UNIVERSITY OF TWENTE.

Creative Technology

Bachelor Thesis

Developing a News Aggregation and Validation System

Author:
Josef Moucachen

Supervisors:
Dr. Angelika Mader
Dr. Ir. Djoerd Hiemstra

July 6, 2017

Acknowledgements

Herewith I would like to thank everyone who has supported me throughout my graduation project. I want to thank both my supervisors for providing me with the opportunity to work on this project. To that end I want to thank Lorenzo Gatti for supporting me heavily and providing me with great explanations.

Abstract

The goal of the bachelor thesis is to research and develop a news aggregation and validation system. In order to do so research on related work and literature is conducted, allowing to create a base for the following points of discussion. Here the focus is on researching properties of the user interface and methods of news verification. After the researched elements are used in order to specify and conceptualise the platform and its design. Moreover it is concluded which method regarding news verification is used. Next the realisation of and implementation of the news aggregator and news verifier are discussed. Here each step is discussed individually that is necessary to realise and complete the project. Finally an evaluation takes place which incorporates the creation three scenarios, a survey, and usability testing. The results shows that the system in general is perceived positively. However the feedback shows as well that a more detailed focused approach can enhance the total experience.

Contents

List of Figures

Introduction

The arrival of the web and the social web brings with it a tremendous amount of news sources. The accessibility of these news sources generates a large wave of information which can often times be contradicting and confusing. Facebook, for example, can be seen as a social platform that allows individuals and groups of individuals to freely exchange thoughts and opinions. When this information travels the social web, it is difficult to distinguish between valid and unsupported news. Looking at recent event such as the U.S. election 2016 and the refugee situation in Europe, fake news have played a big role. Fake news can be used for various reasons: gaining political influence, financial, religious are among those reasons. The social web, e.g. Facebook, Google and Twitter, is one of the reasons for fake news to be spread easily and reach millions of people. However those platforms firstly chose to play the issue down or even deny its existence. Now due to pressure from experts and public those companies have decided to implement functions and methods to create awareness, however those are mostly passive or do not provide a big enough range, meaning they can't cover all news.

This issue leads to the challenge of creating a news aggregation platform that allows users to distinguish between valid news and false news. Therefore various aspects need to be researched and investigated.

1. Firstly it is crucial to research methods of validating news. Therefore it is important to investigate what methods exist and which one provides the most reliable outcome.

2. Secondly the aggregation aspect needs to be researched. The focus here will be on how to gather content from several websites and how to display them on one platform.

Background Information

The topic of declining news views in newspaper and TV has developed over the last decades. Instead of using conventional options young people tend to use the social web as a method of news [1]. Six out of ten American people choose social media over conventional methods [1]. Gottfried and Shearer observed [1], that 66% of Facebook of the questioned 4654 Americans retrieve their news from the website, 59% of the Twitter users choose to gather their news on

Twitter and 70% of Reddit users retrieve their news from that platform. Here it is important to point out that Facebook's 67% corresponds to 44% of the general population, meaning that 67% of the U.S. populations uses Facebook and 44% gather their news on Facebook [1]. Due to that trend news available on social media have increased. However not only established news publisher user this social media as mean to spread news but individuals as well [2]. Lee and Ma [2] state that those individuals seek different aspects by spreading news on social media. Firstly those individuals seek 'status attainment' [2, p.331], meaning that they are looking for attention. Secondly another driving factor for individuals to spread news is 'information seeking' [2, p.331].

The goal of this literature review is to find out what medium people use in order to retrieve their news. In order to find a conclusion it will be investigated whether a decrease in conventional news publication is observable. In addition to that it will be researched whether social media are used as a substitute for conventional methods.

Literature Review

Decrease in Conventional News publication methods

As stated by Brown, Jones, Patterson and Casero-Ripolles [3–6] the usage of printed and broadcasted news has decreased among young people. Mindich [7] points out that 80% of America's population under 30 do not retrieve news form newspaper on a daily basis, where 70% of American people above 30 do not as well. Furthermore Mindich [7] supports his statement by claiming that the median age of American people gather information from broadcasted sources is 60, hence hinting at a low value of young people watching news shows. Young people have various reasons that lead to such a decline: lack of time, preferences in different media or content that does not meet the people's interests [6]. In addition to that Casero-RipollÃl's [6] claims the the lack of relevance can be traced back to the missing connection to experiences and interest. In 2015 the Pew Research Center [8] has conducted a research, observing a decline in cable news compared to rise in the year 2013. Moreover they have observed that newspaper circualtion has declined from 2013 to 2014 [8].

In contrast Barnhurst,Wartella and Raeymaeckers [9, 10] claim that young people are generally interested in news, however the method of conventional spreading does not apply to them. As described by Raeymaeckers [10] young people find news in newspaper and on TV too difficult to understand. In addition to that Barnhurst and Wartella [9] approve Casero-Ripolles [6] statement of the news not being reflective of young people's lives. Nonetheless instead of stating that young people are not interested in political news and such, Reaymaeckers concludes that the news' language does not fit with the youth [10]. They [10] propose that conventional news publisher and producer should focus on helping the youth to understand the background and context of the news better. This conclusion gets supported

by Meijer and Irene [11]. Meijer and Irene [11] state that news producers are required to develop a new of standard of news in order to appeal to young people.

Finally it is possible to say that a decline in newspaper and broadcasted news is observable. Various factors play here an important role, nevertheless the method of translating news for the youth is the biggest issue. Young people do not feel connected to the news any more due to their presentation and language.

Social media as a substitute for conventional news

As stated by Tsagkias, Rijke and Weerkamp [12] a big part of content discussed on social media can be traced back to news. For instance 85% of status updates on Twitter are news related [12]. In addition to that Tsagkias, Rijke and Weerkamp [12] argue that social media such as blog posts and tweets are often linked explicitly or implicitly to articles. Explicitly in this case means that a hyperlink to the original article is provided, where implicitly means that only content of the article is discussed without linking to it [12]. In addition to that as already discussed the Pew Research Center [1] has observed that the majority of social media users use their chosen platform as a mean of gathering news information. Another study by the Pew Research Center [13] observed that news websites and social media are the most common digital methods of gathering news, supporting the idea of social media substituting conventional news.

All in all it is possible to say that social media and news deeply connected. Nowadays social media is not only a method of staying connected to people but also to stay connected to the direct and indirect environment of people. Moreover due to the decline of cable news and news paper social media and news websites have become an important source of news. Thus it is possible to state that social media is a substitute for conventional news.

Conclusion

As discussed throughout the literature review a general decline in newspaper and cable news is observable. Especially the youth has lost interest in those methods of news gathering. The lack of interest is not due to the irrelevance of news to their lives, but as argued by Barnhurst,Wartella and Raeymaeckers [9, 10], due to the language which does not appeal to the youth. Moreover research conducted by the Pew Research Center [8] approves that a decline in newspaper circulation and cable news views is existent. In addition to that social media can be seen as a substitute for the newspaper and cable news. As discussed the majority of tweets on Twitter are related to news. In addition to that research conducted by the Pew Research Center [1] concluded in that the majority of social media users use their chosen platform for news gathering.

Finally it is possible to say that the majority of people chose digital news over newspaper

and cable news. Here news websites and social media are the most chosen methods of gathering news information.

Problem Analysis

Throughout the 2016 presidential election an issue has been observed on various social media platforms and news publishers: Fake News [14]. Fake News are defined as news information that are intentionally false and are produced in order to manipulate and mislead readers [15].

Fake news have their origin in various digital sources. Some websites focus entirely on spreading on publishing fake news, where other websites produce a mixture of false and valid news [15]. In addition to that other websites produce satires that if taken out of context might be interpreted as factual by the reader [15]. Nonetheless websites that publish fake news do tend to exist for only a short period of time, as claimed by Allcott and Gentzkow [15]. Various reasons behind publishing fake news seem to exist. Firstly the creation of fake news, which spreads over social media, such as Facebook and Twitter, generates clicks which leads to profit due to advertisements for the creator [15]. Secondly reasons can be based on ideology, meaning in order to support their own idea, people decide to create fake news and spread them [15].

As claimed by Silverman [16] fake news have been more present on social media than true news during the end of the U.S. elections 2016. In regards to that Mark Zuckerberg, a co-founder of Facebook, has expressed that the issue of fake news does not interfere with the elections [17], nonetheless Zuckerman's statement can be dismissed, according to Berghel [14],due to the factors of group-thinking and herd mentality. In addition to that Marchi [18] reports that teenagers receive their news information from various sources: newspapers, TV, trusted adults and social media. Nonetheless according to Marchi's research [18] social media is the most popular source for teenagers due to reasons such as ease of use, efficiency and the wide range of opinions within the comment section. Regarding social media several platforms are used to obtain news. Those are Facebook, Youtube and blogging websites [18].

Google and Facebook had been ignoring the issue of fake news on their platforms, however have decided to implement functions that allow the users to detect fake news [19]. Google's attempt is to implement a product called 'Fact Check' into it's search system [19]. By doing so labels will appear underneath the results, telling the user whether information within the article are true or not [19]. Nevertheless as stated by Smith [19] it is not possible for Fact Check to check all articles. In addition to that Google News is not supported. By contrast Facebook's attempt is to create awareness for the user by providing them with tips on their news feed [19].

Finally it is possible to state that fake news do play a big role in people's daily life. Throughout various media people get into contact with them, however sufficient methods

of suppressing them do not exist. To that end social media is the most one of the most important news distributor allowing people to not only gather news information but also share opinions. Nevertheless responsible companies such as Google and Facebook first chose to deny the problem, and then provide solutions that are not sufficient enough.

Research Question

In order to solve the issues discussed in several issues need to be solved. Firstly the main goal of this graduation project is to find a solution for discrediting fake news. In order to do so a platform will be developed that aggregates news and personalises them to the user's interests. Hence researching existing methods of aggregating information from various websites and personalizing them according to a user's interests is necessary. Thus the following research question can be posed:

What is necessary in order to develop system that aggregates and validates news and displays them collectively on a platform ?

In order to answer the research question following subquestions are used:

1. *What does related work offer in regard to validation of news?*
2. *What methods of news validation do exist?*
3. *How to aggregate news from different sources?*
4. *What platform is suitable for a news aggregation system?*
5. *What aspects need to be regarded in order to evaluate the final system?*

Definitions

Due to the fact that the terms *fake news* and *real news* can be vague and situation depended, a definition in regard to how those terms will be handled throughout this report will be provided.

Fake News: *News that intentionally spreading false information (i.e. Propaganda).*

Real News: *News published by generally highly trusted publishers.*

Part I

Research

Chapter 1

Ideation Phase

Throughout this chapter various aspects related to the ideation are going to be discussed. In order to do so related work is going to be researched and analysed for their properties, in order to find out whether the proposed system exists. To that end research will on which platforms suit the proposed system best and which design choices lead to a aesthetically pleasing and practical layout. In addition to that the possibilities and methods of news verification are going to be researched. Finally several personas will be composed, in order to get an idea for potential users and their needs.

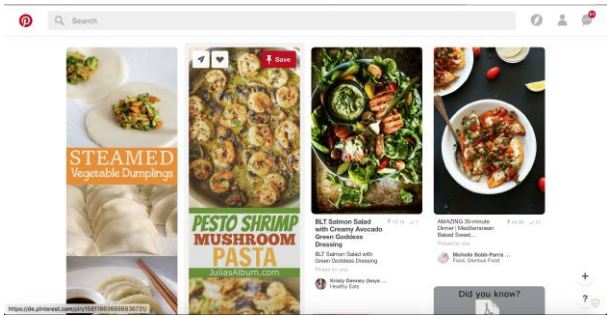
1.1 State Of The Art

1.1.1 Related Work

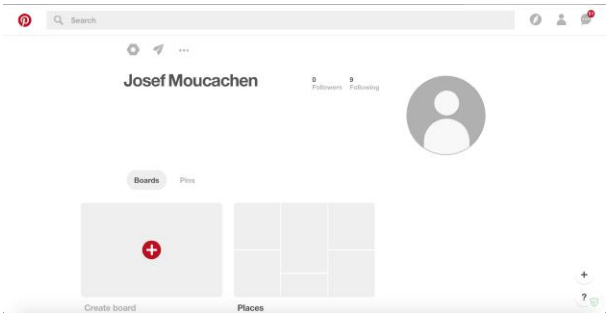
Pinterest

Pinterest.com can be described as a social curation website, which allows users to collect digital images and videos of different content and metaphorically pin them onto their pinboard [20] [21] [22]. Those pins contain a description of the substance produced by the user and a link to the source [20] [21]. The substances can vary extensively, however most common topics are Food and Drink, Home and Garden Decor and Design, and Apparel and Accessories [20]. In addition to that users are able to 'like', 'repin' and comment on the pins. Each action generates concludes in a different action. Liking content resolves in displaying it on the users 'like page'; *repinning* leads to displaying the content on the users own pinboard, and commenting allows users to share their opinion on the content underneath the pin itself. Moreover Pinterest contains further social media characteristics, in terms of users can follow each other, depending on their liking of the content or only certain pinboards [22]. Transparency is also part of Pinterest characteristics. This means information such as usernames, profiles, pinboards and pins and additionally statistics such as repins and comments, are

accessible for all web users, without being a signed up member.



(a) Pinterest Homepage



(b) Pinterest Profile

Figure 1.1 Pinterest Website

Concluding several characteristics can be used as inspiration for a news hub. Firstly a function similar to commenting can be used in order to generate interaction between users based on the article and its content, therefore enabling them to share thoughts and opinions. In addition to that transparency is a desirable feature as well. This is due to the assumption that transparency allows unregistered users to view statistics and content, which could evoke interest in people to join the community.

Google News

Google News is news website, which accumulates articles and reports from more than 4,500 news sources [23]. It is able to group analogical content and display them based on the users interests [23]. The home page of Google News contains several several categories starting with 'Top Stories', then 'World', 'U.S.', etcetera [23]. Those categories are displayed in a vertical menu on the left of the website, which link each category to a page with matching content [23]. Google News provides additional options for signed-in users. Those are firstly the ability to easily access old news stories previously viewed by the user. Secondly the option for recommended news is available for users, which are based on the userś search history [23]. In addition to that Google News uses the hybrid filtering model, meaning recommendations are made based on the userś search history and explicit ratings [24].

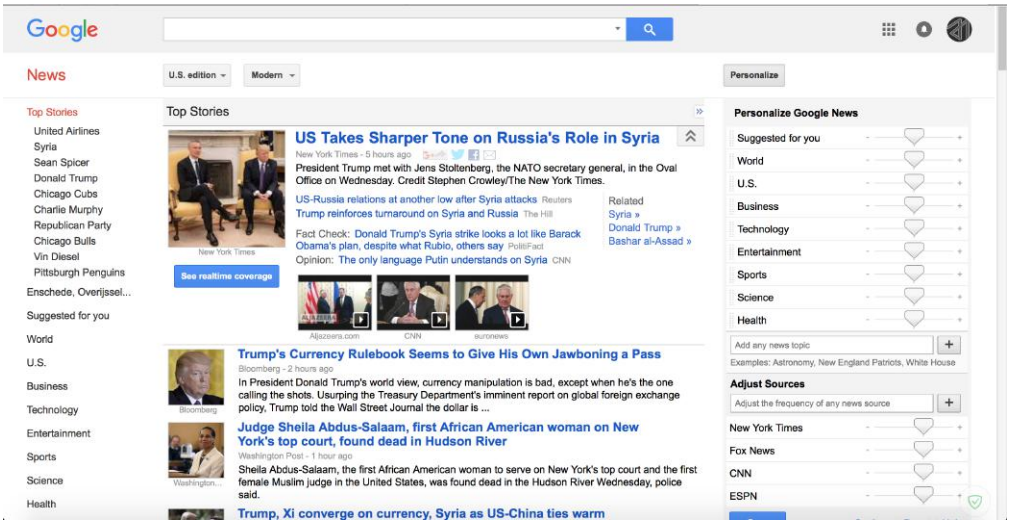
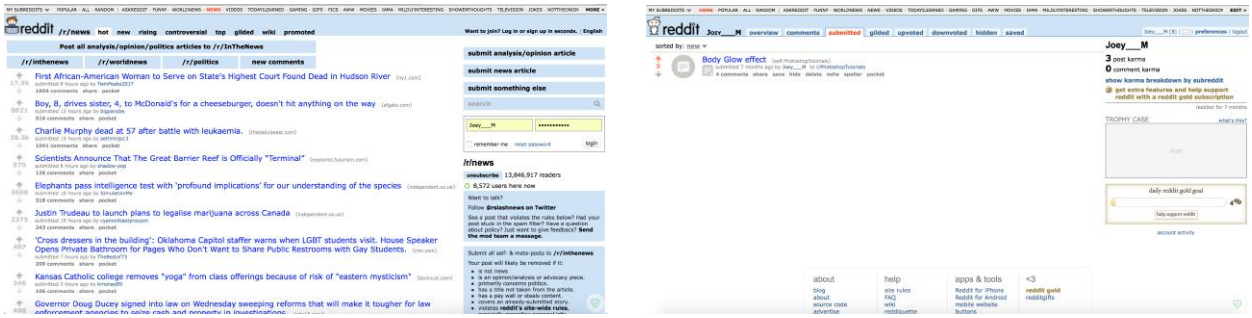


Figure 1.2 Google News Website

Given these points Google News does provide a solution for a recommender system. Instead of using a content based filter system or a collaborate filter system, Google News takes advantage of both system by combining them into a hybrid. An implementation of such a system is considerable due to its advantages and little disadvantages.

Reddit

Reddit.com is social news aggregation website, allowing people to exchange opinions and thoughts [25]. Its framework allows to keep the community and the website structured. In order to do so the website uses *subreddits*, which can be compared to categories. Any signed-in user can create a subreddit and chair, nevertheless Reddit uses administrators who can decide to delete and close subreddits. The creator can decide upon rules [25], such as what content is allowed to be posted. In addition to that users can subscribe and unsubscribe from subreddits [25], allowing users to keep track of subs their are interest in. To that end users are able to post, comment and vote, allowing interactivity with other users. Posting here means to contribute to a subreddit, by submitting content such as links or self-posts. Commenting allows users to comment on those posts, and voting enables users rate a users contribution, by down-voting or up-voting. Additionally users get karma point based on their voting of the posts and comments [25]. The more karma points a user has the more frequently a user is allowed to contribute.



(a) Reddit Homepage (b) Reddit Karma System

Figure 1.3 Reddit Website

All facts considered Reddit does provide a method that reduces exploitation, false information spreading or disrespectful behaviour. By allowing users to up- and down-vote posts and comments they are able to rate a user based on behaviour. Those ratings appear on the user's profile summarised as karma points, which can be used by other users as an indication for the user's behaviours. In an environment a similar system is desirable since it increases the users awareness of the other members of the community.

Facebook

Facebook.com is a social media network firstly launched in 2004, as *thefacebook.com*, which at that time focused on Harvard students only [26]. Later on it expanded to students in whole of the U.S. and in 2007 it continued expanding further to non-academic users worldwide [26].

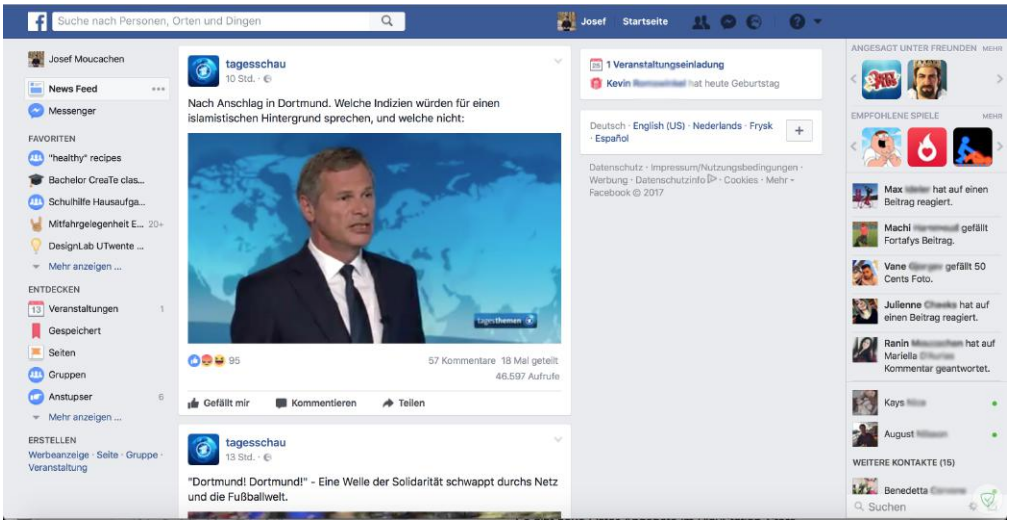


Figure 1.4 Facebook Website

Facebook consists of various API calls, of which each is dedicated to a category [26]. The categories contained within the Facebook environment are as following: *authentication*;

photos; *friends*; *notifications*; *profile*; *users*; *events*; *groups* and *feed* [26]. Each API's call function is cited from [26] and provided within table 1.1.

API Call	Function
Authentication	provides basic authentication checks for Facebook users.
Photos	provides methods to interact with Facebook photos.
Friends	provides methods to query Facebook for various checks on a userâĂŹs friends.
Notifications	provides methods to send messages to users.
Profile	allows you to set FBML (HTML tags) in a user's profile.
Users	provides information about your users (e.g when they are logged in).
Events	provides ways to access Facebook events.
Groups	provides methods to access information for Facebook groups
Feed	provides methods to post to Facebook news feeds.

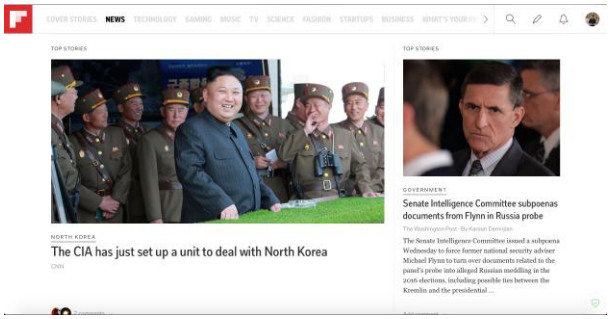
Table 1.1 Facebook API Calls

When creating an account users are asked to provide various personal information (e.g. gender, age, domestics) in order to generate a user profile. When generated the user is able to connect to (or *friend*) other users, which needs to be accepted by the targeted user. Once a connection is established each user is able to see the other user's activities on the respective profile or on their news feed. Those activities include posts; and liked, shared and commented content. Each activities concludes in different outcomes. *Posting* allows users to share their thoughts independently from other content, hence allowing users to create their own content. *Liking* allows users to express their positive attitude towards a content content, resulting in a numerical value underneath the post, showing the amount of liking activities. The *sharing* function enables users to post other people's content on their own profile. This activity concludes then in displaying that content on the news feed of the user's connection. The *commenting* option allows users to comment on content and express their thoughts and opinion, freely.

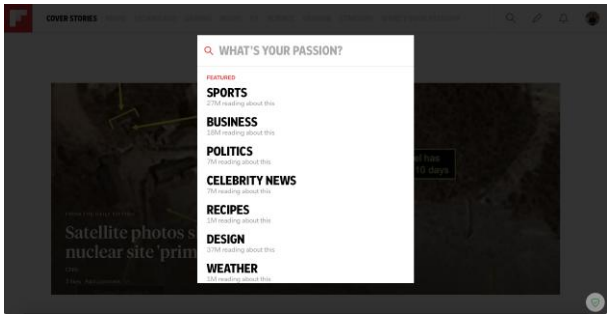
All in all it is possible to claim that Facebook uses functions that considerable for a news hub. As mentioned before in the case of Pinterest functions such as commenting and sharing (repinning in case of Pinterest) are worthwhile. In addition to that Facebook's like-function can be seen as an alternative for Reddit's up- and down-vote system, which as well can be used as an indication for the post's reputation.

Flipboard

Flipboard is a website and smartphone application specialised on personalised news aggregation [27], [28]. It is capable of collecting articles,video and social media and make them accessible in one digital magazine [27], [28]. In order to use Flipboard a registration is needed, which can be done using a Facebook, Google or Twitter account [27], [28]. In order to suggest the user content according to their interests Flipboard suggests the user to chose categories, which when chosen open subcategories [27]. Various news outlets cooperate with Flipboard allowing the system to accumulate videos, articles and other content, however some outlets require to sign up in order to view their content [27]. In addition to that the aggregator is able to suggest related content based on the topics viewed by the reader [27].



(a) Flipboard Homepage



(b) Flipboard Category Menu

Figure 1.5 Flipboard Website

To that end Flipboard enables users to follow their favourable news publisher [27]. In general content can be shared through various media [27]. For instance it is possible to share an article via e-mail or social media platforms [27]. In addition to that information on methods regarding the recommendation mechanism are not published, hence cannot be determined.

All in all it is possible to say that Flipboard provides various interesting functions. Firstly the ability to follow certain news publisher should be considered. In addition to that its method of collecting information on the users preference is an interesting way of collecting preferences of the user, as well.

Discussion

Given these points various characteristics and functions from the related work can used as inspiration. Firstly Pinterest and Facebook share similar features, more specifically both are able to spread content by sharing/repinning it, and both allow the users to indicated their liking of content by interacting with a button. Sharing or repinning allow users to spread

content based on their interest. Moreover Facebook and Pinterest support a connection function, meaning it is possible to connect with other users on those platforms. Looking at those functions individually it is possible to determine a difference. In order to establish a connection on Facebook both parties need to accept the connection, meaning one user needs to actively send a request, which needs to be accepted by the user it is aimed towards. By contrast Pinterest uses a one sided connection function, meaning a user can chose to follow a user, thus get access to the user's content. In case of a news hub the latter connection method does seem like a more significant solution. This is due to the fact that it allows users to connect with each other based on various factors such as common interests, interest in increasing their scope or pure curiosity.

In terms of the liking function both Pinterest and Facebook use similar methods. Both methods allows a user to indicate their liking of a content, however neither supports an expression of disliking. Reddit provides a different solution, by allowing users to up- and down-vote content, hence allowing users to express both liking and disliking of content. Moreover those votes result in karma points indicating a users behaviour. Such a system is preferable in such a system due to the fact that it can give insight into a users behaviour and reliability, hence helping other users to distinguish between reliable and unreliable content.

Regarding Flipboard's functions it is possible to say that requesting the user's interest when creating a profile a useful method of collecting. Doing so allows the user to have a better start and can help to keep up the user's interest.

1.1.2 Creating a new System

Thanks to the analysis and discussion of the related work it is possible to claim that a personalised news aggregator, that discredits fake news does not exist as of now. Hence the research and development will be continued on the chosen topic.

1.2 Properties of the User Interface

1.2.1 Platform

In order to develop the system it is important to research, which platform suits the proposed system the best. On the basis of the *related work* research it is possible to state that mainly to platforms need to be taken into account: websites and mobile apps. This is due to the fact that each discussed system does use the web and apps as their chosen platform. As investigated by Wong [29] both websites and mobile apps have a significantly big user base. Nonetheless a website has various advantages over mobile apps, which are based on pure characteristics of the platform, but also on personal preferences.

Firstly looking at mobile apps a big disadvantage is the fact that they are exclusive to mobile devices. However apps cannot be developed universally, meaning a mobile app needs to be developed for a specific operating system, namely Android or IOS. This factor leads to a limited user base, which is not desirable. A solution therefore is to develop the system for both Android and IOS, however due to workload and time constraints this options will be neglected. In regard to the web as a platform it is possible to state that it is possible to make it accessible on any device with an internet connection, which makes it universally accessible. However it is important to make it web page mobile friendly, in order to make it operate in a practical manner. Due to its accessibility more people can potentially use the system, allowing a bigger user base. In addition to that speaking for a web based platform is the factor that developing a website have already been researched and can be seen as already acquired knowledge.

1.2.2 Design

The user interface design of a platform is crucial in regard to if a user decides to chose a system or not. Thus it is of importance to research what makes a website appealing to the user. In order to make design choices general requirements based on literature and design ideas of related work are going to be discussed.

Requirements

In regard to the layout of the website Boulton [30] suggests the usage of the rule of thirds, as a substitute for golden selection rule. The rule of thirds connotes that a certain length is divided into three parts. This is due to simplicity reasons such as that for fixed width designs (960px) it can be separated in three columns of 320px, where the golden rule would result in unworkable numbers [30]. Nonetheless this rule can only be applied for the horizontal width of the website, not the vertical height, since it is not possible to determine how long the website will be [30]. However if a page scroll is implemented, this issue is resolved. By applying the golden selection or in this case the rule of thirds it is possible to recreate natural patterns, making the user interface more natural for the user [30]. In addition to that the factor of cluttering needs to be taken care of [31]. Cluttering occurs when the load of items on screen hinders the user's performance regarding finding information [31]. Hence in order to keep the user interface uncluttered important information need to be easily detectable [31]. Another rule to keep in mind is to place important items such as the navigation, constantly [31]. Doing so allows the user to learn and remember their location, which increases their performance on the website [31]. Furthermore the layout should be structured in such a way that it allows the user to easily compare content [31]. This helps the user to analyse various factors (e.g. similarities, differences, trends and relationships) [31], enabling the user to make a decision

regarding the importance of content. To that end Shneiderman [31] states that an important factor for the layout is the display density. In order to create an organised and clearly arranged user interface, the information displayed on the screen need to be restricted [31]. By doing so according to a study the users are able to find information easier and more quickly [31]. This is due to the fact that people tend to look for less dense areas more, than looking for dense areas [31]. Moreover participants of the research used less fixation points at the dense areas, where in the less dense areas participants tend to have more fixation points [31].

To that end Tullis, Catani, Chadwick-Dias and Cianchette [32] describe site-level issues, can be described as issues that have to do with the whole website rather than only an individual page [32]. Here the issue of 'depth versus breadth' occurs, which describes the total amount of information versus the total amount and depth of pages [32]. According to Tullis, Catani, Chadwick-Dias and Cianchette a conducted concluded in that if a website is only two pages deep, users are able to find their goal more quickly [32]. Nonetheless during one test an unnatural page layout was used leading to a significant increase of time needed [32]. This supports Boulton's [30] suggestion of using the rule of thirds. In addition to that various other tests were conducted, which concluded in three results [32]:

1. In complex situations breadth is superior to depth
2. In clear and simple situations fewer choices are better
3. Fewer choices in middle levels are better than in top or bottom levels might be better

In addition to that the loading time of a page needs to be discussed. The load time is closely related to the issue of depth versus breadth, due to the fact that based on that decision the amount of information in the page will be determined, leading concluding in a certain loading time for the page [32]. According to a study the loading time of a website plays a role in regard to the user's perception of the website [32]. In general the study concludes that the loading time of a website should be around 10 seconds maximum [32]. Otherwise the user might perceive the website as bad quality, get frustrated and quits using it [32].

Related Work Designs

Looking at related work it is possible to observe various aspects. Here the focus will be on Pinterest and Flipboard due to the fact that those websites represent a news aggregator system the most. When investigating the design of Pinterest (Figure 1.1) it is possible to observe that it uses *cards* in order to keep the page organised. Doing so the website is able to display a large amount of content on a page and keep an organised structure. The card's content is restricted to images, which can vary in height but not in width. This method leads to having a minimalistic representation of content, which cannot be supported by text.

To that end by restricting width and allowing height to vary within a certain range, makes the layout dynamic and keeps its structure organised at the same time. In addition to that it is possible to click on a card, leading to an enlarged view with additional information in form of text and a link to the original website. Only when the user clicks on the link a new tab opens, otherwise the user stays on the same page, allowing a quick interaction with the website. Nonetheless the image-based card approach has its disadvantages, making it unsuitable for a news hub. Due to the usage of only images topic recognition can be demanding and difficult, disadvantaging users by not allowing them to gain a quick overview of the content. Flipboard (Figure 1.5) on the other hand is based on a conventional news website layout. It uses a box layout in order to present content, however the sizes vary in two ways. Top stories of the day take about two thirds of the available width where other stories take one third. The content is represented by using an image, a title, a publisher and the beginning of their article. However the top stories do not provide a the beginning of their article at all. In order to have a clear division between stories Flipboard uses white space and thin lines, helping the user to separate content easily. To that end organised articles according to their date of publication, but does not indicate it using dates.

1.3 Methods of News Verification

News verification can be described as a technology with the aim of identifying fake news, by predicting the probability of being false [33]. Various methods of news verification have been researched and developed. Conroy, Rubin and Chen [33] provide a recent approach, where they distinguish between a linguistic approach and a network approach.

1.3.1 Linguistic Approach & Network Approach

Linguistic Approach

The Linguistic Approach can be described as a method where the content of an item gets extracted and analysed regarding language patterns [33]. As stated by Conroy, Rubin and Chen [33] producers of fake information tend to use language in a strategic manner. Nonetheless the authors [33] describe that sometimes a "leakage" occurs, meaning that a break in the pattern is observable. Thus the following aspects need to be discussed: *Data Representation, Deep Syntax, Rhetorical Structure and Discourse Analysis and Classifiers.*

Data Representation: In order to represent text the "bag of words" approach can be used [33]. Here each aggregated word is regarded as one unit, where the units are equally relevant [33]. Doing so allows to find cues of deception by analysing the frequency of individual words or multiple words (n-grams) [33]. Moreover by the "tagging of words into respective lexical

cues" [33, p.2] such as for instance: part of speech or 'shallow syntax', affective dimensions, or location-based words it is possible to provide frequency sets in order to disclose linguistic cues of deception [33]. Due to simplicity of this method major drawbacks come with it [33]. Since this method is based on individual or multiple words, the contextual background is lost and ambiguity does not get incorporated at all, possibly leading to a false interpretation of content [33].

Deep Syntax: Here the sentence structures (i.e. syntax) are analysed [33], by using Probability Context Free Grammars (PCFG) [33]. A PCFG transforms sentences into rewrite rules concluding in a parse tree in order to define sentence pattern [33]. This method provides, according to Conroy, Rubin and Chen [33], a deception detection with 85-91% accuracy. Nonetheless the authors state that this method alone is not sufficient in order to detect deception [33]. Hence a combination with other techniques is beneficial [33].

Classifiers: As stated by the authors [33] sets of words and category frequencies are supportive in regard to the training of classifiers (i.e. Support Vector Machines or SVM, and Naive Bayesian models). Here pre-coded examples train the mathematical model, allowing it to predict other instances of deception using numeric clustering and distances [33]. Support Vector Machines (SVM) can be described as a machine learning method, that uses two classes with the maximal distance in order to determine a decision boundary [34]. In order to increase the accuracy of SVM different methods of clustering and distance functions can be used [33]. To that end Naive Bayesian algorithms use accumulated proof of correlation between a given variable (e.g syntax) and other variables within the model in order to create classification [33].

Additionally it is possible to say that in order to classify bias the use of "[...] unintended emotional communication, judgment or evaluation of affective state" [33, p.3] by the imposter are assumed. To that end the authors [33] suggest the usage of semantic patterns, in order to differentiate between factual statements and biased ones. Studies conducted by a study of business communication have concluded that compared to random guesses, this method out-performances it by 16% [33]. Moreover it has been observed that deceptive texts hold a lower occurrence rate of "[...] non-extreme positive emotions" [33, p. 3].
Concluding the authors [33] state the Linguistic Approaches provide reliable results, when used in hybrid manner.

Network Approach

As stated by Conroy, Rubin and Chen [33] the Network Approach is complementary to the content based Linguistic Approach. Especially due to the rise of micro-blogging websites (e.g. Twitter), which allow real-time updates, the Network Approach has become more

important [33]. It uses information such as metadata or structured knowledge network queries in order to uncover fake news [33].

Linked Data: The authors [33] believe that knowledge networks are a possibility to achieve scalable computational fact-checking. Here false information can be used in order to be compared to known facts (e.g. collective human knowledge), allowing to asses the truthfulness of content [33]. Such a method relies on making an inquiry about knowledge networks or public structured data such as DBpedia ontology and the Google Relation Extraction Corpus (GREC) [33]. Doing so allows to reduce fact checking to the computation of the simple shortest path [33]. Here queries are used that are build upon factual statements, which then are assigned to syntactic vicinity as a function, which serve as temporary relationship between subject and predicate through other nodes [33]. In this case a statement has a high chance of being true, the closer the nodes are to each other [33].

1.3.2 Additional Linguistic Techniques

Stop Words: As stated by Christopher D. Manning, Prabhakar Raghavan and Hinrich Schuetze [34] Some words do not add value to the context of a text, thus those words do not need to be processed by the system. Those words are described as common terms or *stop words* [34]. Nonetheless it is also stated that in some circumstances the usage of stop words are essential in order to keep the context [34]. An example used here is : 'President of the United States' [34, p. 27], which contains two stop words. When removing those stop words the leftover words do not connote the original meaning, possibly leading to a false interpretation by the system [34]. However due to the usage of the bag of words approach, a loss in meaning does not lead to a affect the outcome negatively.

Stemming and Lemmatization: Articles and other forms of text use different forms of a word (e.g. conjunctions) due to grammatical reasons [34]. Examples therefore are 'am; are; is;' which are different forms of 'be' and 'cars; cars'; car's', which results in car [34]. The process of *Stemming* connotes that the affixes are removed from a word, meaning the ends of words are cut off, potentially resulting in the base form of a word [34]. *Lemmatization* on the other hand uses 'vocabulary and morphological analysis of words' [34, p. 32] in order to reach the goal of finding the base of a word. Thus lemmatization is going to be used.

1.3.3 Additional Classifier Models:

Decision Trees can be described as non-parametric method for classification and regression, which is supervised [35]. Its aim is to learn decision rules based on the provided data, in order to be able to make a prediction regarding other data, based on those rules [35].

1.3.4 Vectorization:

In order to convert textual data into readable information for the classifier vectorization is needed. Regarding text vectorization Scikit Learn provides four different methods: *CountVectorizer, HashingVectorizer, TfidfTransfomer and TfidfVectorizer.*

CountVectorizer converts textual data into token counts [36], meaning it creates a matrix that describes the frequency of occurrences of words. Such a matrix is called a document-term matrix. However a CountVectorizer uses a vocabulary dictionary, which strains the memory of the computer [36].

HashingVectorizer uses feature hashing in order to convert textual data [37]. Doing so provides a significant advantage over the CountVectorizer method, such as low memory usage since it is not necessary to store a vocabulary dictionary in memory [37]. Nonetheless this method does not provide any inverse document frequency (idf) weighting [37].

TfidfTransfomer can be described as a transformer, that transforms a count matrix into tf (term frequency) or tfidf (term-frequency times inverse document-frequency) representation [38]. Tfidf's goal is to provide each token with a weight for its occurrence frequency, meaning that if a word occurs often in a lot of different documents the word gets a low value, indicating that it is not of value [38].

TfidfVectorizer is the combination of the CountVectorizer and the TfidfTransfomer [39].

1.4 Personas

The creation of personas allows to identify potential interest groups and their needs. Hence three personas of fictional characters are going to be created. Their description contains information on their background, their needs, and a short description of their biography. In addition to that the personas will be used during the evaluation phase.

1.4.1 Persona 1

Background

Name: John Tyler

Age: 31

Gender: Male

Job: Economic Analyst

Location: Manhattan, New York

Education: International Business, Master Degree

Family: Wife and 2 children

Characteristic: Ambitious

Needs

1. Valid information on US President's decisions regarding economy (e.g. taxes, import and export tax plans, insurance plans)

2. Valid information on public's reaction towards decisions of US President

Bio

John Tyler is a young ambitious businessman specialised in economic analytics. In order to perform his job he needs access to trustworthy news sources allowing him to analyse the current economic situation.

1.4.2 Persona 2

Background

Name: Haily Bringston

Age: 45

Gender: Female

Job: Mother

Location: Dallas, Texas

Education: Psychology, Master Degree

Family: Divorced

Characteristic: Naive, Lazy

Needs

1. Method of easily finding valid news

2. Quick access to various news

Bio

Haily Bringston is a single mother, living in Dallas, TX. Due to news on terrorism and the refugee crisis in Europe she has become more and more doubtful about immigrants from Arabic countries.

1.4.3 Persona 3

Background

Name: Ben Ali

Age: 22

Gender: Male

Job: Creative Technology Student

Location: Enschede

Education: High School Diploma

Family: Single

Characteristic: Politically Interested

Needs

1. Unbiased news on current political events

Bio

Ben Wang is a Creative Technology student in the middle of his exam period. Due to current political events in the US, Russia, Turkey and the Arabic world, Ben has developed a big interest in politics.

Chapter 2

Specification Phase

Within this chapter the previously researched ideas of the ideation phase are going to be specified. To that end a visual concept of the website is going to be created in order to create a clearer vision. Moreover setup schemes for the system are going to be created in order to create an overview for the necessary processes.

2.1 Conceptualising the User Interface

Based on the research of the ideation phase is possible to state that the proposed system will be web based. Nonetheless other platforms such as a smartphone application suit the system and can be regarded in future work. In addition to that by applying the rule of thirds it is possible to make the layout look and feel natural to the user. Moreover during development it is important to constrain the amount of content displayed on the screen at a time, in order to avoid an overload of information. In case of a news hub it is also important to keep the layout structured. Doing so allows the user to easily compare articles, helping to decide which one to read. In order to keep an organised structure it is important to focus on the content density. Here it is important to not place too much content at too close to each other, but to have a reasonable amount of distance between them. Moreover it is important to make the website only tow pages deep, in order to keep it simple and allow the user to find quickly their intended item. Additionally only a limited amount of information on a page should be used in order to keep the page loading time as short as possible. In regard to the here discussed related work it is possible to say that both provide suitable ideas. Pinterest's card-based layout provides an appropriate overview, allowing the user to skim through content and chose the most interesting stories. Flipboard on the other hand provides more information by combining images, headlines and the introduction of an article. Doing so allows the user to have a sneak peek into the article and decide based on that if they want to read it. Thus a combination of Pinterest's card layout and Flipboard's information provision are desired elements for the platform.

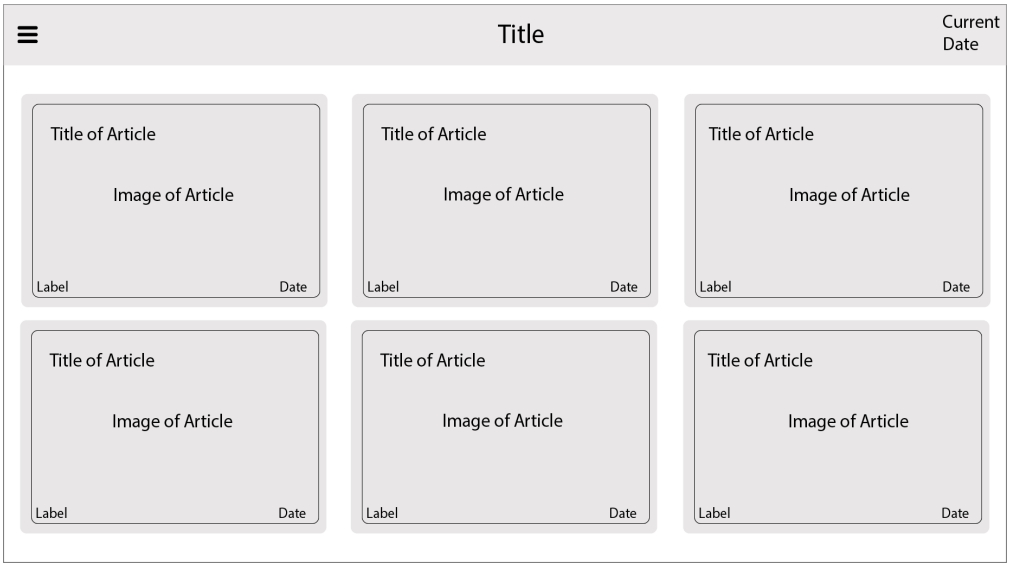


Figure 2.1 Website Concept

Looking at the concept of the website (Figure 2.1) the implementation of the earlier mentioned aspects is observable. Firstly it is possible to see that a card layout has been implemented, which is divided according to the rule of thirds. Moreover the amount of cards is restricted to six per page, making the page arranged clearly and easy to read. In addition to that looking at each card individually the implementation of a title, image and a label are is apparent. The title and image refer to the article's title and image and change according to the article. The label here refers to probability of the article being fake indicated as a percentage. To that the title of the link, when clicked, links to the original article. Doing so allows to keep the depth of website at one level. Finally at the top right corner the publishing dates of the articles is going to be shown.

2.2 Specifying the Method of News Verification

In regard to news verification is possible to say that the of various linguistic methods is necessary in order to develop a reliable system. Based on the Conroy, Rubin and Chen [33] literature the following list is compiled, listing the linguistic methods that are needed in order to develop a hybrid system.

1. Bag of Words
2. Stop Words
3. Part of Speech
4. Lemmatization

To that end a list of classification models has been compiled, showing which models will be used. Those will be weighted against each other in order to find out which one provided the best outcome. Regarding Naive Bayes two variations have been chosen due to the fact that they are the most commonly used classification model [40].

1. Naive Bayesian models (e.g. Multinomial Naive Bayes, Bernoulli Naive Bayes)
2. Support Vector Machines (SVM)
3. Decision Trees (DTs)

In terms of vectorization Scikit Learn's TfidfVectorizer is going to be used. This is due to the fact that it calculates the frequency of words throughout all available documents, giving each word a weight value according to that. Doing so allows to easily filter out words that are not of importance allowing better and faster processing.

In the figure below (Figure 2.2) the setup has been depicted. Looking at it it is possible to see that the whole process is going to run offline. Firstly a corpus is created by scraping articles from the web. More details regarding this aspect will be discussed later on in this chapter. After the corpus has been created NLP needs to be applied to each individual article collected. During the NLP process articles will be chopped into sentences and then words. After that the stop words will be removed and the each words left over will be lemmatized. To that end tfidf-vectorization is going to be applied. Finally the vectorized tokens will be sent to the classifier in order to train it and later on the trained classifier will be saved for later usage.

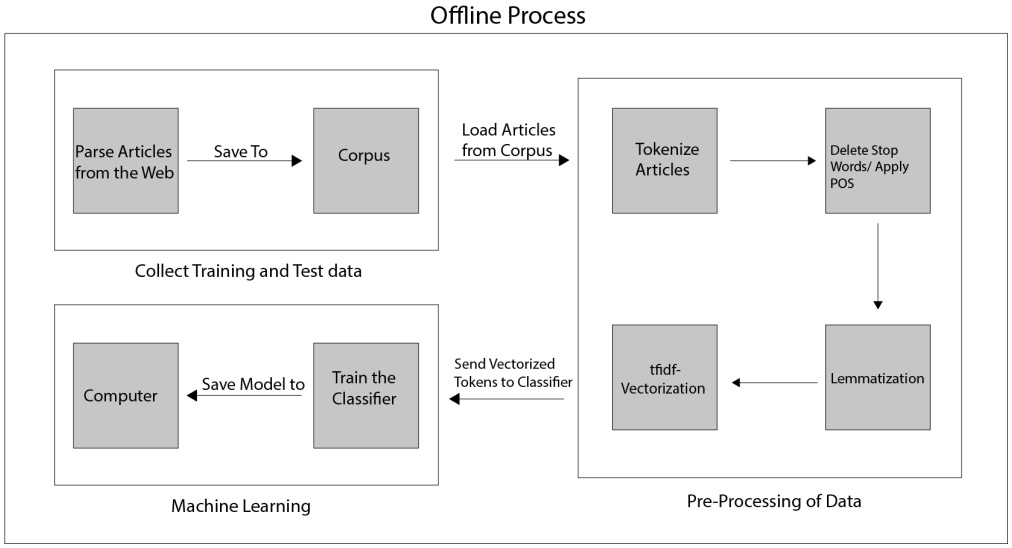


Figure 2.2 Setup Scheme 1: Process for Training of Classifier

Moreover in regard to the Network Approach it is possible to say that it will be disregarded. This is due to the fact that microblogs, such as Twitter, are not within the scope of this graduation project.

2.3 Corpus Creation

Applying the methods of news verification requires a collection of news articles (corpus), which need to be divided into reputational news and fake news. In order to do so two lists

need to be compiled containing URLs of reputational news websites and fake news websites (Appendix B.0.2). The list containing reputational news websites is based on a research conducted by the Pew Research Center [41], who have researched the trust levels of news sources among the American population. Furthermore OpenSources (http://www.opensources.co/), who have generated a list ranging from credible news sources to misleading or fake news sources, is used as well. To that end OpenSources is also used in order to compile the second list containing fake news web sources. In order to provide the classifier with enough training material, the sample size should be quite big, to be more specific 2000 articles need to be parsed and processed. Additionally the amount of articles parsed from each listed website is going to differ. This is due to the fact that each publisher updates their content on a different pace leading to different amounts of articles. The parser will parse as many article as are available. To that end it is possible to say that the difference does not effect the outcome of the training procedure as long as the amount of articles is 50% real and 50% fake.

2.4 Basics of Supervised Machine Learning

Supervised Machine Learning is a method where the machine learns the link between two datasets [42]. X represents the observed data and y and external variable, where the aim is to predict y [42]. In the case of the graduation project X are the articles and y are the labels of the articles (e.g. fake or real). Moreover it impossible to say that this graduation can be described as a classification task, not a regression task. This is due to the fact that the articles are going to be classified to a set of labels [42].

Part II

Developing the News Aggregator

Chapter 3

Realisation Phase

Throughout this chapter the progress of the creation of system will be described. For the training of the classifier the setup scheme in figure 2.2 is going to be used. In addition to that a second setup scheme (Figure 3.1) has been created, which describes the server sided process.

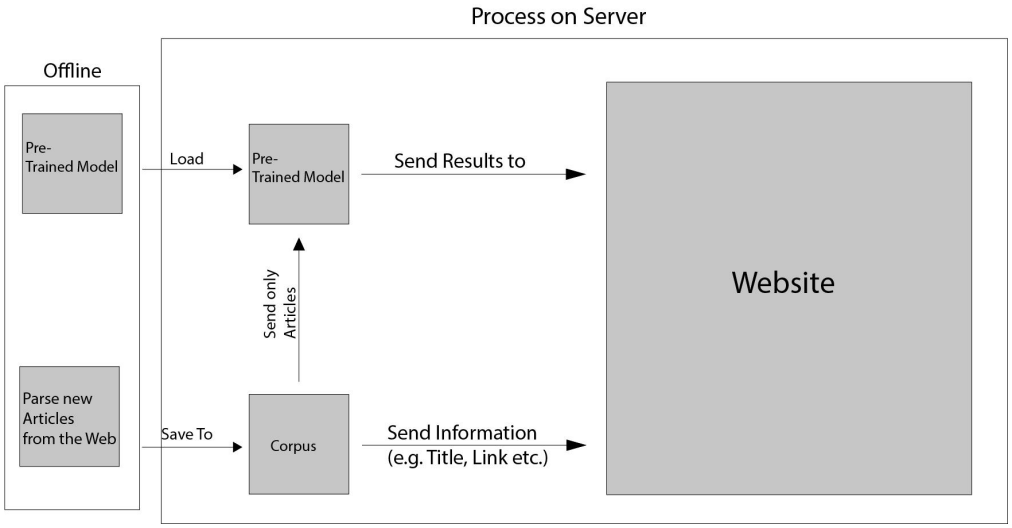


Figure 3.1 Setup Scheme 2: Server Sided Process

3.1 Scraping, Storing & Reading Articles

Scraping Articles

In order to scrape websites the programming language Python provides several modules (i.e. plug-ins or libraries) that allow the extraction of certain information from a website. After conducting research on which modules exist and which ones provide the most reliable outcome the following modules were found: *Beautiful Soup 4; lxml; Scrapy and Selenium.* However due to the fact that Beautiful Soup 4 is the most used module for web scraping,

it was decided to use it, as well. After installing and importing it into Python the code in Appendix C.0.1 was written in order to scrape content from a website. Looking at the code it is possible to see that a list was created containing hyperlinks of articles. Using the 'Urllib' module it is possible to fetch the websites, which gets used by the Beautiful Soup module in order to read the websites' source codes. Doing so allows to search for specific tags, such as the paragraph tag ([...]), in the source code, which than can be printed into the terminal. Looking at the output in Python's terminal (B.0.1) various aspects can be observed. Firstly it is possible to see that it was able to find and print out the article, successfully. However looking deeper into the printed text it is possible to see that not only the article got printed, but other aspects such as references to social media, as well. This is due to the fact that in the it was stated to print all paragraph tags. Hence the information retrieved depend on the source code of the website and how it was written. A work around for that issue is shown in the following code (see 3.1):

```
for eachparagraph in body.find_all('p', class_='[...]'):
    print(eachparagraph.text)
```

Listing 3.1 Beautiful Soup: Class

Here it is stated to find all paragraph tags with a certain class, allowing to retrieve the wanted information. Although this is a solution for retrieving article content from a website, it is not a scalable solution, due to the fact that each website uses different tags and class names for their content, thus not allowing to scrape specific information, easily. Because of that reason a more detailed research has been conducted, researching on how the previously found modules scraping modules work, and what alternatives exist. Doing so lead to finding a module called *Newspaper*. This module is specifically designed to retrieve articles, authors, dates and other data from news websites, allowing to scrape information on a big scale. Investigating the provided code in Appendix C.0.2 it is possible to see that a list containing hyperlinks to news websites was created. In order to iterate through each website and download the available articles a loop was created. After an article was downloaded Newspaper's parse() function is called, allowing to retrieve information, such as the URL, the title, the text (article itself) and the authors, of the article.

Those information are then appended to an empty list, enabling later usage. After looping multiple times and adding the retrieved information to the list, the information are stored in a DataFrame using the module *Pandas*. The following step is to store the DataFrame as a file.

Storing & Reading Data

Regarding the storage of cumulated data various options do exist. Research concluded in the possible usage of three options: *SQLite; JSON; CSV*. SQLite is a slimmed down version of

SQL, meaning it is capable of creating databases containing multiple tables with thousands of information. Nonetheless such a platform is not necessary for the storage of article data, thus it was disregarded. Looking at the other two options, JSON and CSV, it is possible to say that they are more suitable for the presented situation. This is due to the fact that of each article only the text, the authors, the title and the hyperlink need to be stored, which can be done in one table. Hence the usage of nested data is not necessary, leading to decision of using CSV as a manner of storing data.

```
with open('Sources_RF', 'w') as f:
    dl.to_csv(f, sep="/")
```

Listing 3.2 CSV: Write data

In order to write data onto a CSV file the code 3.2 is added to the write() function. Doing so opens the CSV file and allows to append data to it. Moreover it is declared to use a slash symbol (/) as a separator instead of a comma. Usually a CSV file uses commas to separate the content, however there is a articles use commas in their titles and text, as well, possibly leading to a false interpretation of separation. The stored data is divided into five columns: 'LINK', 'TITLE', 'TEXT', 'AUTHOR' and 'CATEGORY', where 'CATEGORY' indicates whether the article is real or fake. Another possibility would have been to only store the links to articles, which could the be downloaded and parsed when it is needed, however this methods lacks a lot in regard to efficiency, meaning reading a file is noticeably faster than downloading and parsing each article individually. This lead to the usage of the former method.

```
import pandas as pd
dr = pd.read_csv('Sources_RF.csv',  delimiter="/", dtype=str)
ds = pd.DataFrame(dr)
```

Listing 3.3 CSV: Read data

Reading the data from the previously generated CSV file can be done by using the following code(see 3.3), which also converts the data to a DataFrame, by using the Pandas module.

3.2 Natural Language Processing

In order to conduct Natural Language Processing (NLP) the Python module NLTK is used. Here the articles are tokenized, stop words are removed, part of speech is applied to each token and then each word is lemmatized. To do so a class called PreProcessing() has been created, which contains the methods:__init__(), tokenize() and lemmatize(). The method __init__() (see C.0.3) is a method that acts as a constructor. It initializes several classes

from the NLTK and the String module. The lower method's task is to return a copy of the received received strings, where each character is lowercased. In addition to that a strip method is initialized, which removes white-space at the beginning and at the end of a string. Moreover the stopword() method, which removes stop words from the articles, the punctuation method which recognizes punctuation in a string, and the WordNetLemmatizer() are initialized.

The method lemmatize() (see C.0.3) is used in order to make the lemmatization compatible with the part of speech tags. Here it is necessary to note that NLTKs lemmatization method is only able to recognize four different tags, which are nouns, verbs, adjectives and adverbs. However the part of speech method from NLTK has 36 different tags. However various tags are just different versions of the former mentioned part of speeches (noun, verb, etcetera), meaning NLTKs part of speech method distinguishes between for instance a noun in singular form and a noun in plural form. Hence the code tells all part of speech tags, received from NLTKs part of speech tagger, starting with a 'N' are nouns, 'V' are verbs, 'R' are adverbs and 'J' are adjectives.

The tokenize method (see C.0.3) uses two for loops, which are nested. The first for loop chops the article into sentences, which due to the second for loop are chopped into single words, of which each has the part of speech applied to. To that end each word is lowercased and the white space of at the beginning and end of each string is removed. Furthermore two if statements are used, where the first one states that if a stop word appears it should be ignored. The second if statement tells the system to check whether a punctuation is receives, which if so is removed. Finally the lemma variable applies the lemmatization to each word.

3.3 Supervised Machine Learning

After the data has been preprocessed using natural language processing, it can be used to teach the system to distinguish between the data, regarding whether it is real or fake.

In order to start the content of the 'TEXT' column of the DataFrame is converted from an dtype object to unicode (see 3.4). This step is necessary to make the data readable for the vectorizer. In addition to that Scikit Learn's LabelEncoder() is used (see 3.4). This translates the 'CATEGORY' column, which contains information on whether an article is fake or real, into numerical values.

```
X = ds['TEXT'].values.astype('U')

y = np.asarray(ds['CATEGORY']).ravel()
le = LabelEncoder()
le.fit(y)
```

```
y = le.transform(y)
```

Listing 3.4 Conversion of X and transformation of y

In the next step a pipeline has been created (3.5). The pipeline arranges the elements in such a way that the first elements output is the second element input. In this case TfidfVectorizer outputs its data to the MultinomialNB. Here two aspects are important to note. Firstly TfidfVectorizer is able to conduct natural language processing such as tokenization and lowercasing characters. However since NLTK, due to higher reliability factor, has been used the vectorizer's parameters are set to use previously in the PreProcessing class defined tokenize method. To that end the vectorizer's lowercase parameter is set to false, since as stated previously the __init__ method contains a method that lowercases characters. The second aspect that stands out is the MultinomialNB classifier that is used, however it can be easily switched out by changing it to for instance BernoulliNB. Hence based on the results, the best classifier will be chosen, however this aspect will be discussed later on in this document.

```
text_clf = Pipeline([
    ('vect', TfidfVectorizer(tokenizer=r.tokenize, lowercase=False)),
    ('clf', MultinomialNB())
])
```

Listing 3.5 Pipeline

After that the next step is to declare that the available data is split into training data, which allows the classifier to learn, and a test set in order to test how well the classifier is able to distinguish between the data . In order to do so Scikit Learn provides a model, called train_test_split (3.6). Looking at its parameters it is possible to see that the test_size has been set to 0.2, leaving 80% of the data for training. Those numbers have been chosen due to the fact that the more training data the classifier is fed, the better it can estimate the test data.

```
X_train, X_test, y_train,
y_test = train_test_split(X, y, test_size=0.2)
```

Listing 3.6 Split Data into training and test set

Finally the data needs to be fitted into the pipeline, which can be done by .fit() as seen in 3.7. More it is possible to see that a variable has been created, called predicted. Its task is to predict the labels (e.g. fake or real) for each article. Finally classification_report function is printed, which build text report. It prints out the results in regard to precision, recall and f1-score. In information retrieval precision indicates the ability of a classifier not to label a

sample as positive although it is negative. The recall value describes the ability of a classifier to find all positive values. Finally the f1-score is the harmonic mean of precision and recall.

```
text_clf = text_clf.fit(X_train, y_train)
predicted = text_clf.predict(X_test)

print(classification_report(y_test, predicted, target_names=['Fake',
'Real']))
```

Listing 3.7 Train and Test Classifier

3.3.1 Evaluating the Classifiers

Summing up it is possible to say that 1823 articles have been parsed and stored, using the method described in Section 3.1. To that end those have been divided into training and test data, where 80% of the data is used to train the classifier and and 20% is used to test the classifier. After that the classier needs to be evaluated in order to gather information on its efficiency. In order to evaluate the different classifiers Scikit Learn's classification_report() function is going to be used. For each classifiers it calculates the precision, recall, f1-score and support. Precision describes the ratio using $tp/(tp+fp)$, where tp describes the amount of *true positives* and fp and the amount of *false positives* [43]. In regard to the recall it is possible to say that it can be described as: $tp/(tp+fn)$. Here tp again refers to the true positive values and fn to the number of *false negatives* [43]. In addition to that the f1-score (or F-beta) can be described as a harmonic mean value, which uses the scores of the precision and the recall in order to be calculated [43]. Here the better the value to closer it is to 1. Finally the support value describes the number of true responses [43], however during the evaluation this number will be disregarded.

Multinomial Naive Bayes

Looking at the outcome of the classification report (3.2) it is possible to observe that the precision for labelling fake news as fake is at 93% and real news as real at 77%, leading to an average of 85%. The recall for fake articles is at 69%, where for real articles it is at 95% leading to an average of 83%. This leads to a average f1-score of 82%.

```
Print results:
              precision    recall  f1-score   support

        Fake       0.93      0.69      0.79       174
        Real       0.77      0.95      0.85       191

avg / total       0.85      0.83      0.82       365
```

Figure 3.2 Classification Report: Multinomial Naive Bayes

Bernoulli Naive Bayes

Looking at the classification report (3.3) for the Bernoulli Naive Bayes classifier it is possible to see that the precision average is at 82%, the recall average at 79%, leading to an f1-score of 79%.

```
Print results:
              precision    recall  f1-score   support

        Fake       0.72      0.93      0.81       174
        Real       0.91      0.67      0.77       191

avg / total       0.82      0.79      0.79       365
```

Figure 3.3 Classification Report: Bernoulli Naive Bayes

Support Vector Machines

The results for the Support Vector Machines (3.4) show that an average of 87% in precision and recall, concluding in an average of 87% for the f1-score.

```
Print results:
              precision    recall  f1-score   support

        Fake       0.89      0.83      0.86       174
        Real       0.85      0.91      0.88       191

avg / total       0.87      0.87      0.87       365
```

Figure 3.4 Classification Report: Support Vector Machines

Decision Trees

Here the classification report (3.5) shows and average of 74% for precision and recall. This concludes in a f1-score of 74%.

```
Print results:
              precision    recall  f1-score   support

        Fake       0.75      0.70      0.72       174
        Real       0.74      0.79      0.76       191

avg / total       0.74      0.74      0.74       365
```

Figure 3.5 Classification Report: Decision Trees

Conclusion

Finally it is possible to say that the Support Vector Machine classifier provides the highest f1-score. This means that SVM classifier is the most reliable option from all tested classifiers, regarding the differentiation of real and fake articles. Hence it will be saved as a pickle-file for later usage.

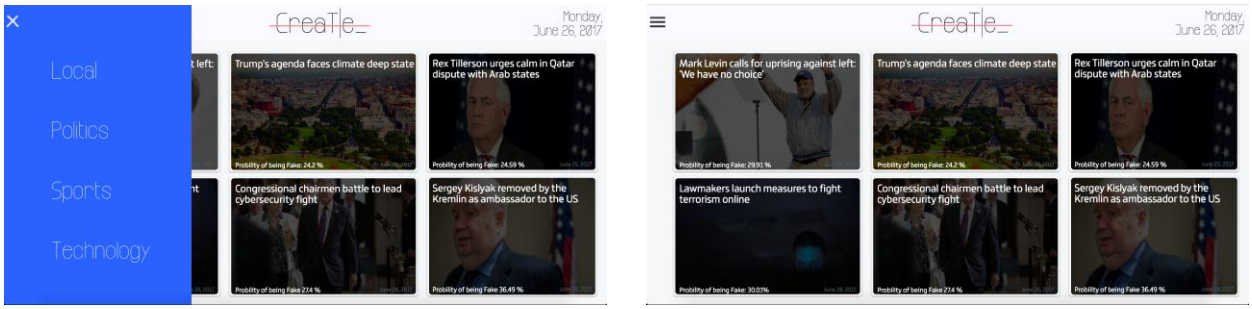
3.4 Platform Development

As stated earlier in this report the the chosen platform is a website. In order to develop the website HTML (Hypertext Markup Language), CSS (Cascading Style Sheets) and jQuery, a cross-platform JavaScript library, are used.

Process of Website Development

The website development process started with the HTML markup in combination with CSS in order to provide the website with content and stylistic elements.

Firstly the focus was on the creation of basic the elements, such as a header and a vertical menu. The header's task is to carry the name of the website, in order to give it an identity and a 'hamburger' button to interact with the menu. By pressing the button the menu slides out on the left side of the screen providing the user with the following additional options: *Local, Politics, Sports, Technology*. However it is not possible to interact with those options due to the fact that it is out of the scope of this graduation project to implement multiple pages, nonetheless those can be added afterwards. Moreover the rule of thirds is applied to the menu. When open it takes one third ow the whole horizontal width of the screen (See Figure 3.6(a)). After that the main section of the website was created (See Figure3.6 (b)). It uses a variation of Pinterest's card layout, where one page contains two rows of which each contains three cards. Additionally the cards are restricted in height and width, and are constructed according to the rule of thirds. To that end each card provides the title, date and the image of an article, and a 'Read More' button enabling the user to access the original article.



(a) Activated Menu

(b) Deactivated Menu

Figure 3.6 Design of the Website

In addition to that the website uses a one page scroll. Instead of having a continuous scroll it scrolls two rows at once, meaning when the user scrolls their are provided with two new rows of content. The one page scroll was implemented by using the jQuery plug-in fullPage.js. The whole code for the website is provided in appendix B.0.1.

Inserting Data into the Website

In order to connect the previously developed system to the website the Flask module can be used. Looking at the code 3.8, it is possible to observe various aspects. Firstly looking at the second line of the code the route() decorator tells Flask at which URL is supposed to trigger the function. Moreover it is possible to see that the method has been set to 'GET', which tells the server to get the information, that are contained on that page. Secondly looking at return statement the render_template() method which is a part of the Flask module is used. Within this method the index file of the website previously developed is loaded In addition to that all information that are going to be displayed on the website are included.

```
app = Flask(__name__)
@app.route('/',methods=['GET'])
def index():
    return render_template('index3.html', df=predict, dg=a, db=df['TITLE
        '],date=df['DATE'], di=df['IMAGE'], pred=np.around(prob[:, 0] *
        100, decimals=2),link=df['LINK'], today=today.date())


if __name__ == '__main__':
    app.run()
```

Listing 3.8 Connection to Website

Despite telling Flask which data is going to be used, it is necessary to implement them into the HTML file. In order to do so slightly modified python statements can be used. An example for the implementation can be seen in the code 3.9.

```
{% for j in dg %}
{% if j == 7 %}
 <img src={{di.ix[j]}} alt="pic">
  <div class="wrapper">
  <a href={{link.ix[j]}} target="_blank"><h3>{{ db.ix[j] }}</h3></a>
   </div>
   <h5>Probility of being Fake: {{pred[j]}} %</h5>
```

Listing 3.9 HTML Code

Chapter 4

Evaluation Phase

In this chapter the prototype that was created is going to be evaluated. To begin with the methods of evaluation are going to be discussed and described. Later on those will be executed and their outcomes are going to be used in order to form a final conclusion.

4.1 Evaluation Methods

In order to evaluate the prototype two methods are going to be used: *Scenarios* and *Usability Testing*. The Scenarios method incorporates stories and context in regard to why a user would use the product. Within the story the user's questions and goal need to be clarified and how the interaction with he product helps to achieve those goals. Here the previously created personas will be incorporated. When conducting a usability test participants are asked to execute tasks. The tester's task is to observe the participants behaviour and take notes. The aim of Usability Testing is to find out whether usability issues exist and to determine the user's satisfaction. The data that is going to be collected is listed below:

- Rate of success of completing task
- Time to complete task
- Requirements for improvement
- User's satisfaction

Therefore a list of tasks needs to be compiled. Here the core functions of the news aggregator are going to be tested. The list of tasks is listed below:

1. Find the probability of an article being fake.
2. Open an article.

3. Find articles the publish date of an article.

Finally a short survey will be conducted. The goal here is to find out whether people are interested in using news aggregation website that is capable of verifying news. Moreover it is used to find out whether there is a difference in interest in different age groups. The survey is provided in appendix D.0.1.

In regard to the Usability Testing ten people will be gathered as participants, where for the survey 20 people will be questioned. In both cases most age variation will be covered, meaning people from different age groups will be tested and questioned.

4.2 Scenarios

4.2.1 Scenario 1: John Tyler

John is a 31 year old economic analyst at the Bank of America. In order to execute his job he relies on heavily on software, however current news play an important role during his decision making process. Shortly before the outcome of the 2016 U.S. election John was asked to analyse and forecast the rate of the U.S. Dollar. Those can differ highly depending on the new president, their view and how people and other countries perceive them. Despite using the software John gathers information by reading various news for different sources. Here he is confronted with different reputational news sources and data, however most information indicate that Hillary Clinton has the significant higher probability of winning the election. Hence he creates a forecast for the U.S. Dollar rate based on that outcome. After election day it has been announced that despite the positive forecast for Hillary Clinton, Donald Trump has won the election and hence becomes the next president of America. Due to that unexpected outcome John's forecast does not reflect the situation at all and failed at doing his job.

4.2.2 Scenario 2: Haily Bringston

Haily is a 45 year old mother, living in Dallas, Texas. There she lives in a average income apartment, in a mid class part of the city. She went through multiple marriages and divorces. Haily has been following the news actively for the last couple of years, where observed an increase in terrorist acts in Europe. According the news she gathers those acts can be traced back to refugees who have entered the countries within the last couple of years. Due to that she has developed a fear of refugees and foreigners from the eastern world. Due to that fear she does not allow her youngest son to meet up with a friend who is a refugee, since she believes that the family of the friend could influence her son negatively or even do something

bad to him. Thus she has also got in contacted with the children's teacher in order to tell her that she should not allow her son to pay with the refugee.

4.2.3 Scenario 3: Ben Ali

Ben is a 22 year old Creative Technology student at the University of Twente. He generally interested in technology, but has developed recently an interest in political activities. Reasons therefore are that due to his foreign background the political situation within Europe can have a big effect on him and his family. Due to the increase of terror attacks in Europe, or more specifically France, Belgium and England right winged parties have gained support. His fear of racism being acted upon him has increased by a lot. Hence he is looking for a method to convince people around him to not believe everything that is being published online. Thus he tried to convince people to start reading and comparing various news sources, however after some time he realised that most people are not interested in doing so.

4.2.4 Conclusion

All in all in regard to the different scenarios it is possible to say that the developed system differs in its effects. In John's situation the system does not provide any benefits. This is due to the fact that John has used news from reliable and reputational publishers in order to gather information on the election and the projections. If you would have used the system it would have led to the same outcome. Looking at Haily's situation it is possible to say that a news aggregator with news verification can help her to obtain a more thought-out and reasoned opinion. Using such a system allows her to cover various news sources easily and see the probability of the content of the article being invalid. Hence in her situation it allows to create a less negative opinion, and develop a more reasonable one. In regard to Ben the developed news aggregator has an positive effect as well. This is due to the fact that the news aggregator allows people to easily compare news articles. In addition to that since the system gives each article a certain probability of being fake Ben's friends do not need to compare the articles actively, but only by looking at the labels. Due to its efficiency his friends have also a higher probability of using it, thus reducing Ben's issue.

4.3 Survey Results

Looking at the results (Appendix D.0.2) it is possible to observe that the most age group related to the project are covered, however it cannot be denied that a big part of the participants are in their twenties. To that end it is possible to see that the majority of the participants (65%) are interested, and only a small part (5%) is not interested in news at all. The rest (30%) occasionally checks the news in order to obtain information on local and

world events. In addition to that the survey resulted in that above half of the participants (55%) check the news daily. 30% gather news information two to three times a week and 10% twice a week. One participant answered that they do not check the news at all. Moreover it was asked which medium is used by the participants in order to gather news (Figure 4.1). Here the results show that the majority (50%) uses social media in order to obtain news, where only 7,14% or 2 participants answered with that they would use a news aggregation system. The second highest votes go to conventional methods, such as the radio, TV and newspaper, which has a value of 21,41%. Moreover 17,88% of the participants use specific news website, such as telegraaf.nl, in order to read news.

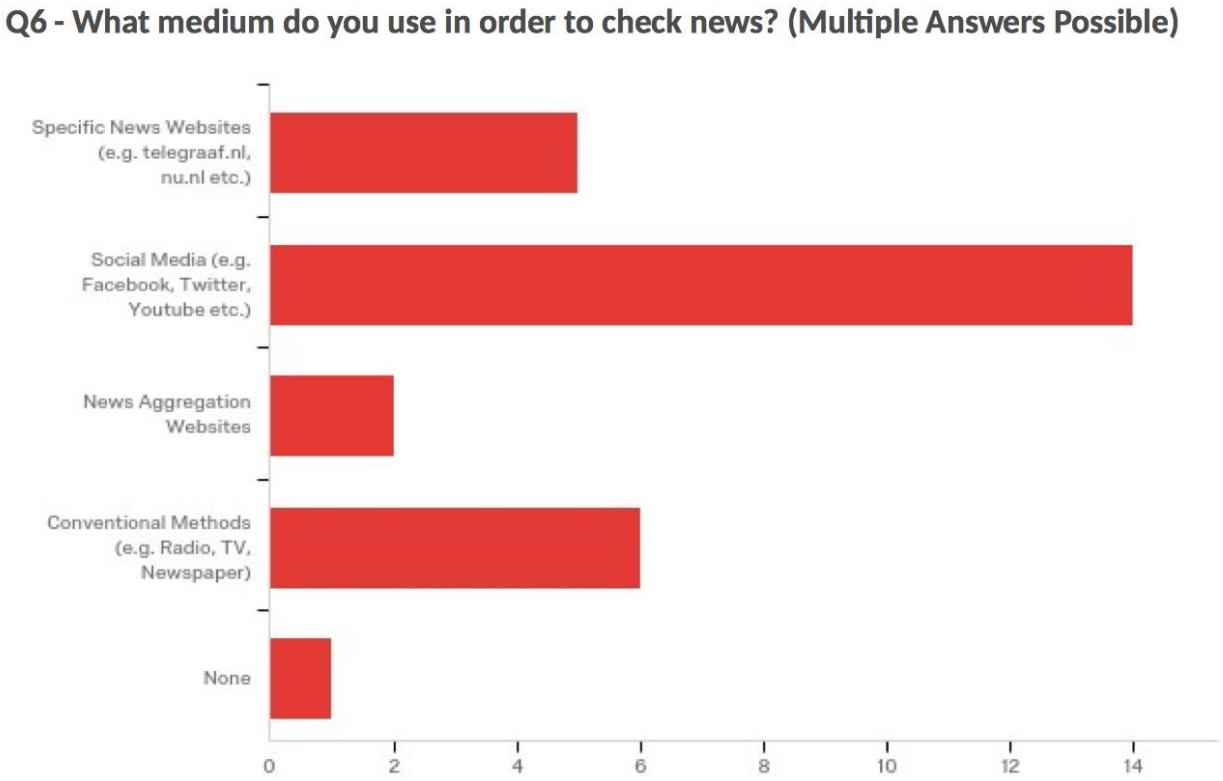


Figure 4.1 Survey: Medium of Choice

In addition to that during the survey participants were asked for their trust in the news media. The results (Figure 4.2) show that the majority (55%) do not know whether to trust news spread by the media. Moreover 10% provided the answer that they do not trust the media at all and another 10% answered with he exact opposite, meaning they do trust the media fully.

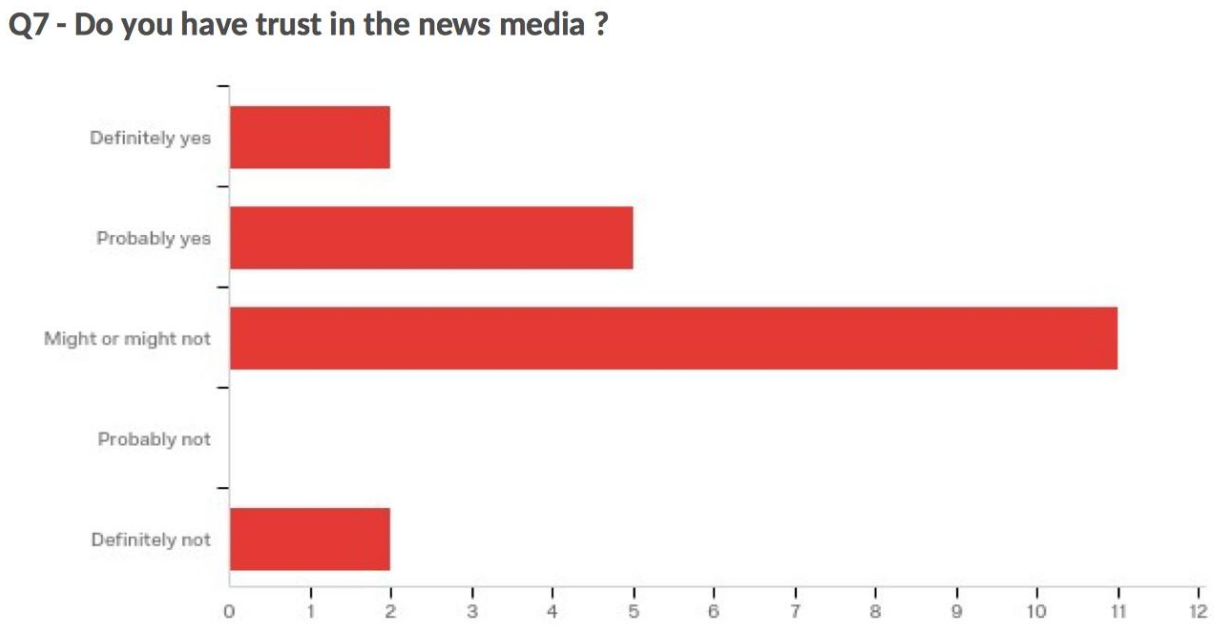


Figure 4.2 Survey: Trust in News Media

Furthermore in regard to their capabilities of distinguishing between fake and real news, 55% of the participants assume that they would be capable of doing so. 35% of them answered that they are not sure but that there would be a possibility of successfully distinguishing between fake and real articles.

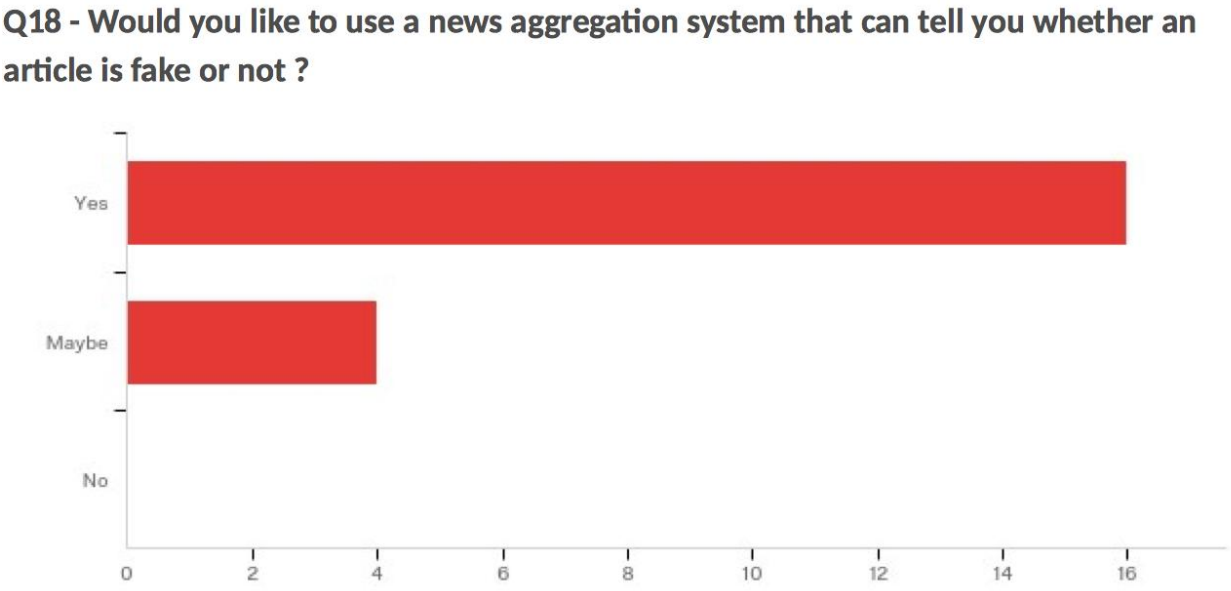


Figure 4.3 Survey: Potential Usage of News Aggregation & Verification System

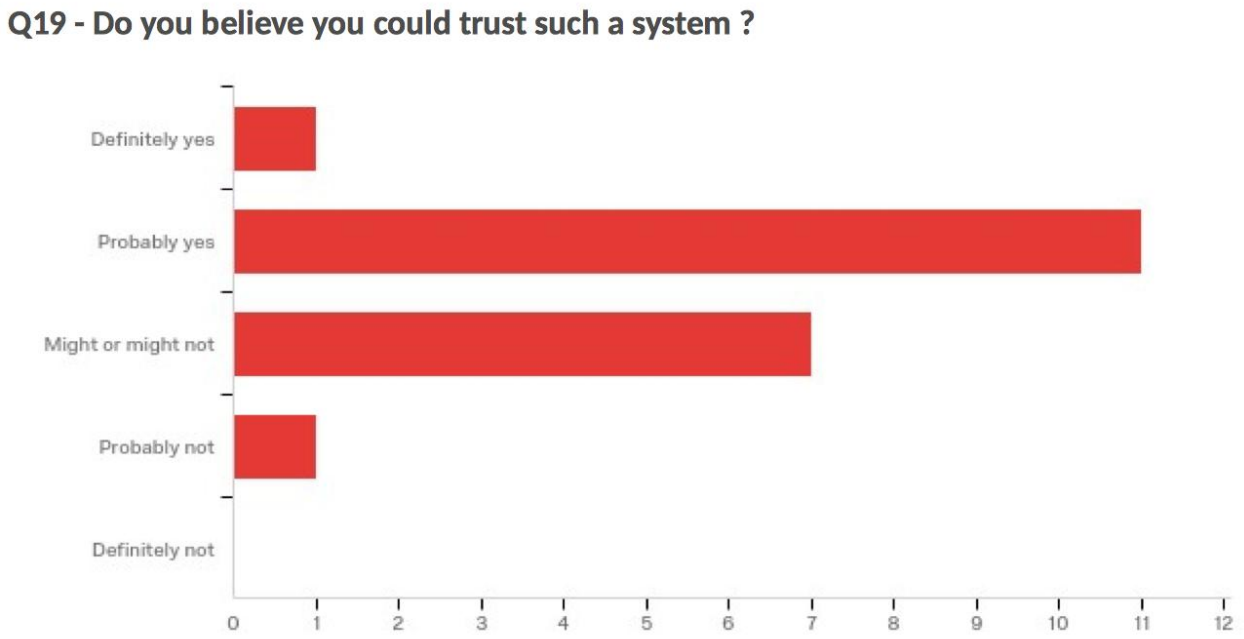


Figure 4.4 Survey: Trust in News Aggregation & Verification System

To that end participants were asked whether they would use a news aggregation system that is capable of news validation (Figure 4.3). The results here indicate that there is a demand for such a system since 80% of the participants replied with yes. The rest of the participants showed uncertainty in whether they would use a system, however none replied with that they would not. Finally the participants were question about their thoughts on whether they could trust such a system (Figure 4.4). Here more that half (55%) believe that they could trust the system, on 5% showed that they are certain that they could trust the system. Moreover it is possible to observe that 35% are uncertain regarding the trustworthiness of the system. Only 5% believe that they could probably not trust the system.

All in all it is possible to say that the survey shows that there is a certain demand for a news aggregation and verification system. This is due to the majority of the users do not know whether they could trust news media or not. Moreover the survey shows that people believe that they could trust a news verification system, indicating that are open minded towards such an idea.

The results of the whole survey are provided in Appendix D.0.2.

4.4 Usability Testing

The results of the Usability Testing show diverse outcomes in different aspects. Moreover it is important to state that three of the tested users are in their twenties, where the other two participants are 48 years old and 68 years old. First the users were given three different task to conduct of which the time to execute them was recorded (Figure 4.5). The overview

shows that the average time for conducting a task are 9 sec, 12 sec, and 6 sec. At first glance the numbers evoke the pretence that participants needed a relative high amount of time in order to execute a task. Nonetheless due to the 48 and 68 year old participants the average increased, since all given maximum values are based on their results. For people at that age with little to medium experience with computers the values are reasonable. Hence it is possible to say that in general the results show a positive outcome, regarding layout, arrangement and overview of the website.

Q1 - Time to complete task:

Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
Task 1 : Find the probability of an article being fake.	3.00	9.00	5.20	2.04	4.16	5
Task 2: Open an article.	2.00	12.00	5.20	3.87	14.96	5
Task 3: Find articles the publish date of an article.	1.00	6.00	3.20	1.60	2.56	5

Figure 4.5 Usability Testing: Time to execute task

In addition to that the participants were asked to provide their thoughts on the website. Here the focus points are: coherence, design, arrangement, and the amount of information. The matrix table (Appendix D.0.3) indicates that the majority of perceived the website positively. In terms of coherence the matrix shows that the participants view it as above average. Regarding the design of the website two participants view the it as average, where the other three perceive its design as good and perfect. To that end one participant perceives the arrangement of the website as bad, where another one views it as average. The other participants viewed the arrangement of the website as good. In regard to the amount of information shown on the website the majority of the participants (4) as good. Thus it is possible to state that although the results show in most instances a positive outcome, slight improvements are necessary in order to improve the experience.

Moreover three additional questions were asked in order to find out, whether certain aspects of the website self-explanatory. The focus point here were the menu, the current date, and the scroll feature. In regard to the menu the participants were asked whether they were aware of the existence of a menu. Three out of five participants stated that they were

not aware of its existence. In terms of the availability of the current date the majority of the participants (4) were aware of its existence. Moreover the participants were questioned about their awareness of the website being scrollable. Here the four out of five participants did state that they were aware. Those results show that in most of the elements are self-explanatory. Nonetheless In regard to the menu additional work needs to be done in order to increase awareness of its existence.

Finally the participants were asked for general comments about the website. The participants perceived the layout, although its aesthetically pleasing appearance, as badly distinguishable. This refers to the layout of the individual cards, since all have the same dimensions. It was suggested to use a layout based on hierarchy, which indicated the importance of an article by the size of the card. Another point of criticism was the publish date of the the individual articles. Here the participants perceived it as hard to see and unnoticeable. According to them this was due to colour choice of the date. To that end the participants suggested to implement an archive for user who want to view older articles and a search bar to allows users to find articles, quickly.

The results of the Usability Testing are provided in Appendix D.0.3.

Chapter 5

Conclusion

TO begin it is possible to say that an additional research question can be asked which is: *How precise and accurate can fake news be identified?*. This question gets answered in Section 3.3.1. There 3.4 it is possible to see that the harmonic mean of of the accuracy and precision is equal to 87%. The results of the survey and the Usability Testing show positive results in regard to the the system in general, its design, its usage and its acceptance by the people. To that end the developed scenarios depict three different situations of which two show off system's benefits. Hence it is possible to say that the conducted survey and three scenarios show a demand for a news aggregation and validation system does exist. Moreover the results of the survey show that the participants in generally tend to believe that they can trust the system, showing that they are open minded and not partisan about it. In addition to that looking at the results of the Usability Testing it is possible to say that that the website is perceived positively. Design, arrangements and coherence seem to fit to today's standards. Nonetheless the general feedback by the user's showed that there are still points that need to be regarded in order to create an overall improved experience.

5.1 Future Work

This part of the report is used in order to discuss which aspect can be changed, implemented or improved in order create an improved version of the news aggregation and validation system. This section will discuss possible improvements for each instance of the system i.e. the news verifier algorithm, the corpus and the website itself.

Firstly it is possible to say that some of the previously discussed aspects in Section 1.1 can be implemented. Especially the like and dislike function can be viewed as an additional approach of validating news. Here modified version, which contains fake and real labels instead of like and dislike labels, which allows users to vote manually whether an article is real or not. In combination with the already used algorithm, such an approach might lead to better and more trustworthy results for the user. In addition to that another future

implementation that needs to be considered are social media features, such as sharing articles and following users and publishers. Doing so gives users the opportunity to customize their content by following users and publishers that share content that are the users are interested in. Furthermore based on Flipboard, it is possible to state that the implementation of recommender systems should be regarded in future work. Here the system would learn overtime the preferences in topics of the user. This allows users to have a customizable and tailored experience, increasing its practicality for the user. In addition to that based on the feedback of the Usability Testing additional features such as a search bar and an archive. Doing so allows users to search for certain articles, that are older. Moreover an implementation of an order into the layout of the website potentially leads to an improved experience for the user. Here the hierarchy can be based on importance of an article or publishing dates. This allows the user to find content that they want to read more easily. To that end the usability test showed that details such as a scroll indicator or better choice of colour can enhance the overall experience. Hence a more detailed focused implementation should be regarded. Furthermore in regard to the linguistic techniques implemented into the system the additional implementation of Probability Context Free Grammars, allows to enhance the algorithm that distinguishes between real and fake news, as stated by [33]. Hence the implementation of such should be regarded in future work. In terms of the corpus it is possible to say that such can be enhanced by cleaning the data. Here articles such as sport articles, TV shows articles etcetera, can be removed from the corpus. This is due to the fact that the parsed articles from the fake news websites contain political information, where the parsed articles from real news websites provided various categories of articles such as sport and TV shows. This could have had an influence on the training of the classifier and if corrected potentially lead to and improved outcome.

Appendices

Appendix A

Website

A.0.1 HTML

```
<!DOCTYPE html>
<html>

<head>
  <meta charset="utf-8">
  <title>Creat|e</title>
  <link rel="stylesheet" href="static/assets/css/main.css">
  <!--jquery-->
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>
  <script src="static/scripts/script.js"></script>

  <!--PageScroller-->
  <script type="text/javascript" src="static/scripts/jquery.fullPage.js"></script>
  <link rel="stylesheet" type="text/css" href="static/assets/css/jquery.fullPage.css">

  <!-- dotdotdot -->
  <script src="static/scripts/jquery.dotdotdot.min.js"></script>


</head>

<body>
  <header>

    <div class="bannerMenu">
      <h1>CreaT|e_
            <span></span>
          </h1>
      <!-- <ul>
                <a href="#">
                    <li>Youtube_</li>
                </a>
                <a href="#">
                    <li>Twitter_</li>
                </a>
                <a href="#">
                    <li>Facebook_</li>
                </a>
                <a href="#">
                    <li>E-Mail_</li>
                </a>
            </ul>-->
      <h4>{{ today.strftime('%A,')}}</h4>
      <h4>{{ today.strftime('%B %d, %Y')}}</h4>

    </div>

    <!--NAVIGATION-->

    <nav role="navigation" id="menufixed">
      <div id="menuToggle">
        <input type="checkbox" />
```

```
      <!--Burger-->
      <span></span>
      <span></span>
      <span></span>

      <ul class="menu">
        <div>

          <a href="#">
            <li class="pad">Local</li>
          </a>
          <a href="#">
            <li>Politics</li>
          </a>
          <a href="#">
            <li>Sports</li>
          </a>
          <a href="#">
            <li>Technology</li>
          </a>
        </div>
      </ul>
    </div>
  </nav>


</header>
<!--Main-->
<!--Page 1-->

<div id="fullpage">

  <div class="section active ">

    <div class="row ">

      <!--   -->
      <div class="col-1-3">

        <div class="card1">

          {% for j in dg %} {% if j == 1 %}
          <img src={{di.ix[j]}} alt="pic">
          <div class="wrapper">
            <a class='over' href={{link.ix[j]}} target="_blank"><h3>{{ db.ix[j] }}</h3></a>
          </div>
          <h5>Probility of being Fake: {{pred[j]}} % </h5>
          <h6>{{date.ix[j].strftime('%B %d, %Y')}} </h6>
        </div>


      </div>
      <div class="col-1-3">
        <div class="card2">

          {% elif j == 2 %}
          <img src={{di.ix[j]}} alt="pic">
          <div class="wrapper">
            <a class='over' href={{link.ix[j]}} target="_blank"><h3>{{ db.ix[j] }}</h3></a>
          </div>
          <h5>Probility of being Fake: {{pred[j]}} %</h5>
          <h6>{{date.ix[j].strftime('%B %d, %Y')}} </h6>




        </div>
      </div>

      <div class="col-1-3">

        <div class="card3">

          {% elif j == 3 %}
          <img src={{di.ix[j]}} alt="pic">
          <div class="wrapper">
            <a class='over' href={{link.ix[j]}} target="_blank"><h3>{{ db.ix[j] }}</h3></a>
```

```
        </div>

        <h5>Probility of being Fake: {{pred[j]}} %</h5>
        <h6>{{date.ix[j].strftime('%B %d, %Y')}} </h6>
        {% endif %} {% endfor %}

      </div>

    </div>

  </div>

  <div class="row ">

    <!--   -->
    <div class="col-1-3">

      <div class="card4">

        {% for j in dg %} {% if j == 4 %}
        <img src={{di.ix[j]}} alt="pic">
        <div class="wrapper">
          <a class='over' href={{link.ix[j]}} target="_blank"><h3>{{ db.ix[j] }}</h3></a>
        </div>
        <h5>Probility of being Fake: {{pred[j]}}% </h5>
        <h6>{{date.ix[j].strftime('%B %d, %Y')}} </h6>

      </div>

    </div>
    <div class="col-1-3">
      <div class="card5">

        {% elif j == 5 %}
        <img src={{di.ix[j]}} alt="pic">
        <div class="wrapper">
          <a class='over' href={{link.ix[j]}} target="_blank"><h3>{{ db.ix[j] }}</h3></a>
        </div>
        <h5>Probility of being Fake {{pred[j]}} %</h5>
        <h6>{{date.ix[j].strftime('%B %d, %Y')}} </h6>


      </div>
    </div>

    <div class="col-1-3">
      <div class="card6">

        {% elif j == 6 %}
        <img src={{di.ix[j]}} alt="pic">
        <div class="wrapper">
          <a class='over' href={{link.ix[j]}} target="_blank"><h3>{{ db.ix[j] }}</h3></a>
        </div>
        <h5>Probility of being Fake {{pred[j]}} %</h5>
        <h6>{{date.ix[j].strftime('%B %d, %Y')}} </h6>
        {% endif %} {% endfor %}

      </div>
    </div>

  </div>


</div>

<!--Page 2-->

<div class="section ">

  <div class="row ">

    <!--   -->
    <div class="col-1-3">
      <div class="card1">

        {% for j in dg %} {% if j == 7 %}
        <img src={{di.ix[j]}} alt="pic">
        <div class="wrapper">
```

```
        <a class='over' href={{link.ix[j]}} target="_blank"><h3>{{ db.ix[j] }}</h3></a>
      </div>
      <h5>Probility of being Fake: {{pred[j]}} % </h5>
      <h6>{{date.ix[j].strftime('%B %d, %Y')}} </h6>


    </div>
  </div>
  <div class="col-1-3">
    <div class="card2">

      {% elif j == 8 %}
      <img src={{di.ix[j]}} alt="pic">
      <div class="wrapper">
        <a class='over' href={{link.ix[j]}} target="_blank"><h3>{{ db.ix[j] }}</h3></a>
      </div>
      <h5>Probility of being Fake: {{pred[j]}} %</h5>

      <h6>{{date.ix[j].strftime('%B %d, %Y')}} </h6>




    </div>
  </div>

  <div class="col-1-3">

    <div class="card3">

      {% elif j == 9 %}
      <img src={{di.ix[j]}} alt="pic">
      <div class="wrapper">
        <a class='over' href={{link.ix[j]}} target="_blank"><h3>{{ db.ix[j] }}</h3></a>
      </div>

      <h5>Probility of being Fake: {{pred[j]}} %</h5>
      <h6>{{date.ix[j].strftime('%B %d, %Y')}} </h6>
       {% endif %} {% endfor %}

    </div>

  </div>

</div>

<div class="row ">

  <!--   -->
  <div class="col-1-3">

    <div class="card4">

      {% for j in dg %} {% if j == 10 %}
      <img src={{di.ix[j]}} alt="pic">
      <div class="wrapper">
        <a class='over' href={{link.ix[j]}} target="_blank"><h3>{{ db.ix[j] }}</h3></a>
      </div>
      <h5>Probility of being a fake Article: {{pred[j]}}% </h5>
      <h6>{{date.ix[j].strftime('%B %d, %Y')}} </h6>

    </div>

  </div>
  <div class="col-1-3">
    <div class="card5">

      {% elif j == 11 %}
      <img src={{di.ix[j]}} alt="pic">
      <div class="wrapper">
        <a class='over' href={{link.ix[j]}} target="_blank"><h3>{{ db.ix[j] }}</h3></a>
      </div>
      <h5>Probility of being Fake {{pred[j]}} %</h5>
      <h6>{{date.ix[j].strftime('%B %d, %Y')}} </h6>


    </div>
  </div>
```

```
    <div class="col-1-3">
      <div class="card6">

        {% elif j == 12 %}
        <img src={{di.ix[j]}} alt="pic">
        <div class="wrapper">
          <a class='over' href={{link.ix[j]}} target="_blank"><h3>{{ db.ix[j] }}</h3></a>
        </div>
        <h5>Probility of being Fake {{pred[j]}} %</h5>
        <h6>{{date.ix[j].strftime('%B %d, %Y')}} </h6>
         {% endif %} {% endfor %}

      </div>
    </div>

  </div>


</div>
<!--Page 3-->

<div class="section ">

  <div class="row ">

    <!--   -->
    <div class="col-1-3">
      <div class="card1">

        {% for j in dg %} {% if j == 13 %}
        <img src={{di.ix[j]}} alt="pic">
        <div class="wrapper">
          <a class='over' href={{link.ix[j]}} target="_blank"><h3>{{ db.ix[j] }}</h3></a>
        </div>
        <h5>Probility of being Fake: {{pred[j]}} % </h5>

        <h6>{{date.ix[j].strftime('%B %d, %Y')}} </h6>

      </div>
    </div>
    <div class="col-1-3">
      <div class="card2">

        {% elif j == 14 %}
        <img src={{di.ix[j]}} alt="pic">
        <div class="wrapper">
          <a class='over' href={{link.ix[j]}} target="_blank"><h3>{{ db.ix[j] }}</h3></a>
        </div>
        <h5>Probility of being Fake: {{pred[j]}} %</h5>

        <h6>{{date.ix[j].strftime('%B %d, %Y')}} </h6>

      </div>
    </div>

    <div class="col-1-3">

      <div class="card3">

        {% elif j == 15 %}
        <img src={{di.ix[j]}} alt="pic">
        <div class="wrapper">
          <a class='over' href={{link.ix[j]}} target="_blank"><h3>{{ db.ix[j] }}</h3></a>
        </div>

        <h5>Probility of being Fake: {{pred[j]}} %</h5>
        <h6>{{date.ix[j].strftime('%B %d, %Y')}} </h6>
        {% endif %} {% endfor %}

      </div>

    </div>

  </div>

  <div class="row ">
```

```
    <!--   -->
    <div class="col-1-3">

      <div class="card4">

        {% for j in dg %} {% if j == 16 %}
        <img src={{di.ix[j]}} alt="pic">
        <div class="wrapper">
          <a class='over' href={{link.ix[j]}} target="_blank"><h3>{{ db.ix[j] }}</h3></a>
        </div>
        <h5>Probility of being a fake Article: {{pred[j]}}% </h5>
        <h6>{{date.ix[j].strftime('%B %d, %Y')}} </h6>

      </div>

    </div>
    <div class="col-1-3">
      <div class="card5">

        {% elif j == 17 %}
        <img src={{di.ix[j]}} alt="pic">
        <div class="wrapper">
          <a class='over' href={{link.ix[j]}} target="_blank"><h3>{{ db.ix[j] }}</h3></a>
        </div>
        <h5>Probility of being Fake {{pred[j]}} %</h5>

        <h6>{{date.ix[j].strftime('%B %d, %Y')}} </h6>

      </div>
    </div>

    <div class="col-1-3">
      <div class="card6">

        {% elif j == 18 %}
        <img src={{di.ix[j]}} alt="pic">
        <div class="wrapper">
          <a class='over' href={{link.ix[j]}} target="_blank"><h3>{{ db.ix[j] }}</h3></a>
        </div>
        <h5>Probility of being Fake {{pred[j]}} %</h5>

        <h6>{{date.ix[j].strftime('%B %d, %Y')}} </h6>
        {% endif %} {% endfor %}

      </div>
    </div>

  </div>


</div>

<!--Footer-->

<div class="section␣">
  <!--Footer-->
  <div class="footer">
    <div class="row">
      <!-- <div class="col-1-2">
                  <h3 class="about">about_</h3>
                  <ul class="about_List">
                      <a href="#">
                          <li>Contact Us</li>
                      </a>

                  </ul>
              </div> -->

      <div class="col-1-2">
        <h3 class="social">social_</h3>
        <ul class="social_List">


          <li>Twitter</li>


          <li>Facebook</li>
```

```
                <li>E-Mail</li>

              </ul>
            </div>
          </div>
        </div>
      </div>
    </div>

</body>

</html>
```

A.0.2 CSS

```
    /* http://meyerweb.com/eric/tools/css/reset/
   v2.0 | 20110126
   License: none (public domain)
*/

html, body, div, span, applet, object, iframe, h1, h2, h3, h4, h5, h6, p, blockquote, pre, a, abbr, acronym
     , address, big, cite, code, del, dfn, em, img, ins, kbd, q, s, samp, small, strike, strong, sub, sup,
     tt, var, b, u, i, center, dl, dt, dd, ol, ul, li, fieldset, form, label, legend, table, caption, tbody
     , tfoot, thead, tr, th, td, article, aside, canvas, details, embed, figure, figcaption, footer, header
     , hgroup, menu, nav, output, ruby, section, summary, time, mark, audio, video {
      margin: 0;
      padding: 0;
      border: 0;
      font-size: 100%;
      font: inherit;
      vertical-align: baseline;
    }
    /* HTML5 display-role reset for older browsers */

    article, aside, details, figcaption, figure, footer, header, hgroup, menu, nav, section {
      display: block;
    }

    body {
      line-height: 1;
    }

    ol, ul {
      list-style: none;
    }

    blockquote, q {
      quotes: none;
    }

    blockquote:before, blockquote:after, q:before, q:after {
      content: '';
      content: none;
    }

    table {
      border-collapse: collapse;
      border-spacing: 0;
    }
    /* resets */

    *, *:before, *:after {
      box-sizing: border-box;
    }

    .clearfix:after {
      content: "";
      display: table;
      clear: both;
    }
    /*
================================================
Load Font
================================================
*/
```

```
    @font-face {
      font-family: 'infinityregular';
      src: url('infinity-webfont.woff2') format('woff2'), url('infinity-webfont.woff') format('woff');
      font-weight: normal;
      font-style: normal;
    }

    @font-face {
      font-family: 'aileronbold';
      src: url('aileron-bold-webfont.woff2') format('woff2'), url('aileron-bold-webfont.woff') format('woff
          ');
      font-weight: normal;
      font-style: normal;
    }

    @font-face {
      font-family: 'bronkoh-boldbold';
      src: url('bronkoh-bold-webfont.woff2') format('woff2'), url('bronkoh-bold-webfont.woff') format('woff
          ');
      font-weight: normal;
      font-style: normal;
    }

    @font-face {
      font-family: 'bronkoh-regularregular';
      src: url('bronkoh-regular-webfont.woff2') format('woff2'), url('bronkoh-regular-webfont.woff') format
          ('woff');
      font-weight: normal;
      font-style: normal;
    }

    @font-face {
      font-family: 'bronkoh-extralightregular';
      src: url('bronkoh-extralight-webfont.woff2') format('woff2'), url('bronkoh-extralight-webfont.woff')
          format('woff');
      font-weight: normal;
      font-style: normal;
    }
    /*
================================================
body
================================================
*/

    body {
      margin: 0;
      padding: 0;
      background: #F6F7F8;
      height: 100%;
      max-width: 100%;
    }

    header {
      z-index: 1
    }
    /*
================================================
menu
================================================
*/

    #menufixed {
      position: fixed;
      z-index: 2;
    }

    #menuToggle {
      /*display: block;*/
      position: relative;
      z-index: 1;
      -webkit-user-select: none;
      user-select: none;
    }

    #menuToggle input {
      /*display: block;*/
      width: 33px;
      height: 32px;
```

```
    position: absolute;
    top: -60px;
    left: 5px;
    cursor: pointer;
    opacity: 0;
    z-index: 2;
    -webkit-touch-callout: none;
  }

  #menuToggle span {
    display: block;
    width: 33px;
    height: 4px;
    top: -50px;
    left: 8px;
    margin-bottom: 5px;
    position: relative;
    background: #31373E;
    border-radius: 3px;
    z-index: 1;
    transform-origin: 4px 0px;
    transition: transform 0.5s cubic-bezier(0.77, 0.2, 0.05, 1.0), background 0.5s cubic-bezier(0.77,
        0.2, 0.05, 1.0), opacity 0.55s ease;
  }

  #menuToggle span:first-child {
    transform-origin: 0% 0%;
  }

  #menuToggle span:nth-last-child(2) {
    transform-origin: 0% 100%;
  }
  /*
* Transform all the slices of hamburger
* into a crossmark.
*/

  #menuToggle input:checked~span {
    opacity: 1;
    transform: rotate(45deg) translate(-2px, -1px);
    background: #E8E8E8;
  }
  /*
* But let's hide the middle one.
*/

  #menuToggle input:checked~span:nth-last-child(3) {
    opacity: 0;
    transform: rotate(0deg) scale(0.2, 0.2);
  }
  /*
* Ohyeah and the last one should go the other direction
*/

  #menuToggle input:checked~span:nth-last-child(2) {
    opacity: 1;
    transform: rotate(-45deg) translate(0, -1px);
  }
  /*
* Make this absolute positioned
* at the top left of the screen
*/

  .menu {
    position: absolute;
    width: 35vw;
    height: 100vh;
    top: -12vh;
    margin: 0 0 0 -50px;
    padding-right: 13vw;
    background: #2962ff;
    list-style-type: none;
    -webkit-font-smoothing: antialiased;
    /* to stop flickering of text in safari */
    transform-origin: 0% 0%;
    transform: translate(-100%, 0);
    transition: transform 600ms cubic-bezier(0.68, -0.45, 0.265, 1.45);
  }
```

```
    .menu a {
      text-decoration: none;
      font-family: 'infinityregular';
      color: #FAFAFF;
      text-align: left;
      transition: all 300ms ease-out;
    }

    .menu a:hover {
      color: black;
      transition: all 300ms ease-out;
    }

    .menu li {
      margin: 0 0 0 7vw;
      margin-left: 150px;
      font-size: 4vw;
      margin-top: 12vh;
    }

    .pad {
      padding-top: 6vh;
      display: inline-Block;
    }
    /*
 * And let's fade it in from the left
 */

    #menuToggle input:checked~ul {
      margin-left: -50px;
      transform: scale(1.0, 1.0);
      opacity: 1;
    }
    /*
=============================================
bannerMenu
=============================================
*/

    .bannerMenu h1 {
      font-family: 'infinityregular';
      color: #232528;
      text-align: center;
      font-size: 60px;
      position: absolute;
      margin-left: 45%;
    }

    .bannerMenu {
      position: ;
      width: 100%;
      height: 12vh;
      background-color: #FAFAFF;
      text-align: right;
      padding: 10px 0 10px 0;
    }

    .bannerMenu h4 {
      font-family: 'infinityregular';
      color: #232528;
      text-align: right;
      font-size: 30px;
      position: relative;
      padding-right: 10px;
    }

    .bannerMenu span {
      position: absolute;
      width: 110%;
      border-top: 2px solid #FF6978;
      opacity: 1;
      border-radius: 5px;
      left: -1vw;
      top: 55%;
    }

    .bannerMenu a {
```

```
    text-decoration: none;
    font-family: 'infinityregular';
    color: #232528;
    position: inherit;
    font-size: 4vh;
  }

  .bannerMenu li::before, .bannerMenu li::after {
    content: '';
    position: absolute;
    width: 0%;
    height: 1px;
    top: 50%;
    margin-top: -0.5px;
    background: #FF6978;
  }

  .bannerMenu li::after {
    right: 0px;
    background: #FF6978;
    transition: width 0.8s cubic-bezier(0.22, 0.61, 0.36, 1);
  }

  .bannerMenu li:hover:before {
    background: #FF6978;
    width: 110%;
    left: -0.5vw;
    transition: width 0.5s cubic-bezier(0.22, 0.61, 0.36, 1);
  }

  .bannerMenu li:hover:after {
    background: transparent;
    width: 110%;
    left: -0.5vw;
    transition: 0s;
  }

  .bannerMenu li {
    position: relative;
    margin-right: 10px;
    display: inline-block;
    margin-top: 1.5%;
    margin-right: 1.2%;
  }
  /*
==========================================
Grid
==========================================
*/

  .row {
    display: flex;
    flex-flow: row wrap;
    margin: -10px 10px;
    margin-bottom: 0px;
  }

  .row:last-child {
    margin-bottom: 0;
  }

  [class*="col-"] {
    padding: 10px;
    width: 100%;
  }

  @media all and (min-width:600px) {
    /*set col widths */
    .col-1-3 {
      width: 33.33%
    }
    .col-1-4 {
      width: 25%
    }
  }
  /*
==========================================
Page 1 Cards
```

```
==========================================
*/

    img {
      filter: brightness(30%);
      z-index: -1;
    }

    a {
      text-decoration: none;
    }

    h3 {
      font-family: 'bronkoh-regularregular';
      font-size: 1.8vw;
      padding: 2% 1% 0px 3%;
      color: black;
      /*display: inline-block;*/
      text-align: left;
    }

    .over h3 {
      position: relative;
      top: -250px;
      z-index: 3;
      color: white;
      left: 0

    }

    h5 {
      position: relative;
      font-family: 'bronkoh-regularregular';
      font-size: 1.2vw;
      padding: 2% 1% 0px 3%;
      color: white;
      /*display: inline-block;*/
      text-align: left;
      text-overflow: ellipsis;
      margin-top: -170px;
      z-index: 4
    }

    h6 {
      position: relative;
      font-family: 'bronkoh-regularregular';
      font-size: 1vw;
      padding-right: 3%;
      bottom: -75px;
      color: #455A64;
      /*display: inline-block;*/
      text-align: right;
      text-overflow: ellipsis;
      margin-top: -90px;
      z-index: 4
    }

    .wrapper {
      width: 100%;
      height: 55%
    }

    .card1 {
      width: 100%;
      height: 40vh;
      /*background: rgba(55,63,81, 0.8) ;*/
      border-radius: 5px;
      margin-left: 3vw;
      margin-top: -60px;
      box-shadow: 0 1px 3px rgba(0, 0, 0, 0.2), 0 1px 2px rgba(0, 0, 0, 0.24);
      /*transition: all 0.3s cubic-bezier(.25, .8, .25, 1);*/
    }

    .card1 img {
      padding: 1% 1% 1% 1%;
      width: 100%;
      height: 100%;
      border-radius: 10px;
```

```
}

.card2 {
  width: 100%;
  height: 40vh;
  /*background: #373F51;*/
  border-radius: 5px;
  margin-left: 3vw;
  margin-top: -60px;
  box-shadow: 0 1px 3px rgba(0, 0, 0, 0.2), 0 1px 2px rgba(0, 0, 0, 0.24);
  transition: all 0.3s cubic-bezier(.25, .8, .25, 1);
}

.card2 img {
  padding: 1% 1% 1% 1%;
  width: 100%;
  height: 100%;
  border-radius: 10px;
}

.card3 {
  width: 90%;
  height: 40vh;
  /*background: #ff1744;*/
  border-radius: 5px;
  margin-left: 3vw;
  margin-top: -60px;
  box-shadow: 0 1px 3px rgba(0, 0, 0, 0.2), 0 1px 2px rgba(0, 0, 0, 0.24);
  transition: all 0.3s cubic-bezier(.25, .8, .25, 1);
}

.card3 img {
  padding: 1% 1% 1% 1%;
  width: 100%;
  height: 100%;
  border-radius: 10px;
}

.card4 {
  width: 100%;
  height: 40vh;
  /*background: #FB3640;*/
  border-radius: 5px;
  margin-left: 3vw;
  margin-top: 0vh;
  box-shadow: 0 1px 3px rgba(0, 0, 0, 0.2), 0 1px 2px rgba(0, 0, 0, 0.24);
  transition: all 0.3s cubic-bezier(.25, .8, .25, 1);
}

.card4 img {
  padding: 1% 1% 1% 1%;
  width: 100%;
  height: 100%;
  border-radius: 10px;
}

.card5 {
  width: 100%;
  height: 40vh;
  /*background: #f48e7e;*/
  border-radius: 5px;
  margin-left: 3vw;
  margin-top: 0vh;
  box-shadow: 0 1px 3px rgba(0, 0, 0, 0.2), 0 1px 2px rgba(0, 0, 0, 0.24);
  transition: all 0.3s cubic-bezier(.25, .8, .25, 1);
}

.card5 img {
  padding: 1% 1% 1% 1%;
  width: 100%;
  height: 100%;
  border-radius: 10px;
}

.card6 {
  width: 90%;
  height: 40vh;
  /*background: #272727;*/
```

```
    border-radius: 5px;
    margin-left: 3vw;
    box-shadow: 0 1px 3px rgba(0, 0, 0, 0.2), 0 1px 2px rgba(0, 0, 0, 0.24);
    transition: all 0.3s cubic-bezier(.25, .8, .25, 1);
  }

  .card6 img {
    padding: 1% 1% 1% 1%;
    width: 100%;
    height: 100%;
    border-radius: 10px;
  }
  /*
================================================
Page 2 Cards
================================================
*/
  /*
================================================
footer
================================================
*/

  .footer {
    height: 100%;
    width: 100%;
    background-color: rgba(25, 25, 25, 1);
  }

  .about, .about_List li {
    position: absolute;
    font-family: 'infinityregular';
    color: #fafafa;
    margin-top: 7vh;
    text-align: left;
    margin-left: 10vw;
    font-size: 2vw
  }

  .about {
    font-size: 5vw;
    /*margin-right: 10vw;*/
  }

  .about_List li {
    font-size: 2vw;
    text-decoration: none;
  }

  .about_List li {
    margin-top: 5%;
  }

  .social, .social_List li {
    font-family: 'infinityregular';
    color: #fafafa;
    margin-top: 7vh;
    text-align: center;
    margin-right: 5vw;
    font-size: 2vw
  }

  .social {
    text-align: center;
    font-size: 5vw;
    /*margin-right: 10vw;*/
  }

  .social_List a {
    font-size: 2vw;
    text-decoration: none;
  }

  .social_List li {
    margin-top: 5%;
    line-height: 3%;
  }
```

A.0.3 JavaScript/jQuery

```
$(document).ready(function() {
    $('#fullpage').fullpage();
    $(".wrapper").dotdotdot({
                //      configuration goes here
        ellipsis        : '... ',
        wrap            : 'word',
        height          : null,
        fallbackToLetter: true,
        remove          : [ ' ', ',', ';', '.', '!', '?' ]

        })

      //        by using the return-value...
        var isTruncated = $("#wrapper").triggerHandler("isTruncated");
        if ( isTruncated ) {
                //      do something

        }


});
```

Appendix B

Corpus Creation

B.0.1 Beautiful Soup 4: Extract of Terminal Output

Share this with
Email
Facebook
Messenger
Messenger
Twitter
Pinterest
WhatsApp
LinkedIn
Copy this link
These are external links and will open in a new window The man who carried out a suicide attack in Manchester was "likely" to have not acted alone, Home Secretary Amber Rudd says. Salman Abedi killed 22 and injured 64 when he blew himself up at the Manchester Arena on Monday night - 20 people are in critical care. The UK terror threat level is now up to its highest level of "critical", meaning more attacks may be imminent.[...] Terms and conditions The BBC is not responsible for the content of external Internet sites Home Secretary Amber Rudd says the Manchester bomber was known to the security services.

B.0.2 List of News Sources

```
https://www.washingtonpost.com/
http://www.npr.org/
http://www.pbs.org
https://www.wsj.com/
https://www.theguardian.com/
https://www.bloomberg.com/
```

```
http://edition.cnn.com/
http://www.economist.com/
http://www.bbc.com
http://www.reuters.com
```

Listing B.1 List of Reputational & Trusted News Sources

```
http://beforeitsnews.com/
http://www.zerohedge.com/
http://www.americanlookout.com
http://www.washingtonexaminer.com/
https://www.infowars.com/
http://www.redflagnews.com/
http://www.conservativespirit.com/
https://makeamericagreattoday.com/
https://www.politicalcult.com/
http://www.realtimepolitics.com/
```

Listing B.2 List of Fake News Sources

Appendix C

NLP & Classifier Training

C.0.1 Beautiful Soup

```
import bs4 as bs
import urllib.request
data = ['http://www.bbc.com/news/uk-40023488', 'http://www.bbc.com/news/
    live/uk-england-manchester-40007967']
# BeautifulSoup
for eachLink in data:
    #print(eachLink)
    sauce = urllib.request.urlopen(eachLink).read()
    soup = bs.BeautifulSoup(sauce,"lxml")
    body = soup.body
    ###Print only Text of p-tags
    for eachhead in body.find_all('p'):
        print(eachhead.text)
```

C.0.2 Newspaper

```
import newspaper
import pandas as pd

data = ['https://www.washingtonpost.com/', 'http://www.npr.org/', [...]]

def write():
     for eachWebsite in data:
         web = newspaper.build(eachWebsite, memoize=False)
         appended_data = []
         try:
```

```
            for article in web.articles:
                article.download()
                article.parse()
                article.nlp()
                d = {'LINK': article.url,
                    'TITLE': article.title,
                    'TEXT': article.text,
                    'AUTHOR':article.authors}
                de = appended_data.append(d)
        except Exception as e:
            print('ERROR_READING_DATA')
            print(e)
        #print(appended_data)
        df = pd.DataFrame(appended_data)
        print(df)
write()
```

C.0.3 Natural Language Processing

Method 1: __init__()

```
    def __init__(self, lower=True, strip=True):
        self.lower = lower
        self.strip = strip
        self.stopwords = set(sw.words('english'))
        self.punct = set(string.punctuation)
        self.lemmatizer = WordNetLemmatizer()
```

Method 2: Lemmatization

```
def lemmatize(self, word, tag):
        tag = {
            'N': wn.NOUN,
            'V': wn.VERB,
            'R': wn.ADV,
            'J': wn.ADJ
        }.get(tag[0], wn.NOUN)
        return self.lemmatizer.lemmatize(word, tag)
```

Method 3: tokenize

```
    def tokenize(self, article):
```

```
        for sent in sent_tokenize(article):
            for word, tag in pos_tag(wordpunct_tokenize(sent)):
                word = word.lower() if self.lower else word
                word = word.strip() if self.strip else word
                if word in self.stopwords:
                    continue
                if all(char in self.punct for char in word):
                    continue
                lemma = self.lemmatize(word, tag)
                yield lemma
```

C.0.4 Flask:Connection to Website

```
import pickle
import pandas as pd
import string
import datetime as dt
from flask import Flask, render_template, request
import random

import numpy as np

from sklearn.metrics import classification_report

from nltk.corpus import stopwords as sw
from nltk.corpus import wordnet as wn
from nltk import WordNetLemmatizer
from nltk import sent_tokenize
from nltk import wordpunct_tokenize
from nltk import pos_tag

from sklearn.preprocessing import LabelEncoder

"""PreProcessing the Data"""

class PreProcessing():
    def __init__(self, lower=True, strip=True):
        self.lower = lower
        self.strip = strip
```

```
        self.stopwords = set(sw.words('english'))
        self.punct = set(string.punctuation)
        self.lemmatizer = WordNetLemmatizer()


    def lemmatize(self, word, tag):
        tag = {
            'N': wn.NOUN,
            'V': wn.VERB,
            'R': wn.ADV,
            'J': wn.ADJ
        }.get(tag[0], wn.NOUN)
        return self.lemmatizer.lemmatize(word, tag)

    def tokenize(self, article):
        for sent in sent_tokenize(article):

            for word, tag in pos_tag(wordpunct_tokenize(sent)):
                word = word.lower() if self.lower else word # Lowercases
                    words
                word = word.strip() if self.strip else word # strips
                   away whitespace at end and beginning

                # if word is stop word igonore & continue
                if word in self.stopwords:
                    continue

                # If punctuation, ignore token and continue
                if all(char in self.punct for char in word):
                    continue

                # Generator
                lemma = self.lemmatize(word, tag)
                yield lemma


def print_full(x):
    pd.set_option('display.max_rows', len(x))
    print(x)
    pd.reset_option('display.max_rows')
```

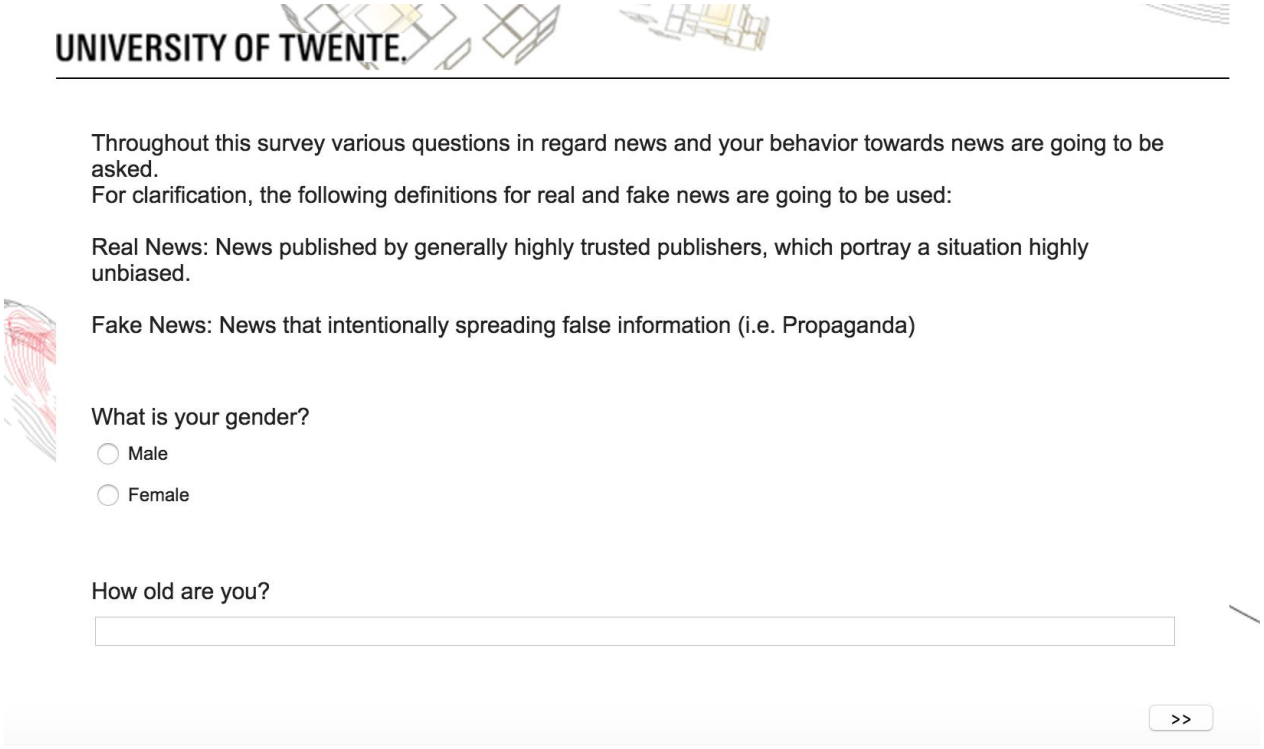
```
svm = pickle.load(open('model.pkl', 'rb'))

data = pd.read_csv('Mixed_Sources',  delimiter="/")
df = pd.DataFrame(data)
df = df.sort_values(by='DATE', ascending=False)
date = pd.to_datetime(df['DATE']).apply(lambda x: x.date())
#date = pd.to_datetime(date)
df['DATE'] = date
#print_full(df['DATE'])
today = dt.datetime.today()
#print(date)

X = df['TEXT'].values.astype('U')
a = df.index.values


predicted = svm.predict(X)
predict = predicted.tolist()
prob = svm.predict_proba(X)

#
# for eachDate in df['DATE']:
#     if eachDate.strftime('%A, %B %d, %Y') == today.date().strftime('%A
    , %B %d, %Y'):
#         print(df['DATE'].ix[0].strftime('%A, %B %d, %Y'))
#         break

# if df['DATE'].ix[13].strftime('%A, %B %d, %Y') == today.date().
    strftime('%A, %B %d, %Y'):
#     print(df['DATE'].ix[12].strftime('%A, %B %d, %Y'))
#
# elif df['DATE'].ix[11 + 1].strftime('%A, %B %d, %Y') == today.date().
    strftime('%A, %B %d, %Y'):
#     print('kaf')



app = Flask(__name__)
```

```
@app.route('/',methods=['GET', 'POST'])
def index():

    return render_template('index3.html', df=predict, dg=a, db=df['TITLE
        '],
                           date=df['DATE'], di=df['IMAGE'], pred=np.
                               around(prob[:, 0] * 100, decimals=2),
                           link=df['LINK'], today=today.date())


# Start the Web Server
if __name__ == '__main__':
    app.run()
```

Appendix D

Survey

D.0.1 Survey


UNIVERSITY OF TWENTE.

Throughout this survey various questions in regard news and your behavior towards news are going to be asked.
For clarification, the following definitions for real and fake news are going to be used:

Real News: News published by generally highly trusted publishers, which portray a situation highly unbiased.

Fake News: News that intentionally spreading false information (i.e. Propaganda)

What is your gender?

Male

Female

How old are you?

>>

UNIVERSITY OF TWENTE.

Are you interested in news?

- Yes
- No
- Occasionally

How often do you check the news?

- Daily
- 4-6 times a week
- 2-3 times a week
- Once a week
- Never

What medium do you use in order to check news?

- Specific News Websites (e.g. telegraaf.nl, nu.nl etc.)
- Social Media (e.g. Facebook, Twitter, Youtube etc.)
- News Aggregation Websites
- Conventional Methods (e.g. Radio, TV, Newspaper)
- None

Do you have trust in the news media ?

- Definitely yes
- Probably yes
- Might or might not
- Probably not
- Definitely not

How many different sources do you check?

- ○ 1
- ○ 2
- ○ 3
- ○ 4 and more
- ○ None

Do you believe that you are capable of distinguishing between fake and real news on your own ?

- ○ Definitely yes
- ○ Probably yes
- ○ Might or might not
- ○ Probably not
- ○ Definitely not

Would you like to use a news aggregation system that can tell you whether an article is fake or not ?

- ○ Yes
- ○ Maybe
- ○ No

Do you believe you could trust such a system ?

- ○ Definitely yes
- ○ Probably yes
- ○ Might or might not
- ○ Probably not
- ○ Definitely not

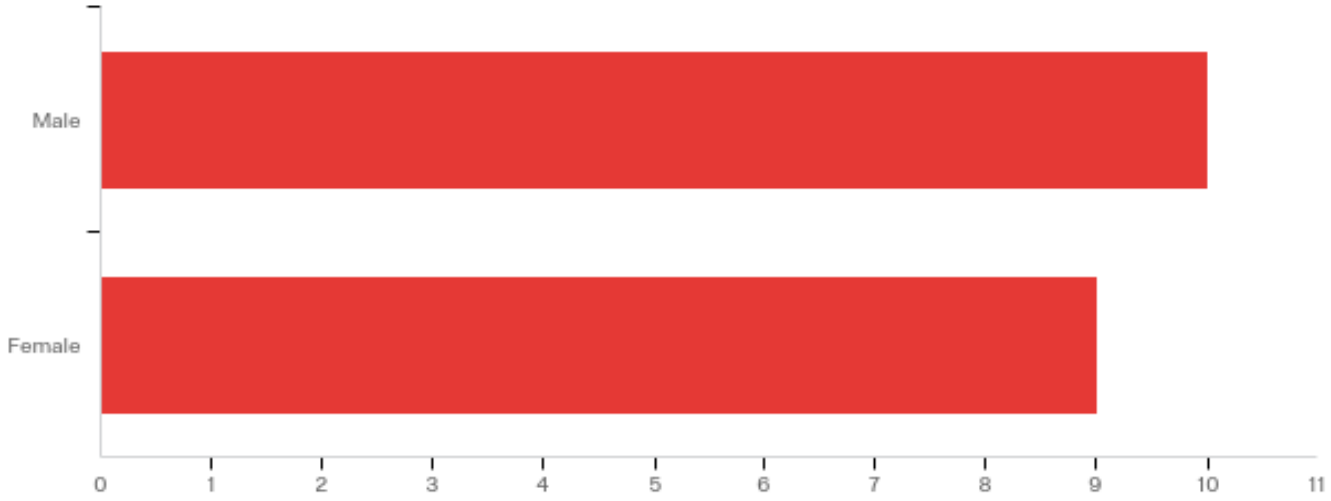
D.0.2 Survey Results

Default Report

Graduation Project

June 27th 2017, 8:12 am MDT

Q2 - What is your gender?

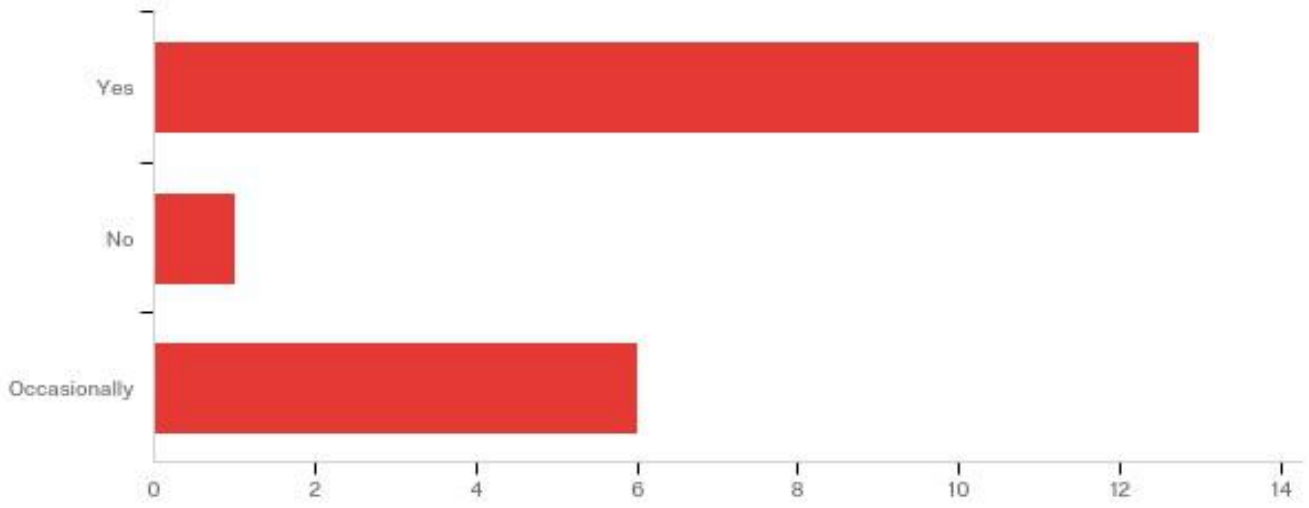


#	Answer	%	Count
1	Male	52.63%	10
2	Female	47.37%	9
	Total	100%	19

Q3 - How old are you?

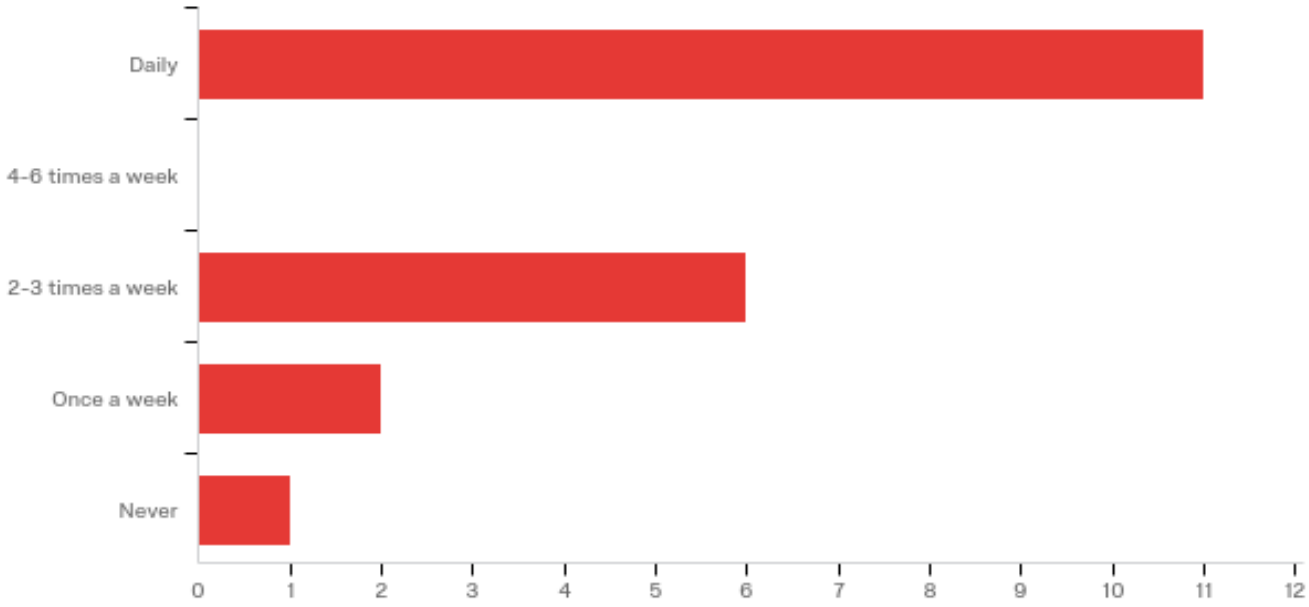
How old are you?
23
23
30
25
25
47
53
26
27
25
25
24
24
23
17
23
21
21
23
21

Q4 - Are you interested in news?



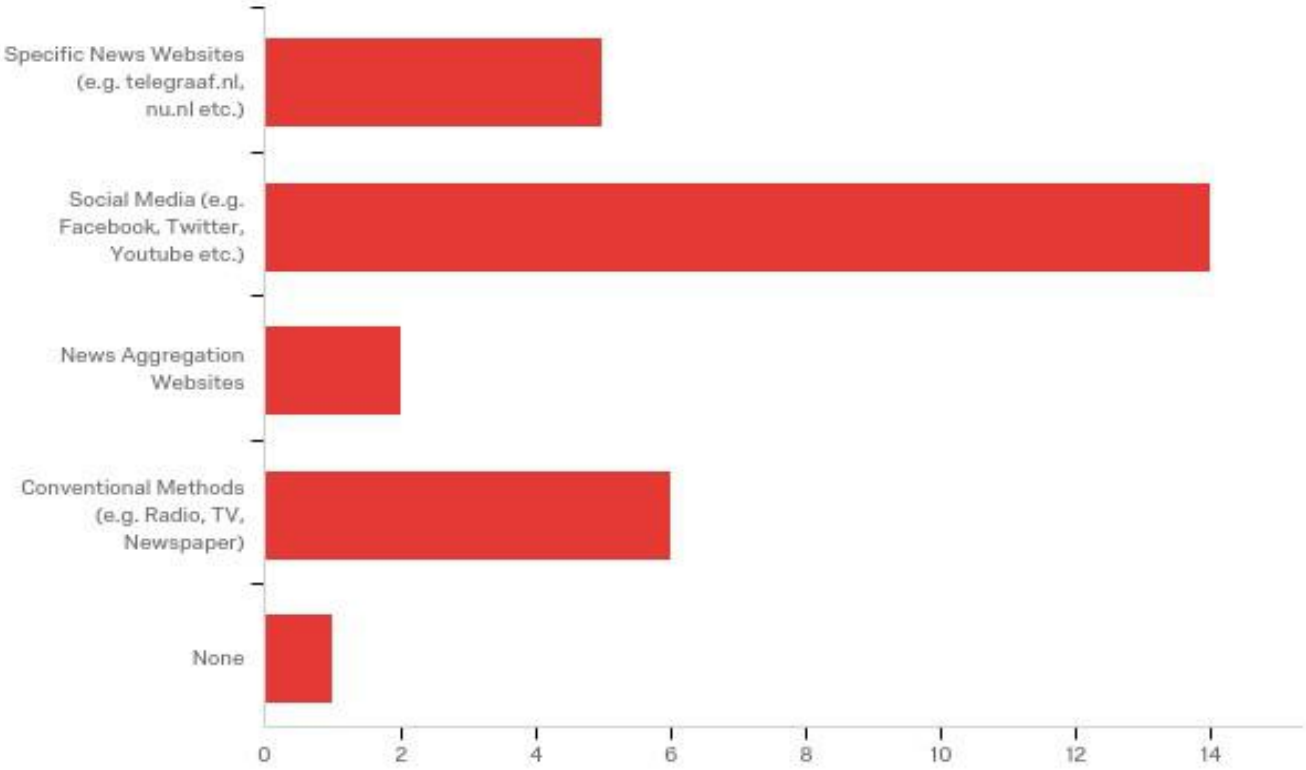
#	Answer	%	Count
1	Yes	65.00%	13
2	No	5.00%	1
3	Occasionally	30.00%	6
	Total	100%	20

Q5 - How often do you check the news?



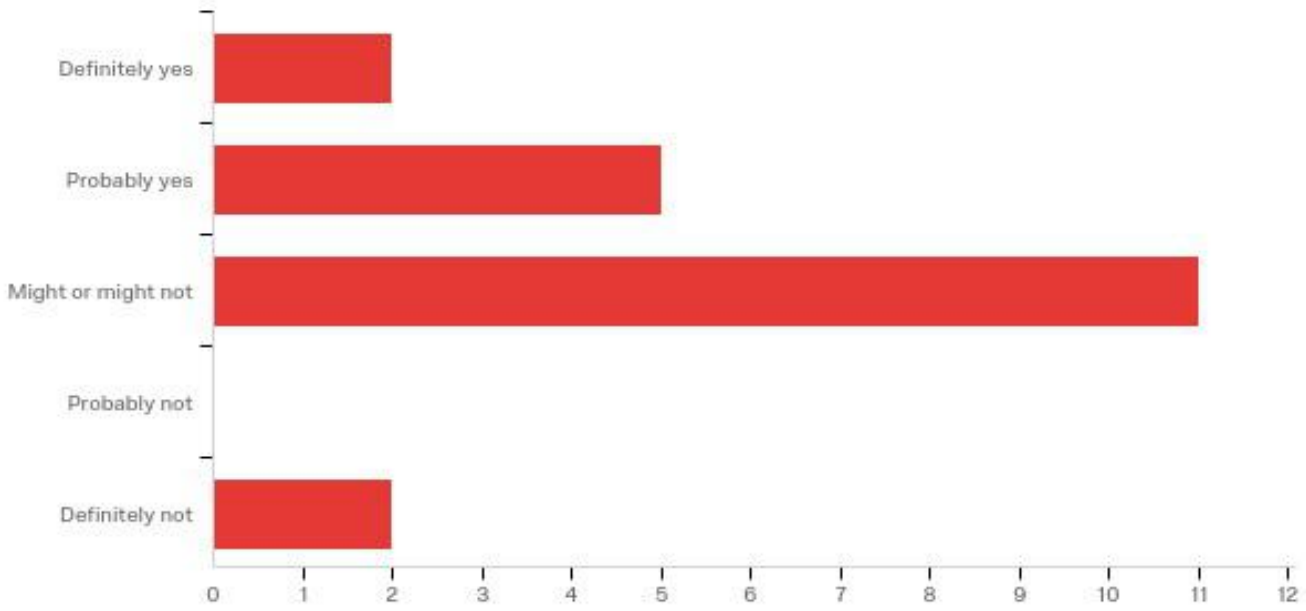
#	Answer	%	Count
1	Daily	55.00%	11
2	4-6 times a week	0.00%	0
3	2-3 times a week	30.00%	6
4	Once a week	10.00%	2
5	Never	5.00%	1
	Total	100%	20

Q6 - What medium do you use in order to check news? (Multiple Answers Possible)



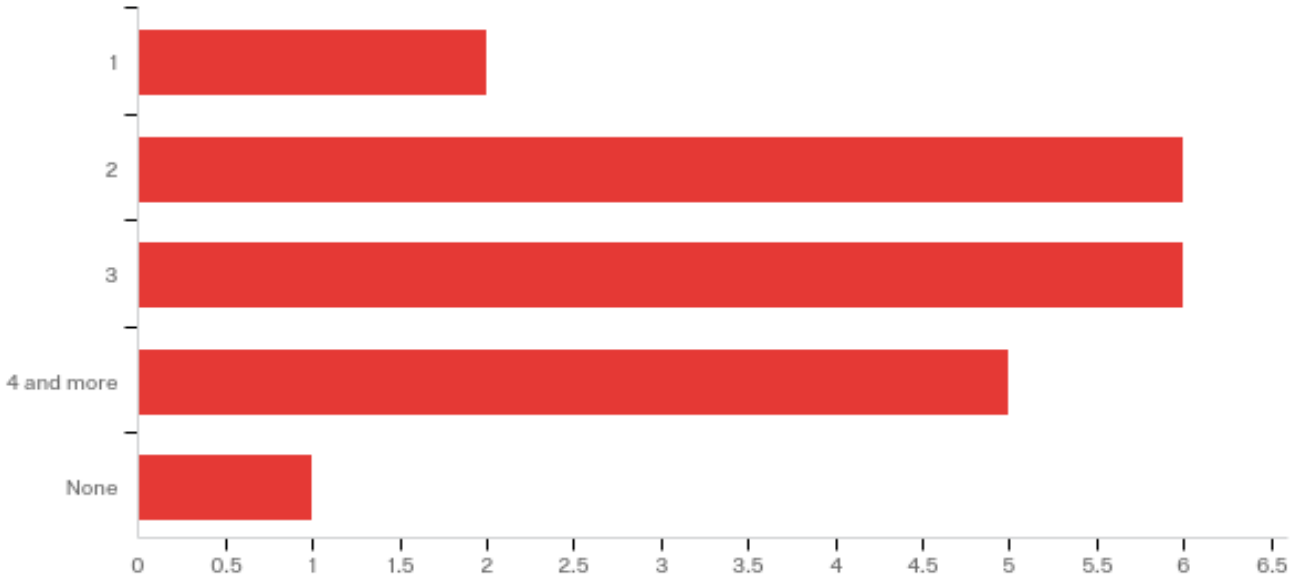
#	Answer	%	Count
1	Specific News Websites (e.g. telegraaf.nl, nu.nl etc.)	17.86%	5
2	Social Media (e.g. Facebook, Twitter, Youtube etc.)	50.00%	14
3	News Aggregation Websites	7.14%	2
4	Conventional Methods (e.g. Radio, TV, Newspaper)	21.43%	6
5	None	3.57%	1
	Total	100%	28

Q7 - Do you have trust in the news media ?



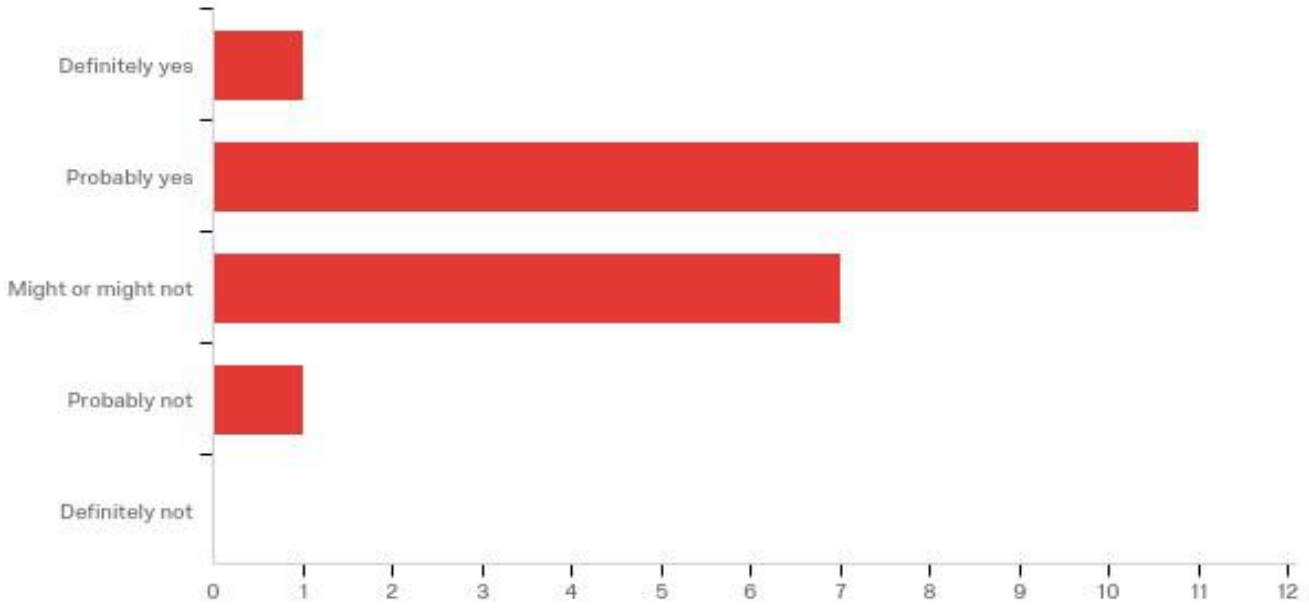
#	Answer	%	Count
1	Definitely yes	10.00%	2
2	Probably yes	25.00%	5
3	Might or might not	55.00%	11
4	Probably not	0.00%	0
5	Definitely not	10.00%	2
	Total	100%	20

Q8 - How many different sources do you check?



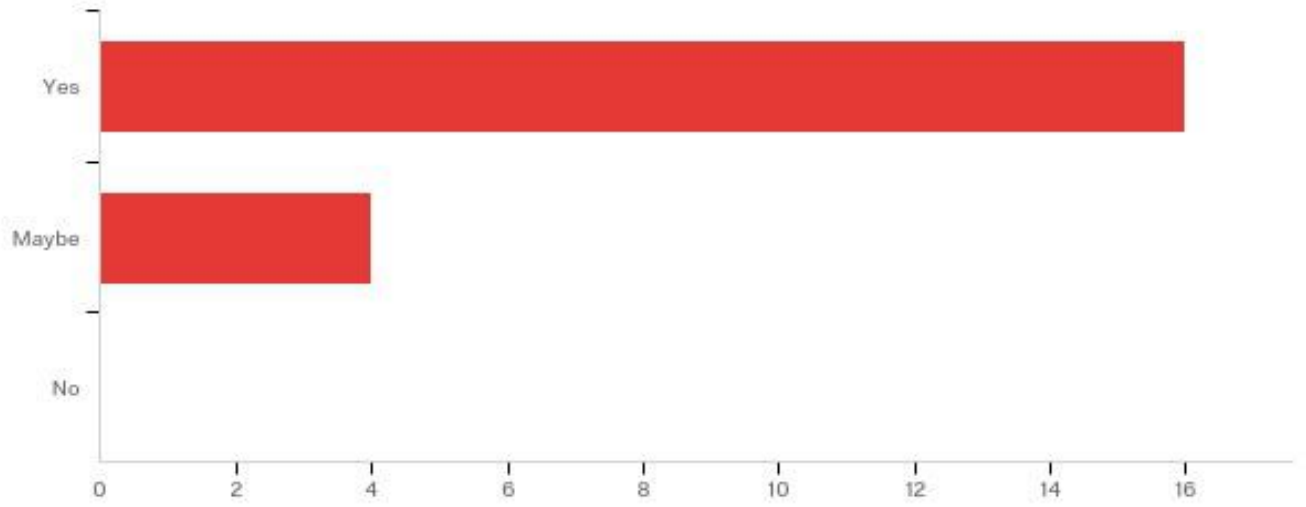
#	Answer	%	Count
1	1	10.00%	2
2	2	30.00%	6
3	3	30.00%	6
4	4 and more	25.00%	5
5	None	5.00%	1
	Total	100%	20

Q9 - Do you believe that you are capable of distinguishing between fake and real news on your own ?



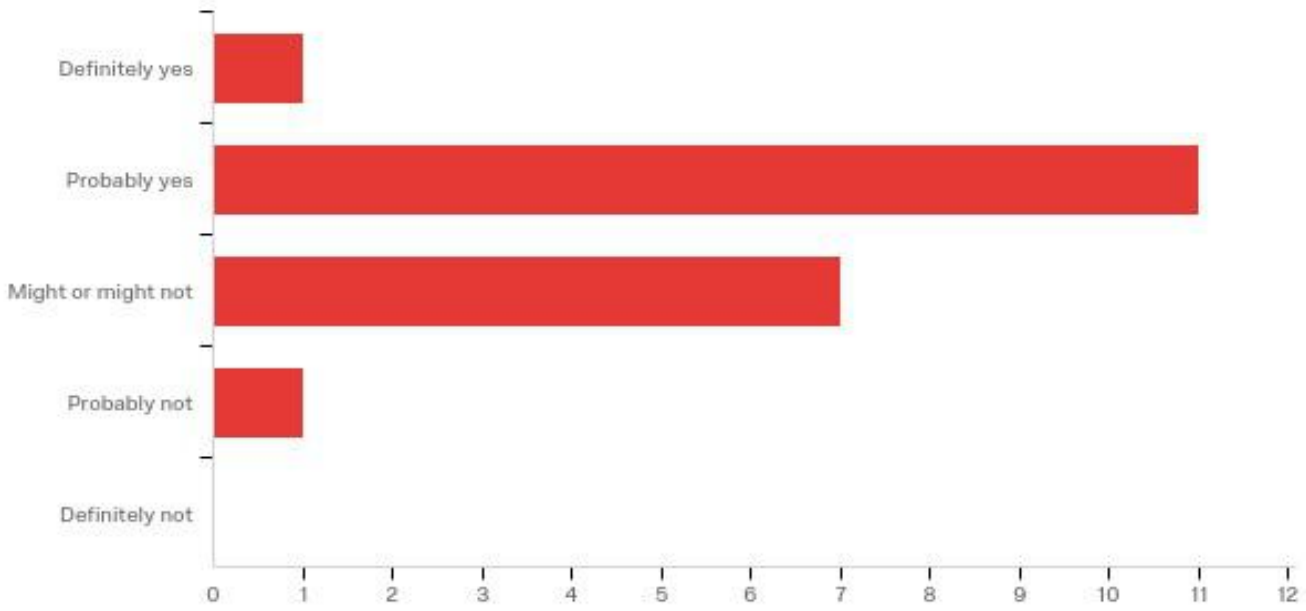
#	Answer	%	Count
1	Definitely yes	5.00%	1
2	Probably yes	55.00%	11
3	Might or might not	35.00%	7
4	Probably not	5.00%	1
5	Definitely not	0.00%	0
	Total	100%	20

Q18 - Would you like to use a news aggregation system that can tell you whether an article is fake or not ?



#	Answer	%	Count
1	Yes	80.00%	16
2	Maybe	20.00%	4
18	No	0.00%	0
	Total	100%	20

Q19 - Do you believe you could trust such a system ?



#	Answer	%	Count
1	Definitely yes	5.00%	1
2	Probably yes	55.00%	11
3	Might or might not	35.00%	7
4	Probably not	5.00%	1
5	Definitely not	0.00%	0
	Total	100%	20

D.0.3 Usability Testing Results

Default Report

Usability Testing

June 29th 2017, 7:45 am MDT

Q1 - Time to complete task:

Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
Task 1 : Find the probability of an article being fake.	3.00	9.00	5.20	2.04	4.16	5
Task 2: Open an article.	2.00	12.00	5.20	3.87	14.96	5
Task 3: Find articles the publish date of an article.	1.00	6.00	3.20	1.60	2.56	5

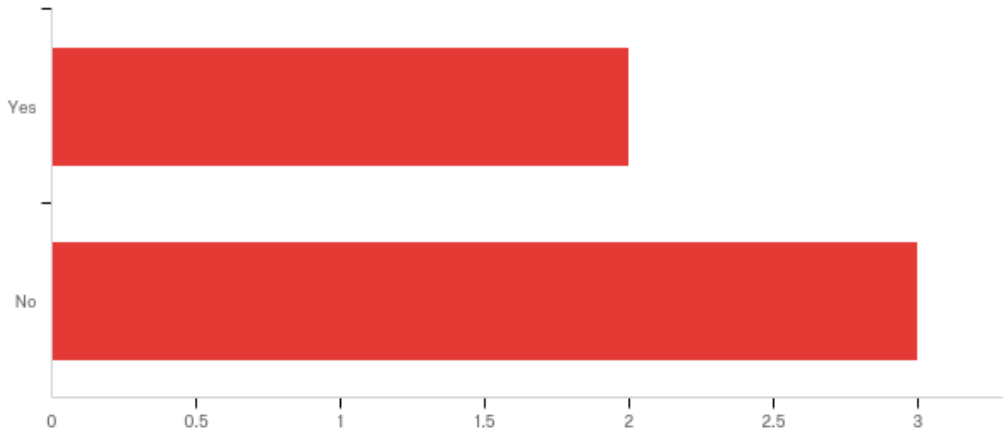
Q2 - Rate of how many tasks were completed successfully.

Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
Completion Rate of Tasks	3.00	3.00	3.00	0.00	0.00	2

Q3 - What do you think of the Website ?

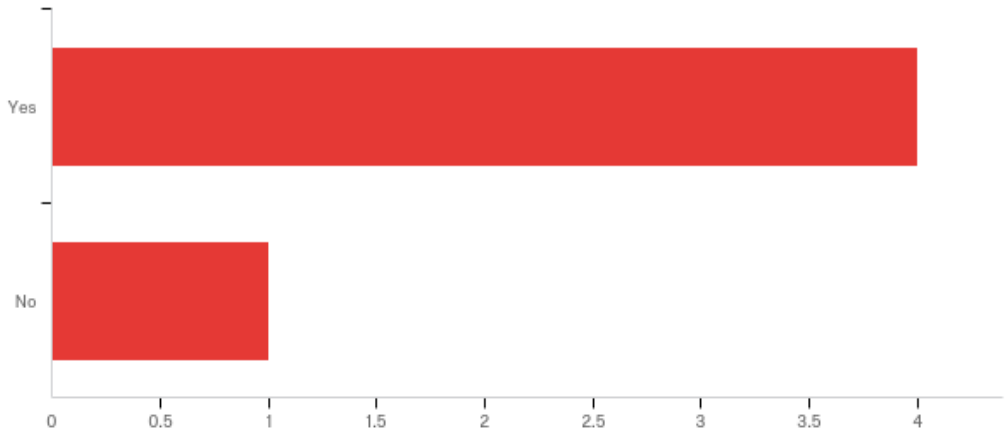
#	Question	Really Bad		Bad		Average		Good		Perfect	
1	Coherence	0.00%	0	0.00%	0	0.00%	0	20.00%	2	60.00%	3
2	Design	0.00%	0	0.00%	0	50.00%	2	10.00%	1	40.00%	2
3	Arrangement	0.00%	0	100.00%	1	25.00%	1	30.00%	3	0.00%	0
4	Amount of Information	0.00%	0	0.00%	0	25.00%	1	40.00%	4	0.00%	0
	Total	Total	0	Total	1	Total	4	Total	10	Total	5

Q4 - Did you see that the website has a menu ?



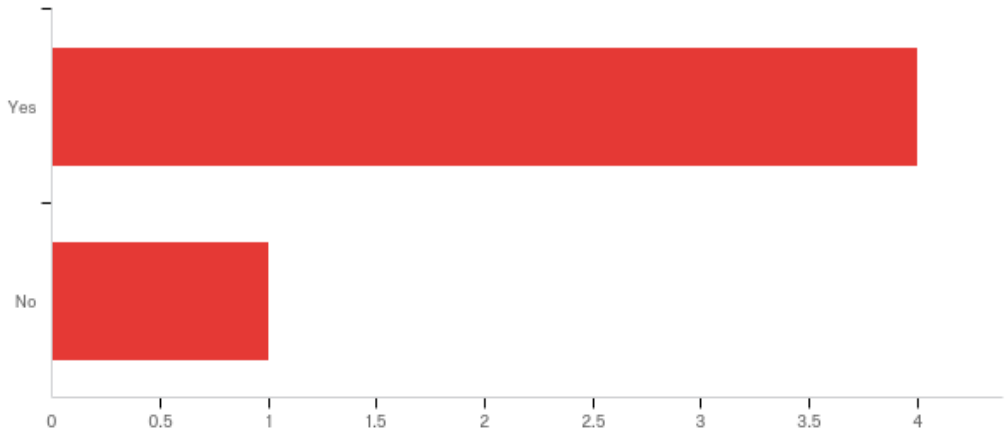
#	Answer	%	Count
1	Yes	40.00%	2
2	No	60.00%	3
	Total	100%	5

Q5 - Did you see that the website provides the current date ?



#	Answer	%	Count
1	Yes	80.00%	4
2	No	20.00%	1
	Total	100%	5

Q6 - Did you know that the website is scrollable ?



#	Answer	%	Count
1	Yes	80.00%	4
2	No	20.00%	1
	Total	100%	5

Bibliography

[1] J. Gottfired and E. Shearer, "News use across social media platforms 2016," p. 19, 2016.

[2] C. S. Lee and L. Ma, "News sharing in social media: The effect of gratifications and prior experience," *Computers in Human Behavior*, vol. 28, no. 2, pp. 331–339, 2012.

[3] M. Brown, "Abandoning the news," *Carnegiereporter Spring 2005*, pp. 1–16, 2005.

[4] A. Jones, *Losing the news: The future of the news that feeds democracy.* New York: NY: Oxford University Press, 2008.

[5] T. Patterson, "Young people and news," *Joan Shorenstein Center on the Press Politics and Public Policy Report Harvard University*, 2007. [Online]. Available: http://www.hks.harvard.edu/presspol/research/carnegie-knight/young_people_and_news_2007.pdf

[6] A. Casero-Ripollés, "Beyond newspapers: News consumption among young people in the digital era," *Comunicar*, vol. 20, no. 39, pp. 151–158, 2012.

[7] D. T. Mindich, "Tuned out: Why Americans under 40 don't follow the news," *Journal of Communication*, vol. 57, no. 1, pp. 175–176, 2007. [Online]. Available: http://onlinelibrary.wiley.com/doi/10.1111/j.1460-2466.2006.000339_2.x/abstract

[8] A. Mitchell and D. Page, "State of the News Media 2015," *Pew Research Center*, vol. 53, no. 9, pp. 1–97, 2015.

[9] K. G. Barnhurst and E. Wartella, "Newspapers and citizenship: Young adults' subjective experience of newspapers," *Critical Studies in Mass Communication*, vol. 8, no. 2, pp. 195–209, 1991.

[10] K. Raeymaeckers, "Newspaper editors in search of young readers: content and layout strategies to win new readers," *Journalism Studies*, vol. 5, no. 2, pp. 221–232, 2004. [Online]. Available: http://www.tandfonline.com/doi/abs/10.1080/1461670042000211195%5Cnhttp://www.tandfonline.com.proxy1.lib.umanitoba.ca/action/doSearch?type=simple&filter=multiple&stemming=yes&searchText=page+layout&publication=&searchType=journals&ajaxpagination=standard&a

[11] I. Costera Meijer, "The paradox of popularity. How young people experience the News," *Journalism Studies*, vol. 8, no. 1, pp. 96–116, 2007.

[12] M. Tsagkias, M. de Rijke, and W. Weerkamp, "Linking online news and social media," *Proceedings of the 4th ACM international conference on Web search and data mining*, pp. 565–574, 2011. [Online]. Available: http://doi.acm.org/10.1145/1935826.1935906

[13] A. Mitchell, J. Gottfired, E. Shearer, and K. Lu, "How Americans Encounter, Recall and Act Upon Digital News," 2017.

[14] H. Berghel, "Lies, Damn Lies, and Fake News," *IEEE Computer Society*, 2017.

[15] H. Allcott and M. Gentzkow, "Social Media and Fake News in the 2016 Election," 2017.

[16] C. Silverman, "Viral Fake Election News Outperformed Real News on Facebook in Final Months of the US Election," 2016. [Online]. Available: www.buzzfeed.com/craigsilverman/viral-fake-election-news-outperformed-real-news-on-facebook?utm_term=.kq3Zz2Wxa#.rbBZBjgdx

[17] O. Solon, "Facebook's Fake News: Mark Zuckerberg Rejects 'Crazy Idea' That It Swayed Voters," 2016. [Online]. Available: https://www.theguardian.com/technology/2016/nov/10/facebook-fake-news-us-election-mark-zuckerberg-donald-trump

[18] R. Marchi, "With Facebook, Blogs, and Fake News, Teens Reject Journalistic "Objectivity"," *Journal of Communication Inquiry*, vol. 36, no. 3, pp. 246–262, 2012.

[19] C. Smith, "Facebook and Google reveal how they plan to fight fake news," 2017. [Online]. Available: http://bgr.com/2017/04/09/facebook-vs-google-anti-fake-news/

[20] C. Hall and M. Zarro, "Social curation on the website Pinterest.com," *Proceedings of the ASIST Annual Meeting*, vol. 49, no. 1, 2012.

[21] C. Padoa, D. Schneider, J. M. De Souza, and S. P. J. Medeiros, "Investigating social curation websites: A crowd computing perspective," in *Proceedings of the 2015 IEEE 19th International Conference on Computer Supported Cooperative Work in Design, CSCWD 2015*, 2015, pp. 253–258.

[22] C. Bernardini, T. Silverston, and O. Festor, "A Pin is worth a thousand words: Characterization of publications in Pinterest," in *IWCMC 2014 - 10th International Wireless Communications and Mobile Computing Conference*, 2014, pp. 322–327.

[23] A. Das, M. Datar, A. Garg, and S. Rajaram, "Google news personalization: scalable online collaborative filtering," *Proceedings of the 16th international conference on*, pp. 271–280, 2007. [Online]. Available: http://portal.acm.org/citation.cfm?id=1242610

[24] O. Phelan, K. McCarthy, M. Bennett, and B. Smyth, "Terms of a feather: content-based news recommendation and discovery using twitter," *Proceedings of the 33rd European conference on Advances in information retrieval*, no. 07, pp. 448–459, 2011. [Online]. Available: http://dl.acm.org/citation.cfm?id=1996889.1996947

[25] T. Weninger, X. Zhu, and J. Han, "An exploration of discussion threads in social news sites: a case study of the Reddit community," *Advances in Social Networks Analysis and Mining (ASONAM), 2013 IEEE/ACM International Conference*, vol. 579, no. 2, pp. 579–583, 2013. [Online]. Available: http://dl.acm.org/citation.cfm?id=2492646

[26] W. Graham, *Facebook API Developers Guide*, 2008. [Online]. Available: http://www.springerlink.com/index/10.1007/978-1-4302-0970-6

[27] J. L. Willsion, "Flipboard (for Android)," 2017. [Online]. Available: http://www.pcmag.com/article2/0,2817,2406226,00.asp

[28] J. Parker and J. Cabebe, "Flipboard for Android review: Beautify your RSS and social feeds," 2014. [Online]. Available: https://www.cnet.com/products/flipboard-android/review/

[29] S. H. R. Wong, "Which platform do our users prefer: website or mobile app?" *Reference Services Review*, vol. 40, no. 1, pp. 103–115, 2012.

[30] M. Boulton, "A Practical Guide to Designing for the Web," *Design*, 2009. [Online]. Available: https://marketing.conference-services.net/resources/327/2342/pdf/AM2011_0257.pdf

[31] B. Shneiderman, "Research-Based Web Design & Usability Guidelines," *Igarss 2014*, no. 1, pp. 1–5, 2014.

[32] R. W. Proctor and K.-P. L. Vu, *Handbook of Human Factors in Web Design*, 2005, vol. 2005. [Online]. Available: http://books.google.com/books?hl=en&lr=&id=d0N6mumoLbAC&oi=fnd&pg=PR11&dq=HANDBOOK+OF+HUMAN+FACTORS+IN+WEB+DESIGN&ots=AGoty49lL8&sig=mbtOIcMNduk8b2E9qrUqXuEEklw

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

[34] C. D. Manning, P. Raghaven, and H. Schuetze, *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[35] "Decision Trees." [Online]. Available: http://scikit-learn.org/stable/modules/tree.html

[36] "sklearn.feature_extraction.text.CountVectorizer." [Online]. Available: http://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.CountVectorizer.html#sklearn.feature_extraction.text.CountVectorizer

[37] "sklearn.feature_extraction.text.HashingVectorizer." [Online]. Available: http://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.HashingVectorizer.html#sklearn.feature_extraction.text.HashingVectorizer

[38] "sklearn.feature_extraction.text.TfidfTransformer." [Online]. Available: http://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfTransformer.html#sklearn.feature_extraction.text.TfidfTransformer

[39] "sklearn.feature_extraction.text.TfidfVectorizer." [Online]. Available: http://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html#sklearn.feature_extraction.text.TfidfVectorizer

[40] A. Kibriya, E. Frank, B. Pfahringer, and G. Holmes, "Multinomial Naive Bayes for Text Categorization Revisited," *In AI 2004: Advances in Artificial Intelligence*, pp. 488–499, 2005. [Online]. Available: http://link.springer.com/chapter/10.1007/978-3-540-30549-1_43

[41] A. Mitchell, J. Gottfired, J. Kiley, and K. E. Matsa, "Political Polarization & Media Habits," 2014. [Online]. Available: http://www.journalism.org/2014/10/21/political-polarization-media-habits/

[42] "Supervised learning: predicting an output variable from high-dimensional observations." [Online]. Available: http://scikit-learn.org/stable/tutorial/statistical_inference/supervised_learning.html

[43] "sklearn.metrics.precision_recall_fscore_support." [Online]. Available: http://scikit-learn.org/stable/modules/generated/sklearn.metrics.precision_recall_fscore_support.html#sklearn.metrics.precision_recall_fscore_support