Fake News Detection on Twitter

Proposal

Shaheen Karodia
University of Cape Town
Rondebosch
Cape Town, Western Cape 7700
krdsha003@myuct.ac.za

Michelle Lu
University of Cape Town
Rondebosch
Cape Town, Western Cape 7700
lxxwei013@myuct.ac.za

Kristin Kimont
University of Cape Town
Rondebosch
Cape Town, Western Cape 7700
knmkr002@myuct.ac.za

1 PROJECT DESCRIPTION

1.1 Introduction

The role of social media in our day to day lives has increased rapidly in recent years. It is now used not only for social interaction, but also as an important platform for exchanging information and news [17]. Twitter, a micro-blogging service, connects millions of users around the world and allows for the real-time propagation of information and news [17]. These factors have resulted in Twitter playing a critical role in world events, especially crisis events [14], where it has been useful in emergency response and recovery [17], as well as assisting in disaster management [14].

Twitter has, however, not only been used for the spread of valid news, but also deceptive and fake news [5]. This fake news can come in the form of spam [2], astroturf (a technique used in political campaigns to fake support numbers, by making a message appear to have ‘grassroots’ origins when in reality it originated from one person or organisation) [23], clickbait (content that aims to attract attention and get users to click on a link to increase website traffic) [7] and more. The increase in the volume of fake news has even led to our current times being labelled ‘the age of misinformation’ [27] and therefore stresses the importance of assessing the credibility of tweets [18].

Our project therefore focuses on developing a system that can algorithmically assess the credibility of tweets on Twitter, and present the assessment results to the user. A classifier will be trained using an annotated data set, generated through a crowdsourcing mobile application, and results will be displayed in the Twitter interface via a Web browser extension.

1.2 Project Significance

The existence of fake news is not new. Before the use of social media, news was restricted to sources such as the radio, newspapers and TV, where the task of filtering out fake news was assigned to journalists and other news publishers [9]. The rapid increase in user generated content has, however, meant reliance on these traditional filtering techniques is no longer applicable. Research has found that humans are not good at detecting lies in text based on content alone [18, 24] and so there has been a drive to automate news credibility evaluation.

If our project is successful, we hope to help improve people’s critical judgment of online news and encourage people to be more aware of its existence. We hope to reduce the spread of fake news and aid people in their credibility assessment of the content that they consume.

1.3 Project Issues and Difficulties

One of the major problems associated with our project is obtaining a labelled dataset of tweets from Twitter. More specifically, it will be difficult to obtain labels that are not subjective as credibility assessment can be personal. The labels also need to be general enough so that we can apply them to different types of fake news, hence, a broad range of tweets and users need to be used in the crowdsourcing. Annotators who are ill-informed on the tweet topic may also provide inaccurate labels.

Providing credibility assessments in real time is difficult if we want the tweet’s context and historic data to be included in the assessment. This is because other tweets related to the tweet topic may not be available if the tweet is one of the first. Providing credibility ratings only on demand may be a solution.

2 PROBLEM STATEMENT

2.1 Aims and Research Questions

In this project, we aim to develop a labelled dataset of fake and genuine news. We will then use the labelled dataset to extract features and train a multiclass machine-learning classifier. A user interface will be developed to notify users of the credibility rating produced by this classifier. We aim to do this on demand and to develop this solution in a South African context. The research questions of respective team members are presented below:

Shaheen: Can a crowdsourcing mobile application be used to annotate the veracity of tweets?

Michelle: Can distinctive features be extracted from fake and legitimate information on Twitter to train a machine-learning classifier, that can assess tweets with high precision and recall?

Kristin: Can a Web browser extension for displaying tweet credibility on Twitter be developed, which is useful and has a usable interface?

2.2 Requirements

After conducting requirements analysis via interviews with potential users, we have gathered the following requirements:

- Develop a crowdsourcing mobile application that will use humans to annotate tweets.
- The crowdsourcing application should display the tweet and the user profile for users to make an assessment on the credibility.
- The scale used to assess and display credibility should have at least 5 levels: Definitely fake, Likely to be fake, Neutral, Likely to be true, Definitely true.
• Leaderboards should be used to incentivise users to stay on the application.
• The user-interface for the credibility system should be a Web browser extension that augments the Twitter interface.
• Credibility assessment should be on-demand.
• Summarised details on how the credibility rating was calculated should be available.
• Users should be able to give feedback on whether they agree or disagree with the rating, and, if they disagree, they can give their own rating.

For the full interviews, please refer to Appendix D

3 PROJECT OVERVIEW

3.1 System diagrams

The following diagram provides an overview of the systems involved in the project, and how the different components interact with each other. For a more detailed explanation of the 3 main project divisions seen in the diagram, refer to the following sections: (1) Crowdsourcing Application in Section 4.1.1, (2) Tweet Classifier in Section 4.1.2 and (3) Credibility Checking System in Section 4.1.3.

![Figure 1: Systems Overview](image)

3.2 Work Allocation

Through consultation with our supervisors, sections of the project have been divided amongst the team. These sections have been split based on the estimated time required for completion, and allocated based on team member preference. The allocation is as follows:

(1) Shaheen Karodia: Crowdsourcing Application
(2) Michelle Lu: Tweet Classifier
(3) Kristin Kinmont: Credibility Checking System

4 PROCEDURES AND METHODS

4.1 Development Procedure

4.1.1 Crowdsourcing Application. An Android application shall be developed. Due to Android’s ubiquity, we hope to reach a large number of users, hence obtaining a substantial amount of annotated data. It will be developed using the Adobe PhoneGap framework. This technology has been selected as it is both Open Source and uses HTML, CSS and Javascript that Shaheen Karodia, the team member assigned, has experience working with. Furthermore, Google Firebase will be used as a real time database.

A user centered design approach will be followed. The feedback obtained from a series of user focus groups will be incorporated in successive iterations of the application.

The application will present users with tweets, one at a time, and a scale to select how credible they feel the tweet is. The tweets that are displayed to users will be provided from the Firebase database. Initially, the database will be populated with a dataset of tweets and corresponding user profiles that will be scraped from Twitter using Twitter’s APIs. As a user annotates a tweet, a new tweet will be selected from the database and sent to the application. For each tweet, the credibility option selected by users will be saved on the database.

Once a sufficient number of user responses have been recorded, the labelled data will be passed offline, in an agreed upon format, to the Tweet Classifier as training data.

4.1.2 Tweet Classifier. The Tweet Classifier will initially take in data from the crowdsourcing application to train the machine-learning classifier. To assess and analyze tweets, we will take on a content-based and graph-based approach. The content-based approach will include a combination of features, such as the URLs, hashtags, @username mentions and sentiment of the tweet. These features have been selected based off previous work done by Gupta et al., O’Donovan et al. and Chhabra et al. [8, 14, 19], which can be found in Section 6.2.

In the graph-based approach, three different graphs will be constructed: a retweet graph, a reply graph and a like graph. We will then look at the features of these graphs, the number of nodes, edges, centrality and community detection, to see if there are any particular distinct patterns that can be extracted to train a classifier to evaluate the credibility of tweets.

Since the data will be classified into five different credibility categories (as mentioned in Section 2.2), we will then make use of a multiclass machine-learning algorithm, via the WEKA library, which is used when there are more than two categories for classification.

4.1.3 Credibility Checking System. The development of the credibility checking system will be divided into two major parts: a front-end and a back-end.

The front-end will be a Web browser extension that will inject tweet credibility scores into the Twitter UI. It will also scrape Twitter data using Twitter’s APIs, and extract the tweet information.

1http://phonegap.com/products/
2https://firebase.google.com/
3https://dev.twitter.com/overview/api
required by the Tweet Classifier. The Tweet Classifier will be embedded in the back-end server, therefore the front-end will send the tweet information to the back-end server for processing. The credibility ratings calculated will then be sent to the front end to be presented to the user. The back-end server will also store any user feedback on the credibility ratings, sent to it by the front-end.

System development will take a modular approach to help ensure re-usability of the extension. Development will also be user-centered. User feedback, gathered through focus groups, will be used to improve each prototype iteration. Javascript, CSS and HTML technologies will be used for front-end development while Java will be used for back-end.

4.2 Development Methods and Practices

For the duration of the project, we will adopt an iterative approach to development, with 3 iteration phases. Evolutionary prototypes for each component will be developed and improved throughout the project. This methodology has been selected due to the inflexible time constraint. As a result, prototypes of previous iterations will be built on in subsequent iterations. An outline of iteration dates can be found in the Gantt chart (see Appendix B).

4.3 Evaluation Metrics

4.3.1 Crowdsourcing Application. The crowd sourcing application will be evaluated with the System Usability Scale (SUS), a 10 item likert scale questionnaire, that will be used to calculate overall usability and user satisfaction indexes for the applications. The questionnaire will be augmented with additional questions such as how easy users found it to annotate the credibility of tweets and could users make a credibility assessment with solely the information presented to them via the application. Furthermore, additional metrics, such as user retention rates and percentage of skipped tweets will be calculated.

4.3.2 Tweet Classifier. The measures for evaluation of the classifier will be precision and recall. Precision is the ratio of how many of the predicted values are actually correct [31]. Recall is the ratio of how many of the actual truth labels were predicted [31]. These will be calculated as follows [31]:

- \( n \) is the number of tweets.
- \( Y_i \) is the actual truth label assigned to the \( i^{th} \) tweet
- \( x_i \) is the \( i^{th} \) tweet
- \( h(x_i) \) is the predicted truth label for the \( i^{th} \) tweet

\[
\text{Precision} = \frac{1}{n} \sum_{i=1}^{n} \frac{|Y_i \cap h(x_i)|}{|h(x_i)|}
\]

\[
\text{Recall} = \frac{1}{n} \sum_{i=1}^{n} \frac{|Y_i \cap h(x_i)|}{|Y_i|}
\]

Prediction of the classifier will be done via k-fold cross validation. This technique divides the dataset into k random samples, from which k-1 samples are used to train the classifier, and the remaining sample is used to test it. This process repeats k times until each and every sample has been used to test and train the classifier. Once that has been completed, the results are then averaged. For each classification, precision and recall will be recorded.

4.3.3 Credibility Checking System. The front-end browser extension will focus on user based evaluation to test the usefulness and usability of the system. To do this users will be asked a series of questions based on, and including, those in the System Usability Scale (as discussed in Section 4.3.1). This scale was chosen as it evaluates usability, one of the focuses of the associated research question. The extension will also be tested for bugs and whether or not it meets the original requirements.

5 ETHICAL, PROFESSIONAL AND LEGAL ISSUES

Two components of the project, namely, the crowdsourcing application and browser extension, require usability testing. Furthermore, requirements gathering will be conducted with potential users. In both instances, we have an obligation to behave professionally during all interactions with participants.

Throughout the project we must adhere to Twitter’s Developer Policy. Ethical clearance must be obtained from the Faculty of Science Research Ethics Committee for the usability tests and to release the crowdsourcing application to users to obtain annotated data.

The crowdsourcing application and browser extension will be free and available for download from the Google Playstore and Chrome Webstore respectively. Both applications will be Open Source and released under the Creative Commons license in congruence with UCT’s IP policy.

6 RELATED WORK

There are numerous studies on credibility assessment related to Twitter. In this section, we highlight approaches previous studies have used that are relevant to this project.

6.1 Crowdsourcing

Crowdsourcing is a process of outsourcing work to a crowd, i.e. having a large network of individuals collaboratively performing a task [26]. Crowdsourcing has been used in various contexts such as crowdsourcing of geographical content, digital archives, artificial intelligence, data analysis and graphical design[26]. Notable successes of crowdsourcing include, Threadless.com a company that has crowdsourced T-shirt designs via online competition and iStockphoto.com that crowdsources stock photographs and remunerates photographers for every photo sold through the iStockphoto site. The company, InnoCentive, posts research and development problems to the crowd on behalf of companies and rewards the crowd for proposing viable solutions[4]. Many problems rely on utilizing the wisdom of the crowd, where an aggregated answer, from a large number of individuals, produces viable solutions to a problem [4]. A subset of crowdsourcing that relies on the wisdom of the crowd is crowdvoting, whereby individuals are presented with a series of options and their answers aggregated into a single vote [20]. Crowdsourcing has been used in various mobile applications [6]. Two mobile crowdsourcing platforms, Gigawalk and TaskRabbit, provide a market to offer crowdsourcing microtasks.

---

\[ \text{dev.twitter.com/overview/terms/policy.html} \]
\[ \text{www.google.com} \]
\[ \text{http://www.uct.ac.za/downloads/uct.ac.za/about/policies/intellect_property.pdf} \]
Examples of tasks performed include crowdsourcing 3D panoramas for Microsoft Bing and outsourcing household errands [28].

6.2 Network Graphs

Network graphs can be used to graphically show the interconnections between entities and how information is propagated. According to Qazvinian et al. [21], network-based features, such as the number of retweets a user has, or their tweet history, can be a good measure of rumour detection. The Web service ‘Truthy’ developed by Ratkiewicz et al. [22], analyzes the circulation of information in Twitter through the use of directed graphs, with the edges representing retweets between users. Castillo et al. [5] took a similar approach with the use of a propagation tree to observe retweeting behaviour. They found that the depth of the propagation tree is an important feature, as more retweets are linked with more credible news.

6.3 Spam Detection

Spammers and phishers are classified as those who post tweets containing trending words and URLs, obscured by URL shorteners, forcing users to load the website in order to identify its content, and through this, generate website traffic and revenue [3]. In studies done by Gupta et al. and O’Donovan et al. [14, 19], it was shown that URLs are an effective feature for credibility detection as spam and phishing tweets tend to include a higher number of URL links. Chhabra et al. [8] have also found that spammers try to gain more visibility by using hashtags to take advantage of trending topics, and use @username mentions to get more retweets, therefore spam tweets tend to have a higher number of mentions and hashtags.

6.4 Machine-Learning Classification

There are various schemes proposed for spam detection or credibility assessment on Twitter that use machine learning methods as a means of classification. These methods include Naïve Bayesian classifiers [1, 14, 29, 30] and Decision Tree classifiers [1, 5, 14, 29, 30]. Wang’s [30] Bayesian classifier was used to differentiate suspicious with Hurricane Sandy. Many SVM related classifiers were trained, using SVM [2, 23, 29, 30], SVM Rank with Pseudo Relevance Feedback [12] and SVM Rank [13]. Other methods tested in different papers include neural networks [29, 30], k nearest neighbour [29, 30], AdaBoost with Decision Stump [23], Random Forest [1], logistic regression algorithms [16], Coordinate Ascent, Ada Rank and Rank Boost [13].

6.5 Content-based Approaches

Content-based or linguistic approaches are those that base assessment on the actual content of the tweet and aim to identify language ‘leakage’ or ‘predictive deception’ [10]. Mendoza et al. [17] utilizes the idea that fake news is questioned more, and both Castillo et al. and Mendoza et al. [5, 17] use text classifiers, to identify and tally tweets that are asking questions. When enough people are questioning the news item/tweet, Twitter can warn users of the potential deception. Rubin et al. [24] makes use of Natural Language Processing (NLP) with machine learning, to identify satire, 90% precision and 84% recall are achieved by focusing on things such as language patterns, sentiment, rhetorical devices and the number of words (i.e. the content of the tweet). Another method by Ikegami et al. [15] used Latent Dirichlet Allocation to locate tweets on specific topics, and classified whether the opinion on the topic was positive or negative by using a semantic orientation dictionary. The credibility of the information was then based on the ratio of positive to negative opinions. The accuracy of both the topic and opinion classification was only 47.6% [15].

6.6 Credibility Assessment Systems

There are systems that have been developed to assess the credibility of information. Systems that are implemented as Web-browser plug-ins include TweetCred[13] which rates the credibility of tweets and PhishAri[1] which detects phishing in Twitter. Both these plug-ins operate in real-time and augment the Twitter interface for their display. There is also Fake Tweet Buster[25] which is a Web application that allows you to paste in a tweet URL or twitter account URL, and it determines whether it is fake or not. They also make use of crowdsourcing to get annotated data about user accounts[25].

7 ANTICIPATED OUTCOMES

7.1 Systems

7.1.1 Crowdsourcing Application. Some of the key features of the application include displaying tweets, recording the responses of users, and having a leaderboard of top annotators. A few challenges have been anticipated such as how to incentivise users to continue using the application, which tweets should be selected to display to users, and how many responses are required to determine consensus for the annotation of a particular tweet.

7.1.2 Tweet Classifier. We expect that distinctive features can be extracted from fake and legitimate information on Twitter through the content-based and graph-based approaches, which will then allow the classifier to successfully classify tweets.

7.1.3 Credibility Checking System. The front-end Web browser extension will allow users to access the credibility rating of a tweet on demand, by clicking a button next to the tweet. This ease of access, as well as the fact that the extension can be switched on and off, makes the system less invasive. The browser extension will allow users to give feedback on the credibility rating. Feedback will, however, require users to login and will be stored on the back-end server. An overview of the credibility rating rationale will also be available to the user via an information button.

7.2 Expected Project Impact

We hope this project will encourage users to be more aware of the information they receive and spread. Taken collectively, we hope to empower users not to tacitly accept information on Twitter as truthful, and aid their credibility assessment of the content that they consume.
7.3 Key Success Factors

Crowdsourcing Application:
- Users are able to make a credibility assessment of a tweet solely from the information displayed to them in the application.
- A large number of annotated tweets are collected.

Tweet Classifier:
- High precision and recall of the classification of tweets.
- Users are satisfied with the response time of receiving credibility ratings.

Credibility Checking System:
- Users find the system useful and informative.
- Users can receive credibility ratings for individual tweets, on demand, via a button.
- The front-end browser extension is able to extract all the necessary tweet information from Twitter required by the Tweet Classifier.

Overall:
- User feedback can give an indication that our credibility assessment is successful, if the majority of users agree with the credibility rating of the tweet.

8 PROJECT PLAN

8.1 Risk and Risk Management Strategies

Many risks related to this project have been identified. They are outlined together with corresponding strategies for mitigation, monitoring and management in Appendix A.

8.2 Timeline

The project runs from 18 April 2017 to 31 October 2017. The Gantt chart for the project outlines the full project timeline, marking both deliverables and milestones. It can be found in Appendix B.

8.3 Required Resources

People participation is a very important resource in our project. We need volunteers to help us create an annotated dataset for training our algorithm as well as users to test the interface of our Web extension. Other resources can be separated into two components: Hardware and Software.

Hardware Resources:
- Computers/Laptops for development
- Android phones for application testing and evaluation

Software Resources:
- Java IDE
- WEKA Java Library
- Adobe PhoneGap framework
- Twitter APIs
- Google Firebase

The hardware and software components are already available to us.

8.4 Deliverables

The main deliverable for this project is a system that algorithmically assesses and presents the credibility of a tweet to a user. Major deliverables, which are necessary for the completion of the system, are a crowdsourcing application used to label our input data, a machine learning classifier to rate the tweets, and a browser extension to display the credibility ratings to users. The features and specifics of these deliverables are described in the previous sections; however they are subject to change as iterative development is carried out. Other deliverables during the course of the project include:
- Literature review
- Project proposal
- Project Proposal presentation
- Software feasibility demonstration
- Final paper
- Final code
- Final project demonstration
- Poster
- Web page
- Reflection

8.5 Milestones

Project milestones can be seen in our Gantt chart (Appendix B) and Milestones table (Appendix C). Milestones include the ones set by the department as well as internal deadlines set by our supervisor.

REFERENCES


## A RISK MATRIX

<table>
<thead>
<tr>
<th>Risk</th>
<th>Category</th>
<th>Consequence</th>
<th>Probability</th>
<th>Impact</th>
<th>Mitigation</th>
<th>Monitoring</th>
<th>Management</th>
</tr>
</thead>
<tbody>
<tr>
<td>Failure to complete the project by the deadline</td>
<td>Schedule</td>
<td>Overall delay of the project</td>
<td>Low</td>
<td>Catastrophic</td>
<td>Have many intermediate milestones for each deliverable</td>
<td>Monitor and update the Gantt chart throughout the project</td>
<td>Inform our supervisor and readjust the Gantt chart.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Rushing could result in a poor quality final deliverable</td>
<td></td>
<td></td>
<td>Have a project schedule approved at the project proposal stage</td>
<td>Weekly progress meetings with our supervisor</td>
<td>Consider revising the project scope</td>
</tr>
<tr>
<td>Team members being too ambitious with their goals for their sections, leading to scope creep.</td>
<td>Scope</td>
<td>Final product has insufficient focus on the core requirements.</td>
<td>Medium</td>
<td>Critical</td>
<td>Clearly define project requirements, scope and team responsibilities at the onset of the project.</td>
<td>Regular meetings with our supervisor ensuring that we are always working within scope.</td>
<td>Undertake a formal scope change process with our supervisor and drop unimportant features</td>
</tr>
<tr>
<td>Supervisor becomes unavailable</td>
<td>Stakeholders</td>
<td>Can result in overall delay of project and in turn a poor quality final deliverable.</td>
<td>Low</td>
<td>Critical</td>
<td>Schedule standing weekly meetings</td>
<td>Confirm weekly in advance, the available of our supervisor</td>
<td>Reschedule weekly meeting and offer alternative times</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Offer alternative formats of meeting such as via Skype</td>
</tr>
<tr>
<td>Unable to get sufficient labelled data from the crowdsourcing application due to a lack of interest from users.</td>
<td>Resources</td>
<td>Labels collected may be of a poor quality. Lack of interest will also make user testing difficult. Project delay as the classifier cannot be trained with a real data set till one is acquired.</td>
<td>Medium</td>
<td>Critical</td>
<td>Schedule the release of the crowdsourcing tool early in the project to ensure sufficient exposure. Marketing of the application will further increase exposure. This can be done via word of mouth, UCT email or social media platforms.</td>
<td>Consistently check how much data has been collected via the crowdsourcing application.</td>
<td>Increase efforts to promote the data capture tool, such as providing incentives to users. E.g. monetary compensation.</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>Team members can’t complete sections in parallel due to dependencies.</td>
<td>Team</td>
<td>Project progress and development will be slow as individuals have to wait for other team members to complete required sections. This may result in missed deadlines and the project not being completed in time.</td>
<td>High</td>
<td>Critical</td>
<td>Prioritize tasks such that work needed by other teammates is completed first.</td>
<td>Consistent team meetings with supervisor ensuring team members are on schedule and team members have access to required project components.</td>
<td>Develop dummy data sets such that other team mates can demonstrate their respective components. To develop dummy labelled datasets, we can scrape tweets from known fake accounts, and tweets from verified news accounts and label them accordingly.</td>
</tr>
<tr>
<td>Unable to integrate final project components.</td>
<td>System</td>
<td>Final system deliverable doesn’t work.</td>
<td>Low</td>
<td>Moderate</td>
<td>Clearly define how data will be passed between project components so that team members can develop their sections accordingly. This includes defining data formats and deciding on programming languages.</td>
<td>Continuous integration at the end of each iteration.</td>
<td>Ensure team member components can function independently, using synthetic and/or hard coded data.</td>
</tr>
<tr>
<td>ID</td>
<td>Task Name</td>
<td>Start</td>
<td>Finish</td>
<td>Duration</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----</td>
<td>----------------------------</td>
<td>-------------</td>
<td>-------------</td>
<td>----------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Literature Review</td>
<td>2017/04/18</td>
<td>2017/05/12</td>
<td>16d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Draft One</td>
<td>2017/04/18</td>
<td>2017/05/02</td>
<td>1d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Draft Two</td>
<td>2017/05/03</td>
<td>2017/05/05</td>
<td>3d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Final Submission</td>
<td>2017/05/05</td>
<td>2017/05/12</td>
<td>6d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Project Proposal</td>
<td>2017/05/12</td>
<td>2017/06/30</td>
<td>36d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Draft One</td>
<td>2017/05/12</td>
<td>2017/05/19</td>
<td>6d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Draft Two</td>
<td>2017/05/22</td>
<td>2017/05/26</td>
<td>5d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Final Proposal</td>
<td>2017/05/26</td>
<td>2017/06/02</td>
<td>6d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Proposal Presentation</td>
<td>2017/06/14</td>
<td>2017/06/15</td>
<td>3d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Revised Proposal</td>
<td>2017/06/15</td>
<td>2017/06/30</td>
<td>12d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Project Code</td>
<td>2017/06/22</td>
<td>2017/07/02</td>
<td>95d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>Iteration One</td>
<td>2017/07/03</td>
<td>2017/07/14</td>
<td>10d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>Prototype Demo (Internal Deadline)</td>
<td>2017/08/07</td>
<td>2017/08/07</td>
<td>0d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>Iteration Two</td>
<td>2017/07/14</td>
<td>2017/07/18</td>
<td>4d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>Iteration Three</td>
<td>2017/07/22</td>
<td>2017/07/28</td>
<td>6d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>Software Feasibility Demonstration</td>
<td>2017/08/14</td>
<td>2017/08/17</td>
<td>4d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>Final Code Submission</td>
<td>2017/10/02</td>
<td>2017/10/02</td>
<td>0d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>Project Paper</td>
<td>2017/07/17</td>
<td>2017/09/02</td>
<td>59d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>Paper plan and scaffold</td>
<td>2017/03/17</td>
<td>2017/03/24</td>
<td>6d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>First Test Write-up</td>
<td>2017/07/25</td>
<td>2017/08/15</td>
<td>16d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>Background Theory</td>
<td>2017/08/11</td>
<td>2017/08/18</td>
<td>6d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>Final Test Write Up Complete</td>
<td>2017/08/16</td>
<td>2017/08/24</td>
<td>7d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>Implementation and Testing write up</td>
<td>2017/08/22</td>
<td>2017/08/29</td>
<td>6d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>Coding Section write up</td>
<td>2017/08/22</td>
<td>2017/08/29</td>
<td>6d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>1st Draft Due (Internal Deadline)</td>
<td>2017/09/04</td>
<td>2017/09/04</td>
<td>0d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>Outline of final paper (Departmental Deadline)</td>
<td>2017/09/05</td>
<td>2017/09/05</td>
<td>0d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>1st Draft Due (Departmental Deadline)</td>
<td>2017/09/12</td>
<td>2017/09/12</td>
<td>0d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>Additional Project Media</td>
<td>2017/06/11</td>
<td>2017/07/12</td>
<td>96d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>Upload Proposal/Timeline to Website</td>
<td>2017/04/01</td>
<td>2017/04/07</td>
<td>5d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>31</td>
<td>Web page</td>
<td>2017/10/05</td>
<td>2017/10/12</td>
<td>6d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>Web Page Draft due (Internal Deadline)</td>
<td>2017/10/11</td>
<td>2017/10/11</td>
<td>0d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>Project Poster</td>
<td>2017/09/27</td>
<td>2017/10/09</td>
<td>9d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>34</td>
<td>Poster draft due (Internal Deadline)</td>
<td>2017/10/04</td>
<td>2017/10/04</td>
<td>0d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>35</td>
<td>Project Closing</td>
<td>2017/10/02</td>
<td>2017/10/31</td>
<td>22d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>Final Project Demonstration</td>
<td>2017/10/02</td>
<td>2017/10/09</td>
<td>6d</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>37</td>
<td>Write Reflection Paper</td>
<td>2017/10/10</td>
<td>2017/10/31</td>
<td>16d</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## C  PROJECT MILESTONES

<table>
<thead>
<tr>
<th>Description</th>
<th>Due Date</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Literature review</strong></td>
<td></td>
</tr>
<tr>
<td>(internal) Draft 1</td>
<td>12/05/2017</td>
</tr>
<tr>
<td>(internal) Draft 2</td>
<td>02/05/2017</td>
</tr>
<tr>
<td><strong>Project proposal</strong></td>
<td></td>
</tr>
<tr>
<td>(internal) Draft 1</td>
<td>02/06/2017</td>
</tr>
<tr>
<td>(internal) Draft 2</td>
<td>19/05/2017</td>
</tr>
<tr>
<td>Proposal presentation</td>
<td>26/05/2017</td>
</tr>
<tr>
<td>Revised proposal</td>
<td>12/06/2017 – 14/06/2017</td>
</tr>
<tr>
<td><strong>Project Code</strong></td>
<td></td>
</tr>
<tr>
<td>Iteration 1</td>
<td>02/10/2017</td>
</tr>
<tr>
<td>(Internal) Prototype demo</td>
<td>07/06/2017</td>
</tr>
<tr>
<td>Iteration 2</td>
<td>14/07/2017</td>
</tr>
<tr>
<td>Iteration 3</td>
<td>18/08/2017</td>
</tr>
<tr>
<td>Software feasibility demonstration</td>
<td>02/10/2017</td>
</tr>
<tr>
<td>Final code submission</td>
<td>14/08/2017 – 17/08/2017</td>
</tr>
<tr>
<td><strong>Project Paper</strong></td>
<td></td>
</tr>
<tr>
<td>Paper plan and scaffold</td>
<td>22/09/2017</td>
</tr>
<tr>
<td>First test write up</td>
<td>24/07/2017</td>
</tr>
<tr>
<td>Background/theory</td>
<td>15/08/2017</td>
</tr>
<tr>
<td>Final test write up complete</td>
<td>18/08/2017</td>
</tr>
<tr>
<td>Implementation and testing write up</td>
<td>24/08/2017</td>
</tr>
<tr>
<td>Coding section write up</td>
<td>29/08/2017</td>
</tr>
<tr>
<td>(internal) Paper draft</td>
<td>14/08/2017 – 17/08/2017</td>
</tr>
<tr>
<td>Outline of final paper</td>
<td>29/08/2017</td>
</tr>
<tr>
<td>Draft 1</td>
<td>04/09/2017</td>
</tr>
<tr>
<td>Finished paper write up</td>
<td>12/09/2017</td>
</tr>
<tr>
<td><strong>Additional Project Media</strong></td>
<td></td>
</tr>
<tr>
<td>Upload proposal/timeline to website</td>
<td>22/09/2017</td>
</tr>
<tr>
<td>(internal) Web page draft</td>
<td>07/06/2017</td>
</tr>
<tr>
<td>(internal) Project poster draft</td>
<td>11/10/2017</td>
</tr>
<tr>
<td>Project poster</td>
<td>20/09/2017</td>
</tr>
<tr>
<td>Web page</td>
<td>04/10/2017</td>
</tr>
<tr>
<td><strong>Project Closing</strong></td>
<td></td>
</tr>
<tr>
<td>Final project demonstration</td>
<td>12/09/2017</td>
</tr>
<tr>
<td>Reflection paper</td>
<td>04/10/2017</td>
</tr>
</tbody>
</table>
D REQUIREMENTS GATHERING

Crowdsourcing Application:

In addition to the tweet, what information do you think you'd like displayed with the tweet to help you make the assessment?

User 1:
He normally gets his news from sources he thinks are reliable. He wants to be able to see the profile of the source. He takes into consideration the picture and their background, as well as if they follow or have any friends that are familiar or credible.

User 2:
She wants to know by whom the tweet has been made and how reliable that person is. If that person in famous in computer science and then tweets about politics, then check his background if he knows the field. Analyze some tweets he posted and see what impact they made. There should be an indication of their authority. Or if they don’t, then what they tweet should maybe people with authority retweet their tweets.

User 3:
He would like to have some kind of authoritative information. Verified accounts don’t necessarily mean that the information is verified. He wants to be able to compare the information with traditional news outlets to see if it correlates, but this additional information, can be left up to the user to google otherwise it’s too much of a cognitive burden when all the information is shown.

User 4:
Depends on the type of news. If it’s computer related stuff, he can easily distinguish it. It it’s in a field that he’s unfamiliar with, he’ll need information about the source, when it was tweeted, what type/category it falls under e.g. political context.

User 5:
Needs the author and domain knowledge. He stated that feature structures such as the language structure is a classification task - not human task. He would want the original tweet and embedded conversation below it to get a good overview and some context. Otherwise you need perfect domain knowledge. There’s no one definition of fake news e.g. clickbait. Need to pilot it - Mechanical Turk - need to do some iteration.

How do we incentivise you to stay on?

User 1:
Must not take too much of his time or cognitive burden. He’s not very interested in leaderboards or gamification, he would prefer if he were to get some kind of acknowledgement for his contribution.

User 2:
Make crowdsourcing social. Tell them about their friends who are using the app. Get information about how they are doing gives a sense of priority for a topic. Annotate tweets that are you are interested in and look at what your friends are interested in. She is not interested in gamification or monetary incentives, though she thinks acknowledgement is better as it brings more value to her.

User 3:
Ensure that the information is easy to annotate, easily verifiable, and work with trending topics so that it doesn’t put a cognitive burden on the user. Gamification doesn’t work and has its limits with
crowdsourcing. He doesn’t like his name to appear online, so acknowledgement incentive doesn’t work with him, but he thinks it will work in general.

User 4:
Monetary incentive works, but other than that, the ease of access to the application is important. If it is seamless in Twitter, he doesn’t have to go to an extra application. He also enjoys recognition so a points system to see how much he’s contributed is effective and acknowledge is okay.

User 5:
Gamification and acknowledgement are surface level and not sustainable for the long run. Maybe implement it as a service e.g. a search engine, and then by using that, people contribute to the data.

*How do we encourage you to be recruited?*

User 2:
No particular interest in getting the user onto the app.

User 3:
To get people to participate use a point system and ensure users are getting points consistently.

*How would you like the annotation process to go?*

User 1:
The rating must be a scale. Yes or no is too restrictive.

User 2:
Use a scale. Strongly disagree, indifferent etc.

User 3:
Scale is better because most information is not black or white.

User 4:
Scale is a better option. There are instances when they aren’t 100% sure.

User 5:
Need a scale, but it also depends on audience and if they want to have more raw data.

*User Interface:*

If you could be told the level of credibility of a tweet, in which from would you want it presented to you? Examples include a Twitter plugin that highlights non credible/fake news tweets in your timeline, a separate web page one can visit, a mobile app, etc.

User 1:
He would like a plug-in for the Twitter app on his phone because it’s easier to access and because there are too many applications available to start having to make a separate one. Make sure to keep the time-ranking of the tweet.

User 2:
She would prefer a separate website to go find false information because she’s not on Twitter.
User 3:
Best way is to have a search engine that can search a trending topic- easier to implement. But regardless of implementation he would prefer a plug-in that is not invasive. He wants to be able to switch it off and click a button on a suspicious tweet to get the rating.

User 4:
Prefers embedded into Twitter, but it doesn’t matter to him because either way, if he was interested, he would google anyway.

User 5:
Browser plug-in/ web portal.

How would you prefer the credibility to be represented? Eg. A true/false rating, a scale, etc?

User 1:
Prefer a scale out of 4, so that it can be balanced on the two sides.

User 2:
Scale

User 3:
Scale should be descriptive. Probably use numbers and map it to a key

User 4:
Scale out of 10 - makes more sense- can have a midpoint.

User 5:
Scale

Is there any other information on the tweet or assessment you would like to be able to see? ie. Back end feedback. How would you want to view this?

User 1:
He would like the details of how it got the rating. Click a button for more information. How we do it is fine, as long we show it.

User 2:
Interested in knowing how we made this judgement.

User 3:
If you want to change the user’s mind, then put the information. It will work for an academic person, but don’t know if it will work for a regular person. He wants more proof of how we got the answer- more features to be more convincing. Tell him the accuracy of the classifier so he can choose to believe the classifier.

User 4:
Interested in knowing why we got the rating we got. The scope of the information needs to be as simple as much as possible since it is social media, he don’t want to have to think too much. Even as a technical person.
User 5:
Interested in seeing more information. Maybe take the tweet and strip out words, and put it into the search api and show the people with similar sentiment.

Would you like to give the application feedback on the credibility rating given? If so, how would you like this to happen? Would you like this feedback to be incorporated into future ratings or update the current rating? Need to put some control.

User 1:
Feedback to the system, agree or disagree, suggest their own rating. He wants his opinion to be acknowledged. The system must be respond and then use the information to improve the app in the future.

User 2:
Feedback is useful for later updates.

User 3:
Personally doesn’t feel feedback is a good idea. It will affect the results. So not a good idea of basing it on what people say.

User 4:
Wants the opportunity to give feedback. At that point it in time he’ll agree or disagree so no need for real-time feedback. Definitely use the feedback to future improve the algorithm.

User 5:
Feedback will also provide a good set of data later.