **Scrapy**: is a python web scraping[15]framework.It was created with the purpose to extract specific information from websites and has built-in export formats like JSON. Has good documentation and a lot of community support.

Scrapy provides the tools we need to create a web crawler, retrieve information from a web page. Besides it can also work with the Beautiful Soup [16], a Python library to parse a web page if needed. Therefore it seems a logical choice to this project.

2.4 Website Blacklist

Another subproblem that we face is the creation of a List with the websites that are considered fake. As we verified in the previous section 2.2 a Lookup System needs to have a blacklist in order to verify the authenticity of a website [8].

As mentioned in the prior section there is a tool that enables us to create a web crawler, Scrapy. This tool also aid us with the creation of a list. Scrapy can export the results of the website evaluation to a JavaScript Object Notation File. JSON as mentioned in [19] is a text format that is completely language independent but uses conventions that are familiar to programmers. Making it very good for the task that we want to do.

[14] https://outsourceit.today/comparison-open-source-web-crawlers/
[15] https://docs.scrapy.org/en/latest/intro/overview.html
[16] https://www.crummy.com/software/BeautifulSoup/bs4/doc/

With that we can have a visual file with marked websites, that are easier to show. And help to determine if the news article we are reading is reliable or not.

3

Fake Websites Catcher

Contents

To be able to find Fake News websites we design a system with four modules, website retrieval, web crawler, website evaluator and a persistence system to save the evaluated website. We also deployed the application on a remote server in order to make available and easily use the application.

3.1 Overview

The Fake Websites Catcher, is composed by five modules as showed in the Figure 3.1. The entry point, an interface where we post an website to be evaluated; The Web scraper where we scrape the pages of the web site for information; The Feature Extractor where we retrieve the websites attributes; The evaluator where we classify the website with its probability of being fake; Finally the Database where we store our data.

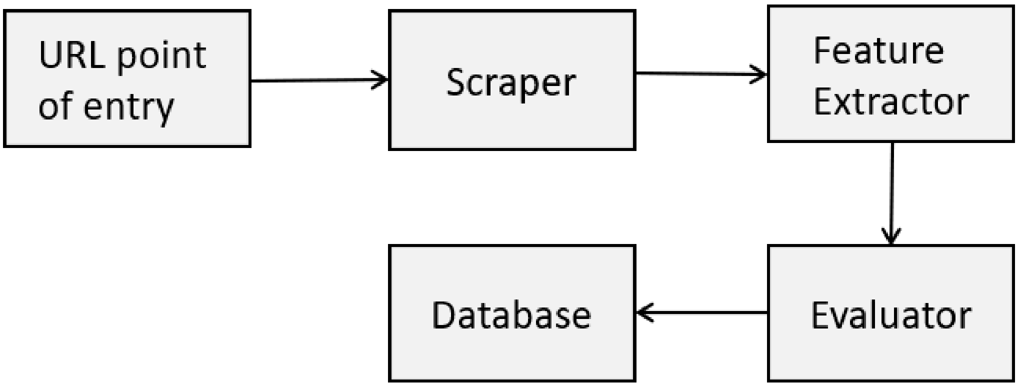
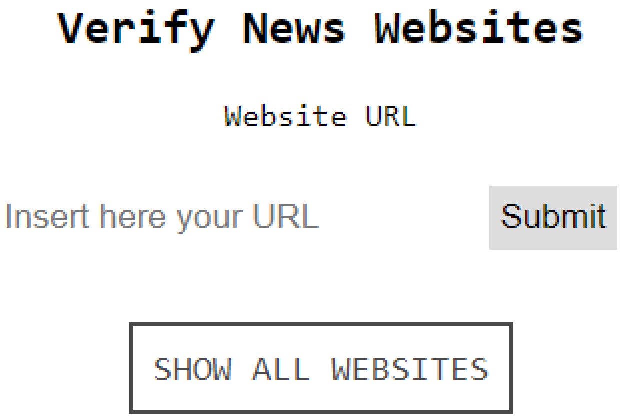


Figure 3.1: Overall view of the system

3.2 Website Retrieval

We created an user interface that receives an url as an input, as showed in the figure 3.2. By accessing the base domain of the app we are prompted by a search text bar where we can insert an url. Clicking submit will start the web site evaluation and display the results on the next web page. We have a go back button that will return to previous page to evaluate another website or show all websites button, were we display all the websites that were evaluated.


Verify News Websites

Website URL

Figure 3.2: Main Page of the fake news website verifier

3.3 Web Scraper

The Web Crawler is responsible for navigating the web page and retrieving the information we need. We use a python package named *Beautiful Soup* to help us scraping the links and information necessary to evaluate the website.

Beautiful Soup [1] is a package commonly use to parse HTML or other markup languages. This is very useful because if a website does not give you a way to download the information you need, it is possible to obtain with *Beautiful Soup*. *Beautiful Soup* navigates through the markup text and remove those markups to return clean text. Not only can remove the markups but it can also search for a specific markup and return the text in that markup. In our project we use that feature to find any tag with <a >and return the hyperlink found. This feature is paramount in order to extract only the meaningful text from the web page. We also use a package named requests to be able to make a request to the web page we are going to evaluate.

We start the scrape of the web site by parsing the input. Because the url string needs to start with http in order to make the request we need to verify and append it to the string in case the url does not have it.

By using the package requests we create a *get request* that returns an object with the page information.

After that request we use *Beautiful Soup* to extract all the links in the page that have *Ficha* or *Ficha Técnica* and save all of the occurrences into a list. In case no page is found we try we a different approach. We use a XPATH parser with the following expression:

```
"//a[contains(@href, 'ficha')]/@href"
```

[1] https://www.crummy.com/software/BeautifulSoup/

This expression returns all the hyperlink references with *Ficha* in the name. XPATH is useful because takes in the XML language and selects the tags that we want.

After this verification, if the list we obtain is empty, we return the website information with a message saying we can't find the news paper editor. Otherwise we return the list with the resulting urls.

The code that retrieves the phrases from the the *Ficha Técnica* web page.

Listing 3.1: Retrieve text from web page

```python
def findNextPhraseInPage(soup, word):
        nextPhrases = []
    for x in (soup.find_all(string=re.compile(word))):
        if len(str(x.next)) > 5:
             phrase = cleanPhrase(str(x.next))
            nextPhrases.append(phrase)
        elif 5 < len(str(x.next.next)):
            phrase = cleanPhrase(str(x.next.next))
            nextPhrases.append(phrase)
    return nextPhrases
```

We check for the length of the phrase and ignore phrases with less then 5 characters because a phrase with less than 5 characters will not have meaningful information.

To be able to find the ERC and editor of the website news we take the urls found in the previous step and construct another http GET request. By using *Beautiful Soup* we try to find all the references of ERC in the page and retrieve the next set of words after that mention.

If it was not possible to find any ERC mentions, we will then proceed to verify the media group editor that the news paper belongs to.

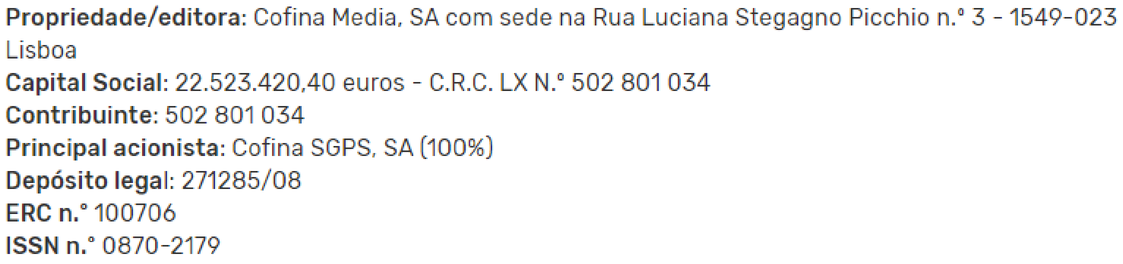
To find the editor we use a similar approach. With the response from the request we use the *Beautiful Soup* to find all the references of "Editor" and take the next set of words and add them to a list. In the end we get a list of possible Editors for the news website.

In the Figure 3.3 we can see an example of the news paper Record page in which we can see the ERC number and Editor. After finding one or both of those features we check them, to know if they are real, against the information we have in the files provider by the ERC entity.

Here we have an example of the row belonging to the Record news website:

```
Title , Registration Number, Editor, Site
Record, 100706, "Cofina Media, S.A.", www.record.pt
```

One field that we would be great to have but the files do not have, is the website name. There are



Propriedade/editora: Cofina Media, SA com sede na Rua Luciana Stegagno Picchio n.º 3 - 1549-023 Lisboa
Capital Social: 22.523.420,40 euros - C.R.C. LX N.º 502 801 034
Contribuinte: 502 801 034
Principal acionista: Cofina SGPS, SA (100%)
Depósito legal: 271285/08
ERC n.º 100706
ISSN n.º 0870-2179

Figure 3.3: Record website legal information

more fields on the files but they are not relevant for our study.

After the ERC and editor phase, we advance for the next step. This step is where we confirm the existence and legitimacy of the numbers found. We can not take for granted that we got is a legitimate number or news group. We take the ERC numbers and start a search for them in two csv files, that we downloaded from the *Entidade Reguladora de Comunicação* website[2]. These files have ERC numbers, editor, location of the editor headquarters and headquarters postal code.

Finally if we do not find an ERC matching the one in the page we set the ERC value of the web site to 0 and we will try to check for the Editor.

For the media group Editor we have the same procedure. We take the words found in the page and search for them in the csv files. However, for the Editor, given it is a word we need to check for degrees of similarity, because there is a possibility that the name can be writing in several forms or with acronyms. To overcome this issue we use the Jaro-Winkler distance metric.

The Jaro-Winkler distance is a string measure between two sequences and a variation of the Jaro distance:

$$sim_j = \frac{1}{3}(\frac{m}{|s_1|} + \frac{m}{|s_2|} + \frac{m-t}{m})$$

Where:

- $|s_i|$ is the length of the string s_i
- m is the number of matching characters
- t is the number of transpositions, that is the number of matching characters but in different order divided by 2.

The closer the value is to 1 the similar are the two words.

The Jaro-Winkler similarity is:

$$sim_w = sim_j + lp(1 - sim_j)$$

[2] https://www.erc.pt/

Where:

- sim_j is the Jaro similarity
- l is the length of the common prefix
- p is a constant scaling factor that determines how much the l factor is valued. The standard value for p is 0.1

We set a Threshold of 0.9 and find any words in the files with a very high degree of similarity to the word found in the page. If we find more than one similar word, we check their Jaro value and return the word with the highest value. If we do not find any match we set the Editor value of the website to an empty string.

We decided to search for the editor in the news website because it is another assurance in order to verify the website credentials. That is, if the parser can not find the ERC it will try to find the newspaper editor instead of automatic ruling out the newspaper has not approved by the Entidade Reguladora Para a Comunicação Social.

We also search the files that we downloaded twice. In the first iteration we try to search in the ERC column for the number retrieved in the web page. The second iteration happens when no ERC is found. Then, we search in the Editor column for the Editor value that was found in the web page.

Here we have the code that loops trough the files with the website information:

Listing 3.2: Retrieve ERC and editor from files

```
for row in reader:
    numberOfCharactersInEditor = len(row[EDITORCOLUMN])
    for possibleEditor in possibleEditors:
        editorReduced = possibleEditor[:numberOfCharactersInEditor].lower()
        if jaroMetric(row[EDITORCOLUMN], editorReduced) >= JAROTHRESHOLD:
            if checkBiggerJaro(row[EDITORCOLUMN].lower(), editor,
                editorReduced):
                editor = row[EDITORCOLUMN]
                erc = row[ERCCOLUMN]
return erc, editor
```

3.4 Interface

In order to be able to evaluate websites we created a simple graphic interface using a web framework.

There was two choices of frameworks, Django and Flask.

Django is a framework with more features than Flask however is less explicit, an heavier framework and harder to start using. For that reason we decided to use Flask for this application, is a lighter and simpler framework with only the essential to build a web application. Although is a fairly simple framework it is extendable, in case we need to have to implement more features like authentication.

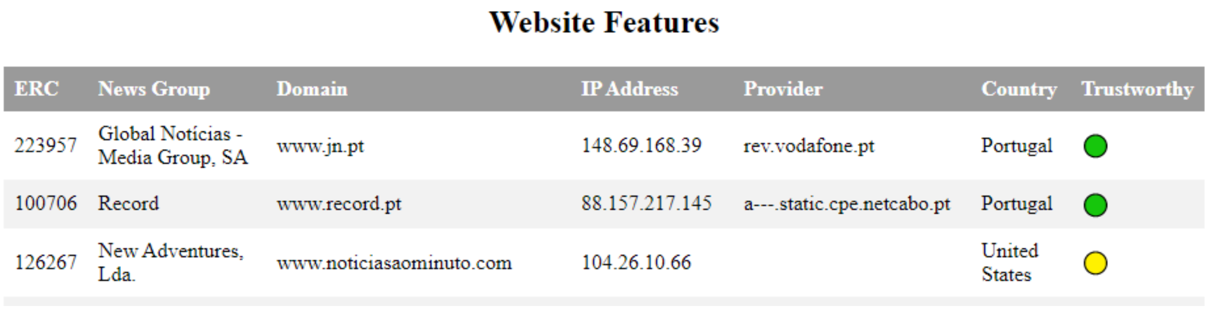
As is written in the article [20]:

> Based on the study, it is evident that Django can be best fit for large-scale projects with the cost of the learning curve. Flask is best fit for the prototyping and smallscale projects but not limited to it.

3.5 Database

The output of the website evaluation had to be saved in some place. To do that we created a table to save the results. For the database we decided to use SQL alchemy because is simpler and it has a plenty of support and a large community using it.

As we can see in the Figure 3.4 our app uses one table with eight columns. The primary key, id, which is a database generated identifier for the entry, the erc which is an integer with default value 0, news group, domain, ip address, provider, country and color.



Website Features

ERC	News Group	Domain	IP Address	Provider	Country	Trustworthy
223957	Global Notícias - Media Group, SA	www.jn.pt	148.69.168.39	rev.vodafone.pt	Portugal	🟢
100706	Record	www.record.pt	88.157.217.145	a---.static.cpe.netcabo.pt	Portugal	🟢
126267	New Adventures, Lda.	www.noticiasaominuto.com	104.26.10.66		United States	🟡

Figure 3.4: Page with all the websites evaulated

- id - is the primary key and is a number generated by the database
- erc - the erc of the website, which is an integer with default value 0
- news group - the news group of the entry
- domain - the website domain, is an unique value and non nullable
- ip adress - the website ip adress, is an unique value and non nullable
- provider - the website provider
- country - the country where the website is located

- color - the color of the website which is inferred using a set of metrics. Green for a legitimate website, yellow the website has some characteristics of a fake website, orange we need to be careful with the website, red the website has all the key characteristics of a fake website

3.6 Website Features

When we evaluate a website we look for some key features. These features will help us decide if the website can be trusted or not. The features are:

- ERC - This field tell us the charter number of the newspaper or the number of the group that they belong. Usually all the newspaper have this information in their information page. However we found a few big newspaper that did not have this information in their page.
- News Group - The news group to which the newspaper belongs. This information is important because it can tell us if the news group is registered and they are legitimate. But just has it happens, in the ERC, some website newspaper do not have the news group information in their page. And therefore we can not obtain this information all of the time.
- Top level Domain - The website top level domain specifies the entity the website is registered in. Usually, Portuguese websites have .pt has their top level domain.
- Communication Protocol - We check if the the website has the https protocol. As we referred in the previous chapter a website is likelier to be have dubious information if it does not use the secure communication protocol.
- Provider - If the provider is not a well know provider in Portugal, it can also be an indicative that the website is not credible.
- Country - the country where the website is located can also be an indicator that the website is not legitimate. Being located in a country that is not Portugal enables the website to not follow the Portuguese law.

These are the features that we weigh when considering the legitimacy of the website.

3.7 Finding the news web site ERC

Portugal has the *Entidade reguladora para a comunicação social*. This organization regulates all entities that have social communication activities in Portugal. That means that all newspaper, radios and television that transmit information to the public need to be approved by the board of the ERC.

This organization also publish all the entities that are registered as we can see here [3]. These lists have the District where the entity is registered, the registry number (ERC), registration date, social designation and the location of their registered office.

For our use case we used the periodic publications and newspaper companies files. These files are downloaded and converted to csv format. We use it to search for the ERC or/and social designation that we found parsing the news website.

3.8 Evaluator

After we retrieve all the information we can from the website, we pass the features retrieved to an evaluator. We defined a weight value for each feature in an environment variables file.

Were we have the algorithm that scores a website to be fake:

Listing 3.3: Evaluation scoring

```
def evaluate(website):
    fake_probability = 0
    if website_country != "Portugal":
        fake_probability += COUNTRY_WEIGHT
    if website_erc != null:
        fake_probability += ERC_WEIGHT
    if website_news_group != null:
        fake_probability += NEWS_GROUP_WEIGHT
    if website_domain != "pt" and website_domain != "com":
        fake_probability += DOMAIN_WEIGHT
    if website_http_protocol != secure_protocol:
        fake_probability += HTTP_PROTOCOL_WEIGHT
    if website_provider != "pt":
        fake_probability += PROVIDER_WEIGHT
return fake_probability
```

We start by setting the score of being an illegitimate website to 0 and depending on the tests that fail we increase that value the set amount for that test. In these tests we compare the website location, where we check if it is in Portugal. We check if we found an ERC value. If it was found an editor for that news website. The website domain, where we see if the top level domain is .pt or .com. The website protocol to check if it uses the secure protocol and finally we check the provider that we verify if it is a

[3] https://www.erc.pt/pt/listagem-registos-na-erc

Portuguese provider. If any of these tests fails we increase the value of the score of the website being fake.

After the tests we return a color matching the result we got from the evaluator. Green for a legitimate website, yellow the website has some characteristics of a fake website, orange we need to be careful with the website, red the website has all the key characteristics of a fake website.

3.9 Web Deployment

In order for the app to be accessible through the web we needed to deploy it in the cloud. We investigate free cloud deployments services to host our project and decided on Heroku.

Heroku is a very well-know platform as a service and it was created in 2007, that means that a a great amount of resources, guides and tutorials are available in the internet. Heroku also has a free pricing model for noncommercial applications that serves our use case, the only down side is that the app sleeps after thirty minutes of inactivity. The next time someone enters the website all the resources need to start up again and that will take a few seconds to load(no more than ten seconds) we did not considered this down side is a major problem. Heroku also has excellent logging with great description in case of errors with the app or the deployment process. This platform also has many plugins to use if we need them. The configuration of the deployment is also very simple and easy and the Heroku official page offers a guide tutorial to deploy python apps in their servers.

Nevertheless, Heroku has some issues with larger scale applications. Their deployment is slow and the cost to maintaining a bigger application is enormous. Thankfully those are not problems for us.

For our database initially we decided to use SQLAlchemy because is light weight and it is easier to learn and start developing.

However, when we were deploying to Heroku we found that our database approach does not work. SQLAlchemy is an ephemeral database. Every time the app restarts, and being in the free tool plan is every thirty minutes of inactivity, the database would not save the registered websites and deleted all the entries.

Thanks to the many plugins that Heroku PostgreSQLoffer, scaling a PostgreSQL instance was very easy and solved our problem. All the entries are now saved and never deleted. In the end, we are using SQLAlchemy in the develop process, locally, and PostgreSQL in the cloud deployment.

3.10 Set Backs

While developing the Fake Website Catcher we were troubled by different problems. The main issue was trying to retrieve the ERC number and Editor of the website.

Different news websites have different ways to expose that information. And some legitimate websites do not have their ERC number written in their website, sometimes they only have the editor. To extract only the page content and finding out the Editor and ERC through text analyze was hard and what usually we got our algorithm over fitted to a specific website not working on others.

This made much more difficult to create a general rule to obtain these parameters.

We tried to make a general rule but the rule was fitted for the websites that we tested. We also obtained all the readable text from a page and then loop trough the text and tried to find the Editor and ERC. In order to do that we look for the words Editor and some variants of the word like, *Editoral*, *Editora* retrieving the words that appear after. But even then some websites place the Editor without any headliner first making the it hard to find it. The same happened for the ERC number.

Other set back that we had was if there was fake websites with ERC numbers copied from a trustworthy website. Fortunately every website that we tested did not present this issue, and for the great part of the websites they did not even had the webpage *Ficha Técnica*. If, eventually, a website copies an ERC from a news website the system can not check for the veracity of the information.

We also has a setback in the deployment phase to the cloud where our ephemeral database would reset the data. We solved that problem by using a PostgreSQL instance.

4

Evaluation

Contents

In the previous chapter, we explained how our system works and how is implemented. In the following section we discuss how we evaluated our system, and what results we obtain from the evaluation.

4.1 Data Gathering

To be able to evaluate our system we gather a list of fake news websites. This list was given to us by Prof. Bruno Martins, initially the list had forty six websites. However, by the time we started our evaluation, seventeen of those websites were already deleted, putting the list at twenty nine websites. We also needed a list of trustworthy websites, to obtain this list we chose well known news websites and check them manually against the list provided by the *Entidade Reguladora de Comunicação*. In the end we obtained twenty two trustworthy websites.

Each entry of the dataset has seven columns, the first column had the information if it was fake or not, the rest of them had the websites properties like the ERC, news Group, Domain, provider, country and security protocol.

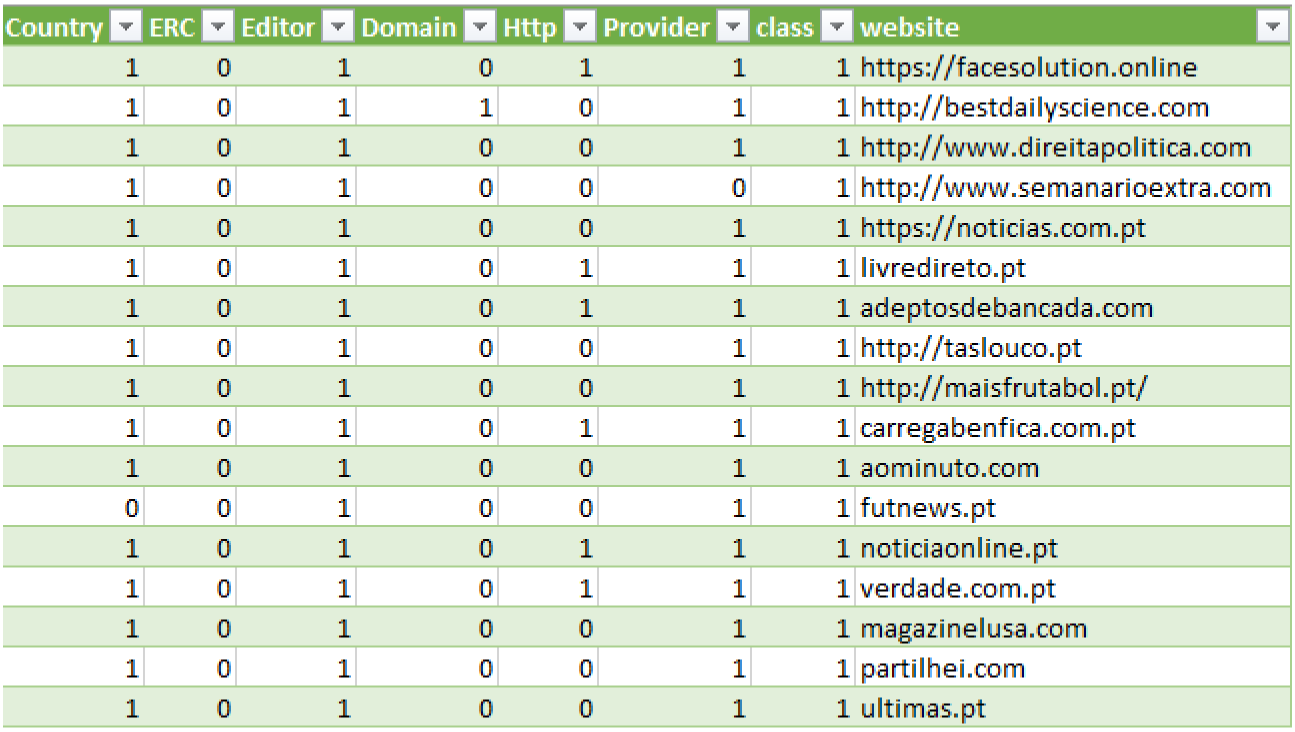
To obtain the features of the website we created a script that calls our system with each of the websites and extracted the features. We then saved the information to a file.

However, in order to be able to do the evaluation using a GA and the Decision tree, we had to convert the values of the dataset to a binary format. The first column had a value of 1 if it was fake and 0 if was trustworthy, and we did the same for the rest of the attributes. If the website had an ERC we change the value to 0, 1 otherwise, if it had the news group the value would be 0 if it did not have it would be 1. If the country was Portugal we change the value to 0 and changed to 1 when was not Portugal. If the domain and provider ended in .pt or .com we would change the value to 0 and finally for the security protocol, we would change the value to 0 if the website was using a secure protocol.

In the figure 4.1 we display a sample of the converted dataset. The class value 1 indicates that the entry is a fake website.

4.2 Procedure

To perform the evaluation we started by choosing at random a training set containing 10 websites, five fake and five trustworthy. Those ten websites were given to the system and their features extracted. With the websites and their features in the database we gave the data to the evaluator. The evaluator change the six parameters weight between a threshold defined apriori (5-40) that would increase in increments of 5 and test the training set against all those combinations. After we extract the four best results and then test all our data set with the weight of the parameters obtained. We also decided to use a Genetic Algorithm to help us find the best possible solution.



Country	ERC	Editor	Domain	Http	Provider	class	website
1	0	1	0	1	1	1	https://facesolution.online
1	0	1	1	0	1	1	http://bestdailyscience.com
1	0	1	0	0	1	1	http://www.direitapolitica.com
1	0	1	0	0	0	1	http://www.semanarioextra.com
1	0	1	0	0	1	1	https://noticias.com.pt
1	0	1	0	1	1	1	livredireto.pt
1	0	1	0	1	1	1	adeptosdebancada.com
1	0	1	0	0	1	1	http://taslouco.pt
1	0	1	0	0	1	1	http://maisfrutabol.pt/
1	0	1	0	1	1	1	carregabenfica.com.pt
1	0	1	0	0	1	1	aominuto.com
0	0	1	0	0	1	1	futnews.pt
1	0	1	0	1	1	1	noticiaonline.pt
1	0	1	0	1	1	1	verdade.com.pt
1	0	1	0	0	1	1	magazinelusa.com
1	0	1	0	0	1	1	partilhei.com
1	0	1	0	0	1	1	ultimas.pt

Figure 4.1: Dataset entries example

We will compare the different set of weights by their F-Score.

We can determine the F-Score of the results with the following formula:

$$F_\beta = (1+\beta^2)\frac{precision * recall}{(\beta^2 * precision) + recall} = \frac{(1+\beta^2) * TP}{(1+\beta^2) * TP + \beta^2 * FP + FN}$$

Where the True Positives (TP) are considered the trustworthy websites that score below or equal to 0.5, the False Positives (FP) are the trustworthy websites that score above 0.5 and the False Negatives (FN) are the fake websites that score below or equal to 0.5, the β is a positive value, where β is chosen such that recall is considered β times more important than precision. For our case we will consider β equals to 1. The True Negatives (TN) while we do not use in the formula means the true negative values, the fake websites that were classify correctly. The recall is the fraction of relevant websites that were retrieved and precision is the fraction of relevant websites among the retrieved websites.

Our dataset is made of twenty nine websites marked as fake and twenty two as trustworthy.

4.3 Tests

In this section we tested the four best results that we got from the step mentioned above.

4.3.1 First set of weight values

For the first set of weight values we tested our dataset with the following weights:

- **Country** - 30
- **ERC Number** - 5
- **News Group** - 10
- **Domain** - 20
- **Http Security Protocol** - 5
- **Provider** - 30

The results we obtain are represented in the figure 4.2.

The websites considered fake are represented in red and the legitimate ones are represented in green. Each one of the websites are given a score between 0 and 1. The website is considered fake if it scores between 0.5 and 1 and is considered trustworthy if it scores below or equal to 0.5.

Websites with a score closer to 1 have a higher probability to be fake and the websites that score closer to 0 are more probable to be legitimate.

As shown in the graph, the evaluator gave a score greater or equal of 0.75 to all the fake websites except three. It gave 0 to Five websites, meaning that these websites are very likely to be trustworthy because it did not had any of the characteristics of a fake website. We have three trustworthy websites are lower than 0.25, meaning some characteristics of a fake website were found in these three websites. Finally, fourteen trustworthy websites had a score between 0.4 and 0.75. This can be explain by the way the web scrapper that finds the website features is built. Specifically the ERC Number and News Group feature.

Every news websites have a different way to present and write the technical information of their business and it becomes really difficult for the parser to find the context of the ERC and News Group in every website. Although in this evaluation the Provider and the Country have a higher weight if one of those features fail will affect greatly the result.

Since the higher values were the Country and the Provider when the parser does not find those features, the evaluator will give them a higher score and consider them as fake.

For the F-score we have the following values:

- **TP** - 15
- **TN** - 27
- **FP** - 2

- **FN** - 7

$$F - Score = \frac{2 * 15}{2 * 15 + 2 + 7} = 0.77$$

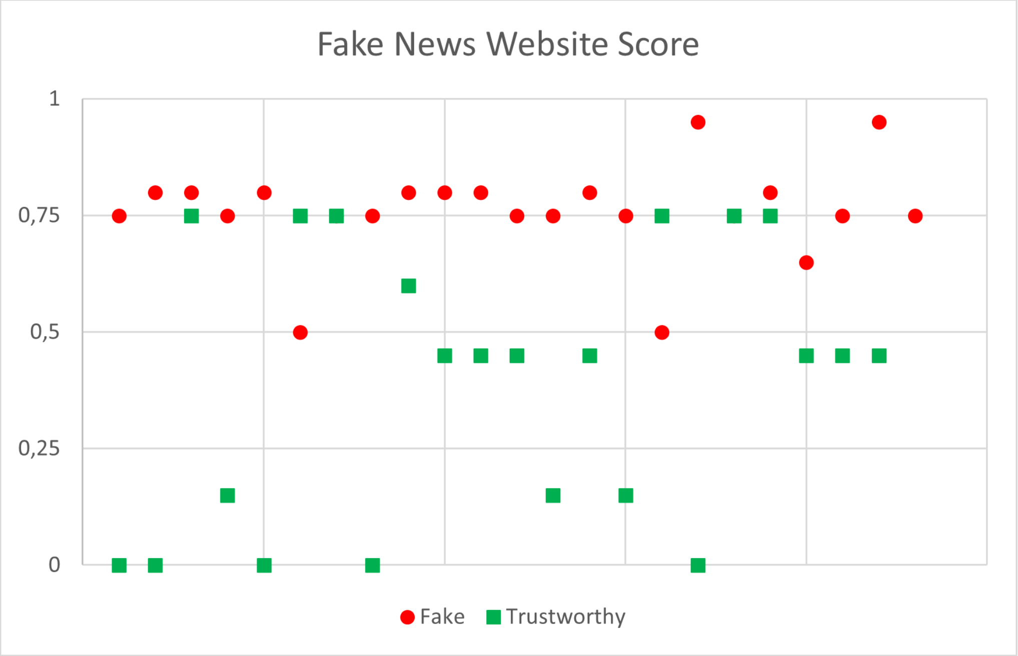


Figure 4.2: Graph for the first test

4.3.2 Second set of weight values

We tested the websites, with these weights values:

- **Country** - 25
- **ERC Number** - 5
- **News Group** - 10
- **Domain** - 20
- **Http Security Protocol** - 5
- **Provider** - 35

The results we obtain are represented in the figure 4.3.

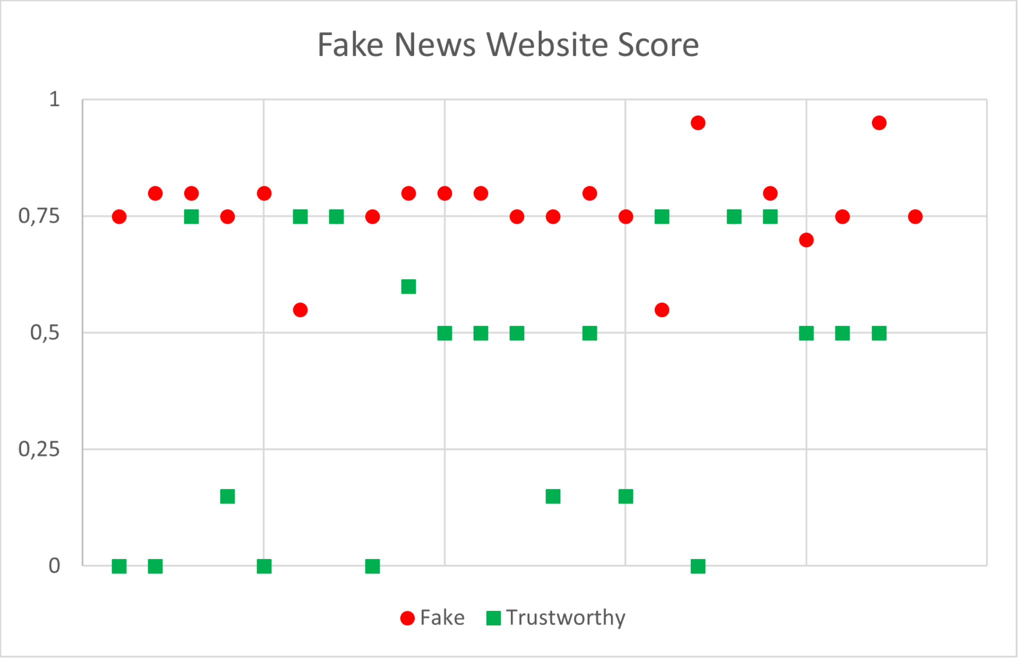


Figure 4.3: Graph for the second test

In the graphic above we can see that is very similar to the first one. But some of the websites are ranked 0.05 to 0.1 higher. This increment makes two of the fake websites that were missed classified in the last test now being classified as fake.

For the accuracy we have the following values:

- **TP** - 15
- **TN** - 29
- **FP** - 0
- **FN** - 7

$$F - Score = \frac{2 * 15}{2 * 15 + 0 + 7} = 0.81$$

The second case has a better score than the first but several websites have a score very high, being 14 of them with a value equal or higher than 0.5.

4.3.3 Third set of weight values

We tested the websites, with these weights values:

- **Country** - 30

- **ERC Number** - 10
- **News Group** - 5
- **Domain** - 15
- **Http Security Protocol** - 25
- **Provider** - 15

The results we obtain are represented in the figure 4.4.

In the third test case we can see that seven trustworthy websites have a value higher than 0.5, with their probability of being fake equal to some fake websites and higher than two of them.

For the accuracy we have the following values:

- **TP** - 15
- **TN** - 28
- **FP** - 1
- **FN** - 7

$$F - Score = \frac{2 * 15}{2 * 15 + 1 + 7} = 0.79$$

4.3.4 Fourth set of weight values

We tested the websites, with these weights values:

- **Country** - 35
- **ERC Number** - 10
- **News Group** - 5
- **Domain** - 20
- **Http Security Protocol** - 5
- **Provider** - 25

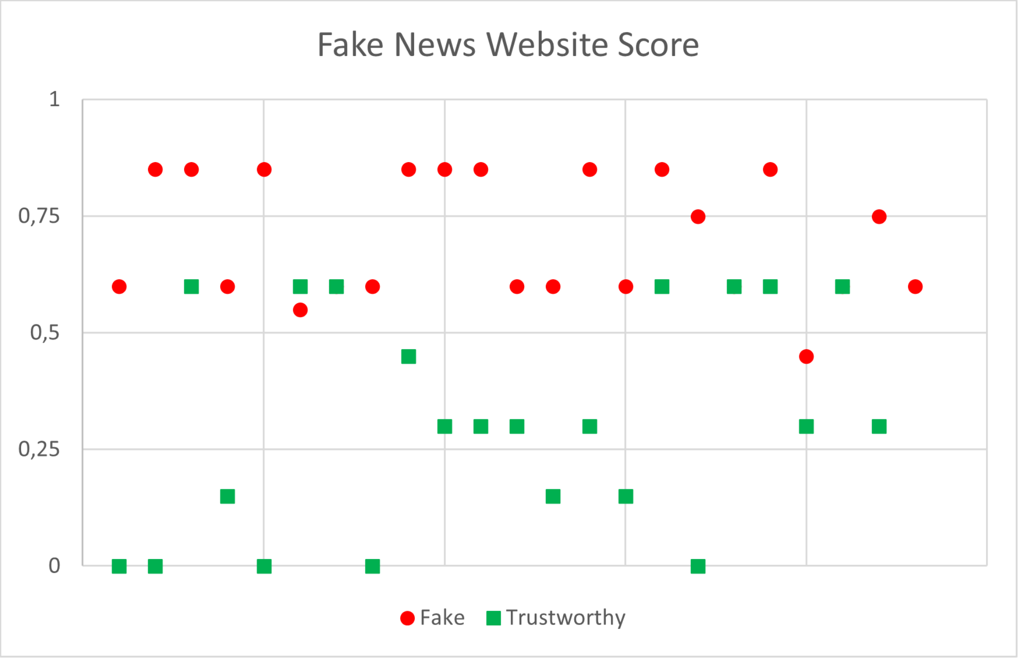


Figure 4.4: Graph for the third test

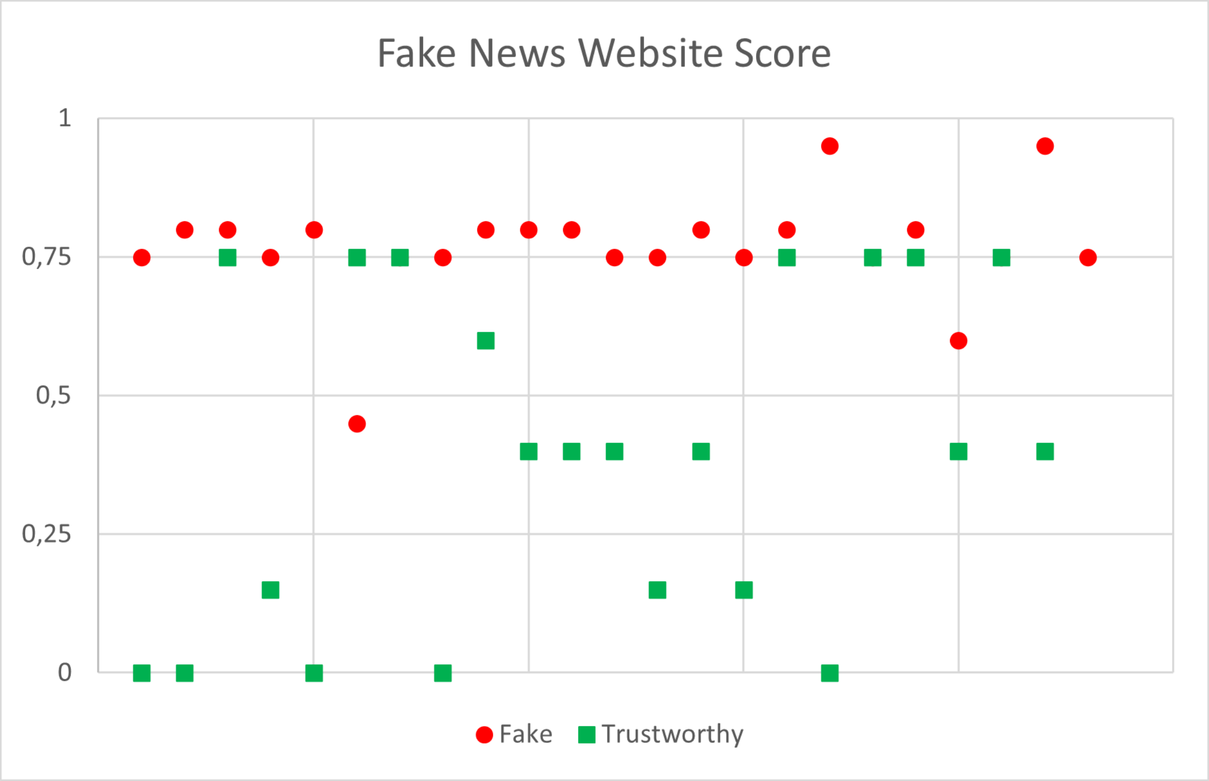


Figure 4.5: Graph for the fourth test

The results we obtain are represented in the Figure 4.5.

In the fourth case we set a higher weight in the country and lower weight in the http security protocol and higher on the provider. That made the probability of being fake of most fake news websites higher than 0.75. However several of the real websites had their probability of being fake also increased. And two of the fake websites have a probability lower than 0.75.

For the accuracy we have the following values:

- **TP** - 14
- **TN** - 28
- **FP** - 1
- **FN** - 8

$$F - Score = \frac{2 * 14}{2 * 14 + 1 + 8} = 0.76$$

4.4 Genetic Algorithm

We decided to use a genetic algorithm to find the best set of values to the attributes and see how it compares to the scores in the previous section. To create the genetic algorithm we use a a python library called PyGAD [1].

We started by creating a fitness function that sees how close the solution provided by the genetic algorithm would be to the real results.

Here we can see our fitness function:

Listing 4.1: Fitness algorithm

```python
def fitness(solution, solutionIndex):
    sumWeight = int(np.sum(solution))
    if 90 <= sumWeight <= 100:
        probList = [0] * 51
        i = 0
        for website in websites:
            fakeProb = 0
            if website.country != "Portugal":
                fakeProb += int(solution[0])   # Country
```

[1] https://pygad.readthedocs.io/en/latest/

```
            if website.erc == 0:
                fakeProb += int(solution[1])  # ERC
            if website.news_group == "No editor Found":
                fakeProb += int(solution[2])  # News Group
            if website.domain[-2:] != "pt" and website.domain[-3:] != "com":
                fakeProb += int(solution[3])  # Domain
            if not website.http_protocol == 0:
                fakeProb += int(solution[4])  # HTTP_PROTOCOL_WEIGHT
            if website.provider[-2:] != "pt":
                fakeProb += int(solution[5])  # Provider
            probList[i] = fakeProb
            i += 1
    else:
        return -999999999
    tp = 0
    tn = 0
    fp = 0
    fn = 0
    #loop for the fake websites
    for j in probList[:28]:
        if j > 50:
            tn += 1
        else:
            fn += 1
    #loop for the trustworthy websites
    for k in probList[29:]:
        if k <= 50:
            tp += 1
        else:
            fp += 1
    fScore = 2*tp / (2*tp + (fp + fn))
    return fScore
```

The fitness functions starts by checking if the sum of the solution is between 90 and 100. We created this condition because the sum of our solution needs to be 100, if we did not had this condition the algorithm would give higher numbers and the score of the websites would no be in the 0-100 range. We also gave the lower threshold the value 90 because we wanted to have some leeway for the algorithm to find some solutions. It would be almost impossible for the solution to be exactly 100. We run the algorithm some times and the sum of the solution would always converged to 100. Therefore, if the solution was not in the range of 90-100 we would give a very low fitness.

The next step would be to evaluate all the websites with the solution provided by the algorithm and saved them in an array with size of the numbers of websites we have to evaluate.

After we evaluate all the websites we find the F-Score for the iteration. We create two loops with two conditions each in order to fin the TP, TN, FP and FN values.

Finally we return the F-score and the highest value that we obtain will be the best generation that our algorithm could create.

We also had to give the genetic algorithm other settings, like number of generations, probability of mutations, low range and high range, mutation type and numbers of genes.

- Number of generations - How many iterations will the algorithm do : 50
- Probability of mutations - Probability of the the population change their parameters : 0.1
- Number of genes - How many attributes does it have : 6
- Low range - Lowest value of an attributes : 1
- High range - Highest value of an attributes : 50
- Mutation type - What attributes will be mutated : random
- Crossover point - When crossing to elements what type of crossover will be : Single point

When running the algorithm we got the following result:

```
Parameters of the best solution :
[33.81577413  3.84281601 10.01569059 31.38444731  6.01378099  6.00516052]

Fitness value of the best solution = 0.7894736842105263
```

With a F-Score value of 0.79, this solution results in the following graphic 4.6.

The result we got is equal to the second best result that we got in the previous evaluations, which is a good result. We could possible have better results by tweaking some settings of the genetic algorithm, like the numbers of generations.

Nevertheless the Genetic Algorithm provided a good solution. Classifying only two of the fakes websites in the wrong place and miss classifying six trustworthy websites.

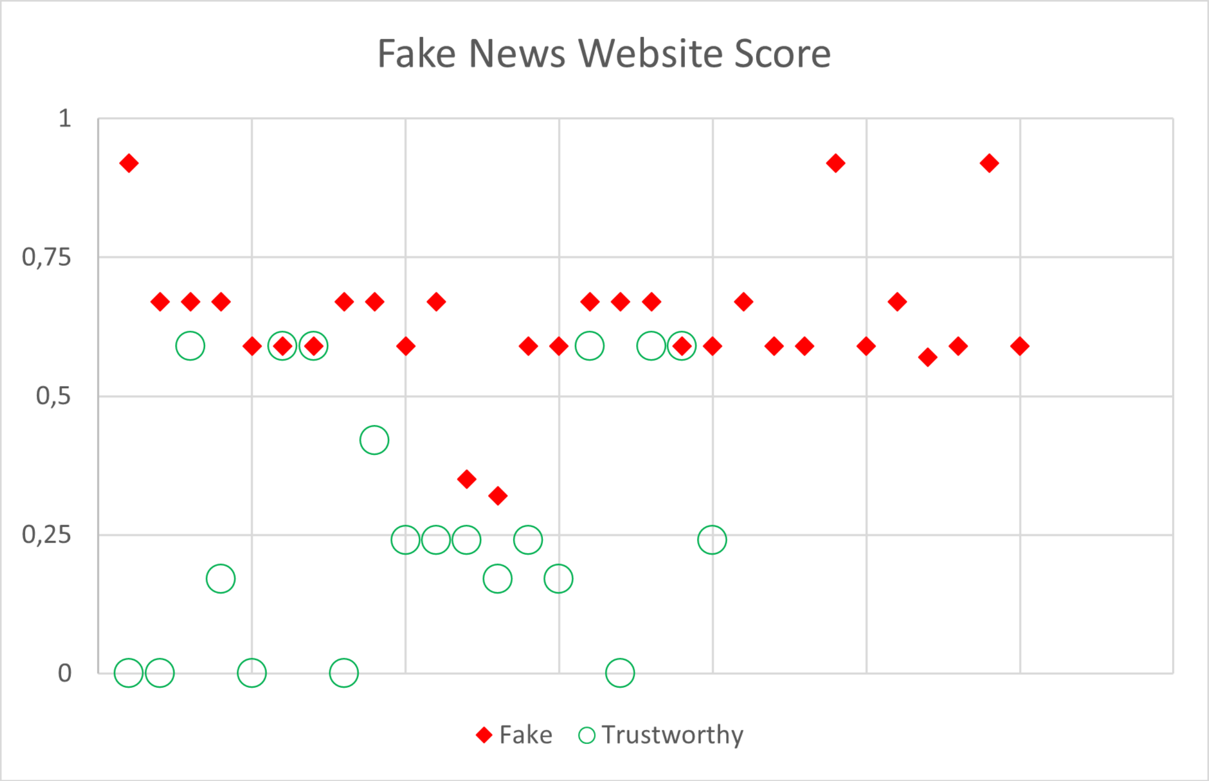


Figure 4.6: Genetic Algorithm results

4.5 Decision Tree

We also generated a decision tree. A decision tree is a flow-chart where each node is a test on a feature and each line represents a binary outcome, true or false.

We decided to create the decision tree with the help of a package in python called sci-kit learn to fit the data to the tree and we use a package named Pandas to arrange the data in a format the sci-kit learn can use it.

In order to use the data we had we converted the results in a binary form. If the Country was not Portugal it would get a 1 and if it was Portugal it would get a 0, that way we could create the decision tree.

We also chose to use a classifier tree because it is better suited for our problem given we have a binary classification, fake or not fake.

The tree we got is displayed in the Figure 4.7.

The features are showed as an array with the name X, below we have what each index of the array represents:

- Country = X[0]

- ERC Number = X[1]
- News Group = X[2]
- Domain = X[3]
- HTTP = X[4]
- Provider = X[5]

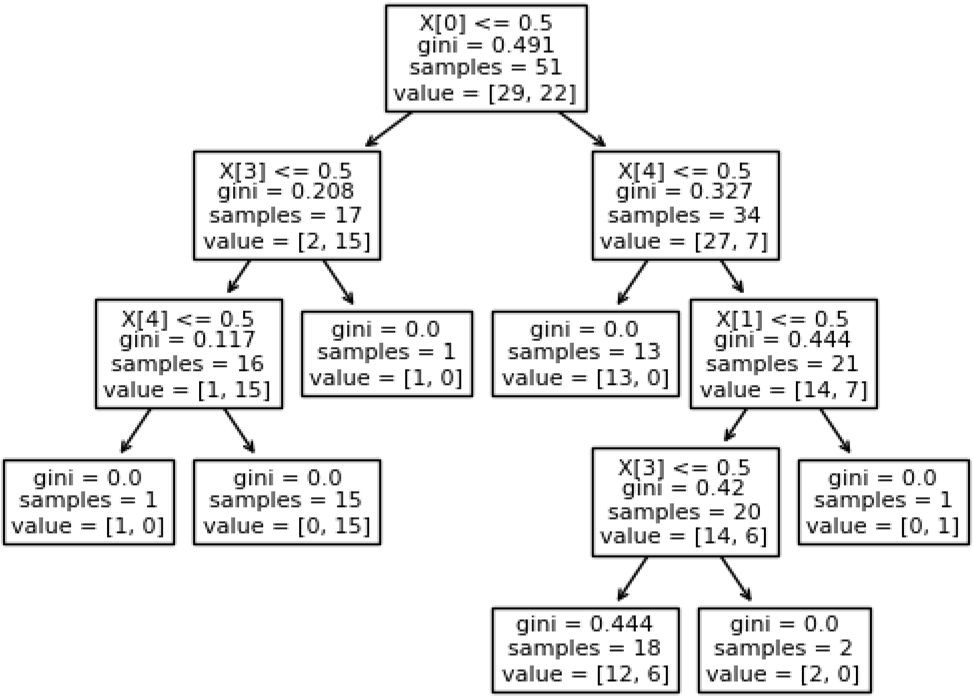


Figure 4.7: Decision Tree

We start on the top of the tree. The first node has analyses the Country and divides the sample between less or equal than 0.5 and more than 0.5. If they are less than 0.5 the samples go to the branch on the left, if they are greater than 0.5 they go to the right. We also have the number of samples, 51, and the value of the samples, 29 fake websites and 22 trustworthy websites. In the first node we can see that 2 out of the 29 fake websites are not hosted in Portugal, and 7 out of the 22 trustworthy websites are not hosted in Portugal.

On the left branch we got the domain feature, where it divides between .com and .pt from the others domain, here we can see that only one fake websites does not have .com or .pt domain.

After that we find a node testing the http protocol where fifteen trustworthy websites have secure http protocol and one fake websites has not.

On the final classifier node we have the Domain where eighteen of the samples have the .com or .pt domain and two of them have other domains. This last node is the only leaf node that we have fake and trustworthy websites. That means that many fake websites shared the same feature as the trustworthy ones. We would need another feature to classify and separate these websites.

The tree also shows the Gini index, this index indicates the probability of a feature that is classified incorrectly when selected at random, the lower the Gini index the most accurate is the classification..

The Gini index can be calculated with the following formula:

$$G = \sum_{i=1}^{C} p(i) * (1 - p(i))$$

Where *C* is the amount of classes, in our case we have two classes and *p(i)* is the probability of selecting a class.

The News Group and the Provider parameter was not used by the classifier to create the tree.

With this tree we can take a website, extract their features and follow the tree path to obtain a result.

We can also observe that a website hosted in Portugal with a domain .pt or .com and using a secure http protocol can be considered trustworthy. It is also possible to identify that a website not hosted in Portugal and without Https protocol is considered Fake.

The other leaf nodes represent only one website and there is one that has 18 websites where those websites are hosted outside Portugal, use Https protocol, no ERC was found and their domain is .pt or .com we have higher Gini. That means that if we find a website with those characteristics, there is a higher possibility that we classify the website wrongfully.

4.6 Discussion

With the evaluation done we can extract several points and get some conclusions.

From the information that we got it is possible to see that in several of the trustworthy websites the ERC Number could not be found. The system found the ERC number in six websites out of the twenty two (DN, Record, Região de Leiria, Fumaca, JN and Jornal de Negocios). This was due to the way websites display information. As there is no specific rule for the websites to expose that information. This problem makes impossible to create a general rule that can enable the system to extract the ERC Number consistently.

For the second test case, which got the best accuracy, it marked seven trustworthy websites has fake. We are going to analyze each of them and see why it failed.

- **Informa Mais** - The website is located in the United States of America, it was not possible to find ERC Number or News Group and the system could not get information about the Provider.

- **Diário de Leiria** - The website is located in the United States of America, it was not possible to find ERC Number or News Group and the system could not get information about the Provider.
- **Gazeta das Caldas** - The website is located in the United States of America, it was not possible to find ERC Number or News Group and the system could not get information about the Provider.
- **Fumaça** - The website is located in the United States of America and could not find information about the Provider. Altough we could find the ERC Number and News Group it was classified as Fake with 0.6, the lowest of the trustworthy websites classified as fake.
- **Jornal Económico Sapo** - The website is located in the United States of America, it was not possible to find ERC Number or News Group and the information found about the Provider was not known by the system.
- **Observador** - The website is located in the United States of America, it was not possible to find ERC Number or News Group and the system could not get information about the Provider.
- **O Setubalense** - The website is located in the United States of America, it was not possible to find ERC Number or News Group and the system could not get information about the Provider.

Finally we can conclude some important points from our evaluation and the results that we obtained:

1. The country is a good indicator of the trustworthiness of a website.
2. We can identify with a high degree of confidence the websites that are fake.
3. The News Group and ERC Number are great metrics however they are complicated to extract automatically from the news websites pages.
4. The case with better results is the second where we got an accuracy of 81% equals to the result that we got with our genetic algorithm. By assessing the sites with a value higher than 0.5 as fake and the websites with a value of 0.5 or lower as trustworthy.
5. The Genetic Algorithm gave good results. When comparing the weights of the values of the Genetic Algorithm (GA) and the third evaluation we can see that they are not that different. The Country got a relatively high weight, the ERC number and News group a low weight, the Domain had a high weight, the Http protocol a low weight and finally the provider had the biggest difference, where in the third evaluation the value was high but in the GA was low. The balance of the weights between the two was close.
6. The decision tree as expected gave the bigger factor to ascertain if the website is fake or not, the location of the website. Although some trustworthy websites are not located in Portugal, if the

websites are not in Portugal and do not have https security protocol they are classified as fake. If they have the https security protocol then they are classified by their ERC number, it would be a great measure but due to the difficult process that it is to obtain them, it only classifies one has a trustworthy website. Finally the last node it classifies with the Domain putting fake and trustworthy websites in the same category, indicating uncertainty in that classification.

7. The decision tree did not use the news group feature and the ERC, because they are very difficult to obtain with a high degree of certainty.

8. The decision tree also did not use the Provider feature which had the best results in our evaluation. This probably happened because in our training set, our samples had a good score with the provider feature. However testing with all the samples, the best weights are not the same as the training set.

5

Conclusion

Contents

5.1 Conclusion

This thesis objective was to evaluate news website and differentiate fake news websites from trustworthy websites in Portugal. In this thesis we created a system capable of solving this problem. We created an web interface with an url as input, a website parser to read an web page and find useful information and a sub system that collects various information from the website in order to be able to successful be evaluated. We learned what are the key features that can show what website can be trusted being the most accurate for our case the location where the website is hosted.

We did four different evaluations, where we gave different weights to the features. On the first evaluation we gave more emphasis to the ERC Number and News Group where all the fakes websites were correctly identified, however twelve trustworthy websites were above the 0.5 mark being incorrectly evaluated.

The second evaluation, being the one that shown better results, we increase the weight of the Country feature and HTTP Security protocol and decrease the ERC Number and the News Group. In this case the system classified all the fake websites above the 0.5 mark and only five trustworthy websites were classified incorrectly, above the 0.5 mark.

The third evaluation, the Country was decreased and the Domain, Provider and News Group were increase. In this case we got twelve wrongly classified trustworthy websites and the fake websites were closer to the trustworthy websites and closer to the 0.5 mark.

The fourth evaluation, the Country feature was increase and the fake websites twenty eight of the thirty websites had a score of 0.85 or higher, however fourteen trustworthy websites had a score higher than 0.5.

/mudar aqui a conclusao

One of the most difficult steps to implement was the retrieval of context from a web page, more specifically the retrieval of the ERC Number and News Group of the website. The parser is unsuccessful at identifying and retrieving those features for some news website. Every website is made differently and the way they expose their information is different from each other. Therefore there is no single rule that the system can use to retrieve that information with 100% accuracy. This problem impacts the evaluation, because some websites actually have the ERC Number and have a real News Group but the system can not find that information. Thus the trustworthy websites have an higher score than they were suppose to have.

5.2 Future Work

The system showed some good results, and could detect with high accuracy the fake news websites, but some aspects can be improved. Although we previously stated that the retrieval of context was difficult.

We believe that the parser can be improved to be more effective in retrieving their ERC Number and News Group. One way of improving this is adding more rules when searching for the ERC and Editor.

This system works in Portugal, because it uses the an identifier given by a Portuguese authority. The system can be extended to work in other countries, if that country has an entity similar to what Portugal has. Having the files with those numbers, changing the value of the feature Country to the country that is being tested, and changing some key words that are exclusively to Portugal, the system would provided similar results for the country being tested.

Bibliography

[1] I. C. Mogotsi, “Christopher d. manning, prabhakar raghavan, and hinrich schütze: Introduction to information retrieval,” *Information Retrieval*, vol. 13, no. 2, pp. 192–195, Apr 2010.

[2] J. R. Andrew M. Guess, Brendan Nyhan, “Exposure to untrustworthy websites in the 2016 us election,” 2020.

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

[4] C. Shao, G. L. Ciampaglia, O. Varol, A. Flammini, and F. Menczer, “The spread of fake news by social bots,” *arXiv preprint arXiv:1707.07592*, pp. 96–104, 2017.

[5] Dan Noyes, “Zephoria Digital Marketing, Facebook Statistics,” https://zephoria.com/top-15-valuable-facebook-statistics/.

[6] K. Singh, H. K. Shakya, and B. Biswas, “Clustering of people in social network based on textual similarity,” *Perspectives in Science*, vol. 8, pp. 570–573, 2016.

[7] N. J. Conroy, V. L. Rubin, and Y. Chen, “Automatic deception detection: Methods for finding fake news,” *Proceedings of the Association for Information Science and Technology*, vol. 52, no. 1, pp. 1–4, 2015.

[8] A. Abbasi and H. Chen, “A comparison of tools for detecting fake websites,” *Computer*, vol. 42, no. 10, pp. 78–86, 2009.

[9] M. D. Molina, S. S. Sundar, T. Le, and D. Lee, ““fake news” is not simply false information: A concept explication and taxonomy of online content,” *American Behavioral Scientist*, vol. 65, no. 2, pp. 180–212, 2021. [Online]. Available: https://doi.org/10.1177/0002764219878224

[10] A. Abbasi, F. Zahedi, S. Kaza *et al.*, “Detecting fake medical web sites using recursive trust labeling,” *ACM Transactions on Information Systems (TOIS)*, vol. 30, no. 4, p. 22, 2012.

[11] N. C. R. L. Y. Teraguchi and J. C. Mitchell, “Client-side defense against web-based identity theft,” *Computer Science Department, Stanford University. Available: http://crypto. stanford. edu/SpoofGuard/webspoof. pdf*, 2004.

[12] Paulo Pena, “Fake news: sites portugueses com mais de dois milhões de seguidores https://www.dn.pt/edicao-do-dia/11-nov-2018/interior/fake-news-sites-portugueses-com-mais-de-dois-milhoes-de-seguidores--10160885.html,” *Diário de Notícias*, 11 2018.

[13] “Como funciona uma rede de notícias falsas em Portugal https://www.dn.pt/edicao-do-dia/21-out-2018/interior/como-funciona-uma-rede-de-noticias-falsas-em-portugal-10046731.html,” *Diário de Notícias*, 10 2018.

[14] F. Johnson and S. K. Gupta, “Web content mining techniques: a survey,” *International Journal of Computer Applications*, vol. 47, no. 11, 2012.

[15] Joao Nuno Martins de Almeida, “2gather4health: Web crawling and indexing system implementation,” *Universidade de Lisboa - Instituto Superior Técnico*, 2018.

[16] M. Kumar, R. Bhatia, and D. Rattan, “A survey of web crawlers for information retrieval,” *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 7, no. 6, p. e1218, 2017.

[17] N. Jain and P. Rawat, “A study of focused web crawlers for semantic web,” 2013.

[18] C. Olston and M. Najork, “Web crawling,” *Found. Trends Inf. Retr.*, vol. 4, no. 3, pp. 175–246, Mar. 2010.

[19] ECMA, “The json data interchange syntax,” 12 2017.

[20] Ghimire, Devndra, “Comparative study on Python web frameworks: Flask and Django https://www.theseus.fi/handle/10024/339796,” 2020.

A

List of Websites

In this appendix you can consult the websites that we use to evaluate the system

- https://facesolution.online
- http://bestdailyscience.com
- http://www.direitapolitica.com
- http://www.semanarioextra.com
- https://noticias.com.pt
- livredireto.pt
- adeptosdebancada.com
- http://taslouco.pt
- http://maisfrutabol.pt/
- carregabenfica.com.pt

- aominuto.com
- futnews.pt
- noticiaonline.pt
- verdade.com.pt
- magazinelusa.com
- partilhei.com
- ultimas.pt
- eu-gosto-e-tu.com
- sogolo.pt
- avozdarazao.com
- lusojornal.com
- portugalglorioso.blogspot.com
- altamente.org
- http://tafeio.com.pt/
- video-divertido.com
- https://www.tuga.press/
- voxpoptv.com
- vamoslaportugal.net
- https://palavrasoltas.com
- www.dn.pt
- www.record.pt
- https://www.informamais.pt/
- https://www.cmjornal.pt/
- https://www.regiaodeleiria.pt/
- http://www.diarioleiria.pt/

- https://gazetadascaldas.pt
- www.jn.pt
- https://fumaca.pt/
- https://24.sapo.pt/
- https://tvi24.iol.pt/
- https://www.iol.pt/
- https://www.abola.pt/
- https://maisfutebol.iol.pt/
- https://www.ojogo.pt/ojogo.asp
- https://jornaleconomico.sapo.pt/contatos
- https://www.jornaldenegocios.pt/
- https://observador.pt/
- https://osetubalense.com/
- https://anoticia.pt/
- https://postal.pt/
- https://infocul.pt/ficha-tecnica/