

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:

# Table of Contents

[Table of Contents 2](#_Toc42245645)

[Table of Figures 4](#_Toc42245646)

[Table of Tables 5](#_Toc42245647)

[Introduction 7](#_Toc42245648)

[1.1. Define Problem 7](#_Toc42245649)

[1.2. Aim and Objectives 9](#_Toc42245650)

[1.3. Chapter Breakdown 10](#_Toc42245651)

[Literature Review 11](#_Toc42245652)

[2.1. How sentences structured 11](#_Toc42245653)

[2.2. How is the meaning captured from phrases? 11](#_Toc42245654)

[2.3. 11](#_Toc42245655)

[Requirements 12](#_Toc42245656)

[Methodology 16](#_Toc42245657)

[Design 17](#_Toc42245658)

[Implementation and Testing 18](#_Toc42245659)

[Project Evaluation 21](#_Toc42245660)

[References 22](#_Toc42245661)

[Appendices 23](#_Toc42245662)

# Table of Figures

[Figure 1 - Project URL Path 20](#_Toc43985879)

[Figure 2 - Website URL Path 21](#_Toc43985880)

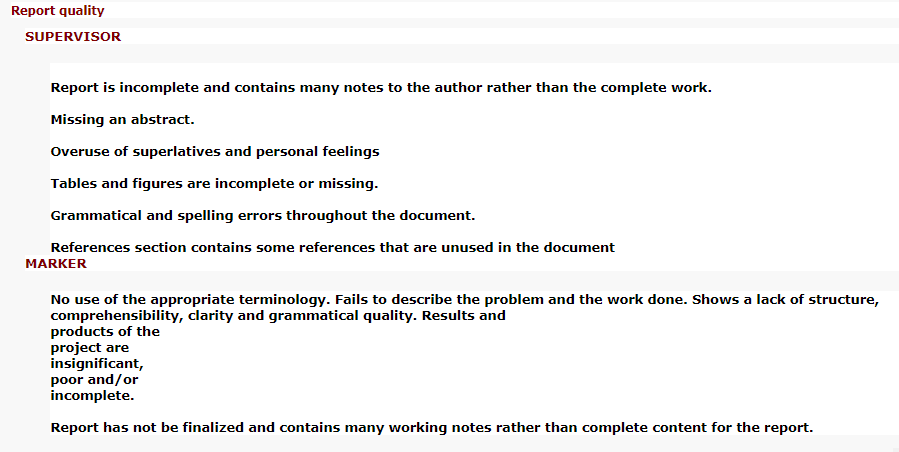
[Figure 3 - Example View 22](#_Toc43985881)

[Figure 4 - Part of Home HTML File 23](#_Toc43985882)

# Table of Tables

[Table 1 - Functional Requirements 13](#_Toc43985884)

[Table 2 - Non-Functional Requirements 14](#_Toc43985885)



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. An eye-catching fact in the Reuters Institute annual report (2019) was how continuously over time the trust level dropping, worst case of 2019 was the United Kingdom with 70% of the people questioning the content from news media. The trust level worsens when it is about politics or when a country is going through eventful circumstances (Reuters Institute, 2019).

It is a fact that fake news is a pernicious problem, it just becomes difficult to determine what degree and how much of an impact it has and will have in the future. A recent event which saw some of the consequence of fake news was the elections of United States of America in 2016. Postelection there were findings of fake propaganda and the usage of fake news to fool Americans, here are some of the reported cases:

* 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.
* 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.

One last point, it terribly sad to see how much money twists people. Really makes some individuals do some outrageous actions. This is exactly what has happened with news companies. What once was their aim to produce honest news is practically non-existent today. Even if there are honest and good journalist, there is always the motive to hesitate whether what you are reading is true. Users should not be worried about what the news publishes as its meant to be re-writing an event. The fact that 1 in 2 people do not trust news is no acceptable and just proves how society crumbles (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 us up to others to catch their lies and make them public. Which will consequently make improve their standards.

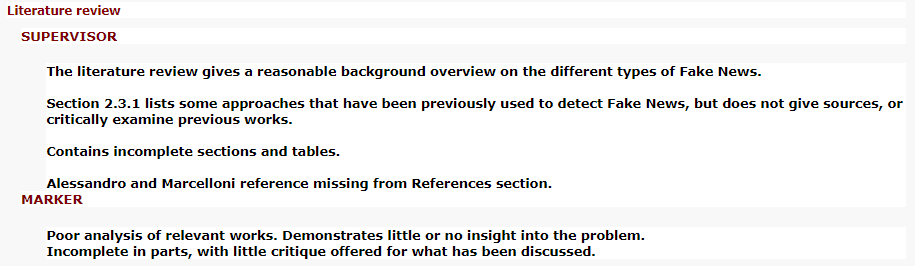
## Aim and Objectives

## 

Aim

Objectives

## Chapter Breakdown



Chapter 2:

# Literature Review

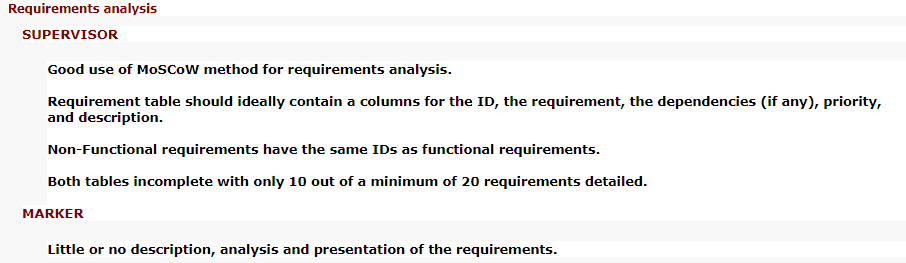
## 2.1.

## How sentences structured

## 2.2. How is the meaning captured from phrases?

## 2.3. Relevant Research

As of now there is no definite answer as to how fake news can be detected



Chapter 3:

# Requirements

## 3.1. MoSCoW Methodology

With this project there is the uncertainty as to what can be completed for the delivery date. Therefore, it is important to create a list which hierarchizes the requirements based upon how important they are for the purpose of the application. This sort of list is known as a requirement prioritization.

|  |  |
| --- | --- |
| **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.* |

A well-known requirement prioritization technique known as **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:

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) |

That being established, the way the project will mainly differentiate each type of requirement is based upon the aim. In this case the projects target is “*to create an intelligent tool that can autonomously be capable from fed text determine whether a given news report is real or fake”.* Whichmeans that functional requirements will around the machine learning software and the detector section within the website, whilst the non-functional requirements will be on the website application (which does not include the detector).

## 3.2. Functional Requirements

Table - Functional Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Dependencies | Priority | Requirement Description |
| FR-01 |  | **Must** | The system must be able through carefully chosen algorithm(s)/model(s) classify a news report as being truthful or deceptive |
| FR-02 |  | **Must** | The system must only accept English written news |
| FR-03 |  | **Must** | The system must only accept URL format as input |
| FR-04 |  | **Must** | The system must be able to retrieve the news content from given URL |
| FR-05 |  | **Must** | The system must identify whether provided URL is an actual news report or not |
| FR-06 |  | **Must** | The Machine Learning application must have a success rate of 75% throughout its thorough testing phase |
| FR-07 |  | **Should** | The system should indicate to the user when the data is being processed and when it is completed |
| FR-08 |  | **Should** | The system should be able to process more than one URL at a time |
| FR-09 |  | **Should** | The website should be able to classify in less than a minute per news report |
| FR-10 |  | **Should** | The systems output should give to some extent some reasoning for its decision |
| FR-11 |  | **Could** | The system should output different content based upon the user privilege on the website |
| FR-12 |  | **Won’t** | The system won’t explore other areas for fake news detection outside the Machine Learning spectrum |

## 3.3. Non-Functional Requirements

Table - Non-Functional Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Dependencies | Priority | Requirement Description |
| FR-01 |  | **Must** | The website must have a responsive design. |
| FR-02 |  | **Must** | The website must fully work on Google Chrome |
| FR-03 |  | **Must** | The project must follow latest industrial practices/techniques with the software being used |
| FR-04 |  | **Should** | The website should fully work on the most popular browser engines |
| FR-05 |  | **Should** | The website should have the capability for other users to be able to extend upon the existing application |
| FR-06 |  | **Should** | The machine learning application should be ready to use for other users to test with other datasets and/or models |
| FR-07 |  | **Should** | The programming should adopt PEP8 format |
| FR-08 |  | **Should** | The machine learning within the website should adopt easy way method to switch between other attempts |
| FR-09 |  | **Could** | The website could have implemented security measures for potential malicious input in |
| FR-10 |  | **Could** | The project could be fully documented |

Chapter 4:

# Methodology

## 4.1. Method

SCRUM is the chosen methodology to help control and organise the development of the 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 (<https://itq.ch/pdf/SCRUM_methodology.pdf>). This is a particular point as to why SCRUM was chosen because it is likely that there would be at some point a delay as there are many concepts being learnt for the first time and other work being done simultaneously.

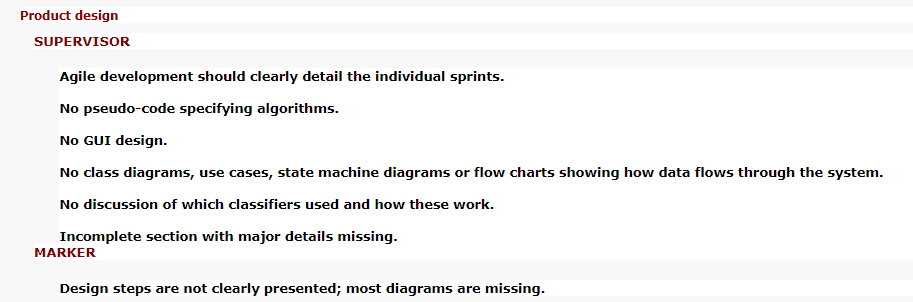
As Verheyen, G. (2013) points out, SCRUM is not a strict process that has obligatory and exhaustive steps. Instead there are proposed techniques that is left for the team to decide whether or not to opt. This is brought up as this project will be using SCRUM as its basis, however, there has been some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

It has been decided that the total project sprints to be divided into three stages: Front-end, Back-end and Machine Learning. All of these contain parts of the requirements that are of great importance (essentially, 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.

Chapter 5:

# Design



## 5.1. Software Development Tools

*Talks about the languages and libraries that are used to develop the program and also need to include photoshop, whatever u use for diagrams*

## 5.2. Project Design

*How the project was structured to solve, basically how the django project is separate to the machine learning*

*Mention the split of the two projects and show diagrams for each. Like separately, maybe one big one to show the global interconnection*

## 5.3. Diagrams

#### 5.3.1. Application Architecture

*About MVC and how the classifier is put into it*

## 5.4. Front-End Design

#### 5.4.1 Website Wireframe

*Put here the*

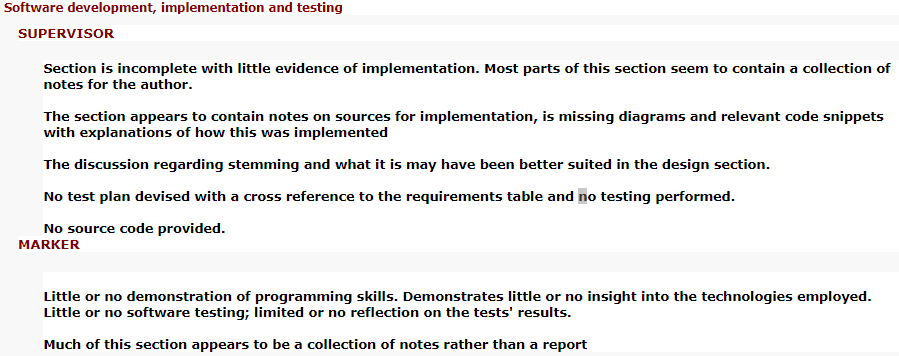
#### 5.4.2 Logo Sketches

## 5.5 Database

## 5.6. Machine Learning

Chapter 6:

# Implementation and Testing



This section is the collective effort of all the prior chapters into developing the application. As previously mention in Chapter 4 and 5 , the approach taken to develop the application was using the agile approach known as a SCRUM. Additionally, the sprints developed can be categorized into stages. Each stage comprises of several sprints and each of them consist on mainly developing an aspect of the project.

## 6.1. Stage 1: Front-End

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

Therefore, the first stage had the following objectives:

* A

The User Interface was built using the following tools:

* HTML5
* CSS
* JavaScript
* Django Front-End Features:
  + Django Template Language (DTL)
  + Template Inheritance
  + Template Snippets

Reasoning behind the use of these tools is mainly due to Django. As seen from the above list, some features of Django were used. What these do is in essence make the code faster to develop, maintainable and scalable.

#### 6.1.1. Django Flow Work

Before going through the used tools, it is important to have grasp how Django runs. 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.

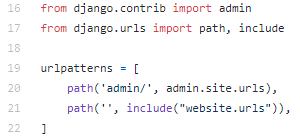
Figure 1 demonstrates the URL path for the entire project. However, it is a bit different to what is mentioned above, but overall works similar.

Figure - Project URL Path

Explaining the URL paths for the project:

|  |  |
| --- | --- |
| ***“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. |
| ***“ ”*** |  |
| 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. |

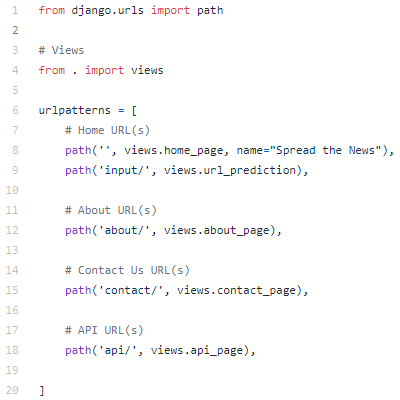
Figure 2 is the URL file that is called in Figure 1, line 21. This contains URL paths that are linked with a specific View.

Figure - Website URL Path

All of these paths have the same principle in functionality. Therefore, giving instead a generic example to explain how they work.

Example:

|  |  |
| --- | --- |
| ***“ ” (Homepage)*** | |
| How to access: | Localhost:8000  *(This is if using local machine to run the server with port 8000)* |
| Understanding the path: | When the above link is accessed then the View named “views.home\_page” is called. |

Once called the View, then the view will have certain statements to execute. Some of which will consist of data that will be sent to the User Interface. Figure 3, is an example View.



Figure - Example View

As seen in Figure 3, line 24 this is the corresponding view from the previous example. Within this view there is a number amount of processes that need to be performed, this is what many views do. In Figure 3, line 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.

#### 6.1.2 Django Template Language

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

Once the data has successfully been sent to the HTML file, then it is possible to access that data with the use of Django’s language, which is triggered by using different combinations of curly braces followed by another symbol, such as “ {{ }} ”, “ {% %} ”, and more.

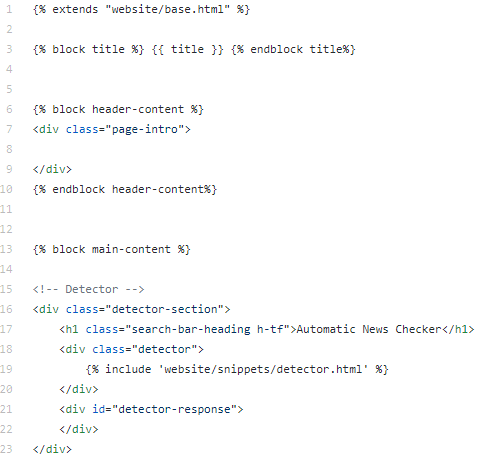
For instance, a common piece of data that is sent to the templates is to feed the <title></title> HTML tags. This piece of data can be seen being prepared to be sent in Figure 3, line 43.

Figure - Part of Home HTML File

In Figure 4, line 3, this line contains:

* A **Django Variable**, which is surrounded by “ {{ }} ”
* A pair of **Django Block Tags,** which is surrounded by “ {% %} ”. A concept that will be explained in the upcoming 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 and being able to use other iterators, conditional statements to condense and select data being displayed.

#### 6.1.3 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 projects to be easier to program and easier to maintain.

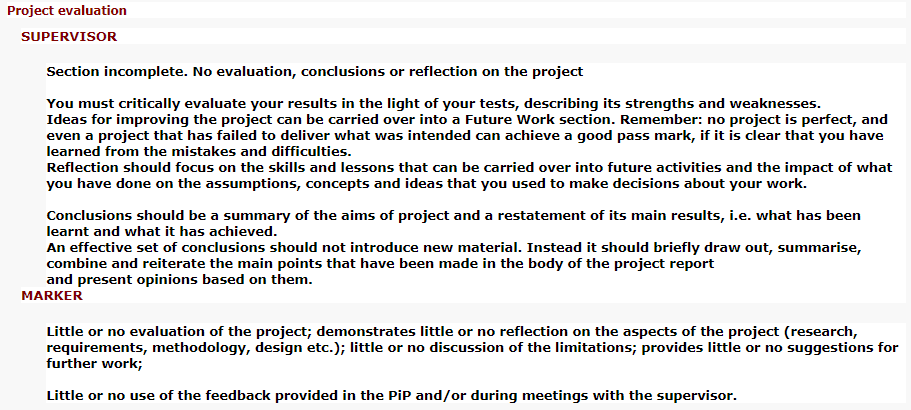
The way Django processes a request from the user, is partially using URL Mapping. Which means that when a URL is loaded then the view for that is called and the view is ion charge of executing designated statements. Some of which states will be on data that will need to be sent to the Template, which in this case it is directed to the HTML file. To have access to this data is when you use Django Template Language.

## Stage 2: Back-End

## Stage 3: Machine Learning

Chapter 7:

# Project Evaluation



# 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].

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].

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].

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

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].

# Appendices

## Appendix A: Tests

|  |  |
| --- | --- |
| Test 001 | Detector – Only Accepts URL |
| Date Tested | 04 June 2020 |
| Objective | Form must only accept URLs |
| Requirements Covered |  |
| Description | It is of importance that what the detector receives as input only URLs. A positive result will allow the detector to continue with the rest of operations |
| Expected Result: | * If not entered a URL, an error message will be shown to the user that the data inserted is not valid * When entered with a URL then the text-field is whipped as the data has been AJAX POST to the back-end to process. However, for testing purposes in the success key-value pair within AJAX in JavaScript has an alert to be triggered if the AJAX procedure is successful |
| Test Result: | **PASS** |
| Evidence: | * When inserted a String      * When inserted a URL |