APPLYING MACHINE LEARNING TO BIG DATA USING SOCIAL MEDIA ANALYSIS TO IDENTIFY PEOPLE WITH HIGH INTELLIGENCE

Dissertation – National College of Ireland

AUGUST 31, 2015

Shane Burke
MSc in Web Technologies
Vikas Sahni

Declaration

I hereby certify that this material which I now submit for assessment of the programme of study leading to the award of MSc in Web Technologies is entirely my own work and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged with the text of my work.

Signature: ____.

Date: ____.

Supervisor: Vikas Sahni

Student No.: x13109839

Email: shaneburke0@gmail.com

Acknowledgement

I would like to take this opportunity to thank my supervisor, Vikas Sahni, who has provided me valuable insights and helped me to complete this work.

I also wish to thank my family and friends for their support during this study, and in particular Lucy, for her help, enthusiasm and patience.

CONTENTS

Executive Summary

Since the rise of social media platforms such as Facebook and Twitter, companies and organisations have performed social media analysis or data mining to help better understand their existing customers and to seek out potential new ones. This research has set about using this technique coupled with machine learning algorithms to explore the question of being able to identify highly intelligent people solely on their social media data. Given that there are millions of people worldwide sharing and collaborating online, the abundance of available data is potentially unlimited. The data collected as part of this research will be stored in a Big Data framework to ensure this work will be able to cope with the vast amounts of data available.

It was concluded that it's not possible to distinguish highly intelligent people by solely analysing their social media data. The observations from analysing over 1 million tweets show that social media users regularly boycott the use correct grammar and punctuation. The findings do suggest that Twitter's character restriction has a large influence on the quality of content posted.

1 Introduction

Big Data, machine learning and natural language processing are just a few of the most popular topics in today's ever changing world of technology. Those topics, combined with possibly the most popular, social media analysis and a research topic is born. The goal of this research is to try and identify highly intelligent people by analysing their social media data using natural language processing with the help of machine learning.

The aim of this research is not to classify people into a particular intelligence bracket, but rather discover those individuals who are deemed to be highly intelligent. Over 1 million tweets from Twitter were analysed and a score awarded to each tweet based on its content.

It is not a goal of this paper to brand people as less intelligent or to identify people of lower intelligence. This research aims to provide a quantitative method of scoring each piece of data collected with a value. At the end of the research a user was awarded a score that was comprised of the average score of all their collected data.

To aid this research, a platform was created to collect and analyse the data. The architecture of the platform is able to handle vast amounts of data or Big Data. Please refer to section 3 for more information regarding the design and specification of the platform. Section 4 describes how the platform was implemented and how each step of the process is linked together.

The major observations have been outlined in section 5. These observations have been based on 1 million tweets collected after they were analysed. Section 6 looks at the evaluation of the data after it has been analysed. The conclusion and further research is outlined in section 7.

2 Background and Related Work

This section will provide a brief synopsis of some of the material used in this research. This material will include various programming techniques, platforms leveraged, software technologies and some algorithms. Also this section describes related research papers and how that research may apply to this study.

2.1 Background

2.1.1 Social Media Analysis

Social media analysis is also known as social web mining, is a sub-set of data mining. Data mining can be described as the process of examining large existing datasets, to extract new information in a different usable structure (Dodd, 2014). Russell has described social web mining as a text-mining technique to extract meaning from human language data (Russell, 2013). There are many meanings which can be extracted. These include, sentiment, who knows whom on a social media platform, how frequent users communicate with each other, what are people talking about and what are people interested in.

For the purposes of this research, social media analysis was used to extract the human generated data on social media platforms. As there are numerous possible social media platforms which can be leveraged, this research will be focusing on analysing user data from Twitter and Facebook. The bulk of the analysis was performed using a technique called machine learning. Before this technique is applied, the social media data must first be stored in a consistent structure. Due to the fact that social media data can originate from different sources the collected data is stored in a consistent way. The goal is to use a data warehouse to store all this data. A data warehouse is a large storage of data acquired from multiple sources. The benefits of using a data warehouse are extremely beneficial when working with huge volumes of data or Big Data. Traditional data stores such as relational databases, text files, and xml files would not offer the same performance or reliability as a data warehouse when working with large datasets.

2.1.2 Machine Learning

An area of artificial intelligence called Machine Learning has been defined as “a field of study that gives computers the ability to learn without being explicitly programmed” by Arthur Samuel (Simon, 2013). Machine learning will make up an extensive part of this research. The machine learning code is used to analyse each piece of data and by a given set of criteria to try to give a score to each piece of data received. At the end of the research, a score will be awarded to each user. This score will be the average score of all their data that has been analysed.

2.1.3 Algorithms

The machine learning code will apply numerous algorithms to each piece of data. The details of each algorithm will explained in further details in the next section of this paper. Essentially the algorithms will be applied to each piece of data received. The algorithms will be responsible for ensuring the piece of data is in English, as for the purposes of this research, only English language data will be assessed. Each piece of data will be broken down into sentences. Other algorithms will evaluate the semantics and syntax of each sentence, before the sentence is parsed into words. Each word will then carry a particular weight depending on the length, how often it used and how specific the word is. A full list of weightings and how they are applied can be seen in the next section.

2.2 Related Work

The goal of this research is to be able to identify highly intelligent people through their social media activity. To accomplish this, the meaning of intelligence needs to be clearly established.

Cognitive ability has been described as the ability to perform a task, which can be physical or mental (Carroll, 1993). According to Carroll, measurements of human ability or intelligence can be performed using an educational test. These tests are a series of cognitive tasks, which assess the participant. These tests also include other information such as time taken or the participant's age among others (Carroll, 1993). For the purposes of this research, the users will only be tested on their written cognitive ability.

Similar research has already been conducted in this area. In 2008, a study was published in the area of trying to find to find high quality content in social media (Agichtein et al, 2008). The research focused on the social media platform called Yahoo! Answers, a leading question/answer platform. Agichtein and the group of Yahoo! employees set about to see if their classification framework could separate high quality social media data from spam with an accuracy similar to humans (Agichtein et al, 2008). The conclusions of this study state that they were able to separate the high quality content with a high level of accuracy, and also made the suggestion that this kind of technique would also be applicable to user generated data on different social media platforms (Agichtein et al, 2008).

The use of machine learning to analysis social media content is not a new topic. There has been plenty of research conducted in this area. Apart of the previously mentioned research by Agichtein et al 2008, the majority of machine learning research with social media data is done to find sentiment in a user's post. Although this is not directly related to this research topic, it is important to understand how fellow researchers gathered their data and performed their machine learning techniques. A previous research topic on "Sentiment Analysis of Twitter" by Anne Hennessey (Hennessey, 2014), analysed tweets by a given hashtag. Hennessey used an API to get the tweets, but did not require any permissions to use the data. Hennessey then stored each tweet in a CSV file and performed the machine learning techniques on each row in this file. This research will take a similar approach by storing the data and then performing the machine learning techniques on it.

Automated Essay Scoring (AES) applied computational techniques using Automated Writing Evaluation (AWE) to grade and assess human writing (Shermis & Burstein, 2003). These algorithms can be applied to user generated text to get a score across multiple factors. These include grammar, comprehension, vocabulary and the users understanding of the topic (Roscoe et al, 2014). The conclusion of this study showed that AES algorithms are effective at grading human writing, but scores which graded humans understanding of a topic were subsequently lower than the same writings graded by humans (Roscoe et al, 2014). This study has showed that AES algorithms are not quite ready to determine humans' knowledge or intellect on a topic they are writing about just yet.

Personality is another form of human behaviour which has been previously analysed through social media. In 2007, Gosling et al set about analysing the personality of 133 Facebook users. The study evaluated the personality of all the users against the Big Five dimensions (Extraversion, Agreeableness, Conscientiousness, Emotional Stability, and Openness to Experience) (Gosling et al, 2007). The assessment involved taking groups of five friends at a time into a meeting room. The group were then asked to complete a series of tasks, which was used to evaluate their personalities. After this method, nine research assistants were asked to review the Facebook profiles of each participant, to score their online personality. It took on average 16 hours to complete over a 5 week period. The main Facebook profile page was analysed along with a random selection of 10 photos. The conclusion of the research showed that some consensus between the Facebook profile and the Big Five dimensions. The impressions also show some accuracy between the recorded personality and the Facebook personality with the exception of Emotional Stability (Gosling et al, 2007).

In 2013, Capiluppi et al set about creating a framework to analyse different social media websites to be assess a candidate's technical skills (Capiluppi et al, 2013). The framework included a set of qualitative signals for each type of social media site. Whether it be a social network site, a code sharing site, a question and answer site or a professional social network site, they created a qualification and reference signal as well as a recommendation for each type of site. They also created a set of signals for profile aggregation sites. These sites, concatenate multiple profiles together and award their score and badges to each user. The research concluded that it advocated the practice of assessing potential candidates through social media. Although the paper does suggest that there are flaws with this method, such as distorted reputation levels or some candidates may be less inclined to use social media as others.

3 Design and Specification

There are multiple components required to make up the complete solution for this research. The main components are a front facing website which is exposed to an end user, external social platforms and a Big Data framework. The entire solution will be described under two headings below; Architecture and Social Media Data Analysis.

3.1 Architecture

The architecture of the solution can be split into three categorised sub sections. The first section is the Social Platform which is an external entity to the solution, the website itself to gather the data, and the big data framework to store the data.

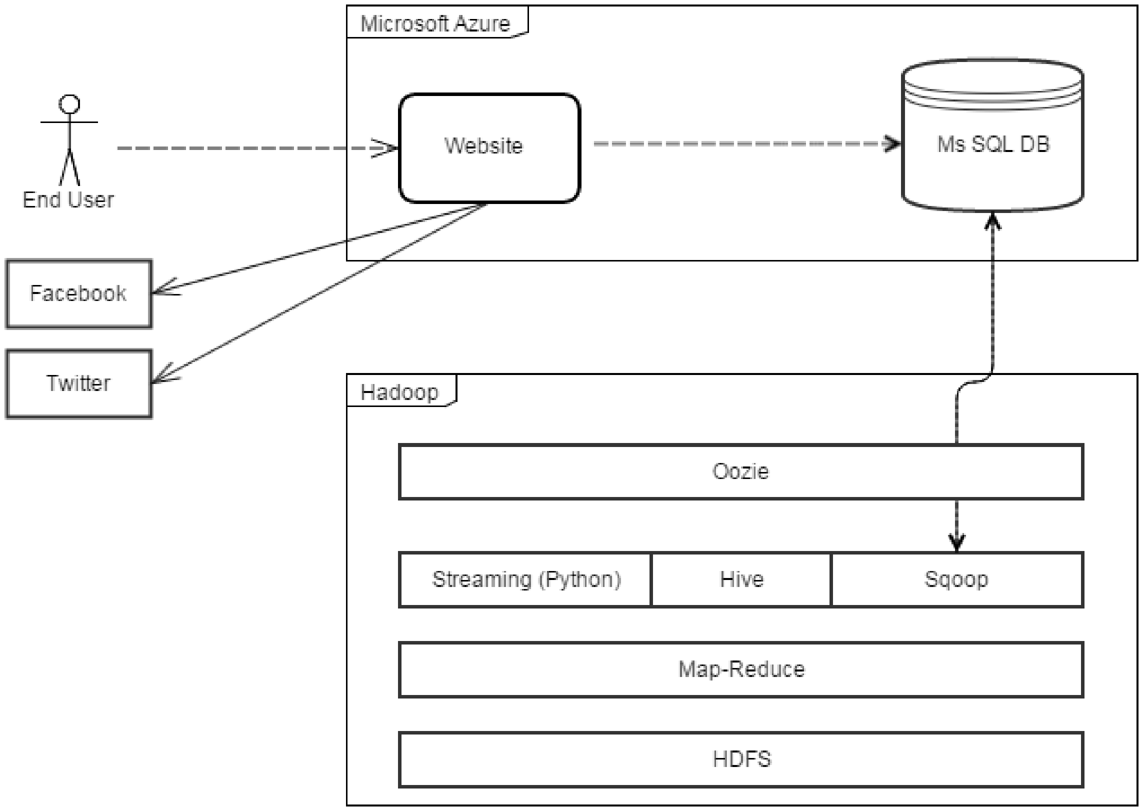
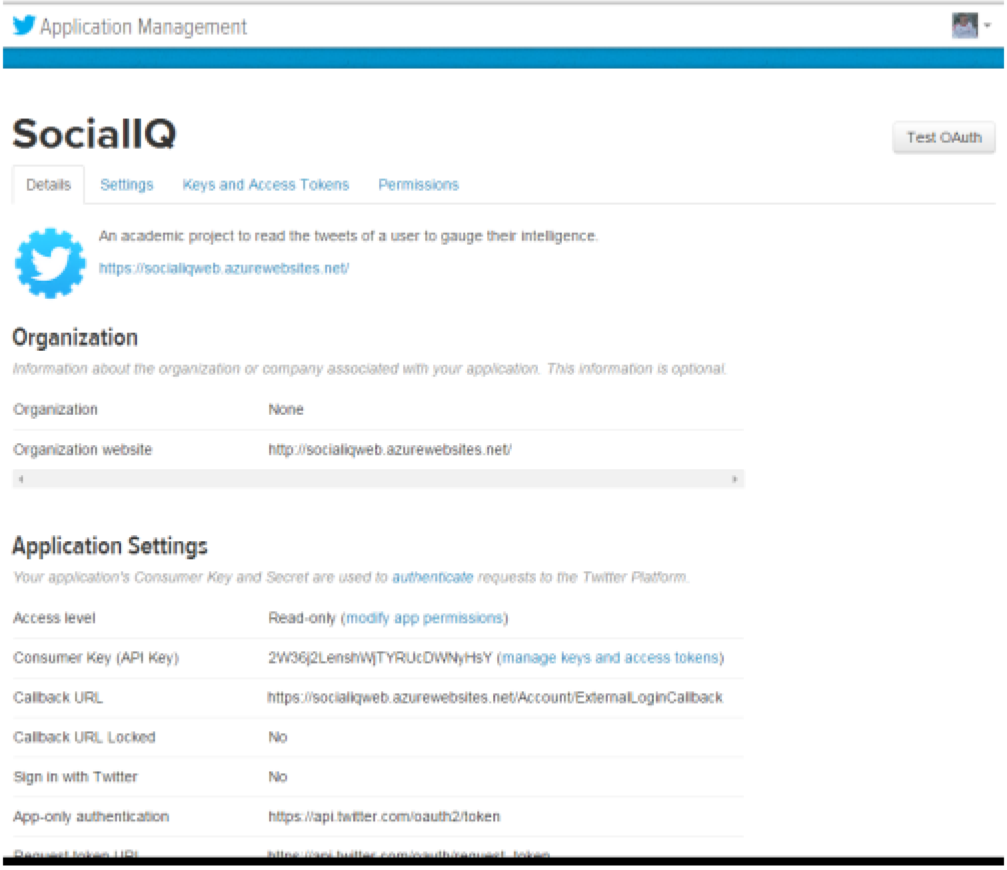


Figure 3-1 - Architecture Overview

3.1.1 Social Platforms

Social platforms can be a website or application which allows a user to hold an account and interact with. Common interactions may be posting messages, which is commonly known as status updates on Facebook or tweets on Twitter. Two of the best examples of social platforms are Facebook and Twitter. According to Statista.com, Facebook currently has 1.490 billion users, whereas Twitter currently has 316 million (Statista, 2015). There are thousands of social platforms but for the purpose of this research, these are the two which will be used.

A main benefit of choosing a popular social platform is not just that it is more than likely going to more available content, but there will also be an API to utilise. An API or Application Programming Interface is a set of functions or methods made available by an application to allow external applications interact with it. In this case, both Facebook and Twitter provide an API which will allow an application to request data. Before data can be requested, an application must first be created for both Facebook and Twitter. An important aspect of creating an application is choosing which permissions to request. When a user is going to authorise an application they can see which permissions the application seeking off them. Only the essential permissions will be sought after for both applications. The social media platform will provide an API key and a secret key. Making a request with both the API key and the secret key, the social media platform will provide an access token. This access token will be sent with each request as part of the header to obtain a user's data.



SocialIQ

An academic project to read the tweets of a user to gauge their intelligence.

https://socialiqweb.azurewebsites.net/

Organization

Information about the organization or company associated with your application. This information is optional.

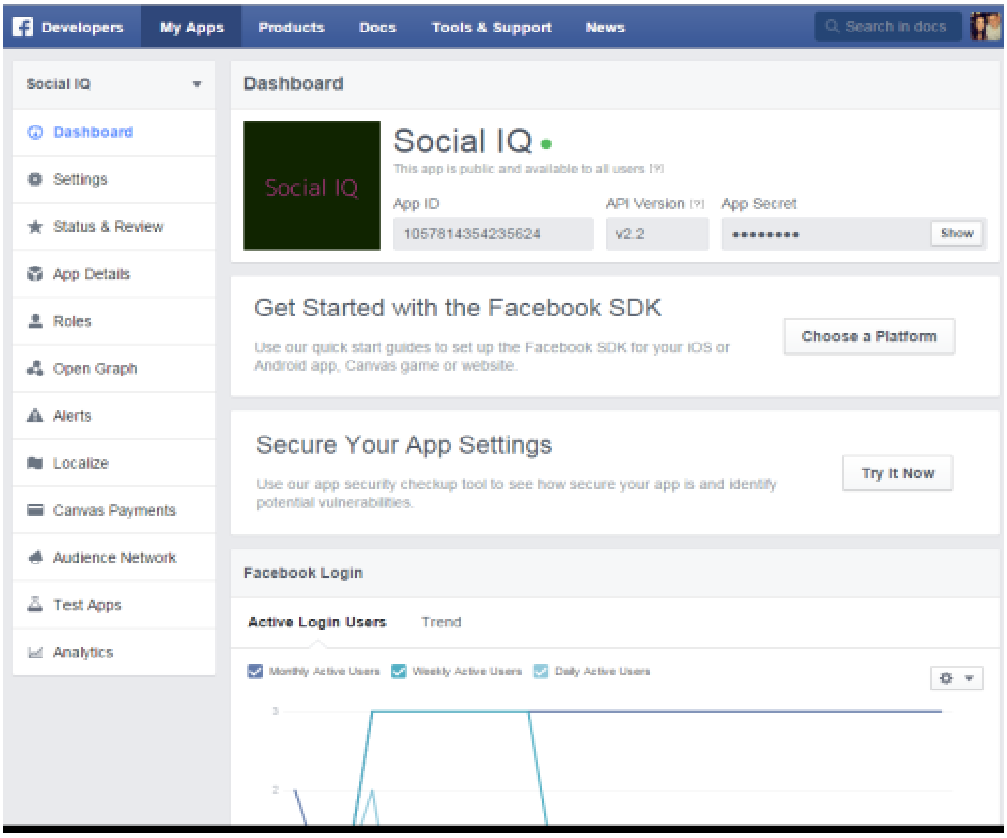
Organization	None
Organization website	http://socialiqweb.azurewebsites.net/

Application Settings

Your application's Consumer Key and Secret are used to authenticate requests to the Twitter Platform.

Access level	Read-only (modify app permissions)
Consumer Key (API Key)	2W36j2LenshWjTYRUcDWNyHsY (manage keys and access tokens)
Callback URL	https://socialiqweb.azurewebsites.net/Account/ExternalLoginCallback
Callback URL Locked	No
Sign in with Twitter	No
App-only authentication	https://api.twitter.com/oauth2/token
Request token URL	https://api.twitter.com/oauth/request_token

Figure 3-2 - Twitter app



Dashboard

Social IQ

This app is public and available to all users [?]

App ID: 1057814354235624

API Version [?]: v2.2

App Secret: ●●●●●●●●●

Get Started with the Facebook SDK

Use our quick start guides to set up the Facebook SDK for your iOS or Android app, Canvas game or website.

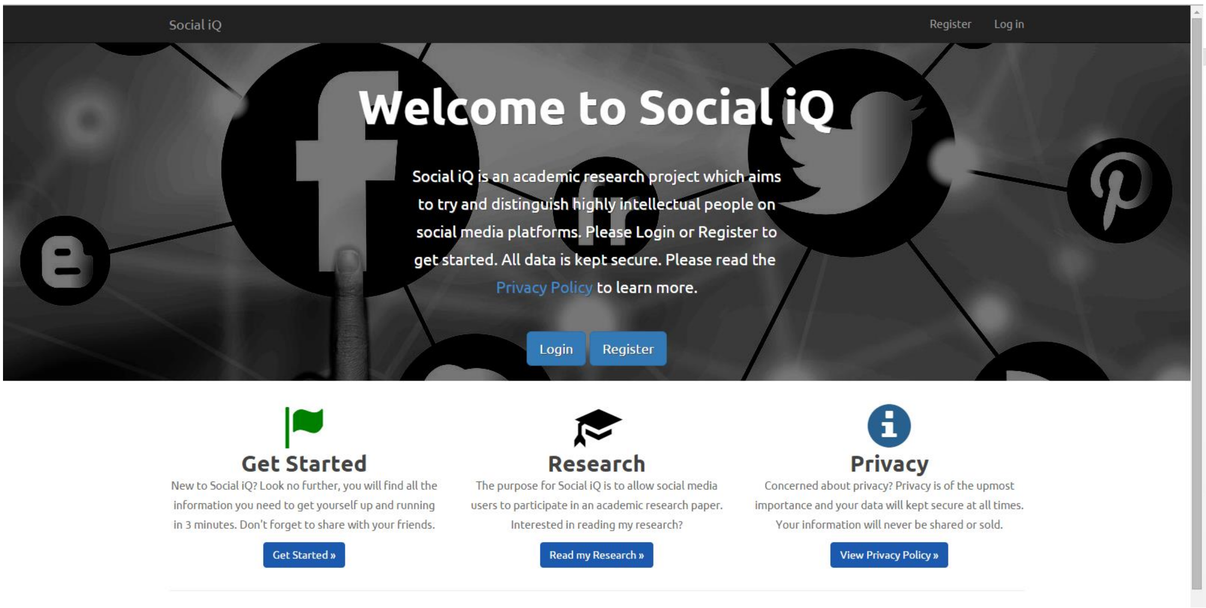
Secure Your App Settings

Use our app security checkup tool to see how secure your app is and identify potential vulnerabilities.

Figure 3-3 - Facebook app

3.1.2 Website

The website's main function is provide a way of a social platform user to authorise the application to request data from the social platform they use. Essentially that means a Facebook or Twitter user can navigate to http://socialiqweb.azurewebsites.net/ and authorise Social IQ to request their data from the social platform they choose.



Welcome to Social iQ

Social iQ is an academic research project which aims to try and distinguish highly intellectual people on social media platforms. Please Login or Register to get started. All data is kept secure. Please read the Privacy Policy to learn more.

Get Started

New to Social iQ? Look no further, you will find all the information you need to get yourself up and running in 3 minutes. Don't forget to share with your friends.

Research

The purpose for Social iQ is to allow social media users to participate in an academic research paper. Interested in reading my research?

Privacy

Concerned about privacy? Privacy is of the upmost importance and your data will kept secure at all times. Your information will never be shared or sold.

Figure 3-4 - Social IQ home page

The website is hosted in Microsoft's Azure Cloud platform and is developed using Microsoft's ASP.NET MVC framework. The programming language used to create the website did not particularly matter, but from previous experience and knowledge, it was quicker to use the Microsoft .NET platform and host in Microsoft Azure.

Once a user logins into the website, a new user record is stored in the SQL Server database which is also running in the Azure Cloud. The website makes asynchronous requests to each social platform the user has authorised. Once the request has returned, the data is stored an SQL table. A user can view all information store in the database related to them from the website.

3.1.3 Big Data Framework

Hadoop was chosen as the Big Data framework for this research. There were several reasons for this but the main two being that it is free and open source and although it is written in Java, it supports other programming languages through it Streaming module. The Hadoop framework is quite vast and can be called an ecosystem as it is often used in conjunction with many other frameworks. One of the main strengths of Hadoop is that it will run comfortably on commodity hardware and does not require expensive servers.

3.1.3.1 HDFS

Hadoop Distributed File System (HDFS) is a distributed file system designed to work on commodity hardware to provide high fault tolerance. The file system stores the application data across multiple data nodes and provides high throughput to the application data.

3.1.3.2 Map-Reduce

Map-Reduce is a software framework for working with large data sets across a distributed network. The Map-Reduce framework as the name suggests is split into two different sections. The "Mappers" and the "Reducers". Map-Reduce runs each task in parallel. Firstly the Mappers are run on the application data to sort the data. Once that has finished the Reducers are run on the input from the Mappers. A Reducer can be set to do many different thing which interact with the data, an example would be to count all the words in the application data, or to return the score for a tweet.

3.1.3.3 Streaming

The Hadoop Streaming utility allows for a scripting programming language to be used to write Map-Reduce jobs instead of the native Java programming language. Python, Perl and Ruby are all supported. Using the Streaming utility increases the time for each job to complete compared to Java. The main advantage of using a scripting language and particularly Python, is that there is a toolkit called the Natural Language Toolkit (NLTK) which provides a helpful platform to interact with human language data.

3.1.3.4 Sqoop

Apache Sqoop is a tool which can import or export from Hive or HDFS to a relational database. Sqoop has a connector for SQL Server databases and can be easily configured to connect to a SQL Server database running in Azure.

3.1.4 Alternative Methods

There are many alternative methods that could have been used to gather, store and analyse the data. Previous research by Dodd used a SQLite database to store tweets, he also stored the cleansed data in the same database instead of preserving the original tweet (Dodd, 2014). Hennessy stored the tweets she collected in a .csv file (Hennessy, 2014). Although this method worked great with a small dataset of 1500 tweets, it could not be considered for Big Data.

Storing data in a database and transferring it to a Big Data framework could be improved by bypassing the database stage and transferring the data directly into the Big Data framework. Microsoft Azure have launched a Machine Learning facility as well as HDInsight which is 100% Apache Hadoop in the cloud. This meaning that a tweet could be gathered from Twitter and stored directly in HDInsight without having to store in a database first. The machine learning facility supports Python code and can be run against the HDInsight dataset. This option would save a lot of time and extra storage and processing of data, but the only downside is not free. Azure provide a 1 hour free experiment duration but once that ends an Azure subscription is required. Although it's not very expensive to run, it took quite a bit of time to get the Map-Reduce jobs to work correctly would have accumulated into substantial fees. The free single-node cluster was a better choice to get started with and learn the Hadoop and machine learning frameworks.

3.2 Social Media Data Analysis

Social media data analysis is essentially the main part of this research. This section will describe how the data will be analysed and the type of process that will be followed to award each piece of data a score. The intrinsic quality score will be obtained by firstly checking the data for Punctuation and Typos, and secondly checking the Syntactic and Semantic Complexity.

This whole process will be run against each piece of data and the code will sit at the Streaming (Python) layer in the Hadoop ecosystem. Please see figure 3-1 above for an illustration of where the social media data analysis code will be.

3.2.1 Punctuation and Typos

It is important to remember that only English text will be evaluated as part of this research. Just analysing English punctuation and typing errors may be a simple process, with many external libraries available. However, social media data is not that straight forward. Jon Reed from Oxford Dictionaries has described the informal way of communicating online as "an alphabet soup of acronyms, abbreviations, and neologisms" (Reed, 2014). Social media platforms and in particular, Twitter, limit how many characters a user can post per message. Given this constraint, a user will likely opt for an informal writing method as opposed to writing multiple message. Reed goes on to describe that social media data today is filled with abbreviations, emoticons, abbreviations and even enclosing physical actions inside an asterix (*). Another common informal writing method nowadays is to use multiple punctuation marks in a sentence. For example, a sentence may contain extra question marks or exclamation marks for extra emphasis. Please see Appendix A for the full listing of acronyms.

Due to the informal writing style on social platforms, a lot punctuation and typing error would need to be excused. In fact, some typical errors may not be an error for a particular social platform. For example, on Twitter the use of '#' and '@' have their own meanings. A '@' typically followed by a word would usually be called a mention. An example might be "Hey @shaneburke22". This tweet is actually a mention to the Twitter user who has the username "shaneburke22". A '#' or hashtag is usually followed by a word or phrase. The hashtag sets the theme of the tweet. Twitter has a facility called trending, where the most common hashtags are listed for people to join the discussion or follow a story.

Bearing in mind the differences listed above against traditional English text, the check for punctuation and typing mistakes will need to be altered to accommodate informal English. The number of typing errors and punctuation mistakes will be still be counted but only after the data is parsed for the most common social media mistakes.

3.2.2 Syntactic and Semantic Complexity

Once the data has been correctly parsed it will move along to the syntactic and semantic complexity check. The syntactic checks each piece of data for sentences, and each sentence is checked for it syntax make-up which would also be called the readability of each sentence.

The semantic complexity looks at each word in a sentence. The Python nltk framework comes with a powerful feature called Part-Of-Speech tagging (POS). The POS tagger reads each work in a sentence and assigns a tag. This tag indicates whether the word is a noun, adjective, verb, etc. Appendix B contains the full list of POS tags and scores awarded. The pre-existing Penn Treebank corpus will be used. A training data corpus could be used instead but it would take a lot of time to create one to work with Facebook and Twitter data. The score awarded for each POS tag was decided upon by the frequency of the tag and the value of each tag in a sentence. The scoring system favours adjectives which describe nouns and pronouns in sentences. Verbs, nouns, pronouns and preposition all carry equal weighting, while all other tags only carry one tenth or one hundredth of that value.

4 Implementation

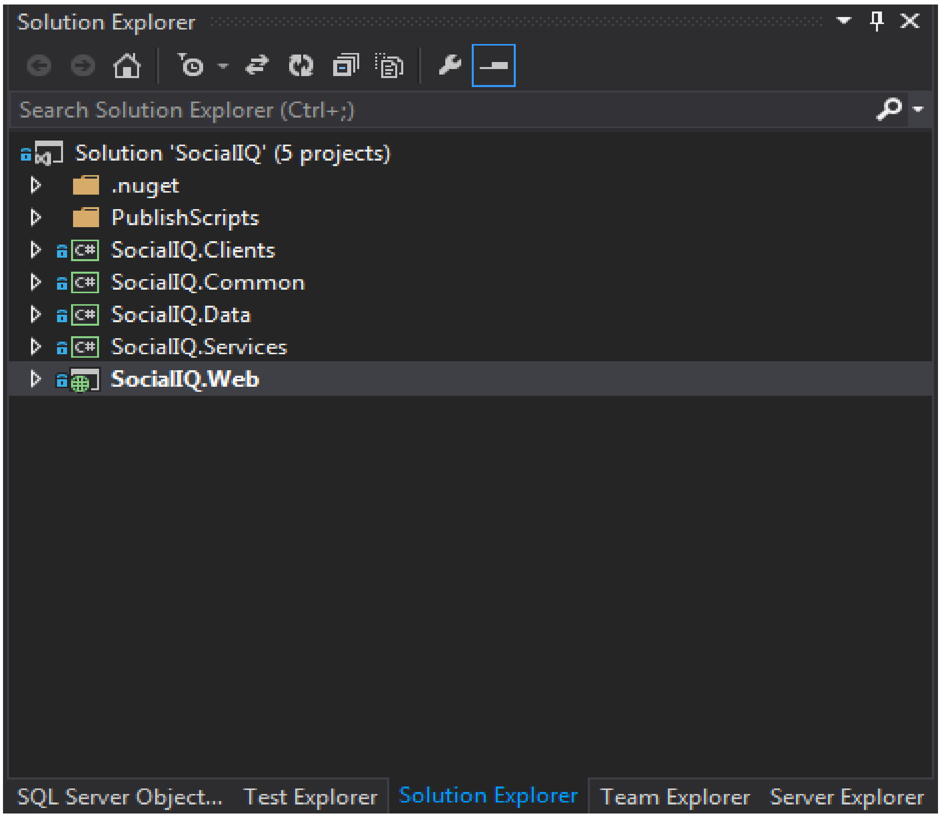
The following section will provide an overview of how a piece of data posted on a Social Media Platform is collected, stored, cleansed, and analysed including which technologies and frameworks have been used.

4.1 Data Collection

The primary step before data can be collection from a Social Media Platform is to create an application on each platform to obtain an API key and a secret key. Figures 3-2 and 3-3 above show the Twitter and Facebook applications respectfully. After both applications were created, it was time to develop a way to retrieve the data from each API. There are multiple ways to do this, but a website method was chosen. As mentioned in section 3.1.1 Social Platforms, both twitter and Facebook were chosen as the platforms to acquire the data. Twitter provides public access to tweets on their site as long as you request those tweets with an access token through their API. Facebook on the other hand do not provide public access to status updates or posts. Special permissions are required, and a user must first authorise an app with those permissions before it is possible to retrieve them. Unfortunately the Facebook app created did not receive those permissions, which will mean that there won't be any Facebook data to analyse.

4.1.1 Website development

A Microsoft .NET MVC website was developed using Visual Studio 2013 on the .NET 4.5.1 framework and the C# programming language. The website was split into five separate projects and following the SOLID principles for best practice. Figure 4-1 below shows the project structure of the website.



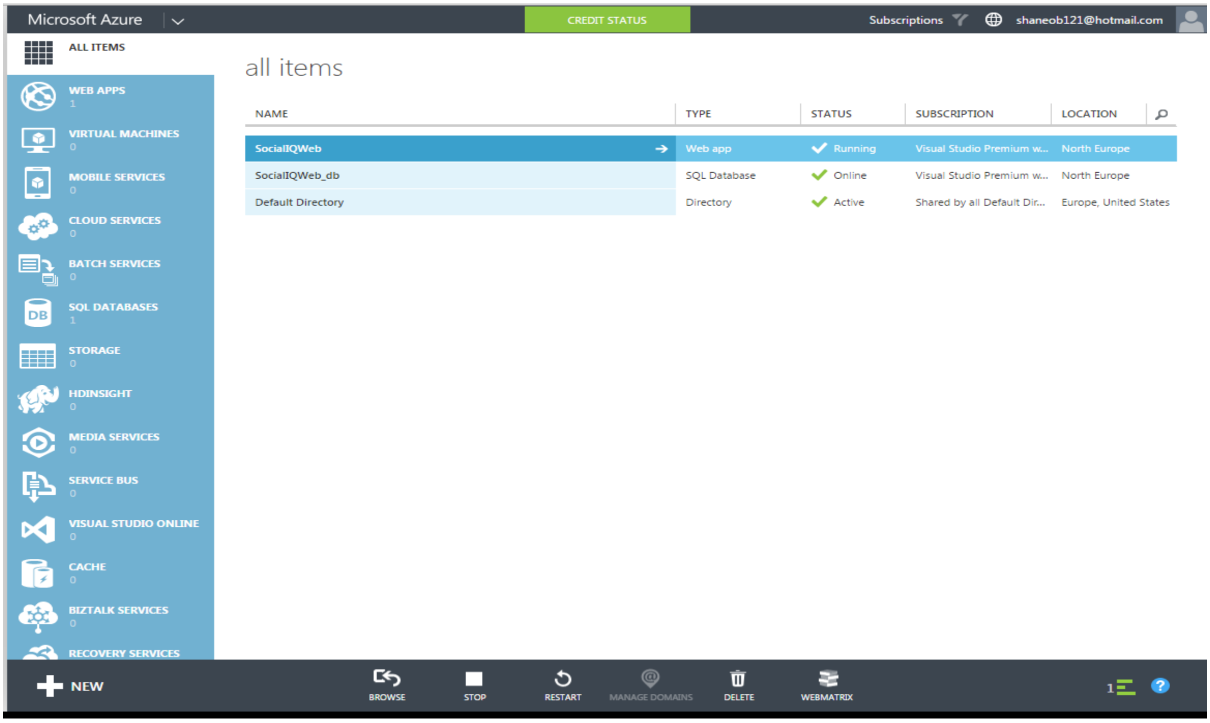
```
Solution 'SocialIQ' (5 projects)
  .nuget
  PublishScripts
  SocialIQ.Clients
  SocialIQ.Common
  SocialIQ.Data
  SocialIQ.Services
  SocialIQ.Web
```

Figure 4-1- Website project structure

SocialIQ.Web contains the traditional MVC part of the website. This project is the presentation part of the website. SocialIQ.Data contains the models used in the project. This project was built using Entity Framework and is separated into a different project because multiple projects in this solution utilise it. The SocialIQ.Clients projects contains a client for each Social Media Platform used. In this case there are two clients, Facebook and Twitter. The API and secret keys are help in this project, and the Social Media Platforms are only accessible from this project. To make it easier to interact with Twitter, a .Net library called CoreTweet was used. This library is a wrapper around the Twitter API to make it much faster to get an application up and running. It is licensed using the MIT License. The SocialIQ.Services projects contains a number of concrete services and their interfaces. The services are called from the presentation tier and either passed to the Data tier or the Clients tier depending on the request. An example would be if a user authorised the Twitter app, the successful request would be passed into the services tier. The service would then update the user object in the Data tier with the users Twitter Id. It would then make a request to the Twitter Client to retrieve the users Tweets based on their Twitter Id. Once the Client returned the Tweets, the service would store the data in the Data tier. The SocialIQ.Common project contains mappers and internal models used within the solution. The mappers are used to transfer one object to another. This is commonly used by a service once the Client has returned back a list of objects, the service maps those objects to another object which is understood by the Data tier.

4.1.2 Hosting

The website has been deployed to Microsoft Azure. Azure is a cloud platform that can host a variety of programming languages, frameworks and databases. Azure provides free hosting for websites, but with limited bandwidth and number of requests. For the purposes of this website, the free hosting is enough. Along with the website, the Microsoft SQL database is also hosted in Azure. Like the website, there are limits on the free database. There are a couple restrictions including a maximum of 1 GB can be stored in the database and the number current connections is limited.



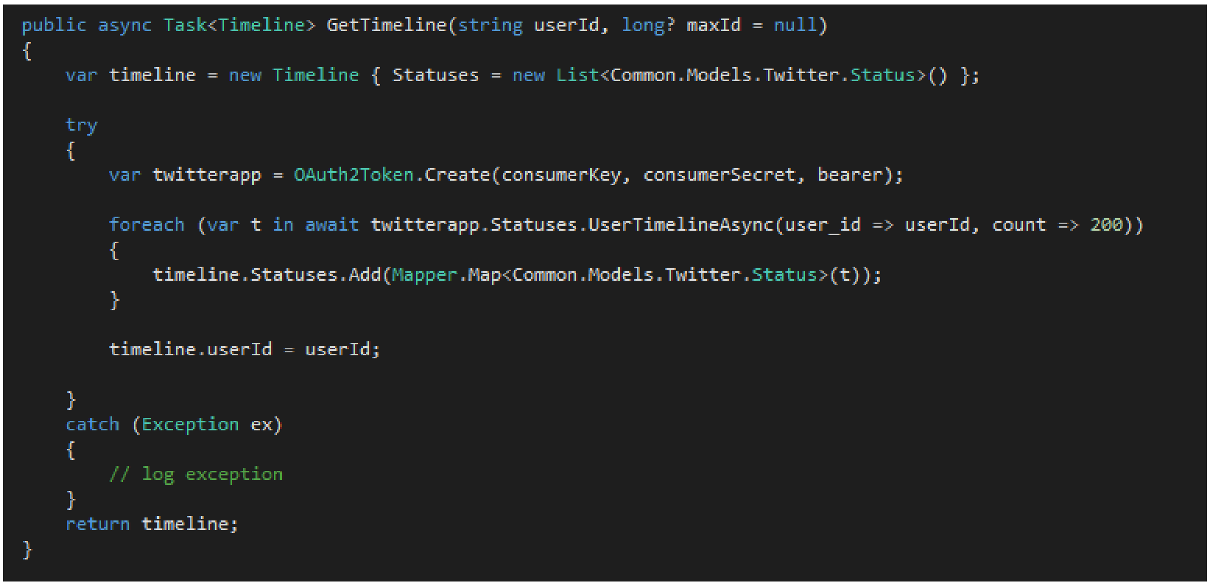
all items

NAME	TYPE	STATUS	SUBSCRIPTION	LOCATION
SocialIQWeb	Web app	✓ Running	Visual Studio Premium w...	North Europe
SocialIQWeb_db	SQL Database	✓ Online	Visual Studio Premium w...	North Europe
Default Directory	Directory	✓ Active	Shared by all Default Dir...	Europe, United States

Figure 4-2 - Azure dashboard

4.1.3 Requesting Social Media data

Once a user login into the website, they can authorize a Social Platform app. After they have authorised an app for the first time, a request is sent to that Social Platform requesting that users data. The code for this can be seen below in figure 4-3.



```
public async Task<Timeline> GetTimeline(string userId, long? maxId = null)
{
    var timeline = new Timeline { Statuses = new List<Common.Models.Twitter.Status>() };

    try
    {
        var twitterapp = OAuth2Token.Create(consumerKey, consumerSecret, bearer);

        foreach (var t in await twitterapp.Statuses.UserTimelineAsync(user_id => userId, count => 200))
        {
            timeline.Statuses.Add(Mapper.Map<Common.Models.Twitter.Status>(t));
        }

        timeline.userId = userId;

    }
    catch (Exception ex)
    {
        // log exception
    }
    return timeline;
}
```

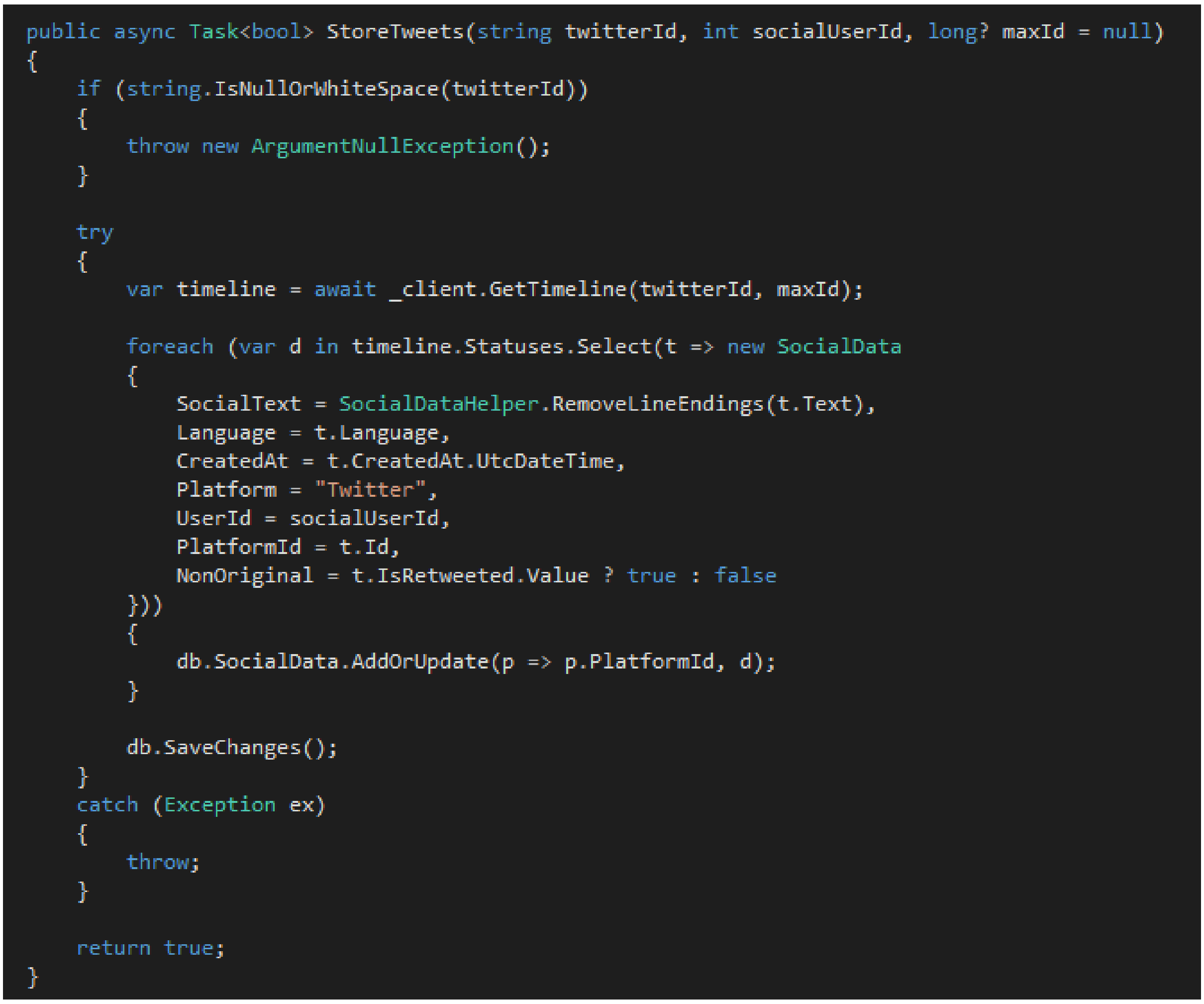
Figure 4-3 - Making Twitter request

The figure above, is a code block used to make a request to Twitter, to obtain the latest 200 tweets for a given Twitter user id. Twitter's recent changes to their API means that a maximum of 200 tweets can be returned in a single request, and a maximum of 3,200 tweets for a user. This unfortunately means that even if a user has tens of thousands of tweets, it is only possible to retrieve the last 3,200 using their API. In the above code block, both the method "GetTimeline" and the request to the Twitter are made asynchronously. Given that the website is running in a cloud infrastructure, using async calls drastically improves the performance. It takes on average about 15 seconds from the time the request to get the latest tweets is sent from the website to the time the request responds to the website. This includes sending a request to twitter to get the latest 200, and storing those 200 tweets in the database.

4.1.4 Storing Social Media data

Figure 4-3 shows how the data is requested, but in figure 4-4, it illustrates how the data is stored. Also it calls the above method to get the data to store. Once the below method executes, Social Media data has now been successfully retrieved and stored in the MS SQL database running in the Azure cloud.

The only modification to the original posted data is to remove line ending and replace with a whitespace. The reason for this is due to how the data is imported and stored in the Big Data framework which will be discussed later in this section.



```
public async Task<bool> StoreTweets(string twitterId, int socialUserId, long? maxId = null)
{
    if (string.IsNullOrWhiteSpace(twitterId))
    {
        throw new ArgumentNullException();
    }

    try
    {
        var timeline = await _client.GetTimeline(twitterId, maxId);

        foreach (var d in timeline.Statuses.Select(t => new SocialData
        {
            SocialText = SocialDataHelper.RemoveLineEndings(t.Text),
            Language = t.Language,
            CreatedAt = t.CreatedAt.UtcDateTime,
            Platform = "Twitter",
            UserId = socialUserId,
            PlatformId = t.Id,
            NonOriginal = t.IsRetweeted.Value ? true : false
        }))
        {
            db.SocialData.AddOrUpdate(p => p.PlatformId, d);
        }

        db.SaveChanges();
    }
    catch (Exception ex)
    {
        throw;
    }

    return true;
}
```

Figure 4-4 - Store tweets in database

4.1.5 Admin section

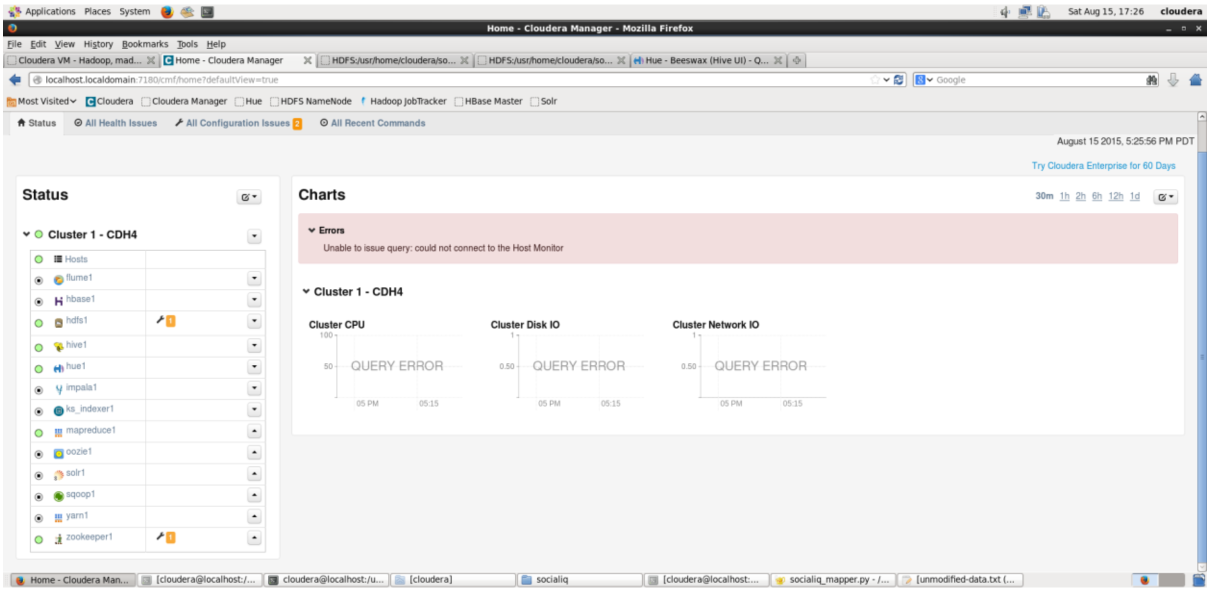
An admin section to the end user website was developed for a couple of reasons. Firstly an admin could create an account on behalf of a social media user. The only piece of information needed is the users Social Platform id. This id is not sensitive information, it would be of the same level as a username. The admin is able to request the latest posts/tweets of all the users, request their previous posts/tweets and update the users profile e.g. profile picture. The admin section makes it easy to update the ever growing database of data while using a smart phone, instead of having to load up a laptop with Visual Studio to access the API.

4.2 Data storage

At this point, each piece of Social Media data is stored in the MS SQL DB in Azure with an associated user id foreign key. This database is a great place to persist the data and display on the website when requested, however it is not the optimum place to analyse the data. There are several reasons for this, firstly because the website is public facing, the data can be requested at any time. The database is usually under load, whether it is storing new data from the website or it retrieving and passing data back to the website, too many requests would severally decrease performance. The main reason is that relational databases were not designed to handle vast amounts of data or Big Data. They cope every well with hundreds or thousands of rows of information, but more than that would cause problems with performance and efficiently analysing the data. For this reason, Hadoop was chosen as the Big Data Framework to analyse the data.

4.2.1 Setting up Hadoop

Hadoop is an ecosystem of components that work together to create a Big Data framework. The main component of Hadoop is the distributed file system called HDFS, but there are other components can store data instead like Hive. For the purposes of this research, it would have taken too long to configure each component to work in a single Virtual Machine (VM). Instead a pre-configured single-node cluster was used from Cloudera. The single-node cluster works with VirtualBox from Oracle and runs on the RedHat operating system. Figure 4-5 below shows a screenshot of the Hadoop cluster and the status of the components.



Home - Cloudera Manager - Mozilla Firefox

localhost.localdomain:7180/cmf/home?defaultView=true

August 15 2015, 5:25:56 PM PDT

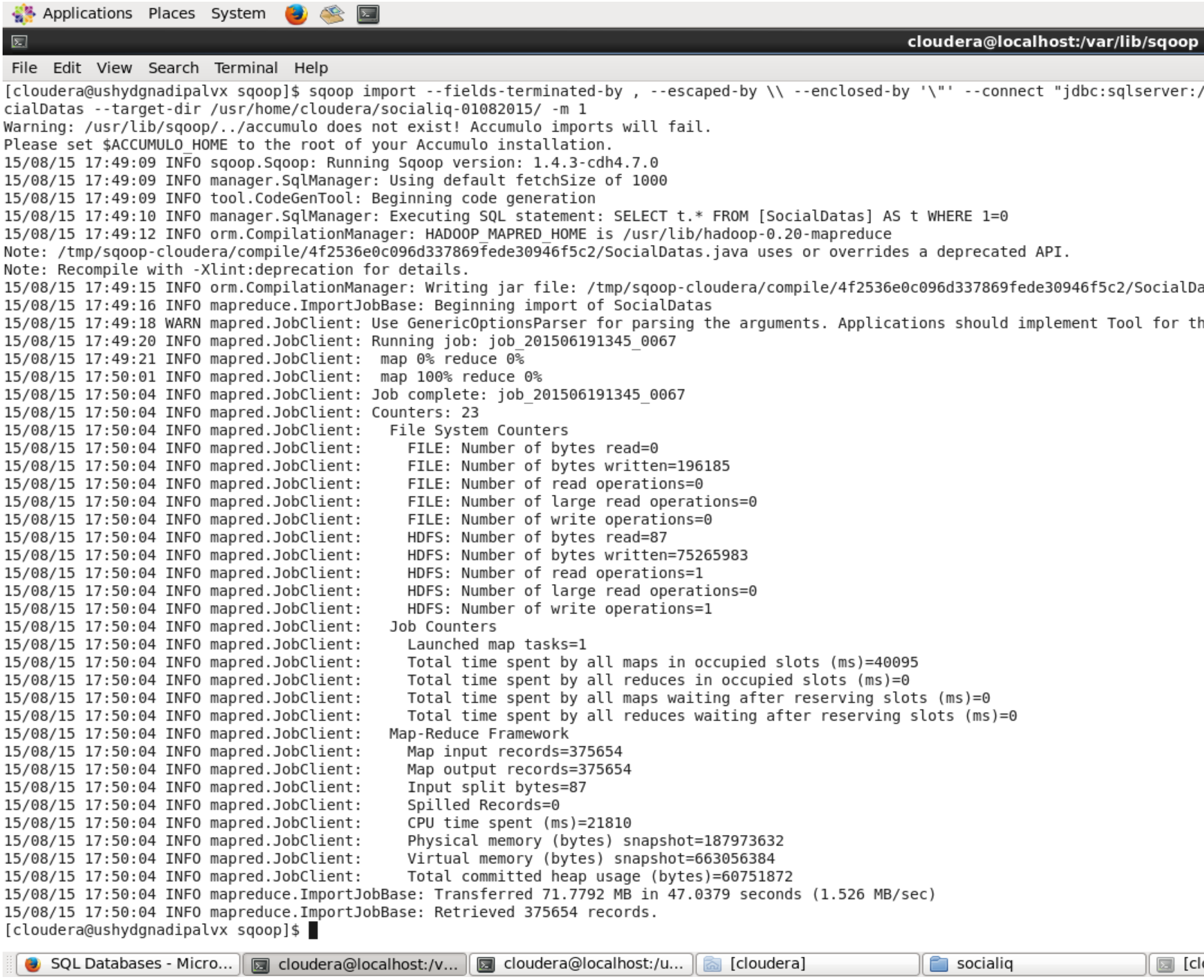
Errors

Unable to issue query: could not connect to the Host Monitor

Figure 4-5- Hadoop Cluster

4.2.2 Importing the data

To transfer the data from the database in Azure into HDFS in the VM an import tool is required. With the Hadoop cluster, it comes with the Sqoop component which is used to import or export database from an external source into Hadoop, or from Hadoop to an external source. The only additional requirement is the MS SQL connector. This is available from the Microsoft website called *Microsoft JDBC Driver 4.0 for SQL Server*. The figure below illustrates running the Sqoop import command through a terminal window.



```
[cloudera@ushydgnadipalvx sqoop]$ sqoop import --fields-terminated-by , --escaped-by \\ --enclosed-by '\"' --connect "jdbc:sqlserver:/
cialDatas --target-dir /usr/home/cloudera/socialiq-01082015/ -m 1
Warning: /usr/lib/sqoop/../accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
15/08/15 17:49:09 INFO sqoop.Sqoop: Running Sqoop version: 1.4.3-cdh4.7.0
15/08/15 17:49:09 INFO manager.SqlManager: Using default fetchSize of 1000
15/08/15 17:49:09 INFO tool.CodeGenTool: Beginning code generation
15/08/15 17:49:10 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM [SocialDatas] AS t WHERE 1=0
15/08/15 17:49:12 INFO orm.CompilationManager: HADOOP_MAPRED_HOME is /usr/lib/hadoop-0.20-mapreduce
Note: /tmp/sqoop-cloudera/compile/4f2536e0c096d337869fede30946f5c2/SocialDatas.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
15/08/15 17:49:15 INFO orm.CompilationManager: Writing jar file: /tmp/sqoop-cloudera/compile/4f2536e0c096d337869fede30946f5c2/SocialDa
15/08/15 17:49:16 INFO mapreduce.ImportJobBase: Beginning import of SocialDatas
15/08/15 17:49:18 WARN mapred.JobClient: Use GenericOptionsParser for parsing the arguments. Applications should implement Tool for th
15/08/15 17:49:20 INFO mapred.JobClient: Running job: job_201506191345_0067
15/08/15 17:49:21 INFO mapred.JobClient:  map 0% reduce 0%
15/08/15 17:50:01 INFO mapred.JobClient:  map 100% reduce 0%
15/08/15 17:50:04 INFO mapred.JobClient: Job complete: job_201506191345_0067
15/08/15 17:50:04 INFO mapred.JobClient: Counters: 23
15/08/15 17:50:04 INFO mapred.JobClient:   File System Counters
15/08/15 17:50:04 INFO mapred.JobClient:     FILE: Number of bytes read=0
15/08/15 17:50:04 INFO mapred.JobClient:     FILE: Number of bytes written=196185
15/08/15 17:50:04 INFO mapred.JobClient:     FILE: Number of read operations=0
15/08/15 17:50:04 INFO mapred.JobClient:     FILE: Number of large read operations=0
15/08/15 17:50:04 INFO mapred.JobClient:     FILE: Number of write operations=0
15/08/15 17:50:04 INFO mapred.JobClient:     HDFS: Number of bytes read=87
15/08/15 17:50:04 INFO mapred.JobClient:     HDFS: Number of bytes written=75265983
15/08/15 17:50:04 INFO mapred.JobClient:     HDFS: Number of read operations=1
15/08/15 17:50:04 INFO mapred.JobClient:     HDFS: Number of large read operations=0
15/08/15 17:50:04 INFO mapred.JobClient:     HDFS: Number of write operations=1
15/08/15 17:50:04 INFO mapred.JobClient:   Job Counters
15/08/15 17:50:04 INFO mapred.JobClient:     Launched map tasks=1
15/08/15 17:50:04 INFO mapred.JobClient:     Total time spent by all maps in occupied slots (ms)=40095
15/08/15 17:50:04 INFO mapred.JobClient:     Total time spent by all reduces in occupied slots (ms)=0
15/08/15 17:50:04 INFO mapred.JobClient:     Total time spent by all maps waiting after reserving slots (ms)=0
15/08/15 17:50:04 INFO mapred.JobClient:     Total time spent by all reduces waiting after reserving slots (ms)=0
15/08/15 17:50:04 INFO mapred.JobClient:   Map-Reduce Framework
15/08/15 17:50:04 INFO mapred.JobClient:     Map input records=375654
15/08/15 17:50:04 INFO mapred.JobClient:     Map output records=375654
15/08/15 17:50:04 INFO mapred.JobClient:     Input split bytes=87
15/08/15 17:50:04 INFO mapred.JobClient:     Spilled Records=0
15/08/15 17:50:04 INFO mapred.JobClient:     CPU time spent (ms)=21810
15/08/15 17:50:04 INFO mapred.JobClient:     Physical memory (bytes) snapshot=187973632
15/08/15 17:50:04 INFO mapred.JobClient:     Virtual memory (bytes) snapshot=663056384
15/08/15 17:50:04 INFO mapred.JobClient:     Total committed heap usage (bytes)=60751872
15/08/15 17:50:04 INFO mapreduce.ImportJobBase: Transferred 71.7792 MB in 47.0379 seconds (1.526 MB/sec)
15/08/15 17:50:04 INFO mapreduce.ImportJobBase: Retrieved 375654 records.
[cloudera@ushydgnadipalvx sqoop]$ 
```

Figure 4-6- Sqoop import

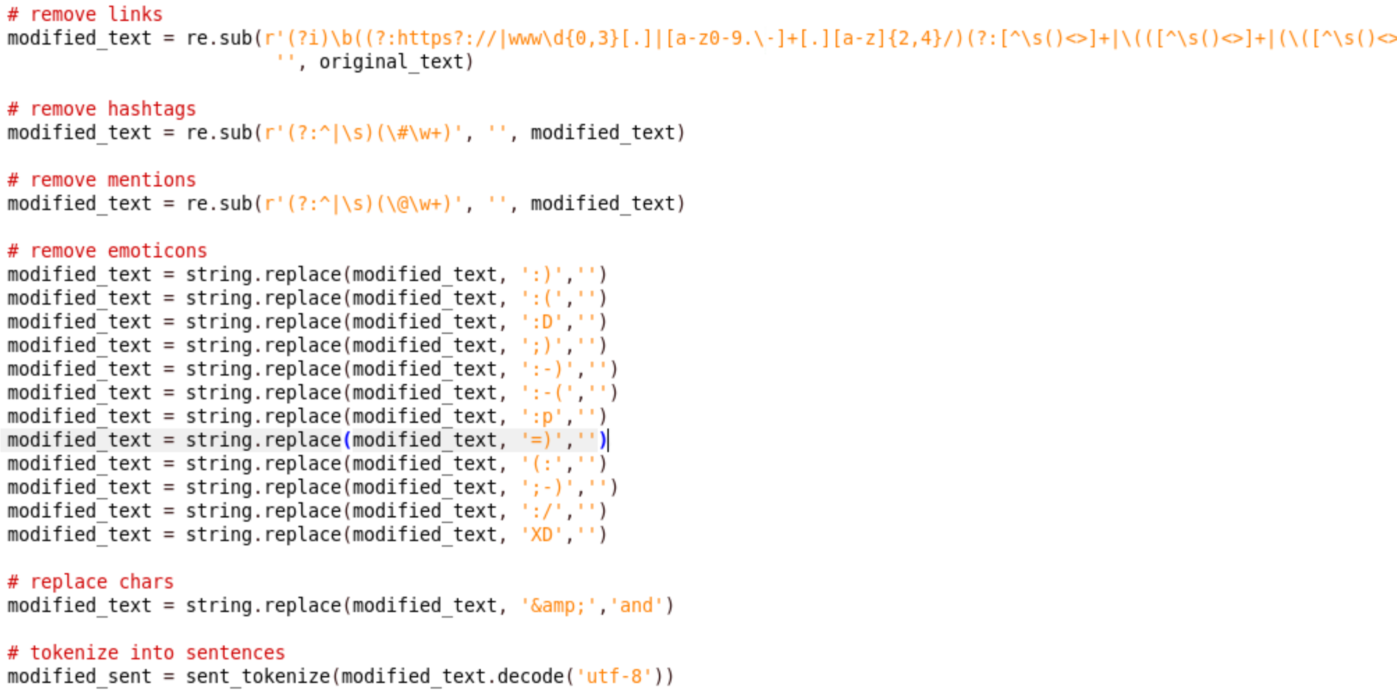
An important note to mention about importing the data is escape characters and enclose strings. The reason for this is because when Sqoop imports the each row in the database, it separates each column by a comma. Essentially it stores the database as a CSV file in HDFS. However, the column which holds the Social Media data can very easily contain a comma and this would cause issues when try to parse the file later on. A solution to this is to enclose each column in quotes. Sqoop allows a parameter to be added to specify which character should be used to enclose each column. Another note about this, although each column can be enclosed in quotes, it is very likely again that Social Media data could contain quotes. This would again cause issues when trying parse the file later on. Sqoop again has another parameter that can be specified to escape characters insides strings.

4.3 Data Cleansing

The main aim of the data cleansing step is to remove any meaningless or noisy data from each input. Meaningless data may refer to any piece of information which does not provide any purposeful information to the text. Noisy data is often the term used to describe meaningless data, but it also incorporates data which cannot be interpreted or understood. As previously mentioned in section

3.2 Social Media Data Analysis, text posted on social media platforms do not always contain words found in a proper English dictionary. For this reason, data cleansing must happen before the data is analysed.

Now that the data is stored in HDFS in the Hadoop cluster, Python is the programming language of choice to cleanse and process the data. A Map-Reduce program will be used for this. The data cleansing process will be executed at the beginning of the mapper. Figure 4-7 below displays the data cleansing code written in Python. The code below very simply tries to remove all links, hashtags and mentions using regular expressions. After that, the modified text then tries to remove eleven of the most popular emoticons. The popular emoticons were analysed by DataGenetics (Berry, 2010) in 96 million tweets. Although there were over 2200 emoticons found, for the purposes of this study, the top eleven will be removed. The next step in the cleansing process is to replace certain characters found. When the data is imported in HDFS, the ampersand '&' character is commonly seen as the character reference value or "&amp;". This value occurs very often so it is replaced with the word 'and'. Once the text has been cleansed, it is decoded to 'utf-8' and tokenised into sentences using nltk library.



```
# remove links
modified_text = re.sub(r'(?i)\b((?:https?://|www\d{0,3}[.]|[a-z0-9.\-]+[.][a-z]{2,4}/)(?:[^\s()<>]+|\(([^\s()<>]+|(\([^\s()<>
                        '', original_text)

# remove hashtags
modified_text = re.sub(r'(?:^|\s)(\#\w+)', '', modified_text)

# remove mentions
modified_text = re.sub(r'(?:^|\s)(\@\w+)', '', modified_text)

# remove emoticons
modified_text = string.replace(modified_text, ':)','')
modified_text = string.replace(modified_text, ':(','')
modified_text = string.replace(modified_text, ':D','')
modified_text = string.replace(modified_text, ';)','')
modified_text = string.replace(modified_text, ':-)','')
modified_text = string.replace(modified_text, ':-(','')
modified_text = string.replace(modified_text, ':p','')
modified_text = string.replace(modified_text, '=)','')
modified_text = string.replace(modified_text, '(:','')
modified_text = string.replace(modified_text, ';-)','')
modified_text = string.replace(modified_text, ':/','')
modified_text = string.replace(modified_text, 'XD','')

# replace chars
modified_text = string.replace(modified_text, '&amp;','and')

# tokenize into sentences
modified_sent = sent_tokenize(modified_text.decode('utf-8'))
```

Figure 4-7 - Data cleansing with Python

4.4 Data analysis

Analysing the data is the single most important part of the research. This step is split over the mapper and the reducer program. This step will be described below as part of the mapper and the reducer.

4.4.1 Mapper

The mapper is a python script that runs across the entire data set. The output of this script is then passed into the reducer. The mapper contains most of the logic for the data analysis step. The mapper reads each line in HDFS which is essentially a CSV format of the MS SQL database in Azure.

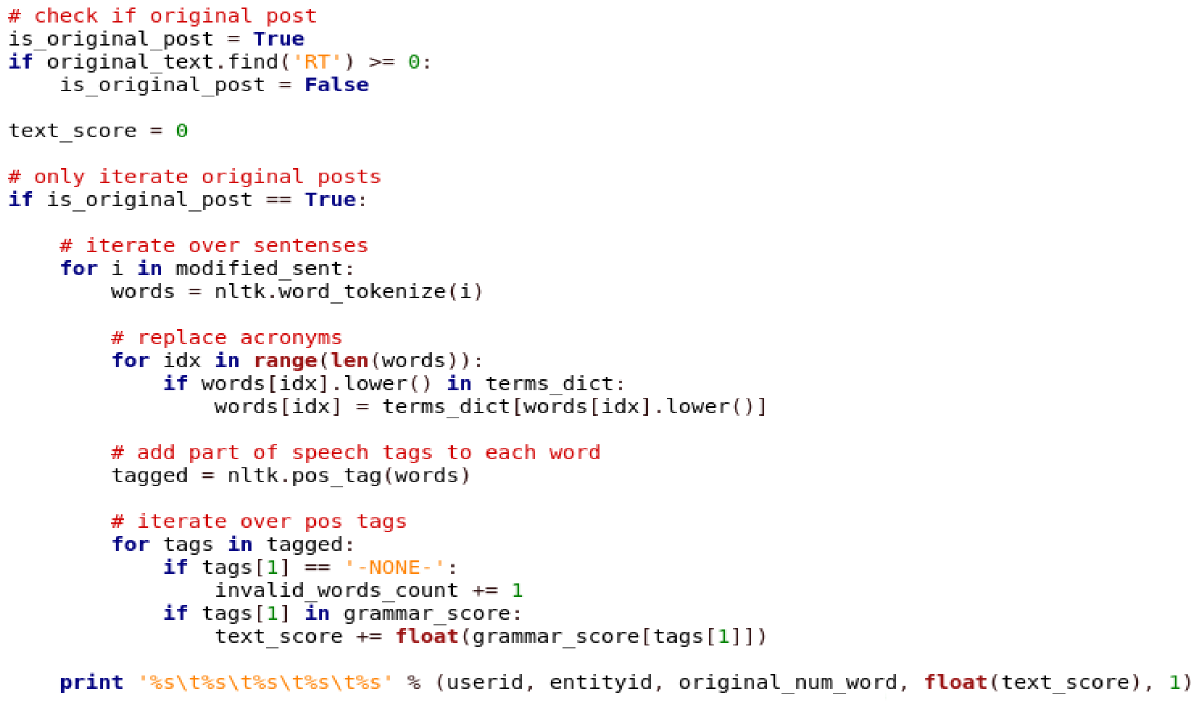
The line is then parsed to extract the social media text column, the user id and the line id or primary key. After the data has been parsed, it is cleansed, as described in section 4.3 Data Cleansing.

The next step checks if the data is an original post or not. A Twitter retweet is not analysed because it was posted by another user so it cannot be awarded a score to the current user. The program does a quick check to find “RT”. If this is present, the programs just moves onto the next line, otherwise this line is analysed. As seen in Figure 4-7, the text is tokenised into sentences using the nltk framework. Each sentence is analysed separately and awarded a score. The score for each sentences is summed at the end to award a score to the text.

The mapper then iterates over each sentence. The sentence is then tokenised into words using the nltk framework. Each word is checked against a csv file containing a list of acronyms and their full text meaning. The full list of acronyms can be seen in Appendix A. If an acronym is found, it is replaced with its full text meaning. This process is used to provide a better quality outcome, as the nltk framework does not understand acronyms.

Once the acronyms have been replaced, the nltk framework is used again to assign each word a part-of-speech tag (POS tagging). This process may also be known as lexical categories or grammatical tagging. A context free parser is used and each word is awarded a tag based on its definition. The tags signify whether a word is a noun, adjective, verb or any English part of speech type. A score is awarded to each work depending on which tag it has. Please see Appendix B for the full list of tags, their meaning and the score awarded for each. Although it is not shown in the list of tags, if a word is assigned the “-NONE-“ tag it is believed to be a misspelt or invalid word.

The mapper then outputs the user’s id, the primary key of the line, the number of words in the text, the score awarded to the text and a counter. These values are tab separated, and will be feed into the reducer. The mapper code can be seen in figure 4-8, with the output on the last line.



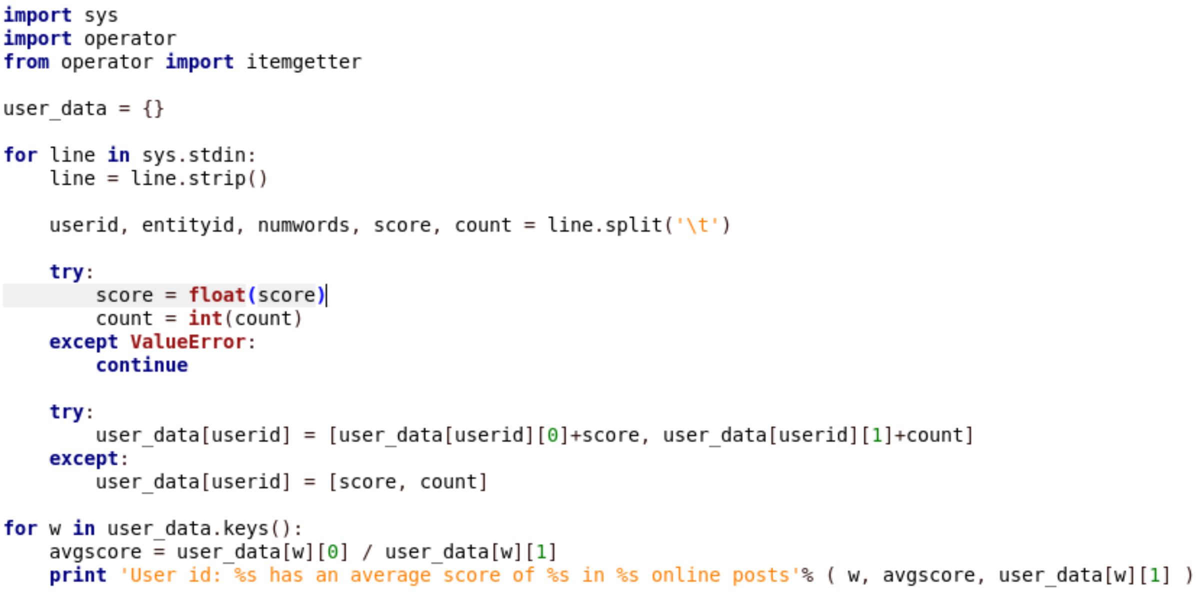
```
# check if original post
is_original_post = True
if original_text.find('RT') >= 0:
    is_original_post = False

text_score = 0

# only iterate original posts
if is_original_post == True:

    # iterate over sentenses
    for i in modified_sent:
        words = nltk.word_tokenize(i)

        # replace acronyms
        for idx in range(len(words)):
            if words[idx].lower() in terms_dict:
                words[idx] = terms_dict[words[idx].lower()]

        # add part of speech tags to each word
        tagged = nltk.pos_tag(words)

        # iterate over pos tags
        for tags in tagged:
            if tags[1] == '-NONE-':
                invalid_words_count += 1
            if tags[1] in grammar_score:
                text_score += float(grammar_score[tags[1]])

    print '%s\t%s\t%s\t%s\t%s' % (userid, entityid, original_num_word, float(text_score), 1)
```

Figure 4-8 - Data analysis mapper

4.4.2 Reducer

The role of the reducer is to take the input from the mapper and perform an action on the data, and output the result. There are two reducers written in Python used within this research, which means there are two separate Map-Reduce jobs. The mapper stays the same for both, but the reducer is slightly different. The first reducer very simply outputs the primary key of each row along with the score value. This is picked up Sqoop to perform an export which will update the MS SQL database running in Azure.

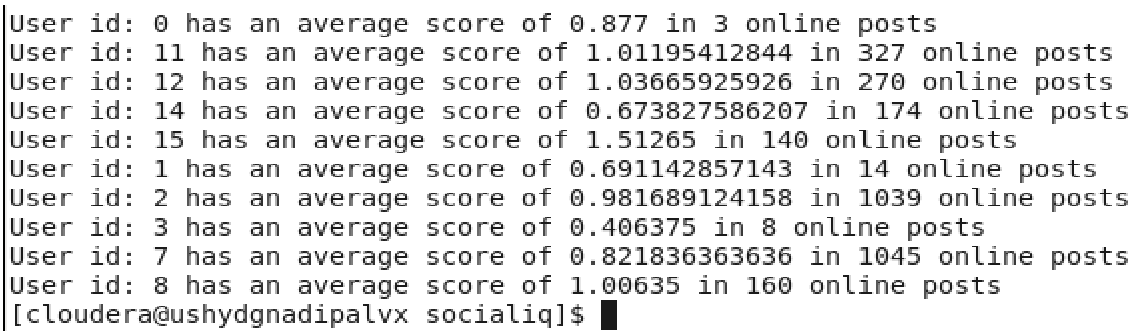
The second reducer gets the average of every unique users score by the user id value sent on by the mapper and summing up all the score values. The reducer also sums up the counter value from the mapper, and once the reducer has finished reading each line, the score value for each user is divided by the counter. This will provide the average score for the number of social media data the mapper has processed. Figure 4-8 is the entire reducer code.



```
import sys
import operator
from operator import itemgetter

user_data = {}

for line in sys.stdin:
    line = line.strip()

    userid, entityid, numwords, score, count = line.split('\t')

    try:
        score = float(score)
        count = int(count)
    except ValueError:
        continue

    try:
        user_data[userid] = [user_data[userid][0]+score, user_data[userid][1]+count]
    except:
        user_data[userid] = [score, count]

for w in user_data.keys():
    avgscore = user_data[w][0] / user_data[w][1]
    print 'User id: %s has an average score of %s in %s online posts'% ( w, avgscore, user_data[w][1] )
```

Figure 4-9 - Data analysis reducer

The output of the reducer above prints out the user id, the average score and the number of online posts analysed. Figure 4-10 below is the output of the reducer run on a sample dataset. As illustrated below, the user with id 15 provides the highest quality of data with a score of 1.51264 in 140 online posts, whereas user 3 only has a score of 0.406375 in 8 online posts.



```
User id: 0 has an average score of 0.877 in 3 online posts
User id: 11 has an average score of 1.01195412844 in 327 online posts
User id: 12 has an average score of 1.03665925926 in 270 online posts
User id: 14 has an average score of 0.673827586207 in 174 online posts
User id: 15 has an average score of 1.51265 in 140 online posts
User id: 1 has an average score of 0.691142857143 in 14 online posts
User id: 2 has an average score of 0.981689124158 in 1039 online posts
User id: 3 has an average score of 0.406375 in 8 online posts
User id: 7 has an average score of 0.821836363636 in 1045 online posts
User id: 8 has an average score of 1.00635 in 160 online posts
[cloudera@ushydgnadipalvx socialiq]$ 
```

Figure 4-10 - Reducer sample output

5 Observation

In total, over one million (1,012,759) tweets were gathered from 390 different Twitter users. Just over 19.73% (199,884) of the total tweets were retweets. On average a Twitter user retweets one in every five of their tweets. For the purposes of this study, retweets were not analysed because they did not originate with user who retweeted it.

The average tweet contains 89.59 characters, 13.4 words and 1.87 sentences. The table below shows the top most popular words in the gathered tweets, excluding "stop words". Stop words are a term used by the nltk framework to describe words which have little lexical content and they don't distinguish one text from another. Please see Appendix C for the entire list of stop words. Although sentiment analysis was not performed on the data set, it appears as if there would be a positive sentiment outcome going by the most popular words.

Word	Occurrences
Love	36023
New	33909
Get	32686
Like	32138
One	29517
See	28811
Thanks	27660
Great	27464
Good	27092
Day	25764

Table 5-1 – Top 10 most popular words

The main observation noticed from Social Media data and in particular data from Twitter, is the language used. Aside from the use of acronyms, emoticons, hashtags and mentions, some words posted are altered so much that they cannot be interpreted correctly. A common trend appears to be removing vowels from words. Given the fact that Twitter limits the number of characters allowed per message, it is understandable that users will come up with ways of tweeting their message inside 140 characters. In contrast, a message posted on Facebook does not need to be altered to be able to post as one message. This has a big impact on the grammar used.

Another observation is the context of tweets, or the lack of context. A user can post a tweet about anything they wish, but when trying to analyse the tweet, there isn't always context around it. An example tweet is "well done". A user tweeting this message is not providing any context around the tweet. Also, it is not a valid grammatical English sentence. It is difficult to analyse this tweet or try to distinguish if it was posted by an intelligent person or not.

Twitter mentions and hashtags are an integral part of the platform. It is very likely that a tweet will contain at least one of these conventions if not both. Users of the platform integrate both of these conventions into their messages and help form part of their sentences. A tweet collected from @Emssilee as part of this research is "Welcome to the amazing world of #twitter @AoifeNagle2, prepare to be addicted!". This tweet appears to be very understandable, the hashtag is referring to twitter itself and the mention is aimed at a friend called Aoife Nagle. This tweet parsed by the mapper described in section 4.4.1 would actually read "Welcome to the amazing world of, prepare to be addicted!" which doesn't actually make sense because we don't know what world the user is referring to. This is why a context free parser is used and each word is assigned a score independently. Albeit, this tweet's score doesn't drastically change when parsed with a context free parser, it only loses two words.

I did considering trying to remove '@' and '#' instead of the entire word and leaving it up to the nltk framework to decide if it would make sense of the word or not, and in most cases it couldn't. In the tweet above, the user AoifeNagle2 would not be recognised as a word. John Dodd attempted to solve this issue with user mentions by replaces the user mention with the text "At user" (Dodd, 2014). This method looks like it would work great for one user mention at the beginning of a sentence, but if there were more than one user mention, then the sentence would no longer make sense. Take this tweet "@shaneburke22 @iBiryukov @conorcost The 28th is a good weekend for me, I should be available to Chaperone all weekend :-)" by @aaronredmond as an example. Using Dodd's technique this would be interpreted as "At user At user At user The 28th is a good weekend for me, I should be available to Chaperone all weekend :-)". I think removing user mentions is a better option, and which leaves you with "The 28th is a good weekend for me, I should be available to Chaperone all weekend :-).

After analysing the collected tweets, a total of 1,020,239 user mentions were found with 31.1% (317,296) of them being unique. The table below displays the top 10 user mentions in order of most used.

User mention	Occurrences
@Angelluisr	1749
@KevinHart4real	1713
@plantetjedward	1656
@iamwill	1606
@TheNotoriousMMA	1446
@TubeSoccerAM	1401
@JLO	1274
@YouTube	1254
@ronanofficial	1245
@elissakh	1243

Table 5-2 - Top 10 most popular user mentions

Another tweet example from @Emssilee looks like "#HappyFriday! My Friday was made even better when I saw #DLRCoCo are fixing the #cyclelanes around #Sandyford!". This tweet example shows how hashtag phrases are used to form part of sentences. Even if the '#' was removed from '#DLRCoCo', nltk would still not be able to make sense of the word. The word is actually an acronym for "Dún Laoghaire-Rathdown County Council". The other hashtag in that sentence for "cycle lanes" is much more legible, but even so, it would still be as difficult to extract and parse into a meaningful phrase. A total of 359,864 hashtags were in found in the entire collection of tweets analysed. Marginally over 30% (109,848) of the hashtags were found to be unique. Table 5-3 below shows the top 10 most occurring hashtags.

Hashtag	Occurrences
#retweet	3639
#artwork	2137
#drawing	2000
#art	1701
#twitart	1523
#MarRef	1435
#mindfulness	1366
#1	1195
#sketch	1118
#ink	1028

Table 5-3 - Top 10 most popular hashtags

Another popular convention used on Twitter are acronyms. As previously mentioned in section 3.2.1, Appendix A contains the entire list of acronyms catered for. There were a total of 245,085 acronyms found within the data, 121 unique acronyms and only 9 acronyms were used once. Table 5-4 shows the top 10 most popular acronyms below excluding the acronym ‘rt’.

Acronym	Meaning	Occurrences
Lol	Laugh out loud	5797
w/	With	3975
Omg	Oh My God	2238
Thx	Thanks	1351
Dm	Direct message	1108
Pm	Private message	794
Cc	Carbon copy	773
Ppl	People	531
Mt	Modified tweet	531
Fb	Facebook	430

Table 5-4 - Top 10 most popular acronyms

6 Evaluation

This research set about trying to identify highly intelligent people by using machine learning on their social media data. After analysing all 812,875 tweets collected, a score was awarded to each tweet. The average score of each user was also collected, with the highest average score being 1.63 from @BryanCranston. The table below shows the top 10 users based on their average score. The entire list of 390 twitter usernames can be found in Appendix D.

Twitter Name	Name	Occupation	Score
@BryanCranston	Bryan Cranston	Actor	1.63171
@rupertmurdoch	Rupert Murdoch	Business magnate	1.62733
@ActuallyNPH	Neil Patrick Harris	Actor	1.54105
@Redknapp	Harry Redknapp	Football manager	1.53596
@Ed_Miliband	Ed Miliband	Politician	1.52174
@StephenAtHome	Stephen Colbert	Television host	1.48432
@ConanOBrien	Conan O'Brien	Television host	1.43993
@GardaTraffic	Garda Traffic	Law enforcement	1.42341
@jeffweiner	Jeff Weiner	CEO LinkedIn	1.41450
@Lord_Sugar	Alan Sugar	Business magnate	1.38676

Table 6-1 - Top 10 users

The table above contains a diverse group of different individuals who all share one common trait, to be recognised as a standout figure in their profession, with the exception of @GardaTraffic. This twitter account is run by the traffic branch of An Garda Síochána, who are the Irish police department. This account is very active with over 15,300 tweets and 162,000 followers. The account frequently tweet information about traffic and major events, as well as tweeting about speeding drivers and major accidents. Top of the list, Bryan Cranston is a well-known American actor, but with only 259, it's not a lot of data to analyse. In second place with 1543 tweets is Australian/American business man Rupert Murdoch. In 2015, Forbes listed Murdoch as the 77th richest man in the world (Forbes, 2015). Neil Patrick Harris comes in third place with over 2300 tweets. He is best known as an actor but also a writer, producer, director, comedian and singer. Just behind Harris, is Harry Redknapp, who has tweeted on 186 occasions. He is a well-known English premier league manager and pundit. The former leader of the English Labour Party, Edward Miliband is in fifth place and has 4807 tweets to his name. Sitting just below is Stephen Colbert with 3683 tweets. Colbert is an American television host of The Colbert Report and an author. Dropping down the list by almost 0.5 is Conan O'Brien with 2169. Conan similarly to Colbert is an American television host of his own show called Conan. Coming in at ninth is Jeff Weiner, the CEO of LinkedIn who has tweeted 2812. Rounding out the top 10 is British business magnate and political advisor Alan Sugar. Sugar by and large the most active twitter in the top 10 with over 33,700 tweets.

The score awarded to each tweet is not based on the users' intellect, instead it is based on the words used in each tweet. It would be impossible to try and assess a user's intelligence based on a collection of out of context words, which is what Twitter appears to be at times. The awarded score identifies that this user consistently tweets long messages and opts for more proper words as opposed to regularly using hashtags for words or phrases. After observing the data collected from Twitter, it was decided that tweets with no punctuation and typos would not be penalised as previously mentioned in section 3.2.1. The reason for this is because tweets generally don't punctuation, and typos are very common. With the character restriction on Twitter, users will often amend a word to make it fit into one tweet instead of posting two tweets. I decided that I would not penalise the user as they are adapting their message to restrictions of the platform.

It is not suffice to say that a user with a higher score is of higher intelligence. A higher score indicates that a user tweets longer messages, using more adjectives and less hashtags or mentions, as well as less incorrectly spelt words. In relation to the original question, the score awarded does not answer the question.

Natural language processing (NLP) is becoming a widely used tool for companies to gain an insight into their customers. There are many applications that NLP can be applied to, the company and customer is just one. NLP is a great tool when it is applied correctly. Firstly, there needs to be context around the data to be analysed, instead of gathering random data and trying to pull meaningful insights from it. The nltk framework provides many great tools like tokenising text into sentences and words, also POS tagging is a very powerful tool. The nltk framework can only do so much, and the more consistent the data used the better the results. For the purposes of this study, nltk was very helpful in parsing the data, but it struggles with the informal language found on Twitter. Better results could be achieved by using a Twitter specific natural language toolkit which could parse hashtags into meaningful words or phrases. As the language used on social media platforms is ever changing, the toolkit would need to be able to evolve to cope with the ever changing content.

The platform created for this research I believe to be stable and easily extendible to another research project around social media data. The website can be extended to connect with any number of other social media platforms. It can be hosted free of charge in Azure with 1 GB of free database storage. It would be relatively inexpensive to improve the performance of the website by adding a new instance to the cloud platform or acquiring extra storage if needed. The Hadoop single-node cluster was very quick to install from Cloudera. It was free of charge and required no setup tasks. The benefit to using a Hadoop cluster is that it can be vastly improved by adding a new node. The new node can be an old pc / laptop that is no longer used or a VM running on a different computer. It does not cost a lot to improve the Big Data framework. The only change that would really need to be made from this research is the Map-Reduce job to process the data. The architecture of this research could very easily be modified to perform sentiment analysis on the collected data instead of awarding a score based on the POS.

The entire dataset of over 1 million tweets analysed was 343.38 megabytes in size. This dataset could not be considered Big Data. The framework created handled this volume of data with ease, and could easily be improved if a larger dataset was obtained. The Sqoop import from the SQL database in the Azure cloud into Hadoop cluster took a little under 2 minutes to complete at about 3 mb/second download speed. The primary Map-Reduce job which analysed each individual tweet took 1 hour and 25 minutes to complete. The job ran one mapper and one reducer. It would have been possible to increase the mappers and reducer to increase the speed, but since it was a single node cluster, having a single mapper and reducer didn't put the cluster under too pressure. The main reason for lengthy time to analyse the tweets was because of the nltk framework. The framework processed the data very quickly, but a limitation with the Hadoop framework meant that the entire nltk library needed to be packaged up and imported by the mapper file for each chunk of data analysed. The library was 2.3 mb when zipped up and 7.8 mb when unzipped. A lot of time was wasted by Hadoop trying to unzip the framework so the map job could import the nltk framework. The processing time could be improved by adding extra nodes to the cluster and running multiple mappers and reducers in parallel or creating a Map-Reduce job using Hadoop's native Java programming language. Hadoop are very open about the fact that Java Map-Reduce jobs out perform their streaming jobs especially when reading and writing to files are involved.

7 Conclusion

In conclusion it has not been possible to identify intelligent people by analysing their social media data and in particular their tweets. There are several reasons for this, firstly there is no context around each tweet. It is often just a collection of words with no punctuation and does not follow correct grammar rules. This is different to the research done by Agichtein and the group of researchers at Yahoo! They were analysing the quality of questions and their answers. Each question provided the context in which to analyse each answer submitted. The second reason is the language used on social media sites, and in particular on Twitter is very different to what would be found in an English essay. Comparing this reason again to Agichtein, the content posted as a question or an answer were not limited to 140 characters. The convention of using hashtags to highlight particular words or phrases did not exist. The study by Shermis & Burstein into using Automated Essay Scoring to grade human writing is not relatable to this research. Shermis & Burstein knew the topic which the question was asked on, this provided the context for each essay. Also each essay was written following correct English grammar rules. Their research did not have to make allowances for hashtags, user mentions, acronyms, URL links or lack of punctuation and grammar.

A person's intelligence is usually assessed by taking an IQ test. These tests are standardised based on the results and the person's age. There is no correlation between an IQ score and the quality of text a person posts online, especially when limited by the number of characters per message. Even if an intelligence quotient was assigned to a user based on their tweets, it could not be related back to an IQ test score they took, as there is no link between them. As concluded by Roscoe et al, using an AES algorithms could not correctly assess a person's knowledge and understanding of a topic, it could then not be possible to assess a person's knowledge and understanding by analysing out of context words posted on Twitter.

I believe this research has showed that Twitter data cannot be used to assert a user's intelligence. It is unfortunate that it was not possible to get Facebook data to analyse. It would have been interesting to compare the data from Facebook and Twitter to see which platform contains the better quality of data. Again it would be important to have context around what is being analysed. As previously stated, Gosling used a hybrid approach to determine a person's personality through Facebook and observing the person. It would be interesting to see if the same results could be obtained by a purely quantitative study.

After completing this research, I believe that any natural language processing requires context. This includes using machine learning algorithms. This is especially valid for sentiment analysis. I would not believe the outcome of a sentiment analysis research unless it stated the topic that was being used as part of the research.

7.1 Improvements

I think this research could be improved by making a few small changes. The first change would be to pick a topic to analyse the data on. It would be far more useful to have a certain topic to analyse instead of having to accommodate any type of text.

I think it would be more useful to research the quality of the data posted by a user on a social media platform, than try to identify highly intelligent people. This change combined with picking a particular topic would provide what I believe to be more meaningful information.

7.2 Follow on research

I have found this research to be a very worthwhile study and after completing this research I believe there are a couple of follow on research topics available.

- It would be worthwhile to analyse the difference in quality between different social media platforms and which users post higher quality content.
- Another area to research would be to analyse the sentiment of multiple users across different social media platforms. Is there a platform a user posts more positive information on than others?
- Another area to research would be to create a natural language parser for Twitter content. If the parser could convert a user mention into the actual full name of that user, and be able to read hashtags as words or phrases. Although reading hashtags could never guarantee one hundred percent accuracy every time, a confidence indicator would be a useful addition. This could greatly improve how content on Twitter is analysed into more meaningful and useful information.

8 Bibliography

1. Dodd, J (2014). Twitter Sentiment Analysis*. Data Mining & Sentiment analysis.* Pg. 3.

2. Russell, M (2013). Mining the Social Web. 2nd ed. O'Reilly Media. pg. Xiv (Preface).

3. Simon, P (2013). Too Big to Ignore: The Business Case for Big Data. Wiley. Pg. 89.

4. Carroll, J (1993). Human cognitive abilities. Cambridge University Press. pg. 3

5. Carroll, J (1993). Human cognitive abilities. Cambridge University Press. pg. 14-15.

6. Agichtein et al. (2008). Finding High-Quality Content in Social Media. Available: http://www.mathcs.emory.edu/~eugene/papers/wsdm2008quality.pdf. Last accessed 9th June 2015

7. Hennessey, A (2014). Sentiment Analysis of Twitter. *Implementation*. pg. 19-25.

8. Shermis, M. D. & Burstein, J. (2003). Automated Essay Scoring: A cross disciplinary perspective. Mahwah, NJ: Lawrence Erlbaum Associates.

9. Roscoe et al (2014). Writing Quality, Knowledge, and Comprehension Correlates of Human and Automated Essay Scoring.

10. Gosling et al. (2007). Personality Impressions Based on Facebook Profiles. Available: http://www.icwsm.org/papers/3--Gosling-Gaddis-Vazire.pdf. Last accessed 28th June 2015.

11. Capiluppi et al. (2013). Assessing Technical Candidates on the Social Web. Available: https://www.academia.edu/2681298/Assessing_Technical_Candidates_on_the_Social_Web. Last accessed 28th June 2015.

12. Reed, J. (2014). How social media is changing language. Available: http://blog.oxforddictionaries.com/2014/06/social-media-changing-language/. Last accessed 11th July 2015.

13. Berry, N., 2010. DataGenetics. Available: http://www.datagenetics.com/blog/october52012/index.html. Last accessed 10th August 2015.

14. Dodd, J (2014). Twitter Sentiment Analysis*. Implementation.* Pg. 17.

15. Forbes. (2015). *The World's Billionaires.* Available: http://www.forbes.com/billionaires/. Last accessed 30th Aug 2015.

16. Statista. (2015). *Leading social networks worldwide as of August 2015, ranked by number of active users (in millions).* Available: http://www.statista.com/statistics/272014/global-social-networks-ranked-by-number-of-users/. Last accessed 30th Aug 2015.

9 Appendix

9.1 Appendix A – List of Acronyms

Acronym	Text	Acronym	Text
afaik	as far as I know	l8	late
aida	attention, interest, desire, action	Li	LinkedIn
ama	ask me anything	Lmao	laughing my ass off
api	application programming language	Lmk	let me know
asl	age/sex/location	Lms	like my status
b/c	because	Lol	laughing out loud
bc	because	lolz	laughing out loud
b2b	business-to-business	mm	music Monday
b2c	business-to-consumer	mt	modified tweet
b4	before	mtfbwy	may the force be with you
bae	before anyone else	Nm	not much
bf	best friend	nsfl	not safe for life
bff	best friends forever	nsfw	not safe for work
brb	be right back	nvm	never mind
btaim	be that as it may	oan	on another note
btw	by the way	omg	oh my god
cc	carbon copy	omw	on my way
cmgr	community manager	ootd	outfit of the day
cms	content management system	Op	original poster
cpc	cost per click	orly	Oh really?
cpm	cost per thousand	otp	one true pairing
cr	conversion rate	p2p	peer to peer
crm	customer relationship management	potd	photo of the day
css	cascading stylesheet	ppc	pay per click
cta	call to action	pm	private message
ctr	click through rate	ppl	people
cx	customer experience	Pr	public relations
dae	does anyone else	Pv	page views
dftba	don't forget to be awesome	qotd	quote of the day
dm	direct message	rofl	rolling on the floor laughing
eli5	explain like I'm 5	roflmao	rolling on the floor laughing my ass off
esp	email service provider	roi	return on investment
f2f	face to face	Rss	really simple syndication
fath	first and truest husband	Rt	retweet
fb	Facebook	Rtd	real-time data
fbook	Facebook	saas	software as a service
fbf	flashback Friday	sem	search engine marketing
fbo	Facebook official	seo	search engine optimization
ff	follow Friday	serp	search engine results page
fomo	fear of missing out	sfw	safe for work
ftfy	fixed that for you	sm	social media
ftw	For the win!	smb	small business
futab	feet up, take a break	smh	shaking my head
fyi	for your information	smm	social media marketing
g+	Google Plus	smo	social media optimization

g2g	good to go	solomo	social, local, mobile
ga	Google Analytics	sov	share of voice
gg	good game	tbh	to be honest
gr8	great	tbt	throwback Thursday
gtg	good to go	tgif	thank goodness it's Friday
gtr	got to run	thx	thanks
hb	happy birthday	tl;dr	too long; didn't read
hbd	happy birthday	tmi	too much information
hmb	hit me back	tos	terms of service
hmu	hit me up	ttyl	talk to you later
ht	hat tip	ttyn	talk to you never
hth	here to help	ttys	talk to you soon
html	hypertext mark-up language	txt	text
ianad	I am not a doctor	ugc	user generated content
ianal	I am not a lawyer	u	you
icymi	in case you missed it	ui	user interface
idc	I don't care	url	uniform resource locator
idk	I don't know	ux	user experience
ig	Instagram	w/	with
ikr	I know, right?	wbu	What about you?
ily	I love you	wcw	woman crush Wednesday
imho	in my humble opinion	wdymbt	What do you mean by that?
imo	in my opinion	wom	word of mouth
io	insertion order	wotd	word of the day
irl	in real life	ymmv	your mileage may vary
isp	internet service provider	yolo	you only live once
jk	just kidding	ysk	you should know
kpi	key performance indicator	Yt	YouTube
afaik	as far as I know	l8	late
aida	attention, interest, desire, action	Li	LinkedIn
ama	ask me anything	lmao	laughing my ass off
api	application programming language	lmk	let me know
asl	age/sex/location	lms	like my status
b/c	because	Lol	laughing out loud
bc	because	lolz	laughing out loud
b2b	business-to-business	mm	music Monday
b2c	business-to-consumer	mt	modified tweet
b4	before	mtfbwy	may the force be with you
bae	before anyone else	Nm	not much
bf	best friend	nsfl	not safe for life
bff	best friends forever	nsfw	not safe for work
brb	be right back	nvm	never mind
btaim	be that as it may	oan	on another note
btw	by the way	omg	oh my god
cc	carbon copy	omw	on my way
cmgr	community manager	ootd	outfit of the day
cms	content management system	op	original poster
cpc	cost per click	orly	Oh really?

9.2 Appendix B – Part-of-Speech Tags

The part-of-speech tags used below can be found on the Python Programming website as this URL: http://pythonprogramming.net/part-of-speech-tagging-nltk-tutorial/

POS Tag	Score	Definition
CC	0.01	coordinating conjunction
CD	0.01	cardinal digit
DT	0.01	determiner
EX	0.01	existential there (like: there is ... think of it like there exists)
FW	0.01	foreign word
IN	0.1	preposition/subordinating conjunction
JJ	0.2	adjective 'big'
JJR	0.2	adjective, comparative 'bigger'
JJS	0.2	adjective, superlative 'biggest'
LS	0.001	list marker 1)
MD	0.001	modal could, will
NN	0.1	noun, singular 'desk'
NNS	0.1	noun plural 'desks'
NNP	0.1	proper noun, singular 'Harrison'
NNPS	0.1	proper noun, plural 'Americans'
PDT	0.1	predeterminer 'all the kids'
POS	0.01	possessive ending parent's
PRP	0.01	personal pronoun I he she
PRP$	0.01	possessive pronoun my, his, hers
RB	0.1	adverb very, silently
RBR	0.1	adverb, comparative better
RBS	0.1	adverb, superlative best
RP	0.1	particle give up
TO	0.001	To go 'to' the store.
UH	0.1	interjection errrrrrrrm
VB	0.1	verb, base form take
VBD	0.1	verb, past tense took
VBG	0.1	verb, gerund/present participle taking
VBN	0.1	verb, past participle taken
VBP	0.1	Verb, sing. present, non-3d take
VBZ	0.1	Verb, 3rd person sing. present takes
WDT	0.01	wh-determiner which
WP	0.01	wh-pronoun who, what
WP$	0.01	possessive wh-pronoun whose
WRB	0.01	wh-abverb where, when

9.3 Appendix C – Stop Words

Whom	Between	Herself	During	Which	Under
A	Same	Ourselves	With	Him	Other
And	He	itself	Most	What	Our
Doing	An	Me	Above	Now	Below
Just	Will	Are	If	So	There
Few	Is	Up	T	This	I
Your	Did	Off	Against	Down	Before
Their	Why	After	Had	Myself	Each
Should	Been	Where	While	Once	Too
We	Very	When	She	Was	Yourselves
Yours	S	Only	Ours	Here	For
The	Be	Over	Any	Own	Then
About	Who	Am	Have	His	Until
Can	Does	You	Or	Both	Further
As	My	Because	They	Do	But
Into	No	That	Of	Theirs	All
How	Those	Where	Such	On	Through
Himself	Don’	Hers	It	By	Having
Nor	From	Has	Than	More	Its
Not	These	To	In	Out	Themselves

9.4 Appendix D – Twitter user names

shaneburke22	minniemelange	danieltosh	bernadettedoyle
Emssilee	adrianweckler	radioleary	IanDempsey
CampionCat	vickinotaro	AlanCarr	liaonet
lucindaalbourke	EamonLeonard	katyperry	derrymcv
EoinEdwards	RosemaryMacCabe	TheEllenShow	Gerard_McCarthy
GMcM_	aplusk	GaryLineker	MarianKeyes
the_crouch	Graeme_McDowell	Desbishop	BigSean
stephenfry	harrygoddard	KearneyRob	Pharrell
daraobriain	barrypj	jamieheaslip	channingtatum
grahnort	iBiryukov	damofitzpatrick	JohnCena
michkeegan	ilkeuygun	csl_	johnlegend
KimKardashian	echancrure	C_McCann	Tatawerneck
GardaTraffic	aaronredmond	kingsalkelly	CherLloyd
MiriamOCal	kenpower	selenagomez	azizansari
JeremyClarkson	JosephPKehoe	KendallJenner	snooki
hollywills	john_logue	KylieJenner	elissakh
ghook	GNev2	MileyCyrus	IAMQUEENLATIFAH
vincentbrowne	NBTGary	ladygaga	ollyofficial
jimmyfallon	KatieOCarroll1	ashleytisdale	jennettemccurdy
BarackObama	ocarroan	SrBachchan	andersoncooper
KevinHart4real	TClohosey	JLo	SethMacFarlane
ConanOBrien	khloekardashian	shakira	50cent
SCMoney21	StephenAtHome	NiallOfficial	JessieJ
charltonbrooker	blakeshelton	MariahCarey	realwbonner
DerrenBrown	ActuallyNPH	RyanSeacrest	VictoriaJustice
Tommedian	taylorswift13	tyrabanks	Ludacris
noelburke6	KingJames	rustyrockets	iamdiddy
kieranmoloney88	CherylOfficial	carlyraejepsen	justinbieber
Paddy__Gorman	cooper_m	SHAQ	rihanna
obie6661	amyhuberman	TheRock	britneyspears
anthonymcg	aoifemcwey	SofiaVergara	NICKIMINAJ
georgiasalpa	IAMKELLYBROOK	victoriabeckham	JimCarrey
bdoylers	JenniferMaguire	jessicaalba	ParisHilton
ClassyCody	conor5reilly	ByronKatie	SnoopDogg
clairedooley000	TheRealKirstyG	donni	NicoleScherzy
laurawhelanx	RobertDowneyJr	safiyyahn	maryjblige
kwood36	KatieTaylor	KenJennings	lilyallen
ProperChurch	LewisHamilton	alyankovic	ParineetiChopra
ChipSArr	Redknapp	OfficialKat	Michael5SOS
StephenJCahill	susannareid100	kellyoxford	ahickmann
andy_murray	OfficiallyGT	badbanana	TomCruise
JKCorden	Nigel_Farage	SarahKSilverman	IGGYAZALEA
iamwill	Lord_Sugar	tomhanks	chelseahandler
FloydMayweather	Ed_Miliband	SteveMartinToGo	jason_mraz

reggiefivehands	MikeTyson	BeckyLynchWWE	DwightHoward
TheNotoriousMMA	GabbyLogan	Marc_MuFc1	tonyhawk
jimmybullard	aemerson_	PaulOMahonyEire	Ashton5SOS
kourtneykardash	LaurenCoffey4	wwebalor	CP3
jimmykimmel	SafaTopal	seancullen95	mindykaling
AP_McCoy	OTooleAsh	EmmettScanlan	tyleroakley
gokathrynthomas	Gordonwdarcy	RonanOGara10	johngreen
UnaFoden	richardajkeys	MarkusFeehily	msleamichele
BryanCranston	docallaghan4	RobMSheehan	gabyespino
louistheroux	RafaelNadal	BrianMcFadden	jamieoliver
EmWatson	Paulmcgrath5	JanetJealousy	souljaboy
JimWhite	PawHennessy	MrEdByrne	bridgitmendler
russellhoward	EvaLongoria	ShaneFilan	HilaryDuff
carolineflack1	DjokerNole	colinodonoghue1	RitaOra
AislingRiordan	Padjii	KianEganWL	mirandalambert
Emma_Carroll	zachbraff	ronanofficial	RedHourBen
oj_byrne	Rachel_Breslin	Glinner	SandyLeah
JoshRadnor	Pat_Guiney	DaithiDeNogla	NickCannon
DavidSpade	DeniseMcCarthyx	AnnieMac	serenawilliams
richardbranson	pjgallagher	Storm_Keating	raisa6690
Carra23	Harry_Styles	planetjedward	Angelluisr
BrianODriscoll	Harmonica26	Evy_Lynch	tomdoorley
loukav22	KKutlesa	paulyhiggins	ThePatrickk
jack	rachie_t	SivaKaneswaran	MCVEYSLLAMAS
rupertmurdoch	wossy	TheScript_Danny	ceegy
jeffweiner	Charlottegshore	WWESheamus	oceanclub
emilychangtv	Bridget_O_Dea	McIlroyRory	john_mcguirk
brooke	JordanDrennan	LWalshMusic	Ragin_Spice
BBCRoryCJ	RACHEL_WYSE	shanelynchlife	Omaniblog
alexia	denise_vanouten	Ryan_Acoustic	Paul_McGovern
JoannaStern	mattletiss7	LPOBryan	LaurenBoothWeb
pkafka	Schofe	IrishSmiley	damienmulley
Benioff	patdebrun	JOCofficial	Tweetinggoddess
shanselman	TheMasterPrawn	clodaghanne	LPOBryan
tim_cook	nackywal9	RossOCK	carol1hamilton4
timoreilly	CalvinHarris	RobertBohan	sofia_lawson5
ezraklein	Ruby_Walsh	KristianNairn	susan1ford1990
BillGates	TubesSoccerAM	SeanOBrien1987	brad61211
davidmcw	halliedhardy	DevonMMurray	RedScareBot
GerryAdamsSF	neiltyson	nairamk	BrianBrownNet
colmtobin	bkav2011	Keithpbarry	GraceyOConnell
FrankSunTimes	bporourke99	JoeMullally	SurrrahV
gnelis	emma_grattan	skooal	AmyGrimesSuxx
fionndavenport	Woolberto	Piyanuch_Model	kalyxo_
marklittlenews	garykingTDN	ImaProudSwifty	CremeEgg95
CarolineAtMCD	mrcianmckenna	cocomairead	Ireland2gether
Ceraward	violethesketh	eoghanmcdermo	sljmcgrath

louisemcsharry	Pink	SarahBolger	UnaMullally
MrNiallMcGarry	BrunoMars	declanwalsh	likemamuse2bake
daniellamoyles	VickyGShore	_DarraghDaly	liam11
RozannaPurcell	Hozier	rickoshea	johnpeavoy
genemurphy	Louis_Tomlinson	AlisonMorris_	K4yleigh_
CKennedyPR	NickyByrne	monella_naija	gavreilly
ArianaGrande		Real_Liam_Payne	