



*A Machine Learning approach to detect Fake News*

University of the West of England

Name: Juan Camilo Rodriguez

Student ID: 16020551

Module: Digital Systems Project

Module Code: UFCFXK-30-3

Word Code:

# Abstract

# Acknowledgments

# Table of Contents

[Table of Figures 8](#_Toc47001357)

[Table of Tables 9](#_Toc47001358)

[Introduction 10](#_Toc47001359)

[1.1. Define Problem 10](#_Toc47001360)

[1.2. Aim and Objectives 12](#_Toc47001361)

[1.3. Chapter Breakdown 13](#_Toc47001362)

[Literature Review 14](#_Toc47001363)

[2.1. News Media 14](#_Toc47001364)

[2.1.1. News 14](#_Toc47001365)

[News Format, Structure and Writing 14](#_Toc47001366)

[2.1.2. What is Fake News 16](#_Toc47001367)

[2.1.3. Types of Fake News 17](#_Toc47001368)

[2.3. Relevant Research 18](#_Toc47001369)

[2.3.1. Fact Checking – Manually 19](#_Toc47001370)

[2.3.2. Automatic Techniques for Fake News Detection 20](#_Toc47001371)

[2.3.3. Style-Based 21](#_Toc47001372)

[Style Representation 21](#_Toc47001373)

[What is Language? 21](#_Toc47001374)

[The Different Features 22](#_Toc47001375)

[Style Classification 28](#_Toc47001376)

[2.3.5. Prior Attempts 30](#_Toc47001377)

[2.4. Gaps Identified 32](#_Toc47001378)

[Requirements 33](#_Toc47001379)

[3.1. MoSCoW Methodology 33](#_Toc47001380)

[3.2. Functional Requirements 34](#_Toc47001381)

[3.3. Non-Functional Requirements 35](#_Toc47001382)

[Methodology 37](#_Toc47001383)

[4.1. Software Development Strategies 37](#_Toc47001384)

[4.1.1. Traditional Development Models 37](#_Toc47001385)

[Waterfall Model 37](#_Toc47001386)

[4.1.2. Agile Development Models 38](#_Toc47001387)

[SCRUM 38](#_Toc47001388)

[4.2. Method 38](#_Toc47001389)

[4.2.1. SCRUM Approach for Project 39](#_Toc47001390)

[Design, Implementation and Testing 42](#_Toc47001391)

[5.1. Software Development Tools 42](#_Toc47001392)

[5.2. Project Architecture 43](#_Toc47001393)

[5.2.1. Web Architecture and Workflow 43](#_Toc47001394)

[What is MVC? 43](#_Toc47001395)

[MVC and MVT 44](#_Toc47001396)

[5.2.2. Machine Learning Architecture 44](#_Toc47001397)

[5.3. Design, Implementation and Testing 46](#_Toc47001398)

[5.3.1. Stage 1: Front-end 46](#_Toc47001399)

[Sprint Backlog 46](#_Toc47001400)

[Planning & Design 47](#_Toc47001401)

[Wireframe 47](#_Toc47001402)

[Implementation 48](#_Toc47001403)

[Development Tools 48](#_Toc47001404)

[Django Template Language 49](#_Toc47001405)

[Template Inheritance 50](#_Toc47001406)

[Django Snippets 52](#_Toc47001407)

[Testing 54](#_Toc47001408)

[Review 60](#_Toc47001409)

[5.3.2 Stage 2: Back-End 61](#_Toc47001410)

[Sprint Backlog 61](#_Toc47001411)

[Planning & Design 62](#_Toc47001412)

[Implementation 63](#_Toc47001413)

[Development Tools 63](#_Toc47001414)

[Testing 67](#_Toc47001415)

[Review 68](#_Toc47001416)

[5.3.3 Stage 3: Machine Learning 69](#_Toc47001417)

[Sprint Backlog 69](#_Toc47001418)

[Planning & Design 70](#_Toc47001419)

[Activity Diagram 70](#_Toc47001420)

[Implementation 70](#_Toc47001421)

[Testing 70](#_Toc47001422)

[Review 70](#_Toc47001423)

[Project Evaluation 72](#_Toc47001424)

[6.1. Reflection 72](#_Toc47001425)

[6.2. Improvements & Further Work 73](#_Toc47001426)

[6.4. Conclusion 75](#_Toc47001427)

[References 76](#_Toc47001428)

[Appendices 79](#_Toc47001429)

[Appendix A: Project Tools 79](#_Toc47001430)

# Table of Figures

[Figure 1 – Techniques used to detect Fake News 17](#_Toc47001431)

[Figure 2 - The Penn Treebank: POS tags 23](#_Toc47001432)

[Figure 3 - Parse Tree 24](#_Toc47001433)

[Figure 4 - Demonstrating terms visually on parse tree 25](#_Toc47001434)

[Figure 5 -Example of PCFG 26](#_Toc47001435)

[Figure 6 – Research conducted in each area for fake news detection 29](#_Toc47001436)

[Figure 7 - Waterfall Model 36](https://d.docs.live.net/b6910bcedc65acd4/Documents/Spread%20the%20News%20-%20Report.docx#_Toc47001437)

[Figure 8 - SCRUM Sprint 38](#_Toc47001438)

[Figure 9 -Project SCRUM Development 39](#_Toc47001439)

[Figure 10 - MVC System Architecture 42](#_Toc47001440)

[Figure 11 - MVT System Architecture 43](#_Toc47001441)

[Figure 12 - Typical Machine Learning System Architecture 44](#_Toc47001442)

[Figure 13 - Project's Machine Learning System Architecture 44](#_Toc47001443)

[Figure 14 - Website Wireframe 46](#_Toc47001444)

[Figure 15 - Part of Home HTML File 48](#_Toc47001445)

[Figure 16 - Base Template (base.html) 49](#_Toc47001446)

[Figure 17 - Example Block tag 50](#_Toc47001447)

[Figure 18 - Template Machine Learning Snippet 52](#_Toc47001448)

[Figure 19 - Where Machine Learning Snippet is added on Home Page Template 52](#_Toc47001449)

[Figure 20 - Sequence Diagram: Generic Interaction 61](#_Toc47001450)

[Figure 21 - Sequence Diagram: AJAX 61](#_Toc47001451)

[Figure 22 - Project URL Path 62](#_Toc47001452)

[Figure 23 - Website URL Paths 63](#_Toc47001453)

[Figure 24 - Example View 64](#_Toc47001454)

# Table of Tables

[Table 1 - Chapter Breakdown 12](#_Toc47001455)

[Table 2 - Attempted results to detect fake news 30](#_Toc47001456)

[Table 3 - Functional Requirements 33](#_Toc47001457)

[Table 4 - Non-Functional Requirements 34](#_Toc47001458)

[Table 5 - SCRUM Sprint Steps 39](#_Toc47001459)

[Table 6 - Requirements in corresponding Stage 40](#_Toc47001460)

[Table 7 - Language: Python 41](#_Toc47001461)

[Table 8 - Framework: Django 41](#_Toc47001462)

[Table 9 - Library: Scikit-Learn 41](#_Toc47001463)

[Table 10 - Requirements for Front-end 45](#_Toc47001464)

[Table 11 - Requirements for Back-end 60](#_Toc47001465)

[Table 12 – Understanding Project URL Paths 63](#_Toc47001466)

[Table 13 – Understanding Website URL Paths 64](#_Toc47001467)

[Table 14 - Requirements for Machine Learning 68](#_Toc47001468)

*Chapter 1:*

# Introduction

## Define Problem

News corporations are critical, necessary, and influential on a global scale. They are the main and most consumed source of communication that informs about events that occur worldwide. Due to this, news organisations are expected to deliver rich, trustworthy, and quality content. However, the way our society functions does not always and necessarily award those that are honest, hard-working, and/or humble, especially when it comes to honesty.

These news companies with time have become more oriented and focused on what will generate the most profit. This has led to a very grievous problem with news reports being misleading and containing false information. Misleading and misinformed news are also known by the term of **fake news**. This is not a new term; it is just one that has been catching attention lately. Nowadays there has been a shift, and many people are starting to question the content from news reports. In fact, there has been survey studies questioning the public about news and their opinions. Reuters Institute produces an annual report on the news media, a section of this report is dedicated to questioning individuals about their trust over their national news companies. In 2019, the United Kingdom scored 70% for the number of individuals that do not trust their national news outlets. It was one of the highest out of all the countries surveyed and with a 12% rise from the previous year. (Reuters Institute, 2019).

It is a fact that fake news is a pernicious problem, there have been many cases in which millions worth in money have been be lost due to this. For instance, back in 2008 a rumour spread that Steve Jobs had suffered a heart attack. This led to a $9 Billion market value dent (Daqing, L. et al, 2018). Yet this is a problem can continues to haunt and cause significant value. Another recent event which saw some of the consequence of fake news was the presidential elections of United States of America of 2016. Postelection there were findings of fake propaganda and the usage of fake news to trick Americans, here are some of the reported cases and the harm it caused:

* Within five months prior to the elections, there was roughly 170 million tweets. Of which 30 million were about the election. It was found that 7.5 million of those 30 million, which equates to 25%, contained fake news and/or were extremely bias (Bovet, A., and Makse, H.A, 2019)
* Guess, A., Nyhan, B. and Reifler J. (2018) found out that 1 in 4 Americans visited a fake news website throughout the presidential elections period.
* Additionally, Guess, A.’s (2018) report also claim that more than 40% of individuals do not know and/or did not conduct fact-checking when reading news about the elections.

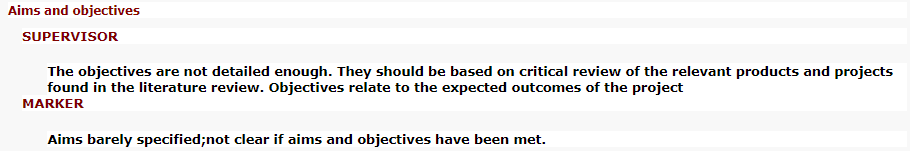
Another motivation for this project is that this year the United States either re-elects or elects a new president. Based off prior presidential run, many companies and individuals have attempted to prevent for this time the spread of fake news. This project is another attempt to aid and counter fake news for the public.

What once news corporations aim to produce honest has now turned into an industry in which many individuals hesitate if what is being produced to be true. It is not of importance as to why some of them went down this route, but more as to why would they go down a route that will in the long run be harmful for their business. Consumers should not be worried about what the news publishes as its meant to be a re-writing of an event. The fact that 1 in 2 people do not trust news is not acceptable and just proves how society has crumbled (Reuters Institute, 2019). Therefore, a standard need to be set, and if news companies will not make the effort to produce such content then it up to others to catch their lies and make them public.

## Aim and Objectives

These following are the aims and objectives set for the following project:

|  |  |
| --- | --- |
| Aim: | *To create an intelligent tool that can autonomously be capable from fed text, determine whether a given news report is real or fake.* |
| Objectives: |  |



## Chapter Breakdown

The remaining of the project will be going through the chapters mentioned in the table below.

Table - Chapter Breakdown

|  |  |
| --- | --- |
| Chapter | Description |
| Literature Review | This chapter goes over content that was relevant for the problem at stake. Terminologies, concepts, and previous attempts are brought up. |
| Requirements | In this chapter the requirements for the desired product are clarified. Additionally, determining their importance towards being int the final release. |
| Methodology | This chapter goes over the strategy that was used for the software development of the project. |
| Design, Implementation & Testing | In this chapter the planning, designing, development and testing of the key areas of the software are explained. This chapter in particular is mainly about the entire creation of the application. |
| Project Evaluation | In this chapter a reflective and critical analysis is made on the project conducted and determining how successful the application was and what future work is to be considered. |

Chapter 2:

# Literature Review

This chapter introduces and explores relevant research that was conducted in order to get a deeper knowledge towards generic news, fake news, and the current methods being used to be able to detect fake news.

## 2.1. News Media

Getting a better understanding of core concepts, techniques and terminologies used in the news will definitely be helpful. As this will help familiarise with the format that will be used throughout this project. Therefore, this section will explore mainly fundamental concepts for generic news and fake news.

### 2.1.1. News

News is defined by Tanikawa, A. (2017) as “*A report of what a news organization has recently learned about matters of some significance or interest to the specific community that news organization serves*”.

The way the news used to be delivered was through newspapers, but ever since the rise of the internet the format of news has changed. As well the amount of news being made, before news would be made on a 24-hour period, while nowadays as soon as there is an occurrence news companies are rushing to deliver the story to the public. This change of delivery has also meant the change of news style and writing (Tanikawa, A., 2017). The following sub-section goes through the ways in which news reports has changed over time.

#### News Format, Structure and Writing

As every piece of writing, there is normally a technique that is used/recommended as it has demonstrated success. Fang, I. (1991) discusses some of the most common formats, writing styles and structures that are used to produce news reports. The following list is a summarisation the key points made and that could be of relevance for this project:

* **Inverted Pyramid –** this is about the newspaper covering the most important part of a story within the first paragraph, also known as *lead*. The rest of the news is about unfolding the details of the event. This is used to be able to capture the attention of the reader to continue reading the rest of the story.
* **Five W’s and One H (5W1H)** – This is another technique used to be able to produce news reports, typically most of these are covered within the lead. Meaning that the reporter should write about the “**Who** was involved, **What** happened, **Where**, **When**, **Why** and **How** did it happen”.
* **Less formal writing style –** it has also been seen that over time newspapers have stopped being extremely formal, and instead have opted to use language that is more commonly used within radio and television.
* **Substance/Content covered has no relationship with writing –** Fang points out that the writing style used is completely independent to the story covered and the length of the news report.
* **Story has an impact on writing –** the difficulty of the story has an effect as to how the journalist writes the news. The example used in the paper is that a story on “mayor's arrest for drunk driving” would be simpler to write than “the mayor's presentation of the city budget” as more simplistic wordings, such as one syllable words, are/could be used.
* **Quote for sentence structuring –** Fang claims that all news writers would agree on the following quote from “Kansas City Star Style” book for sentence structuring:

*“Use short sentences. Use short first paragraphs. Use vigorous English, not forgetting to strive for smoothness. Be positive, not negative.”*

Another study by Tanikawa, A. (2017) also shows how American news, specifically The Washington Post, Los Angeles Times and International Herald Tribune changed their writing and other features over time. Here is another list of the most relevant changes that could be used for this project:

* **More informal:** This paper also found that informal writing is more common. States that papers resemble more magazine writing. Which consists of overextended analysis of stories, including wherever possible well-known people/businesses and increasing the number of visualisations. All for the purpose to catch the attention of the reader.
* **Other techniques employed:** Nowadays news does not necessarily follow the traditional 5W1H and Inverted Pyramid techniques, although still used to some extent. Instead some are using other situation, anecdotal and descriptive types of leads and structures.
* **Impact of the Internet:** with the rise of the internet the amount of content that is in the news has declined nearly 20%.

The way news is formatted, structures and written is of importance as they are a starting indicator as to how potentially fake news could be detect from solely text. In addition, in later sections it is discussed further how these factors are contributors to helping to solve the problem.

### 2.1.2. What is Fake News

The term fake news has been degraded significantly and been given many definitions for many purposes. The reason as why this is the case is based upon what perspective and branch of knowledge you are using fake news from (Clayton, K. *et al*, 2019).

This being said the definition that this project for fake news is by Lanius, D. and Jaster, R. (2018) which says, “*fake news is news that does mischief with the truth. That is because, as we argue, fake news is characterized by two shortcomings: it lacks truth and truthfulness. More specifically, fake news is either false or misleading (lack of truth) and it is propagated with either the intention to deceive or an utter disregard for the truth (lack of truthfulness)”*

Additionally, between papers there is also a confusion for wordings that are similar to for fake news, such as “fraudulent news”, “misleading news”, “junk news” and more. Some papers use these words freely as synonyms, but others have different meanings for each term. Therefore, it is worth noting that this report will treat those variations as synonyms unless mentioned otherwise.

### 2.1.3. Types of Fake News

As seen from the above subsection controversy, the reader may also wonder “what is classified as fake news?”. This is another controversy between papers as there are different opinions as to what could be regarded. This really depends on the research conducted and what the authors decide to portray fake news as in their papers. Therefore, this project will also be using Rubin, Chen, and Conroy (2016) to determine the types/classifiers for fake news:

1. **Serious Fabrications**

This type of fake news can be found mainly in yellow press and tabloids. These are areas of journalism where exaggerated headlines are used to catch the attention of readers. Along with that the story tends to be scandalous and over-dramatic. Some examples of this type of fake news are found in gossip columns, crime stories and astrology.

1. **Large-scale Hoaxes**

This type is similar to “Serious Fabrications”, as the intentions to lie to the audience. However, what differs with this type are the malicious intents of the journalist(s) with their publication. These writers are more careful with the content and wording of their reports, to make them sound as legitimate as possible. This is for any reader that is uniformed or not cautious to be misled in some manner, typically for the benefit of the journalist(s)

The degree of severity with this type is above the other types of fake news. For instance, in 2016, an individual was deceived and entered to a restaurant firing an armed weapon (Tandoc Jr, E.C *et al*., 2017). This was all due to some distinct political views that the journalist had which led them write about a scandalous hoax about one of the candidates for presidency.

1. **Humorous Fakes**

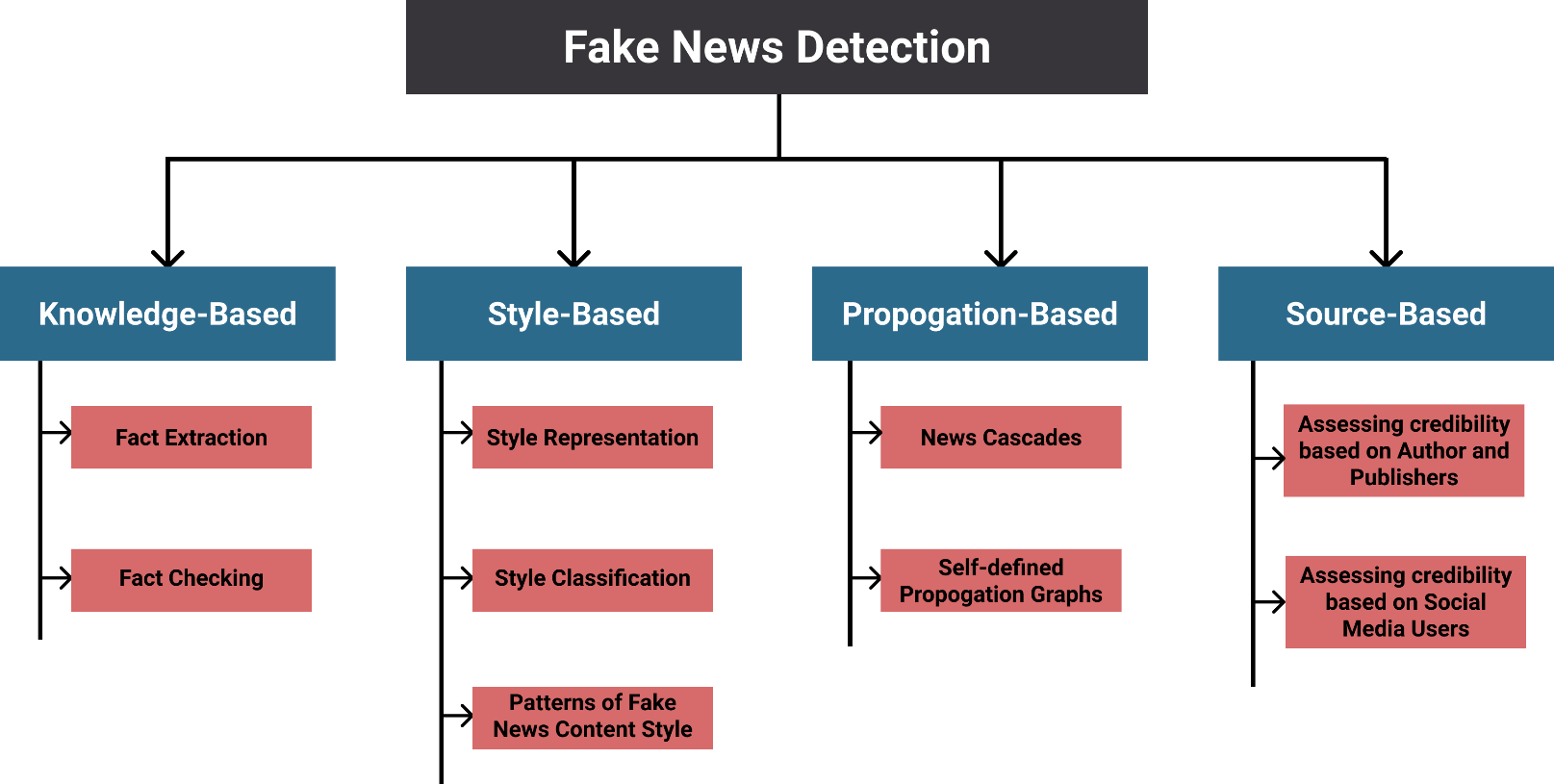
The news produced by this type are for individuals that are conscious of the topic and that can realise that the news article produces was fabricated with the intentions to humour the audience. However, for readers that are unaware of the topic of the written article will most likely take it as a factual news reportage. Which in consequence will have the same effect as the other types of fake news.

## 2.3. Relevant Research

In this section, it will be reviewed some of the current methods that have been created to be able to determine the reliability of news reports. There are many ways in which news credibility can be determined. No method is by means perfect, as all of them have some sort of defect that causes them to not be accurate, feasible or currently not possible.

The following figure illustrates the four main methods that have been employed to be able to detect fake news:

Figure – Techniques used to detect Fake News



The following sections will go in depth into the most common manual way to perform fake-news check, explain as to why this is not an approach that should be used as currently and in the long-run it simply will not be useful. Followed up by a quick review of the techniques shown above. Then reviewing the approach that is the closest to what the project aims to achieve. Finally, will be presenting and reviewing actual results for fake news detection.

### 2.3.1. Fact Checking – Manually

This is when an individual takes time into reading a news report and fact-checking with other resources and then concluding if whether the news article can be trusted or not. This process is also known as Expert Oriented Fact Checking (*Beer, D.B. and Matthee, M., 2020)* This sub-section will discuss how this method works and how it is not a viable option in the long term.

There are few companies that have the economical capability into funding and fully employing people for the sole purpose of fact-checking. PolitiFact, Snopes and FactCheck are some of the few companies and exceptions that can hire fact checkers. The difficulty lies that this is an industry does not have much financial benefit. Many of these companies are not funded economically by any organisation and have to solve the monetary difficulty themselves. Mainly through the use of donations, advertisements, and other non-subsidised methods of income.

Regardless there are a set of individuals known as *guardians* that are people that find enjoyment into fact-checking news reports and help communities by providing this fact-checking service by referring people to links and to sources that have confirmed the legitimacy of news articles, such as the companies mentioned earlier (Vo, Nguyen and Lee, K.. 2018). However, another problem lies with further research checking fact checkers decisions on news reports. Some have shown that the conclusion to many articles are vague and inconsistent between fact checkers (Lim, C., 2018). The paper then proposed that even for articles that have been supposedly fact checked should not be trusted fully and the consumer should verify with other sources. Overall, this simply means that this problem is not being solved as the initial problem continues to persist.

Apart from the above problems, the main one lies with how the task that is being done is unfeasible when compared with the amount of news that is produced. The ratio between fact checkers and news reports is vastly different. Therefore, why automatic computer-based approaches have been created to be able to detect fake news.

### 2.3.2. Automatic Techniques for Fake News Detection

This section overviews the techniques that are shown in figure 1, discusses briefly as to what are some core concepts about them, how they work and some final notes regarding the method used

Approaches used to detect fake news:

* **Knowledge-Based**

Knowledge-based approach mainly consists of two tasks: extracting the data and fact-checking (Zhou, X. and Zafarani, R., 2020).

* **Style-Based**

This approach uses the content of the news report as input. Then this input is fed to a Machine Learning or Deep Learning method to learn about the *style of writing,* as fake news is believed to have different writing approach compared to typically news. This is believed to be because of the malicious intentions behind fake news. Upon reading about this approach, seems to be the path as to what that aim of the project is to complete. Therefore, will be examining this approach in more detail in the following section.

* **Propagation-Based**

This approach relies on the fact that it has been proven by researchers that fake news tends to spread through social media and news organisations way quicker. Therefore, the technique used mainly in this approach is by evaluating the spread speed of a news articles and determine if the news report falls within a fake new or real news dissemination trend (Zhou, X. and Zafarani, R., 2020).

Machine Learning can be used for this technique as well. It can be trained to learn the pattern and trend of fake news spread compared to real news spread. Therefore, the input that is fed to the model is different as to compared to what was used for the style-based approach. In this method *News Cascades* are used to capture the propagation of news throughout a network (social media, news organisations, and more) (Zhou, X. and Zafarani, R., 2020).

* **Source-Based**

The final approach branch that is used to be able to detect fake news is by evaluating from where the news report originated. Which means that the news is classified based upon the source’s reputation.

### 2.3.3. Style-Based

AS previously mentioned, this approach seems to be the most appropriate as to what the aim of the project intends to achieve. Therefore, will go in greater depth about how this technique is used to detect fake news. Firstly, will be identify and learning about the different features that can be used to feed a model, then followed by the Machine Learning models that are more commonly used for this method.

#### Style Representation

The first step to be able to develop a Machine Learning model in order to solve this problem is how is the news report going to be represented? This is a key factor as it will be seen in the following sub-sections that there are many inputs that can be fed to the Machine Learning model as its input/training.

##### What is Language?

Muin, F. (2019) defines language as "a system of arbitrary vocal symbols used for human communication”. The key word from the definition is *arbitrary.* What Muin later in this chapter explains that all the elements that have been created about language were once randomly made and individuals of that time happen to agree upon proposed linguistic forms.

With this definition it demonstrates that language has no quantifiable way to measure it as words are arbitrary. Now, considering that style-based systems depends on the usage of text, then the key point is how is the text input going to be quantified for the model to understand. This section covers different ways in which text has been exploited to extract information that is used to be able to feed to the model.

##### The Different Features

In the above sub-section of *What is Language?* It was seen how language is difficulty when wanting to use it as a measurement. Regardless language itself has its own set of (non-quantifiable) rules. These rules come mainly into use when writing or speaking, as there is an understanding as to how to organise and produce a coherent sentence, it is simply a concept that is not thought out in great detail. This section will explore to some extent some linguistic concepts/rules, then explaining how these are used as features for a Machine Learning models:

* **Lexicon**

From the definition provided for language, it can be inferred that overall language is at least a sentence/phrase. Language can further be broken down into smaller components, such as words and grammar. The term *word* is not commonly used in linguistics as it is to ambiguous to what is being referred to as “word”. Therefore, the term *lexicon* is used instead, as this is the formal term to refer to a vocabulary of units. Lexicons can be further broken down into other components, such as: lexeme, word forms, meaning, and more, these other components are also known as *lexical items* (Aronoff, M., and Rees-Miller, J., 2017) (Mondal, P. , 2018). However, as of now there is no need to go into more depth to understand the lexicon feature.

Using these new terms and understandings, lexicons can be used as a feature input for the Machine Learning model. The technique used here is known as *Bag Of Words (BOW*) (Conroy, N.K., Rubin, V.L. and Chen, Y., 2015). Following are some terms that should be known and how BOW works:

***n-gram*:** this is the number of lexica that want to be considered , depends upon the developer whether a lexicon or lexica should be used (Shu, K., Silva *et al,* 2017). Saying 2-gram is a group of 2 lexica, if 3-gram then of 3 lexica and so on.

***Bag of Words:*** Each n-grams are considered of equal importance. How it works is that it uses the frequencies of lexica within the corpora provided. The reasoning as to why this technique is used is basing what was earlier stated that it is believed that the content within fake news and real news is different. Therefore, basing off that hypothesis it is believed that certain words will appear less/more than in real news than in is in fake news. However, a downside with this model is that it fully disregards and ignores the structure of the corpora.

* **Syntax**

In the previous feature, it was discussed how there are two main parts to create language: there was lexicons and grammar. The syntax feature will now introduce grammar and how it is used as a feature.

Syntax is all about knowing how words should be arranged to be able to structure a phrase (Jurafsky, D. and Martin, J.H., 2019). *Word classes* are a way to determine the type of a given word, three of the most commonly used word classes are: nouns, adjectives, and verbs ( Vogel, P.M. and Comrie, B., 2000). Word classes is the main characteristic that both the following ways to do syntax feature can be developed:

***Shallow Syntax:***

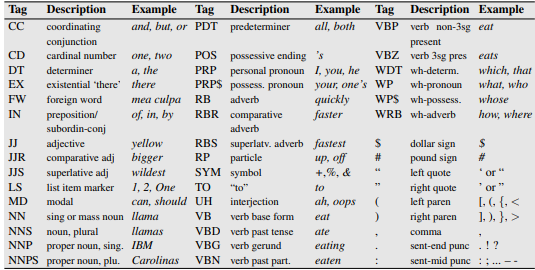
The technique used for shallow syntax is known as *Part-Of-Speech (POS*). Part of speech is a synonym for word classes.

This technique involves assigning the POS tags for each lexicon in the corpora. However, this is not as simple as that as the same word depending on the context can have a different tag, example:

Phrase 1: The car is moving fast

Phrase 2: That is a fast roller coaster

The word fast is used in both example phrases. However, they have different POS tagging. In the first one fast is used as an adjective while in the second it is an adverb. Therefore, identifying the correct POS is of importance. The tags that are given to lexica tends to be a couple of letters or symbols that represent a world class. The following image is part of *The Peen Treebank,* which is one of the most popular POS tags system:

Figure - The Penn Treebank: POS tags

*(Zhou, X. and Zafarani, R. 2020)*

Once the tags have been established then this technique requires to count the frequencies of each tag in each corpus. The frequency amount for each POS is what is used then for the classifiers input and this is how shallow syntax differentiates real from fake news.

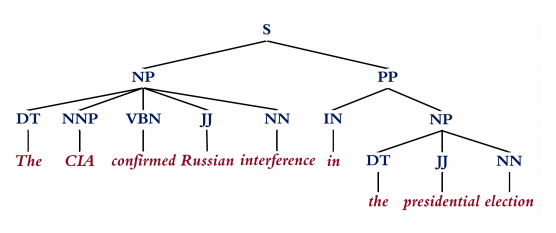
***Deep Syntax:***

The technique used for deep syntax is *Probabilistic Context-Free Grammar (PCFG).* However, before going into more depth about PCFG some prior concepts need to be understood.

###### Constituent Structure, Parse Trees & Context-Free Grammar (CFG)

This concept of being able to set grammar into a constituent structure was proposed by Wilhelm Wundt in 1900 (Jurafsky, D. and Martin, J.H., 2019). What this means it that a clause that is written is made up of components that work together and collectively create the phrase. The important factor here is that word classes can be used to represent as to how the constitute structure for a phrase was reached.

This constituent structure can be visualised through parse trees (Zhou, X. and Zafarani, R. 2020). Generally speaking, going deeper into the tree shows what the parent node was made of. For instance, as it is shown in the figure below that a sentence can be broken down firstly into a *Noun Phrase (NP)* and *Preposition Phrase (PP).* The following section for Noun Phase is the word class for all the words while Preposition phrase has many more layers.

Figure - Parse Tree

*(Zhou, X. and Zafarani, R. 2020)*

Context-Free Grammar (CFG) uses parse tree and constituent structure principles to further explain the syntax of a clause. CFG can be formalized with the following 4 parameters:

Where,

🡪 set of non-terminals

🡪 set of terminals

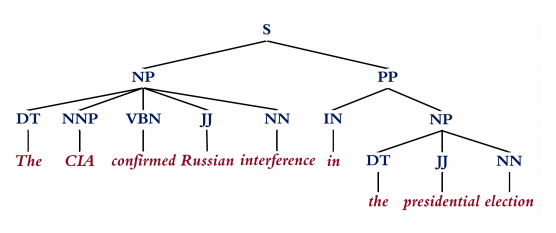
🡪 set of rules, in format

🡪 starting symbol

|  |  |
| --- | --- |
| *Terminals:* |  |
| *Non-terminals* |  |
| *Rules:* |  |
| *Starting symbol:* |  |

To understand further the above terms, the below figure is part of the previous figure, with a key to indicate where each of the above terms are located in this parse tree:

Figure - Demonstrating terms visually on parse tree



🡪

🡪

🡪

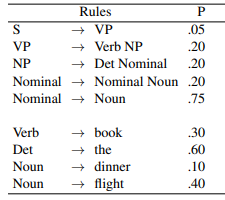
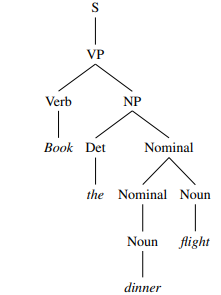
🡪

###### Probabilistic Context-Free Grammar (PCFG)

This is the feature that is used for the model to be fed on. The difference that PCFG and CFG have is for each non-terminal variable there is a probability out of 1 for all the rules it has, therefore:

Here is an example from Jurafsky, D. and Martin, J.H., (2019) showing the parse tree of a corpus and the rules for each non-terminal word class, with their corresponding probability:

Figure -Example of PCFG



*(Jurafsky, D. and Martin, J.H., 2019)*

The rules with its probabilities for each corpus are what is fed to the machine learning model to be able to train it. In essence, the model is being provided the grammar rules and structure of corpora from real news and fakes news to be able to learn from the patterns within them.

* **Disclosure**

The prior features from this one was concerned to extract useful information for the model at a word-level. Discourse is the collective handling and usage of multiple sentences of a corpus (Eisenstein, J., 2018). The way this problem is solved is through the use of *Rhetorical Structure Theory.*

###### Rhetorical Structure Theory (RST)

This is a descriptive framework, which provides features at a clause-level (Mann, W.C and Thompson, S.A., 1988). Features that can be extracted includes:

* Hierarchical structure of a corpus
* The relation between clauses
* Segmentation of the corpus
* Comprehensive analysis rather than selective commentary

Rhetorical Structure Theory formally speaking is made up of 4 elements:

When it comes to discourse there are more uses that can be used to help identify the differences between fake news and real news:

Rhetorical Structure Theory (RST) and rhetorical parsing tools can be used to capture the frequencies of rhetorical relations among sentences as features

* **Semantic**

Finally, at a semantic level, such frequencies can be assigned to lexicons or phrases that fall into each psycho-linguistic category (e.g., those defined in Linguistic Inquiry and Word Count (LIWC) [Pérez-Rosas et al. 2017]), or that fall into each self-defined psycho-linguistic attribute. These attributes can be derived from experience, or be inspired by related deception theories (see news-related theories in Table 2 or [Zhou et al. 2019a] as a typical interdisciplinary fake news study). Based on our investigation, such attributes and their corresponding computational features can be grouped along ten dimensions: quantity [McCornack et al. 2014], complexity, uncertainty, subjectivity [Undeutsch 1967], non-immediacy, sentiment [Zuckerman et al. 1981], diversity [Undeutsch 1967], informality [Undeutsch 1967], specificity [Johnson and Raye 1981], and readability (see Table 4), which are initially developed for identifying deception in computer-mediated communications [Fuller et al. 2009; Zhou et al. 2004b] and testimonies [Afroz et al. 2012], and recently used in fake news detection [Bond et al. 2017; Pérez-Rosas et al. 2017; Potthast et al. 2017; Zhou et al. 2019a].

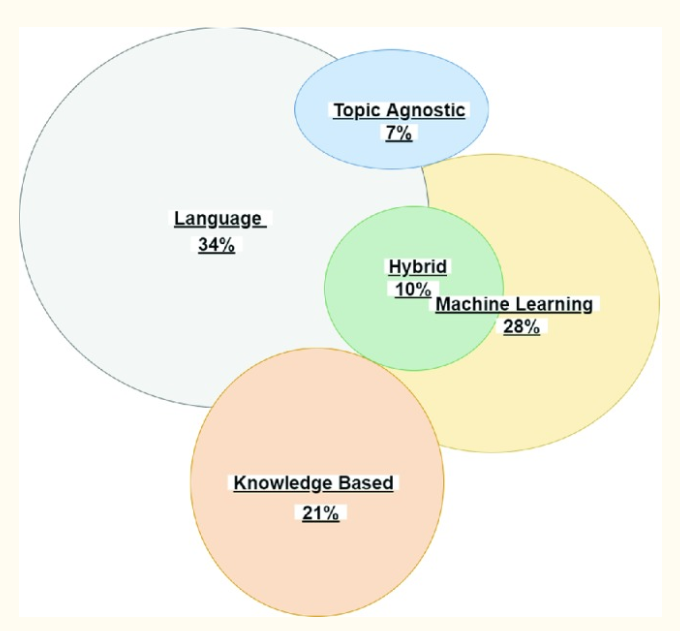
#### Style Classification

Following the representation that can/could be fed to the Machine Learning model. The next part is to choose what classification model will be used.

### 2.3.5. Prior Attempts

As seen from the above sections, there are multiple ways in which news has been tested automatically to check their veracity. A paper by Beer, D.B. and Matthee, M. (2020) investigated the research conducted in this problem domain between 2009-2019 and how techniques have evolved for detecting for fake news, in the hope to inform researchers about the current state and technologies used. Therefore, the following figure has been extracted from their paper demonstrating the approaches used as well as the amount of research done within each area:

Figure – Research conducted in each area for fake news detection



*(Beer, D.B. and Matthee, M., 2020)*

The following table is about attempts towards detecting fake news, with the technique being used and the results that were achieved:

|  |  |  |
| --- | --- | --- |
| Citation | Technique/Approach used | Results and Observations |
| (Marquarft, D. 2019) | - POS tagging  - sentiment analysis  - Using Gunning readability index, an application to determining the complexity of a text, ranging from 1 to 7 | - Ratio between noun and verbs used differed between true and fake news. True news had a ratio of 2.67 while fake news had a 4.27.  - With Gunning readability index, the complex of true news was slight over fake.  However, the key observation was that for true news, the topic with the least difficulty was politics (levels 3-4) while the most complex was on science and legal matters (levels 6-7)  - Negative news was confirmed to be an indicator for fake news. |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Table - Attempted results to detect fake news

Different papers have different positions as to what method is regarded as the best option and path to follow.

## 2.4. Gaps Identified

Chapter 3:

# Requirements

This chapter will go through the requirements that have been set for the development. They have been ordered in a prioritisation manner, which will help complete elements of the project that are of more importance over others. The requirements have also been split into the services that will and will not be used, as well as how such services will be measured.

## 3.1. MoSCoW Methodology

There is the uncertainty of the amount of time that can be invested into this project is unknown. Therefore, it is of importance to prioritise the requirements into essential and non-essential as that will provide a target as to what is being achieved. A common method to be able to hierarchize the requirements is by using requirement prioritisation methods.

A well-known requirement prioritization technique is **MoSCoW**, it is a common practice in the agile development environment (Vestola, M., 2010). This method has four different levels of prioritisation (Kuhn, J., 2009). These are the definitions proposed by Achimugu P. *et al* (2014) for each of the levels:

|  |  |
| --- | --- |
| **Must** | *Requirements are not negotiable; the failure to deliver these requirements would result in the failure of the entire project* |
| **Should** | *Features that would be nice to have if at all possible* |
| **Could** | *Features that would be nice to have if at all possible but slightly less advantageous than the “S” (Should)* |
| **Won’t** | *These requirements are not unimportant, but they will definitely not be implemented in the current software project. They may, at a later stage, be created.* |

In addition, the requirements have been split up into ***functional requirements***and ***non-functional requirements***. This project will be using the following definitions:

|  |  |
| --- | --- |
| **Functional Requirements (FR)** | *“Functional requirements specify the functions of the system, how it records, computes, transforms, and transmits data.”* (Lausen, S., 2002) |
| **Non-Functional Requirements (NFR)** | *“Non-functional requirements describe the nature and limitations on the project instead of its functionality, also this term describes the non-behavior aspects and attributes of the system including usability, portability, security, understandability, reliability, and modifiability. In general, the non-functional requirements highlight the requirements that describe "how good" the software.”* (Hudaib, A. *et al,* 2018) |

## 3.2. Functional Requirements

In this section the services that the final application must, should, could and won’t have are covered. These requirements set out a standard for the application at the end of the project as well as other functions that can be used for further development.

Table - Functional Requirements

|  |  |  |
| --- | --- | --- |
| ID | Priority | Requirement Description |
| FR-01 | **Must** | The intelligent system will use Machine Learning techniques only to be able to determine the legitimacy of news reports |
| FR-02 | **Must** | The system will only accept URLs that link to an English text report |
| FR-03 | **Must** | The only accepted input from the user should be URLs, anything else will be rejected |
| FR-04 | **Must** | The system has to be able to extract the content from the provided URL |
| FR-05 | **Must** | The Machine Learning application after thorough testing should be at least achieving 75% |
| FR-06 | **Should** | There system shall provide feedback to the user regarding the determined legitimacy of the report. |
| FR-07 | **Should** | The system should be able to process more than one URL at a time |
| FR-08 | **Should** | The systems output will provide a reason as to how it came to its output decision |
| FR-09 | **Should** | The website application should be able for other users to then extend upon the current project easily |
| FR-10 | **Should** | The machine learning application should be ready to use for other users to test with other datasets and/or models |
| FR-11 | **Should** | The machine learning within the website should be easily exchangeable if needed |
| FR-12 | **Won’t** | The system will use different artificial intelligent approaches to create a hybrid system |

## 3.3. Non-Functional Requirements

In this section, it will be covered the requirements that will judge the functional requirements. Essentially, these requirements/judgements will allow quantitively and qualitatively measure the application.

Table - Non-Functional Requirements

|  |  |  |
| --- | --- | --- |
| ID | Priority | Requirement Description |
| NFR-01 | **Must** | The website will have a responsive web design for desktop users |
| NFR-02 | **Must** | The websites functionality shall fully work on Google Chrome browser engine. |
| NFR-03 | **Must** | The project should attempt to follow the most up-to-date technologies and practices. |
| NFR-04 | **Should** | The website should work on the most popular browser engines |
| NFR-05 | **Should** | The website should follow recent styling trend |
| NFR-06 | **Should** | The website should process the data for the Machine Learning dynamically |
| NFR-07 | **Should** | The website should be able to classify in less than a minute per news report |
| NFR-08 | **Should** | Operations should be carried out in a time frame that does not cause the user to believe that the system has become unresponsive |
| NFR-09 | **Should** | The programming should adopt Python PEP8 standard format |
| NFR-10 | **Could** | Have security measures for malicious input |
| NFR-11 | **Wont** | The project to be entirely documented |

Chapter 4:

# Methodology

In this chapter different approaches to develop software will be presented and compared. Followed by the technique that this project will use throughout the project, with some modifications to the method.

## 4.1. Software Development Strategies

Software Development Life Cycle (SDLC) is defined as the method that is used to develop software, these models have steps and techniques as to how the software should be developed in order to achieve the highest success (Mishra, A. and Dubey, D., 2013). The SDLC models can generally be categorised into “Traditional Development” and “Agile Development” (Leau, Y.B. *et al*, 2012).

### 4.1.1. Traditional Development Models

The most recognised traditional software development models include: Waterfall, V-Model, and many more. In this section, Waterfall will be explained and later discussed with the agile methodology.

#### Waterfall Model

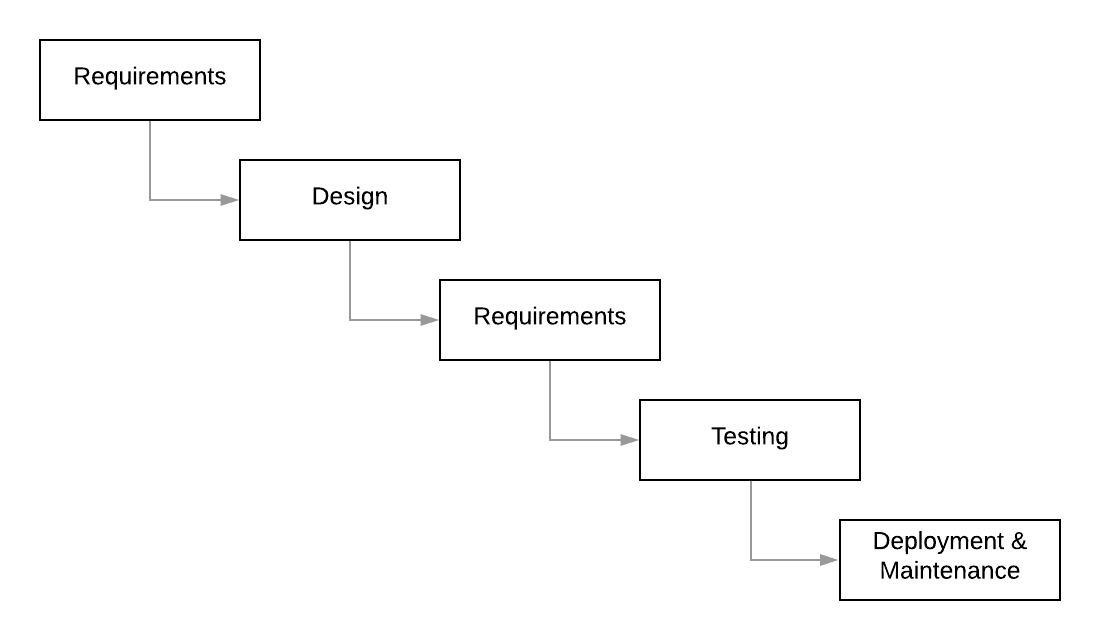


Figure - Waterfall Model

Model introduced by Winston Royce in 1970 (Mishra, A. and Dubey, D., 2013), it a linear development method, what this means is that a phase once completed is not revisited again. It is easy model to understand, however, its rigid architecture creates is hard to be able to correct mistakes from previous phases.

### 4.1.2. Agile Development Models

A group of seventeen individuals, referred to as “the Agile Alliance”, met in 2001 and discussed about the need for an alternative way to develop software. This meeting is now known as “The Agile Manifesto and what sprung was a new series of “lightweight methodologies” and rules for Software Development and now also entering into other business frameworks (Hudaib, A. *et al,* 2001).

Within Agile there are several popular models, most popular ones include: SCRUM, Extreme Programming and more. A benefit with agile methods is the adaptability that can be used if necessary, this allows for unforeseen problems to be addressed and modified on time.

#### SCRUM

SCRUM like other agile models is a dynamic method that revolves around teamwork, working software, customer interaction and adaptability to change (Schwaber and Sutherland, 2017). What is of interest about SCRUM is how the sprints creates small objectives on short scales of time.

Sprints are the main focus when it comes to SCRUM as it creates a flexible Software Development approach. Each Sprint is given a goal, all must work to be able to accomplish this target in the proposed time frame. To be able to achieve the highest success rate the sprint is split into further stages. There is no standard as to what steps need to be taken as this varies with every project. Generally speaking, the recommended stages include planning, development work, sprint review and sprint retrospect (Schwaber and Sutherland, 2017).

## 4.2. Method

SCRUM is the chosen methodology to help control and organise the development of this project. This method is based upon the principles of the Agile Manifesto, along with its own set of techniques. It provides tremendous flexibility and adaptability for the development of the application. This is a particular point as to why SCRUM was chosen because it is likely that there will be at some point a delay as there are many concepts being learnt for the first time and other work being done simultaneously.

As Verheyen, G. (2013) points out, SCRUM is not a strict process that has obligatory and exhaustive steps. Instead there are proposed techniques that is left for the team to decide whether or not to opt. This is brought up as this project will be using SCRUM as its basis, however, there has been some additional decisions that are not particularly part of the SCRUM framework that will disclosed in the following paragraphs.

### 4.2.1. SCRUM Approach for Project

It has been decided that the total project sprints will be divided in these stages: Front-end, Back-end and Machine Learning. All of these contain parts of the requirements that are of great importance (the MoSCoW “must” requirements). Therefore, the idea behind this method is that in the worst-case scenario the end application will contain all of the “must” requirements. Anything else will be an application with additional features, which would improve the User Experience (UX) and the usages for the software.

Each stage is comprised of a number of sprints. Each sprint will follow roughly conventional SCRUM sprint protocol, Figure 4 gives and illustrative representation as to what comprises each sprint and what exactly does each step of the sprint mean.

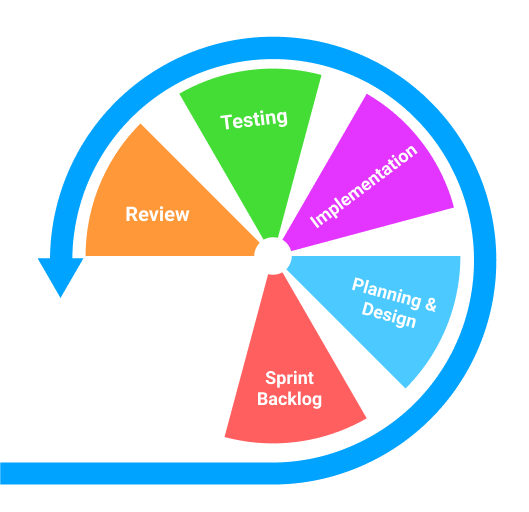
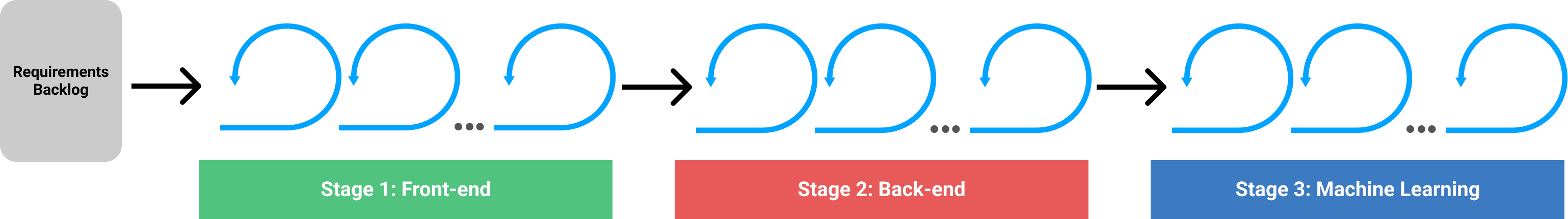
Figure - SCRUM Sprint

Table - SCRUM Sprint Steps

|  |  |
| --- | --- |
| SCRUM Sprint Step | Description |
| Sprint Backlog | **Requirements, from Chapter 2, that are related to the stage name are selected to be completed within the sprint**.  This stage formalizes the specific tasks that need completion. |
| Planning & Design | Focuses on how the current stage will be implemented. Therefore, will have a series of diagrams, sketches, charts, or other relevant planning or designing techniques that have been used to condensate the implementation process. |
| Implementation | Overviews the key implementations, in general will consist mainly around code. This is conveyed through snippets of code and explain the chosen approaches/techniques.  This stage highly focuses on the sprint backlog being developed and completed. |
| Testing | Testing that the program functions adequately. Regardless, most emphasis on the testing will surround that the sprint backlog has been properly implemented. |
| Review | At the end, some observations are left for later development and potential comments that can be reused in “Project Evaluation” chapter. |

Overall, the project will have the following development overlook:

Figure -Project SCRUM Development

In addition, the table below contains the same requirements brought up in *Chapter 3: Requirements.* However, now they have been categorised to their corresponding stage section.

Table - Requirements in corresponding Stage

|  |  |
| --- | --- |
| Stages | Requirement ID’s |
| Front-end | FR-06, FR-09, NFR-01, NFR-02, NFR-03, NFR-04, NFR-05, NFR-06, NFR-08 |
| Back-end | FR-02, FR-03, FR-04, FR-07, FR-08, FR-09, FR-11, NFR-03, NFR-06 |
| Machine Learning | FR-01, FR-05, FR-10, NFR-03, NFR-07 |

Chapter 5:

# Design, Implementation and Testing

## 5.1. Software Development Tools

In this section the most important software development language, framework and library are covered. In *Appendix A: Project Tools* all the software utilities are covered and explained as to how they were used.

Table - Language: Python

|  |  |  |  |
| --- | --- | --- | --- |
| Language – Python | |  | *URL:* https://www.python.org/ |
| *Description:* | Python is an extremely popular language in multiple fields of Computer Science, it is used in Data Science, Web Development and more. Its readability and flexibility create an attraction to it. On top of that Python has a very dedicated community for creating libraries and frameworks for all disciplines. | | |

Table - Framework: Django

|  |  |  |  |
| --- | --- | --- | --- |
| Framework – Django | |  | *URL:* [*https://www.djangoproject.com/*](https://www.djangoproject.com/) |
| *Description:* | This is one of the many frameworks that has been dedicated for Python. It is mainly used for Web Development. Such framework is meant to help developers, mainly with menial tasks that are common within many web development projects. | | |

Table - Library: Scikit-Learn

|  |  |  |  |
| --- | --- | --- | --- |
| Library – Scikit-Learn | |  | *URL:* [*https://scikit-learn.org/stable/*](https://scikit-learn.org/stable/) |
| *Description:* | This is a Machine Learning library. This library mainly covers supervised and unsupervised learning algorithms/models. Similarly to Django, this library is meant for developers to use the pre-defined models, instead of having to write up the model from scratch | | |

## 5.2. Project Architecture

This project is about the production of two different applications and then combining both of them into a single application.

### 5.2.1. Web Architecture and Workflow

Since Django is a framework, then its intentions are to make the life of developers much easier. One of which is having an architectural system for development. The architectural design that Django follows is known as **Model-View-Template (MVT).** It is practically a synonym for the more commonly recognised design pattern known as **Model- View-Controller (MVC).**

#### What is MVC?

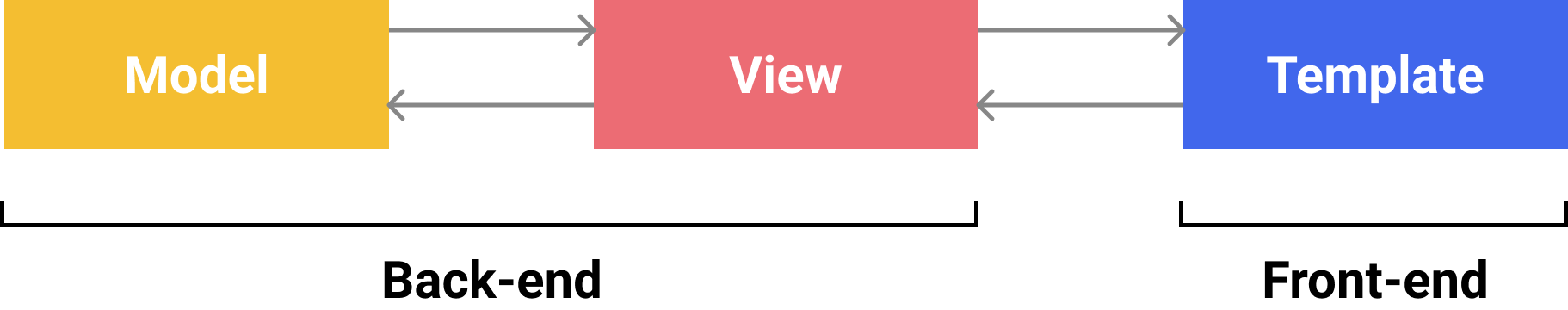
Model-View-Controller is a software system design that is used greatly on the web, due to the typically of the two different roles of Front-end and Back-end. MVC is split up into three components:

* **Model:** This section is meant for all the data and logical operations of the program. (Krasner, G.E. and Pope, S.T., 2014)
* **View:** This is the actual presentation of the application, known as the User Interface. No logical process is done in this area of the application. (Krasner, G.E. and Pope, S.T., 2014)
* **Controller:** This is the connecting point between the View and the Model, can be refered as the middleman. It is what will collect the input from the user and send it to the model. Can also collect an item from the model and present it to the view. (Krasner, G.E. and Pope, S.T., 2014)

Figure - MVC System Architecture

#### MVC and MVT

The main difference between MVT and MVC lies on how each section of the design is named and the task that each section has; the figure below illustrates which section of MVT is paired with MVC.

Figure - MVT System Architecture

In MVT these are the tasks of each section:

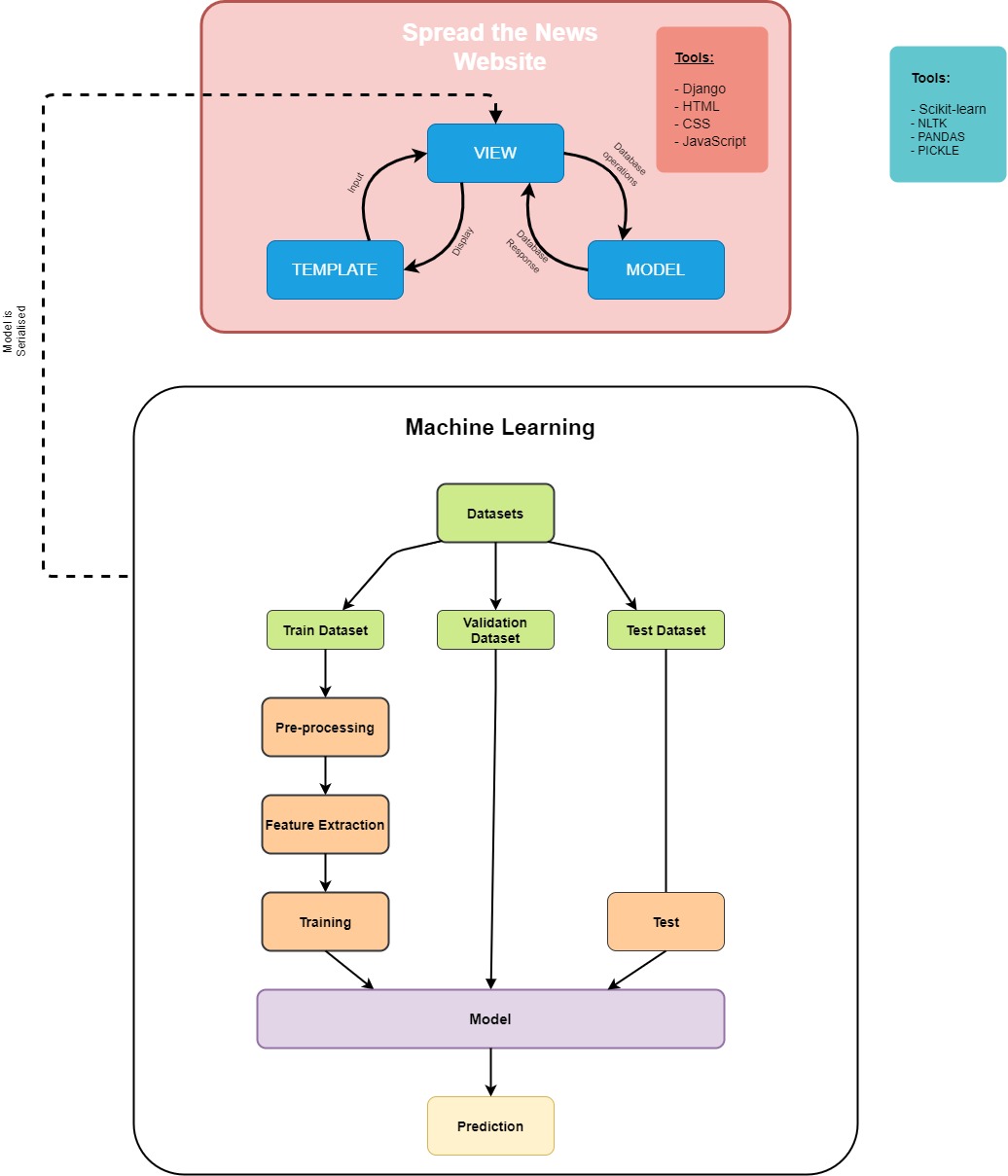
* **Model:** This section is meant for the database of the application.
* **View:** Similarily to the Controller of MVC, the View is the middleman between the Template and the Model. But in addition, it is also the place where the logical operations processes occur.
* **Template:** This just like the View of MVC, is the place that has to do with anythin that has to do with graphical for ther user.

### 5.2.2. Machine Learning Architecture

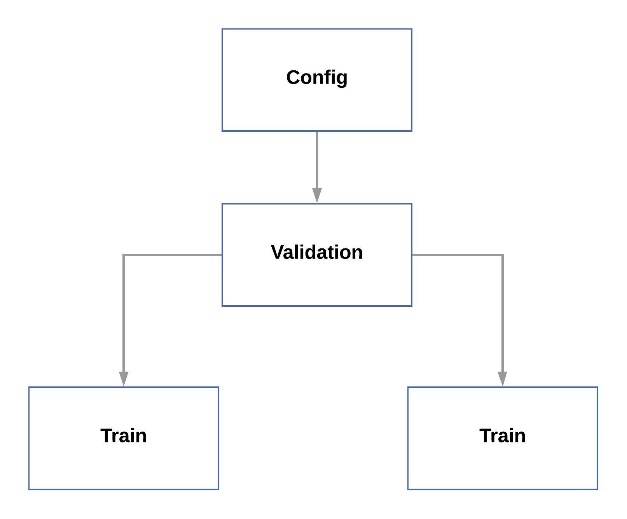
The architecture for the machine learning project will be different than the conventional way as to how other developers implement it. Main reason being that the project has set a requirement for other individuals to be able to use the project to test with other variables.

Conventional Machine Learning applications for this type of problem have the following architecture:

Figure - Typical Machine Learning System Architecture



Whereas this projects main high-level difference lies that the Machine Learning training and testing will be done separately. This will deliver for external users to try out different models and datasets, and so satisfy the due requirement.

Figure - Project's Machine Learning System Architecture

## 5.3. Design, Implementation and Testing

This subsection is the culmination effort of all the prior topics that have been discussed up to this point. As stated, the project has been developed in Front-end, Back-end and Machine Learning stages. Each stage was developed using SCRUM approach, which consists of five steps, for greater detail about each step refer to *Chapter 4: Methodology.*

### 5.3.1. Stage 1: Front-end

This stage focuses on the front-end of the application, that being the User Interface (UI) and User Experience (UX) of the website application.

#### Sprint Backlog

These are the functional and non-functional requirements that were split in Chapter 4 for the Front-end Stage.

Table - Requirements for Front-end

|  |
| --- |
| **Stage 1: Front-end Requirements** |

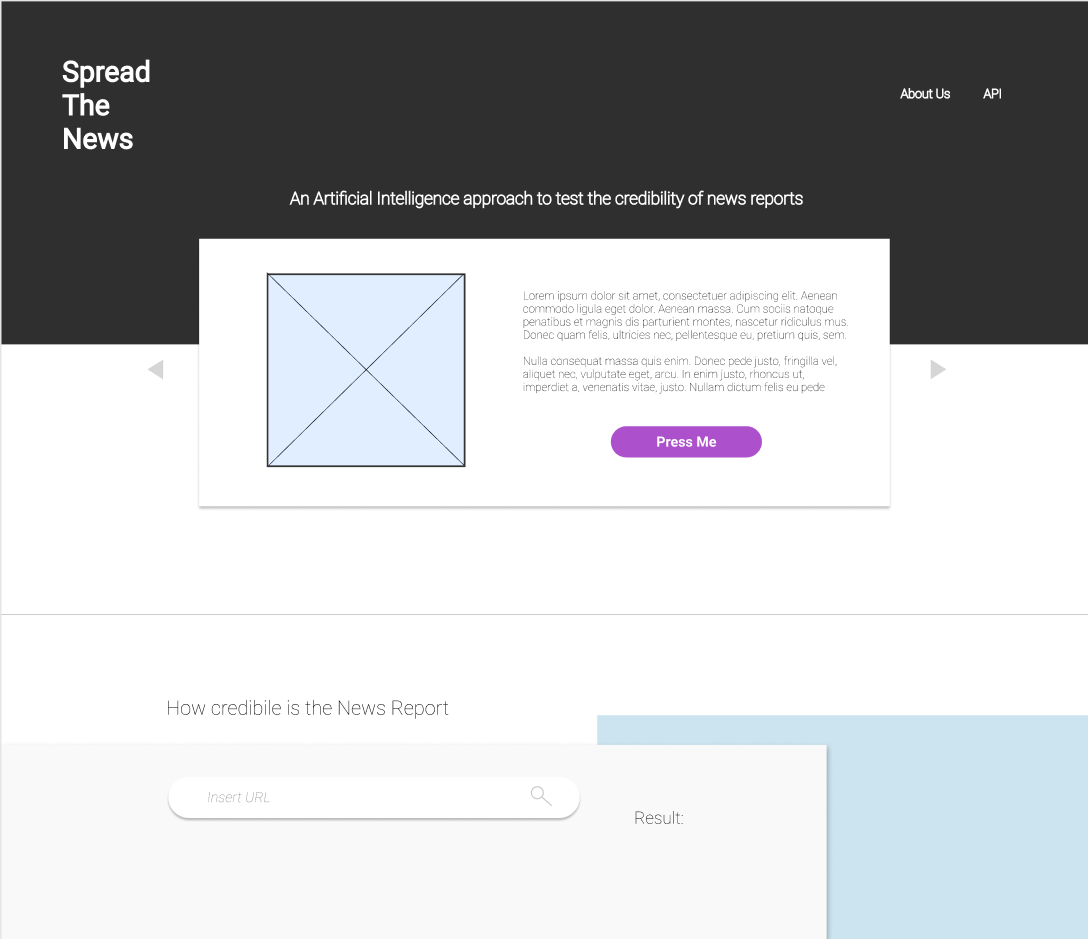
|  |  |  |
| --- | --- | --- |
| ID | Priority | Requirement Description |
| FR-07 | **Should** | There system will announce to the user when the intelligent system has come to a decision with the given URLs |
| FR-11 | **Should** | The website application should be able for other users to then extend upon the current project easily |
| NFR-01 | **Must** | The website will have a responsive web design for desktop users |
| NFR-02 | **Must** | The websites functionality shall fully work on Google Chrome browser engine. |
| NFR-03 | **Must** | The project should attempt to follow the most up-to-date technologies and practices. |
| NFR-04 | **Should** | The website should work on the most popular browser engines |
| NFR-05 | **Should** | The website should follow recent styling trend |
| NFR-06 | **Should** | The website should process the data for the Machine Learning dynamically |
| NFR-08 | **Should** | Operations should be carried out in a time frame that does not cause the user to believe that the system has become unresponsive |

#### Planning & Design

The sub-section covers the planning that was undertaken to be able to develop the User Interface of the website application

##### Wireframe

Wireframe is a sketch/mock-up of the website User Interface. This planning technique allowed for many of the requirements to be thought of and to be able to have an idea as to how to solve them each. After several sketches, the image below shows the final sketch.

Figure - Website Wireframe

#### Implementation

##### Development Tools

The tools that were used to be able to develop the front-end stage of the application includes:

###### HTML5, CSS and JavaScript

The tools of HTML5, CSS and JavaScript are traditional tools for Web Development. Django allows the usage of this tools along with the framework. HTML5 was used to generate most of the static content within the webpage. CSS was used to style the HTML structure and content. JavaScript was used to create dynamic functionality for the website, such as us Asynchronous JavaScript and XML (AJAX) technique to be able to process data in the background and allowing the user to freely scroll around the page, overall delivering a more satisfying user Experience, covered more in the back-end stage.

###### Django Front-End Features

Django Features such as:

* + Django Template Language (DTL)
  + Template Inheritance
  + Template Snippets

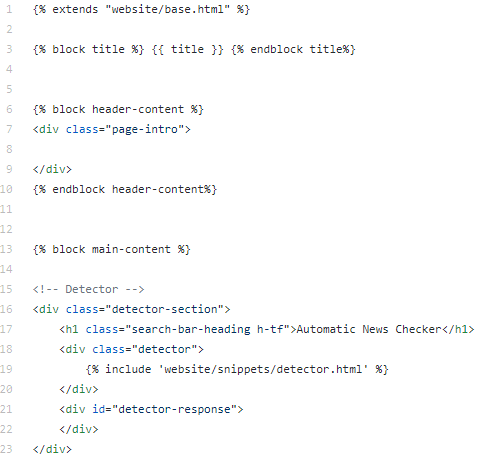
The tools provided by Django promotes highly loose coupling and Don’t Repeat Yourself (DRY) design. Which then allows the program to be scalable, maintainable, and more readable for further development.

The next subsection goes further in depth as to how these Django features works and how they were used within the program. Keep in mind that for now the back-end needs to be considered as a black-box, therefore, must be assumed that the information being exchanged between front-end and back-end are correct.

##### Django Template Language

Once the data has successfully been sent to the Template from the View. It is then possible to access that data with the use of Django’s Template Language (DTL), which is triggered by using different combinations of curly braces and symbols, such as “ {{ }} ”, “ {% %} ”, and more. Each of these symbols have different representations. Will be explaining key ones as they appear in the examples.

For instance, a common piece of data that is fed to the templates is title variable and this goes within the “title” HTML tags. What DTL does is complete that sent data in the tags, this is seen in Figure 11, line 43.

Figure - Part of Home HTML File

In Figure 9, line 3, this line contains:

* A **Django Variable**, which is surrounded by “ {{ }} ”
* A pair of **Django Block Tags,** which is surrounded by “ {% %} ”.
  + concept that will be explained in the upcoming Django features sections.

There are two important things to take from this line:

1. The title data is being called within the Django Variable
2. The entirety of line 3 is using DTL. This is language is what helps communication between Template with the View. This emphasizes the use of the MVT architectural design

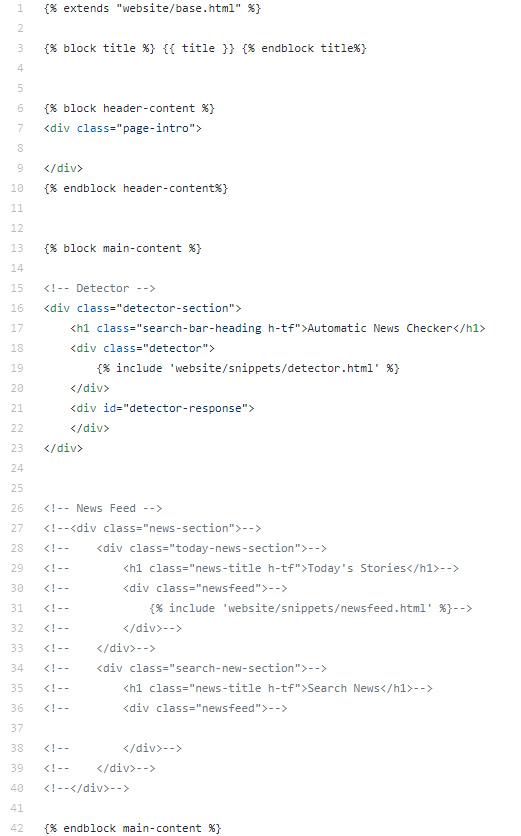
##### Template Inheritance

This is one of the most powerful features about Django and why it is very appealing for many people to use Django. This feature allows massive website to be easier to program and maintain. The reader may have observed that when going through webpages, there always seems to be a trend in design, layout, and style. Template inheritance is what helps accomplish this by using the similar concept of inheritance in Object-Oriented programming.

That being said, there is a template that is typically names **base.html** that will act as the fundamental all or a group of templates, below is the base template for this project:

Figure - Base Template (base.html)

From this file a series of other templates with different contents can be generated easier. This is achieved through the use of Django Block Tags. This works by having some tags as shown in Figure 12, line 24 which states that any child template/inheriting template when using this tag will put its content within this area. For instance, the following HTML Template is the one for home page and look careful at lines 13 and 42. As seen this is the same blocks used in figure 12. Basically, the content of the base template is overwritten by what is in home page HTML.

Figure - Example Block tag

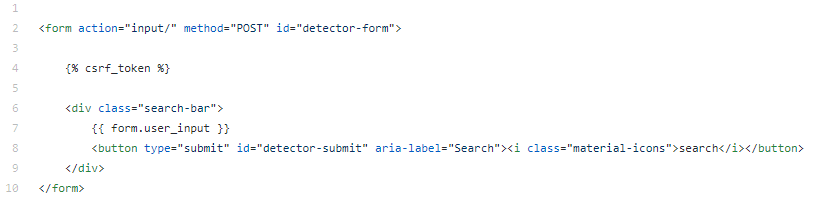
Additionally, this Django feature also helps with the implementation of styling with CSS. This is because all the pages have been strategically designed so that the HTML structure is similar. This allows to re-use most of the CSS selectors for different pages, only in cases when there is unique HTML does the CSS have to be different and more specific. But when it comes for instance with responsive web design, which has been achieved using the combination of Grid Layout and Flexbox Layout, then the CSS is the same for all of them. Overall, once again and maximising reusability, maintenance, and organisation.

##### Django Snippets

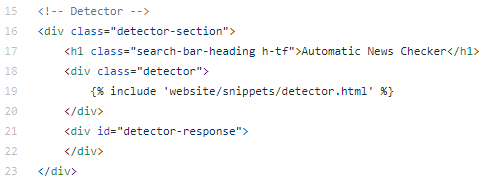
What this feature does is being able to create Template snippets and integrate them anywhere within the designated Template File. For instance, in the project the section that is meant for the user to input the URL and where the Machine Learning responds was turned into a snippet. Meaning that in home page HTML that area is not explicitly there, but more as created a link to the file to extract the HTML from. The two main advantages that this brings is:

* Whenever another developer would want to specifically work on the Machine Learning part of the website then they will know exactly where to find it and how much of the HTML it covers. Instead of perhaps accidentally mixing some other areas of the webpage.
* Another benefit that this feature has is how the part that is snipped can be integrated in any area that the user would desire. Which means for instance if I had also would want to integrate the Machine Learning part into another webpage, instead of having to copy paste the entire HTML, would just had to direct and link the snippet that I would want to be present.

Here is the snippet created for the Machine Learning part of the website. With Figure 14 showing the easiness to integrate the snippet onto the Template.

Figure - Template Machine Learning Snippet

Following figure shows the integration of this component into the Template that is displayed to the user.

Figure - Where Machine Learning Snippet is added on Home Page Template

This concept helps readability, maintenance, and scalability for the website. But most importantly, this concept is to compartmentalise key areas of Templates, in other words makes emphasis on the DRY concept.

#### Testing

This section will cover the testing performed to test the User Interface of the website. The way to test it was by using the requirements defined and categorised for this stage and to determine if the page does what was set to complete.

|  |  |
| --- | --- |
| Test 001 | Detector – Announce about prediction |
| Latest Test Date: | 26 July 2020 |
| Objective | Announce to user when prediction is made |
| Requirement(s) Covered | FR-07 |
| Expected Result: | Expected result would be that there is a pop-up after x amount of time. Depends as to how many URLs are in need to classify. Regardless a message should pop up indicating the detector has reached prediction. |
| Test Result: | **PASS** |
| Evidence: | Message for user: |

|  |  |
| --- | --- |
| Test 002 | Detector – Fails to make prediction |
| Latest Test Date: | 26 July 2020 |
| Objective | Announce to user when detector system has failed to provide a result for the user |
| Requirement(s) Covered | FR-07 |
| Expected Result: | If the detector is not able to make a prediction it should display “The Intelligent System has failed to identify the URL”. |
| Test Result: | **PASS** |
| Evidence: | Evidence provided is through purposely failing the AJAX call: |

|  |  |
| --- | --- |
| Test 003 | Detector – Extend upon current Front-End website |
| Latest Test Date: | 26 July 2020 |
| Objective | Developers should be able to extend upon the existing programs Front-end |
| Requirement(s) Covered | FR-11 |
| Expected Result: | Evidence of all the methods available that the user can/could use for their development. |
| Test Result: | **PASS** |
| Evidence: |  |

|  |  |
| --- | --- |
| Test 004 | Detector – Responsive Web Design |
| Latest Test Date: | 26 July 2020 |
| Objective | The website should be able to respond for desktop purposes |
| Requirement(s) Covered | NFR-01 |
| Expected Result: | Through the changing of screen size, the website should adjust accordingly. |
| Test Result: | **PASS** |
| Evidence: | Full Screen:  *Browser Dimensions: (1920 x 937)*  Half Screen:  *Browser Dimensions: (958 x 927)*  Quarter of a screen:  *Browser Dimensions: (958 x 407)* |

|  |  |
| --- | --- |
| Test 004 | Detector – Extend upon current Front-End website |
| Latest Test Date: | 26 July 2020 |
| Objective | Developers should be able to extend upon the existing programs Front-end |
| Requirement(s) Covered | FR-11 |
| Expected Result: | Evidence of all the methods available that the user can/could use for their development. |
| Test Result: | **PASS** |
| Evidence: |  |

|  |  |  |
| --- | --- | --- |
| ID | Priority | Requirement Description |
| FR-07 | **Should** | There system will announce to the user when the intelligent system has come to a decision with the given URLs |
| FR-11 | **Should** | The website application should be able for other users to then extend upon the current project easily |
| NFR-01 | **Must** | The website will have a responsive web design for desktop users |
| NFR-02 | **Must** | The websites functionality shall fully work on Google Chrome browser engine. |
| NFR-03 | **Must** | The project should attempt to follow the most up-to-date technologies and practices. |
| NFR-04 | **Should** | The website should work on the most popular browser engines |
| NFR-05 | **Should** | The website should follow recent styling trend |
| NFR-06 | **Should** | The website should process the data for the Machine Learning dynamically |
| NFR-08 | **Should** | Operations should be carried out in a time frame that does not cause the user to believe that the system has become unresponsive |

#### Review

This stage covered the extensively as to how the Front-end was developed, as well as the Django concepts that were needed to be able to develop the application. It is believed that the techniques used throughout the entire front-end were ones that

### 5.3.2 Stage 2: Back-End

This stage focus focuses on the back-end of the application. In other words, the functional part of the website and how the data provided by the user is processed.

#### Sprint Backlog

These are the functional and non-functional requirements that were split in Chapter 4 for the Back-end Stage.

Table - Requirements for Back-end

|  |
| --- |
| **Stage 2: Back-end Requirements** |

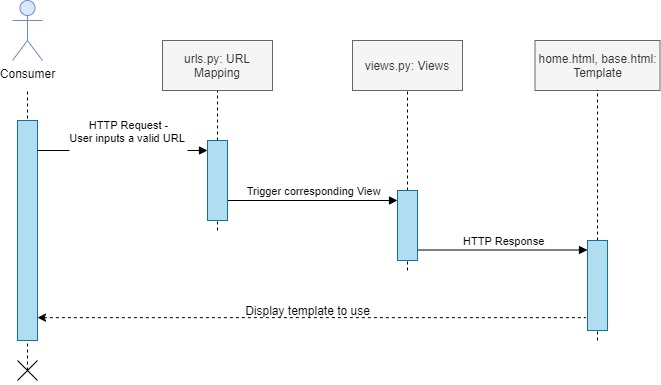
|  |  |  |
| --- | --- | --- |
| ID | Priority | Requirement Description |
| FR-02 | **Must** | The system will only accept English written newspapers |
| FR-03 | **Must** | The only accepted input from the user should be URLs, anything else will be rejected |
| FR-04 | **Must** | The system has to be able to extract the content from the news report |
| FR-05 | **Must** | The system should be able to determine if the given URL comes from a news source |
| FR-08 | **Should** | The system should be able to process more than one URL at a time |
| FR-09 | **Should** | The systems output will provide a reason as to how it came to its output decision |
| FR-10 | **Should** | The website application should be able for other users to then extend upon the current project easily |
| FR-12 | **Should** | The machine learning within the website should be easily exchangeable if needed |
| NFR-03 | **Must** | The project should attempt to follow the most up-to-date technologies and practices. |
| NFR-06 | **Should** | The website should process the data for the Machine Learning dynamically |

#### Planning & Design

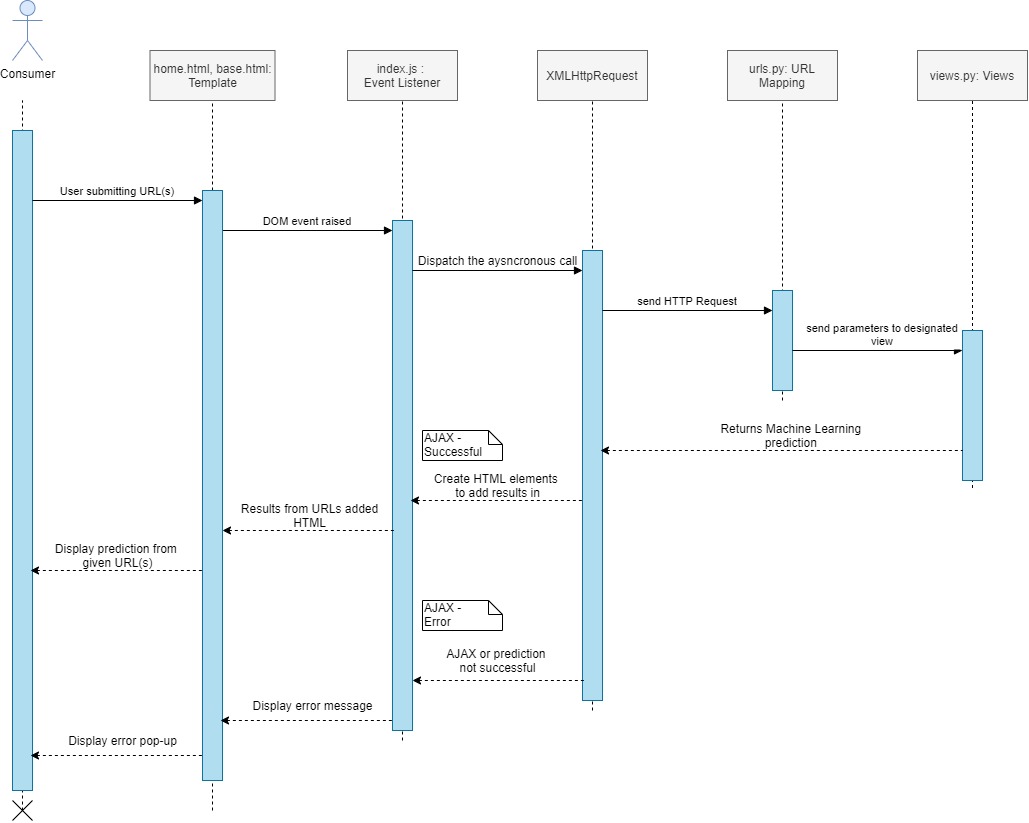
This section provides all the resources that were created in order to design and plan the back-end of the system.

###### Sequence Diagrams

The first sequence diagram is for all the generic/menial operations of the web application. Typically, when the page needs to refresh to be able to then display another part of the website.

Figure - Sequence Diagram: Generic Interaction

The following diagram is the visual representation for the asynchronous process for when the user inputs URL(s) in the detector section and the data has to be sent to the back-end of the application without refreshing it.

Figure - Sequence Diagram: AJAX

#### Implementation

The website application does not have much emphasis when it comes to the Baack-end. Regardless there are some key implementations that are still worth noting about.

##### Development Tools

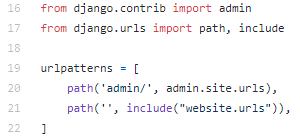
The tools that were used to be able to develop the back-end stage of the application includes:

###### Views

Views are a key component in Django, they are where the logical operations occur. It is also the place that links the Model with the Templates. Therefore, this section will go through as to how many of the operations in the front-end were able to occur. Before going through that it is important to have a better grasp as to who Django workflow is and how Views play a part in it.

Django uses a built-in feature known as **URL Mapping**. When a user enters a URL then Django looks for the URL that matches said input. With each URL there is designated action to it, generally it will link to a View that will then be called.

Figure 1 demonstrates the URL path for the entire project. However, it is a slightly different to what is mentioned above, but overall works similar.

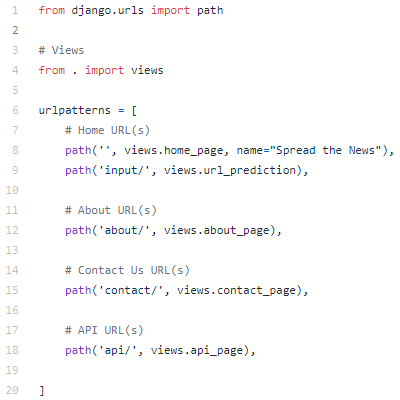
Figure - Project URL Path

Explaining the URL paths for the project:

Table – Understanding Project URL Paths

|  |  |
| --- | --- |
| URL path: Admin | |
| How to access: | Localhost:8000/admin  *(This is if using local machine to run the server with port 8000)* |
| Understanding the path: | This is an inbuilt URL that is provided by Django. Which just contains features for the admin. |
| URL path: *blank* |  |
| How to access: | Localhost:8000  *(This is if using local machine to run the server with port 8000)* |
| Understanding the path: | This blank String URL is for the website application. The keyword “include” essentially extends the URL path to another URL file with more paths. |

In Figure 14, line 21 makes a includes another URL file path. Those URLs are shown below in Figure 15, and these URL paths are the ones that the project uses.

Figure - Website URL Paths

All of these paths have similar principles in functionality. Here is how they work and the purpose for each:

|  |  |
| --- | --- |
| URL path:  *blank* | |
| How to access: | Localhost:8000  *(This is if using local machine to run the server with port 8000)* |
| Understanding the path: | Once again this is intentionally left blank, it is if there was a domain then it would just be the name given instead of localhost. This directs to the homepage view, which later directs to the homepage template. |
| URL path: input |  |
| How to access: | Cannot access this link |
| Understanding the path: | This URL path is for machine learning section in the home page. When the user submits input into the system then JavaScript is used to call the URL path, that then calls the View with the steps to validate the input and if valid then predict what the news report classifies as. |

Table – Understanding Website URL Paths

Once called the View, then it will perform some tasks, some of which will end up in the Front-end. Figure 3, demonstrates the process that the Home page view looks like and its operations

Figure - Example View

As seen in Figure 16, this is the home page view and it is performing a number of statements which give results needed for the Template. At line 39, it is specified as to what is the Template destination. Lines 42-46 the dictionary named “context” will contain the data that is needed to be sent to the template, in this case to the HTML file. Line 48 is when the data is sent to the home page template.

Now that some fundamental Django concepts and workflow have been covered, it will make the following and future concepts much easier to understand.

###### Asynchronous JSON And XML (AJAX) and Views

Asynchronous JSON And XML (AJAX) is a technology in web development that allows a web page to request data asynchronously. The purpose of asynchronous programming in web development is to avoid having to go through a whole web page request to be able to obtain something in return (Ford, J.L.E, 2009). This technology has been one that many high earning companies have invested, such as Google, into their webpages as it results to deliver a more satisfying website experience (Woychowsky, E., 2007).

Although AJAX is regarded more of a front-end technology over a back-end. It is included within this stage as the particular AJAX and View that were used throughout the project is the related with the Machine learning part of the website.

#### Testing

|  |  |
| --- | --- |
| Test 001 | Detector – Only Accepts URL |
| Latest Test Date: | 04 June 2020 |
| Objective | Form must only accept URLs |
| Requirement(s) Covered | FR-11 |
| Expected Result: |  |
| Test Result: | **PASS** |
| Evidence: |  |

This section covers the testing for the requirements stated for Back-end in the Requirements Backlog section of this stage

#### Review

### 5.3.3 Stage 3: Machine Learning

In this stage the focus is the development of the Machine Learning application and explaining as to what route was taken and explaining as to how this path works in detecting fake news.

#### Sprint Backlog

These are the functional and non-functional requirements that were split in Chapter 4 for the Machine Learning Stage.

Table - Requirements for Machine Learning

|  |
| --- |
| **Stage 3: Machine Learning Requirements** |

|  |  |  |
| --- | --- | --- |
| ID | Priority | Requirement Description |
| FR-01 | **Must** | The intelligent system will use Machine Learning techniques only to be able to determine the legitimacy of news reports |
| FR-06 | **Must** | The Machine Learning application after thorough testing should be at least achieving 75% |
| FR-11 | **Should** | The machine learning application should be ready to use for other users to test with other datasets and/or models |
| NFR-03 | **Must** | The project should attempt to follow the most up-to-date technologies and practices. |
| NFR-07 | **Should** | The website should be able to classify in less than a minute per news report |

#### Planning & Design

It has been established that it is desired that the Training and testing of the model will be separate. However, this section will cover the other aspects in which that application will be developed in order function and complete the requirements

##### Activity Diagram

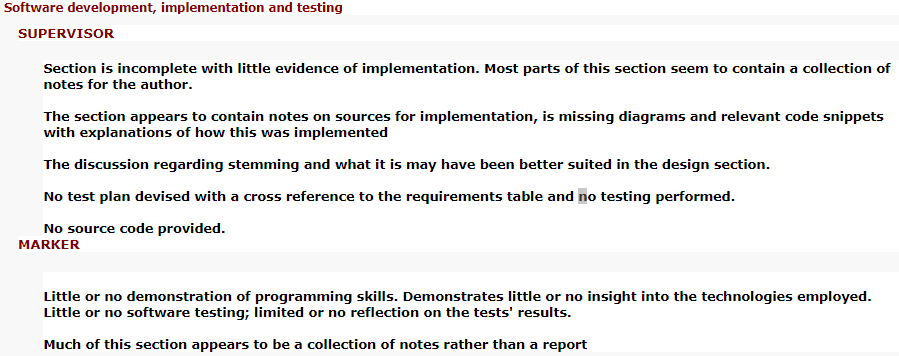
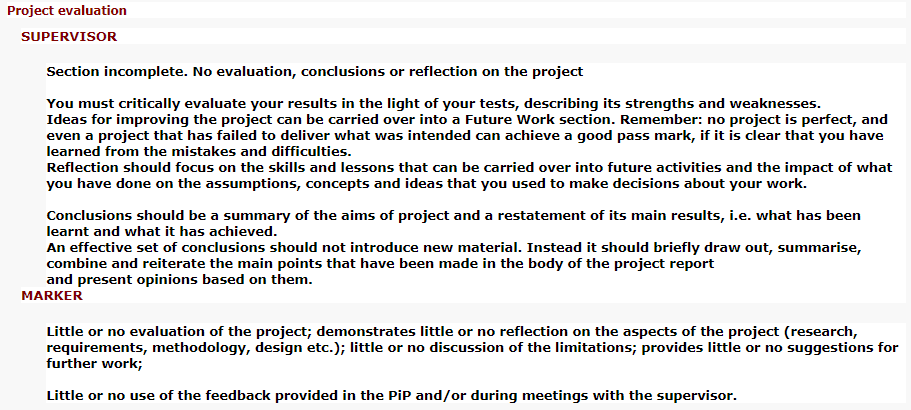
The following diagram will illustrate how objects within the Machine Learning Program interact

#### Implementation

#### Testing

#### Review

𝐴𝑐𝑐𝑢𝑟𝑎𝑐𝑦 = 𝑁𝑢𝑚𝑏𝑒𝑟 𝑜𝑓 𝑐𝑜𝑟𝑟𝑒𝑐𝑡 𝑝𝑟𝑒𝑑𝑖𝑐𝑡𝑖𝑜𝑛𝑠 / 𝑇𝑜𝑡𝑎𝑙 𝑛𝑢𝑚𝑏𝑒𝑟 𝑜𝑓 𝑚𝑎𝑑𝑒 𝑝𝑟𝑒𝑑𝑖𝑐𝑡𝑖�



Chapter 6:

# Project Evaluation

## 6.1. Reflection

It is acknowledged that the task at hand was one of great difficulty. One that has been attempted multiple times, and only until recent year has there been a degree of success by researchers but by no means is there yet a system that can be verified as an absolute source of trust. That being said this caused this final year project to undergo a lot of struggle as this is a dense inter-disciplinary problem to solve, and one that would require a lot more time to be able to develop. Therefore, first reflection would be that perhaps it would have been better to have narrowed down the scope of the project to a particular area.

Second reflection comes along a similar line, in the sense that apart from wanting to solve a difficult problem the project also wanted to be presented in an elegant format. Which meant that a good portion of the project was learning a lot about Web Development, which was a field that there was knowledge to some extent but by no means proficient at it. Once again, an overcomplication that extended the amount of work needed to be done.

Regardless of these two main difficulties, my next reflection comes with having learnt a lot throughout this time. The comfortability and experience gained in both Machine Learning and Web Development has been substantial and would have not thought at the start of the academic year that would have these capabilities.

## 6.2. Improvements & Further Work

As mentioned previously, coming to this project there was to some extent some knowledge in web development and a little bit towards Machine Learning. Which meant that a lot of the things in this project were for the first time. However, while this project some things were learnt throughout the journey which would have been beneficial, but at the time it was either too late and/or too complex, therefore, here is a list of things that could be considerations to improve and further the development of this project:

* During research it was learnt that the news media between countries varies a lot. For instance, in Tanikawa, A., (2017) paper it compares the new reports between countries, an example mentioned is how the United States news contains more factual content while French news was tailored more in providing analytical news. That being said, a further step would be to explore more independently towards each country
* It was learnt late into the development that Artificial Neural Networks (ANN), more specifically that Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN), were an approach that in recent years started to be used for solving this problem. Known as the next advancement in the NLP world. Once this was learnt, it was immediately my intentions to change what has been done up to that point, but once starting to learn about these approaches, it was for sure more complex to implement and time was not in favour. Therefore, a great next step would be to transition to ANN as this field is still naïve in NLP and shows great potential.

Personally, speaking the next approach that I would undertake would be to programmatically change the program to an ANN approach. Followed by using some sort of metaheuristic technique to optimise the ANN, techniques such as Particle Swarm Optimisation or Genetic Algorithms.

* It was learnt in the that there is to some degree some news that are computer generated. There are programs that have been created to use GloVe to generate fake stories based on some keywords given as input. However, there has been methods to counter the computer-generated approach by using the same GloVe unsupervised learning algorithm to determine the likelihood that corpus present was made by a computer. It is believed that this approach can be used in a hybrid system create a more robust application.
* Another problem that was found throughout research is that there have been many attempts for creating these fake news detector systems. However, there is such a scarcity available for the public. This was partly the reason as to why part of the project was developing a webpage. There are formats in which this detector product needs to be available in, such as browser extensions, social media bots, and many more.
* Another potential approach that can be taken, is based upon the structure of the news reports. After reading that not all approaches focus on the actual content of the news report, such as propagation-based and source-based approaches. The perhaps another approach would be to maybe see if there is some correlation between the way real news and fake news are structured. Some initial structural concepts were brought up in *Chapter 2, section 2.1.1 News Media.*
* This next proposition is slightly different. During research, by means was the area of language and linguistics covered. This is an area of great depth and with much research needed. Therefore, another recommendation for further work would be to dive into more detail about linguistics as perhaps there is something being overlooked or something that can be discovered to be a new approach.
* From the relevant reading, it was seen that every period of time a new method is found that helps the overall accuracy of news detection. However, it is becoming more of a struggle to find new and ideas that have success. Therefore, as some other papers suggested, it would be of benefit to create hybrid systems to improve the accuracy. Especially considering that most of the papers encountered were mainly just using a technique.
* Language is a tool that many humans use on a daily basis for multiple purposes. Yet this is a field that has needs a lot of research, and some of the problems that are currently faced with computation linguistic rises due to that. Therefore, it is hard to elaborate a concise plan as to what should be done, since by no means is the author a linguistic expert. However, the message that wants to be delivered is that from what is seen a lot of the problems that emerges comes simply based on how language overall is “incomplete”, in the sense that by it not being from the very start given standards and a structure has caused that nowadays there to be many difficulties that are hard to put into either mathematical, logical and/or organised sense.

## 6.4. Conclusion

# References

Achimugu, P., Selamat, A., Ibrahim, R. and Mahrin, M.N. (2014) A Systematic Literature Review of Software Requirements Prioritization Research. Information and Software Technology [online]. 56 (6), pp. 568-585. [Accessed 22 June 2020].

Arnoff, M.,and Rees-Miller, J (2017) The Handbook of Linguistics. 2nd ed. Unitedd Kingdom: John Wiley & Sons, Ltd.

Beck, K., Beedle, M., Bennekum, A.V., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R.C., Mellor, S., Schwaber, S., Sutherland, K., and Thomas, D. (2001) Manifesto for agile software development. Available from: https://agilemanifesto.org/. [Accessed 15 July 2020]

Beer, D.B. and Matthee, M. (2020) Approaches to Identify Fake News: A Systematic Literature Review. Integrated Science in Digital Age 2020 [online]. 136, pp. 13-22. [Accessed 23 July 2020].

Clayton, K., Blair, J.A., Forster, S., Glance, J., Green, G, Kawata, A., Kovvuri, A., Martin, J., Morgan, J., Sandhu, M., Sang, R., Scholz-Bright, R, Welch, A.T., Woff, A., Zhou, A. and Nyhan, B. (2019) Real Solutions For Fake News? Measuring the Effectiveness of General Warnings and Fact-check Tags in Reducing Belief in False Stories on Social Media. Political Behavior [online]. 1 (23) [Accessed 10 March 2020].

Conroy, N.K., Rubin, V.L. and Chen, Y. (2015) Automatic Deception Detection: Methods For Finding Fake News. Language and Information Technology Research Lab (Lit.rl) [online]. 52 (1), pp. 1-4. [Accessed 13 December 2019].

Daqing, L., Jiali, G., Jichang, Z., Zilong, Z., Levy, O. and Havlin, S. (2018) Repetitive Users Network Emerges From Multiple Rumor Cascades. Arxiv: Physics and Society [online]., pp. 1-22. [Accessed 22 July 2020].

Eisentein, J. (2018) Natural Language Processing. : Mit Press.

Fang, I. (1991) Writing Style Differences in Newspaper, Radio, and Television News. Center For Interdisciplinary Studies in Writing, University of Minnesota [online]. 2 [Accessed 22 July 2020].

Ford, J.L.E (2009) Ajax Programming For the Absolute Beginner. Canada: Cengage Learning.

Haspelmath, M. (2001) Word Classes and Parts of Speech. International Encyclopedia of the Social and Behavioral Sciences [online]., pp. 16538-16545. [Accessed 24 July 2020].

Hudaib, A., Qasem, M.A.H., Masadeh, R.M.T. and Alzaqebah, A.I. (2018) Requirements Prioritisation Techniques Comparison. Modern Applied Science [online]. 12 (2), pp. 62-80. [Accessed 23 June 2020].

Jurafsky, D. and Martin, J.H. (2019) Speech and Language Processing: An Introduction to Speech Recognition, Computational Linguistics and Natural Language Processing. 3rd ed. : Prentice Hall.

Krasner, G.E. and Pope, S.T. (2014) A Description of the Model-view-controller Userinterface Paradigm in the Smalltalk-80 System. Joop - Journal of Object-oriented Programming [online]. [Accessed 18 July 2020].

Kuhn, J. (2009) Decrypting the MoSCoW Analysis. The Workable, Practical Guide to Do It Yourself [online]. 5 (44), pp. 1-2. [Accessed 22 June 2020].

Lanius, D. and Jaster, R. (2018) What Is Fake News?. Versus, Quaderni di studi semiotici [online]. 2 (127), pp. 207-227. [Accessed 22 July 2020].

Lausen, S. (2002) Software Requirements Styles and Techniques. Great Britain: Biddles Ltd.

Leau, Y.B., Loo, W.K., Tham, W.Y. and Tan, S.F. (2012) Software Development Life Cycle Agile Vs Traditional Approaches. 2012 International Conference on Information and Network Technology (Icint 2012) [online]. 37 (1), pp. 162-167. [Accessed 15 July 2020].

Lim, C. (2018) Checking How Fact-checkers Check. Research and Politics [online]. 5 (3), pp. 1-7. [Accessed 22 July 2020].

Marquarft, D. (2019) Linguistic Indicators in the Identification of Fake News. Mediatizations Studies [online]., pp. 95-114. [Accessed 25 July 2020].

Mariani, N., Arief, Y.A. and Muin, F. (2019) An Introduction to Linguistics. 2nd ed. Indonesia: Jurusan Pendidikan Bahasa Dan Seni Fkip Ulm Banjarmasin.

Mishra, A. and Dubey, D. (2013) A Comparative Study of Different Software Development Life Cycle Models in Different Scenarios. International Journal of Advance Research in Computer Science and Management Studies [online]. 1 (5), pp. 64-69. [Accessed 15 July 2020]

Mondal, P. (2018) Lexicon, Meaning Relations, and Semantic Networks. Innl4ai@ Ai\* Ia [online]., pp. 40-52. [Accessed 24 July 2020].

Rashkin, H., Choi, E., Jang, J.Y., Volkova, S. and Choi, Y. (2017) Truth of Varying Shades: Analyzing Language in Fake News and Political Fact-checking. Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing [online]., pp. 2931-2937. [Accessed 23 July 2020].

Reuters Institute (2019) Digital News Report 2019 [online]. Oxford: Reuters institute for the Study of Journalism. Available from: https://ora.ox.ac.uk/objects/uuid:18c8f2eb-f616-481a-9dff-2a479b2801d0 [Accessed: 20 February 2020].

Schwaber, K. and Sutherland, J. (2017) The Definitive Guide to Scrum: The Rules of the Game. The Scrum Guide [online]., pp. 1-19. [Accessed 17 July 2020].

Shu, K., Silva, A., Wang, S, Tang, J. and Liu, H. (2017) Fake News Detection on Social Media: A Data Mining Perspective. Acm Sigkdd Explorations Newsletter [online]. 19 (1), pp. 22-36. [Accessed 22 July 2020].

Tanikawa, A. (2017) What Is News? What Is the Newspaper? the Physical, Functional and Stylistic Transformation of Print Newspapers, 1988-2013. International Journal of Communication 11 [online]., pp. 3519-3540. [Accessed 23 July 2020].

Thompson, G (2014) Introducing Functional Grammar. 3rd ed. Usa, Canada, United Kingdom: Routledge.

Verheyen, G. (2013) Scrum - A Pocket Guide - A Smart Travel Companion. Premedia, Amersfoort – Nl: Van Haren Publishing.

Vestola, M. (2010) A Comparison of Nine Basic Techniques For Requirements Prioritization. Helsinki University of Technology [online]., pp. 1-8. [Accessed 04 June 2020].

Vo, Nguyen and Lee, K. (2018) The Rise of Guardians: Fact-checking Url Recommendation to Combat Fake News. : The 41st International Acm Sigir Conference on Research and Development in Information Retrieval [online]., pp. 275-284. [Accessed 22 July 2020].

Vogel, P.M. and Comrie, B. (2000) Approaches to the Typology of Word Classes. Berlin, Germany: By Walter De Gruyter & Co.

Mann, W.C and Thompson, S.A. (1988) Rhetorical Structure Theory: Toward a Functional Theory of Text Organization. Text [online]. 8 (3), pp. 243-281. [Accessed 27 July 2020].

Woychowsky, E. (2007) Ajax - Creating Web Pages with Asynchronous Javascript and Xml. : Pearson Education, Inc.

Zhou, X. and Zafarani, R. (2020) A Survey of Fake News: Fundamental Theories, Detection Methods, and Opportunities. Acm Computing Surveys (Csur) [online]. 15 (2), pp. 1-37. [Accessed 22 July 2020].

# Appendices

## Appendix A: Project Tools