Adam Flax c1115629

Initial Plan – Project 54 Event Detection From Big Data Using Social Media

Supervisor: Steven Schockaert Moderator: David W Walker

Module Number: CM3203

Module Title: One Semester Individual Project

Credits this module is worth: 40 credits

Section 1.1 Project Description

Social media sites such as Twitter often have a real time element associated to them. As stories and events break out around the world people will generally post content related to these events. For instance during the announcement that Tokyo would host the 2020 olympics more than 26,000 tweets[1] relating to this news was posted in the first minute alone.

Content that is created on social media sites such as Twitter or Flickr often have a lot of descriptive metadata associated with them. Descriptive Metadata was defined by the NISO as "s*tructured information that describes a resource for the purpose of identification or discovery*".

By looking at this metadata associated with the content we can learn things such as when and where it was posted along with data about the person who posted it. These properties are good for attempting to detect trends in social media content as its created.

This project will look at data from Twitter, a microblogging social media platform, and attempt to discover what significant events have happened in a given spatial region area during a given time period.

This project will also look at the problems involved with handling and processing 2 weeks worth tweets at once (this will be around 20 million tweets) as well as the scalability involved (twitter gets over 400 million tweets per day[3]) and attempt to tackle these problems.

Section 2.1 Project aims and objectives

This project can be split into 3 sub programs. The first program will record public data flowing through Twitter. This will be done with Twitter's public data streaming API. In this part of the project consideration will need to be taken to implement a suitable format to store all these tweets. This will be one of our core aims and objectives for this sub program.

An individual tweet's size and associated metadata is around 900 bytes to 1 kilobytes in size. Therefore we are dealing with around a gigabyte of space for every million tweets we store. Accessing all of these tweets needs to be done such that it is easy to sort through and fast to access. While the tweets need to be efficiently stored so they take up as little space as possible. This problem could be solved by simply saving all the tweets in a text file with the rule being 1 tweet per line. As we will often only be reading in a section of tweets at a time and we will only be reading in

that section once there does not seem to be an advantage in using a database manager. This is because we will not be able to make use of the majority of the benefits of a database manager while we will still get the overhead of using one.

Another objective that we will look at for this subprogram is to only collect and harvest tweets that are suitable for event detection in our chosen spatial areas and that are in English. These tweets need to contain metadata information such as the location the tweet came from otherwise it will be harder to detect events. This problem can be solved with the use of filters when we create the streaming connection to twitter. With filters we can ask for tweets to be from a given location or to be only in english.

Section 2.2 looking for trends

The second sub program will read in the sample data and analysis and attempt to look for trends that would point to potential events. Once we have found tweets that could point to potential events we will save these to disk. We will save these tweets to disks as it will act like a cache system. So if we query the program for any tweets in a time spatial area and we have done the processing before we can just return the data rather than perform all the processing again.

We will attempt to look for trends that will point to potential events by by looking at the aggregate tweets of a time period's hashtags and look at which hashtags occur in a higher frequency compared to their frequency in our sample data.

However loading millions of tweets into memory will not be physically possible so another aim and objective will be to research and implement a way of keeping track of all the metadata of all tweets without exceeding a memory threshold which has yet to be decided. The most simple solution to this problem is to only process sections of our sample data at a time.

Another aim and objective for this subprogram is to look for misspellings in hashtags and treat these misspelled hashtags as the intended hashtag. It would make sense that if we had no reasonable doubt that the hashtag "Lpndon" actually meant "London" that we should count it as "London". I believe a reasonable approach to do this would be with Levenshtein distance which tells me how similar 2 strings are based on deletions, insertions or substitution of characters. Once we have identified misspellings we will compare the miss spelling hashtags to hashtags that belong to tweets in the same geographic region at around the same time. By doing this we can improve accuracy and make sure the hashtag is a miss spelling and not another word.

Section 2.3 Detect events based on trends

The final program's job will be to look at tweets in a given geographical area and time and attempt to predict events based on these. This will be done by merging related tweets and hashtags together to build up a story. If we get a large volume of tweets combined our story would show an event has happened. In this project we will find related tweets using Bings synonyms API which will give us a list of alternative ways people refer to products, entities, events and real world locations. Related tweets can also be found with techniques such as cosine similarity and explicit semantic analysis. More research will need to be done to find which additional techniques will be appropriate

Another aim for this project related to this program is that it should be able to look at context of the

spatial area for grouping events. For instance imagine that it's Guy Fawkes night an bonfire night event based in London should probably not be matched with a bonfire night event happening in manchester as these events are local to the area they are happening in.

Section 2.4 additional aims for the project

One aim of this project is for it to be scalable. As we increase the size of the data set it should be as simple as simply running multiple instances of the programs to tackle the extra growth. This project aims to do this in a distributed memory environment due to the scalability problems that shared memory suffer from.

One way we can tackle our scalability problems is for it to be possible to run multiple instances of the same program at the same time. This is because the different programs will be running at different speeds. It will probably take longer to group similar hashtag's together then it will to check which hashtags frequencies are out of their normal distribution. Therefore we could run 2-3 different instances of the final program to keep up with the other programs. This will help tackle the scalability issues as it is often cheaper to create new nodes then it is to improve the hardware on nodes.

Another way in which we will tackle our scalability problem is for us to implement eventual consistency for storage rather than strong consistency. As we are scraping around 17 tweets a second from my Twitter parser if we implemented strong consistency for all of the programs then it would lead to these programs doing a lot of extra work constantly to make sure they are up to date. This extra work is probably not worth the effort for what would be a possibility of the events detected being 40-50 tweets inaccurate.

Another aim of this project is for us to detect events that have happened between the 31/01/2014 and the 14/02/2014 in the 5 largest regions of the United Kingdom. These areas and their latitude and longitude coordinates are :

Latitude Longitude	**Area**
51,31,00,06	London
52,29,01,52	Birmingham
53,30,02,15	Manchester
53,47,01,45	Leeds-Bradford
53,23,03,02	Liverpool-Birkenhead

Section 3.1 Work Plan

Section 3.2 Work I have already completed

Twitter Scraper (project one).

Preliminary research.

Section 3.3 Week 1

Initial plan.

Set up development environment and start running python scraper.

Section 3.4 Week 2

Create program and unit tests for the program that looks at the normal distribution of the frequency of hashtags to detect anomalies.

Look into other ways than just the normal distribution to detect outliers.

Write preliminary research into a background section for my final report.

Section 3.5 Week 3

Research ways on how to implement spellcheck.

Implement spellchecking for hashtags so that obvious misspelled hashtags are counted as the hashtags they were meant to be. Write unit tests for spellchecking for hashtags.

Take a look at if it is viable to implement the porter stemming algorithm to also identify hashtags of different tenses.

Section 3.6 Week 4 and 5

Create program that will pair similar tweets together by bings synonyms api.

Write unit tests for the program.

Look into if its worth implementing explicit semantic analysis.

Section 3.7 Week 6

Review meeting

Adapt event prediction program so we can run multiple instances of the same program at the same time to attempt to get horizontal scalability. Horizontal scalability is the idea that if we need to increase our hardware to handle an increasing amount of work we should be able to simply create more nodes to handle the work rather than for us to increase the hardware resources on each node.

Write integration tests for this program.

Section 3.8 Week 7

Write up the approach section for my final year project.

adjust programs based on review meeting.

Section 3.9 Week 8

Adapt anomaly detection program so we can run multiple instances of the same program at the same

time to attempt to get horizontal scalability.

Write Integration tests for this program.

Section 3.10 Week 9

Write up the implementation details for the programs for the final year report.

Section 3.11 Week 10

Continue working on the final year project report.

Section 3.12 Week 11

Review meeting.

Continue working on the final year project report.

Section 3.13 Easter Break

Continue working on the final year project report.

Section 3.14 Week 12

Work on viva .

1. https://blog.twitter.com/2013/congratulationstokyo

2. http://www.niso.org/publications/press/UnderstandingMetadata.pdf

3.http://www.washingtonpost.com/business/technology/twitter-turns-7-users-send-over-400-million-tweets-per-day/2013/03/21/2925ef60-9222-11e2-bdea-e32ad90da239_story.html

4. http://www.ibmbigdatahub.com/infographic/four-vs-big-data