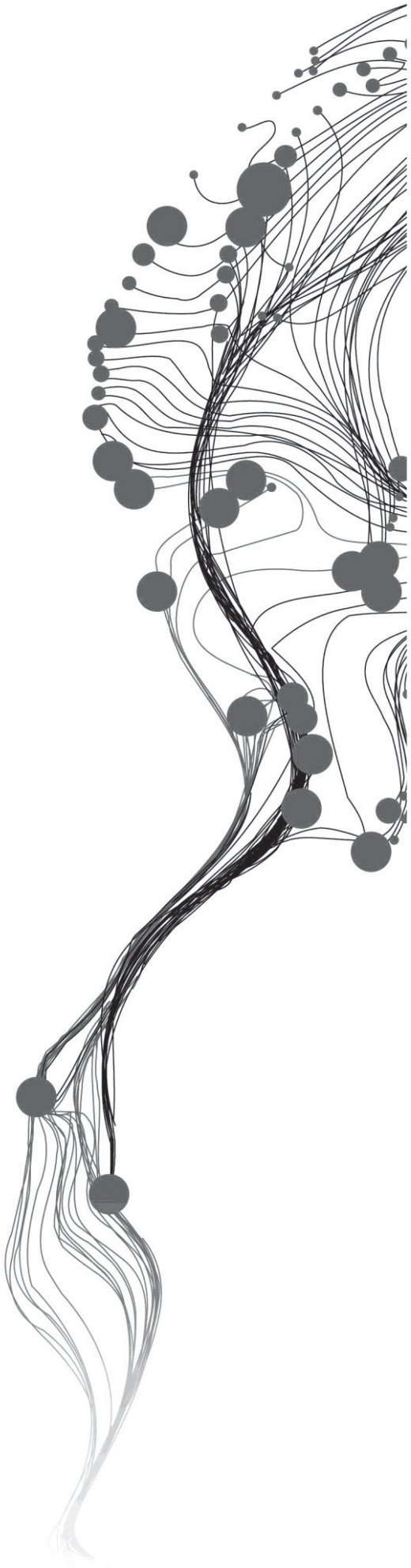
MONITORING URBAN TRAFFIC STATUS USING TWITTER MESSAGES

FATMA AMIN ELSAFOURY
February, 2013

SUPERVISORS:
Dr. UlanbekTurdukulov
Dr. Rob Lemmens



Monitoring Urban Traffic Status Using Twitter Messages

FATMA AMIN ELSAFOURY

Enschede, The Netherlands, February, 2013

Thesis submitted to the Faculty of Geo-Information Science and Earth Observation of the University of Twente in partial fulfilment of the requirements for the degree of Master of Science in Geo-information Science and Earth Observation.
Specialization:Geo-Informatics.

SUPERVISORS:
Dr. UlanbekTurdukulov
Dr. Rob Lemmens

THESIS ASSESSMENT BOARD:
Dr. A.A. Voinov
Dr. O. Huisman, ROSEN Inspection

ABSTRACT

Traffic congestion is a worldwide problem. All large cities and capitals around the world suffer from this problem. Traffic congestions cost money and time. The existing tools used to help in collecting information about traffic are expensive and have limitations. Nowadays, micro-bloggers are being used widely. It allows people to share information, opinions and stories in short messages. Twitter is a very popular micro-blogger. It allows people to share whatever they want in 140 characters. Twitter offers a new source of information for variety of topics.

This research proposes a system to use traffic information shared by Twitter messages (tweets) in a real time manner. It uses a customized Part of speech (POS) tagging method for extracting information from the tweets. POS is also used for Geo-locating the tweets with custom developed locations' dictionaries. Google Geo-Code API is also used in the geo-locating task. It also follows the traffic information sent by @TfLTrafficNews which is an official Twitter account used for reporting about traffic in London. A prototype of the proposed system was implemented. This prototype contains implementation of the proposed POS algorithm and also the implementation of the system work flow.

The result of the system is a map showing a highlighted route. This route is the location (road) mentioned in the tweet. The highlight colour depends on the traffic status which is also mentioned in the tweet. The results were tested by comparing it against Google Maps traffic feature. The results could be helpful for future work.

Keywords: Twitter, Twitter Streaming API, Traffic congestion, POS, Google Geo-Code API, Google Maps API

ACKNOWLEDGEMENTS

This dissertation would not have been possible without the help and support of Allah the most merciful and the most gracious.

I also want to thank my supervisors Dr.UlanbekTurdukulov and Dr. Rob Lemmens for their help and guidance and support throughput the research period.

Special thanks must go to Dr. MS Connie Blok for her understanding, help and support.

The most inspiring thing to me and that encouraged me to continue this research was the values of Islam and the teachings of prophet Muhammad (PBUH) as he said:

"The seeking of knowledge is obligatory for every Muslim."

So, it was important for me to continue my research and try to add something to the world and I hope it will be helpful. That's what real Islam orders us to do as one of the Quran versus is:

"*If anyone saved a life, it would be as if he saved the life of the whole people* ".

Avery special thanks to my friend soha who helped me in every possible way.

I also dedicate this dissertation to the spirit of my mother and to my supportive sisters and brothers in Egypt. Thank you all.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

1.1. Background

Traffic congestion is one of the biggest problems in our modern life. All cities around the world suffer from this problem. The time spent during traffic congestions, is miserably wasted. The cost of driving delays annually comes to 48 billion $ or 640$ per vehicle deriver(Arnott & Small, 1994).To avoid this problem urban societies use hardware sensors like cameras, inductive loops and radars to monitor traffic status. These tools function well; however, they have some limitations. One of these limitations is the high maintenance costs of these tools. Another limitation is that the tools cover only the certain area of the network and are designed to collect specific type of information like vehicles count(Carvalho, 2012).

Micro-bloggers are web applications that allow people to share statuses, information and opinions in short messages. They provide a light weight, easy and fast way of communication between us (Java et al., 2007) . Twitter is a very popular micro-blogging service. It gives people space to express their statuses in 140 characters. Twitter allows friends, family members and co-workers to communicate easily through desktops or mobile phones. There are millions of people that use Twitter to share their daily stories. Actually within 8 months after the release of Twitter on April, 2007, about 94000 users was subscribing to it(Java et al., 2007). The topics that people usually share on Twitter range from daily stories, current events, opinions and others(Java et al., 2007) .

Twitter offers researchers, marketers, activists and decision makers an access to a new source of digital information and data as users share their stories. Some studies have started using this user generated dataset for example;Zook et al. (2010) used Twitter data (tweets) for studying crisis response. Also, tweets were used to cover federal elections in Australia (Bruns & Burgess, 2011).

Mining this source of information may help in solving the urban traffic congestion problems in a near real time manner. Analysing the content of Twitter messages might provide better understanding of the traffic congestions in terms of why, when and where does it happen. This way of extracting traffic information overcomes the limitations of the used hardware sensors. It also allows the easy and free access to such information for all people to help them in avoiding congestion spots.

1.2. Problem Statement

Real time traffic information is important for avoiding traffic congestion spots. Using Twitter messages as a source of information could be helpful. The existing studies that use text mining methods depend only on geo-tagged tweets. Here, this research problem is to combine between semantic analysis method for filtering and extracting tweets data with geo-locating method for the non- geo- tagged tweets.

1.3. Research Identification

1.3.1. Research Objectives

1) Developing a system for filtering the tweets and getting the traffic related tweets.
2) Make use of the official news Twitter accounts that report about traffic.
3) Use text mining method to analyse and extract information from tweets.
4) Geo-locate the tweets based on their content.
5) Plot the extracted location on Goggle maps with different symbols showing the traffic status.
6) Create a system workflow to show traffic status as mentioned in tweets.
7) Asses the system through comparing its results to Google Maps traffic functionality.

1.3.2. Research Question

1. Developing a system for filtering the tweets and getting the traffic related tweets.
    a) How to retrieve the tweets in a real time manner?
    b) How to filter the tweets using specific query?
    c) How to classify the tweets according to the sender intto official and individual classes?

2. Make use of the official news Twitter accounts that report about traffic.
    a) Which official news Twitter account to follow?

3. Use text mining method to analyse and extract information from tweets.
    a) Which text mining method to use or modify?
    b) How to define the problem of the traffic stated in the tweets?

4. Geo-locate the tweets based on their content.
    a) Which method to use or modify to geo-locate the tweets?
    b) How to combine between geo-locating step and information extraction step?

5. Create a system workflow to show traffic status as mentioned in tweets and plot the output on Google maps.

    a) Which system architecture to use?
    b) How to connect between the system work flow and the implemented algorithm?
    c) How to plot the extracted locations on the map?
    d) How to use different line segments' colours and labels to show the traffic status?

6. Asses the system through comparing its results to Google Maps traffic functionality.
    a) How reliable Google Maps is?
    b) What are the comparison criteria?
    c) How to test the system' results?

1.4. Innovation Aimed At

The innovation in this research aims at proposing an algorithm to extract traffic information from tweets and plot it on Google maps. This algorithm is based on the integration between a text mining method and a geo-locating method.

1.5. Research Methodology

The methodology followed in this research started with reviewing the literature for text mining methods and geo-locating methods. The methods that seemed to fulfill the research objectives were selected. Then, the two methods were combined and a prototype of the proposed system was implemented. The results of the prototype were tested to see how reliable the proposed system is. The testing was based on comparing the results with the traffic status feature of Google maps. For the incomparable results some updates were done on the algorithm. Then, the prototype was retested after the changes. The last step was repeated until getting good results compared with Google maps traffic analysis feature.

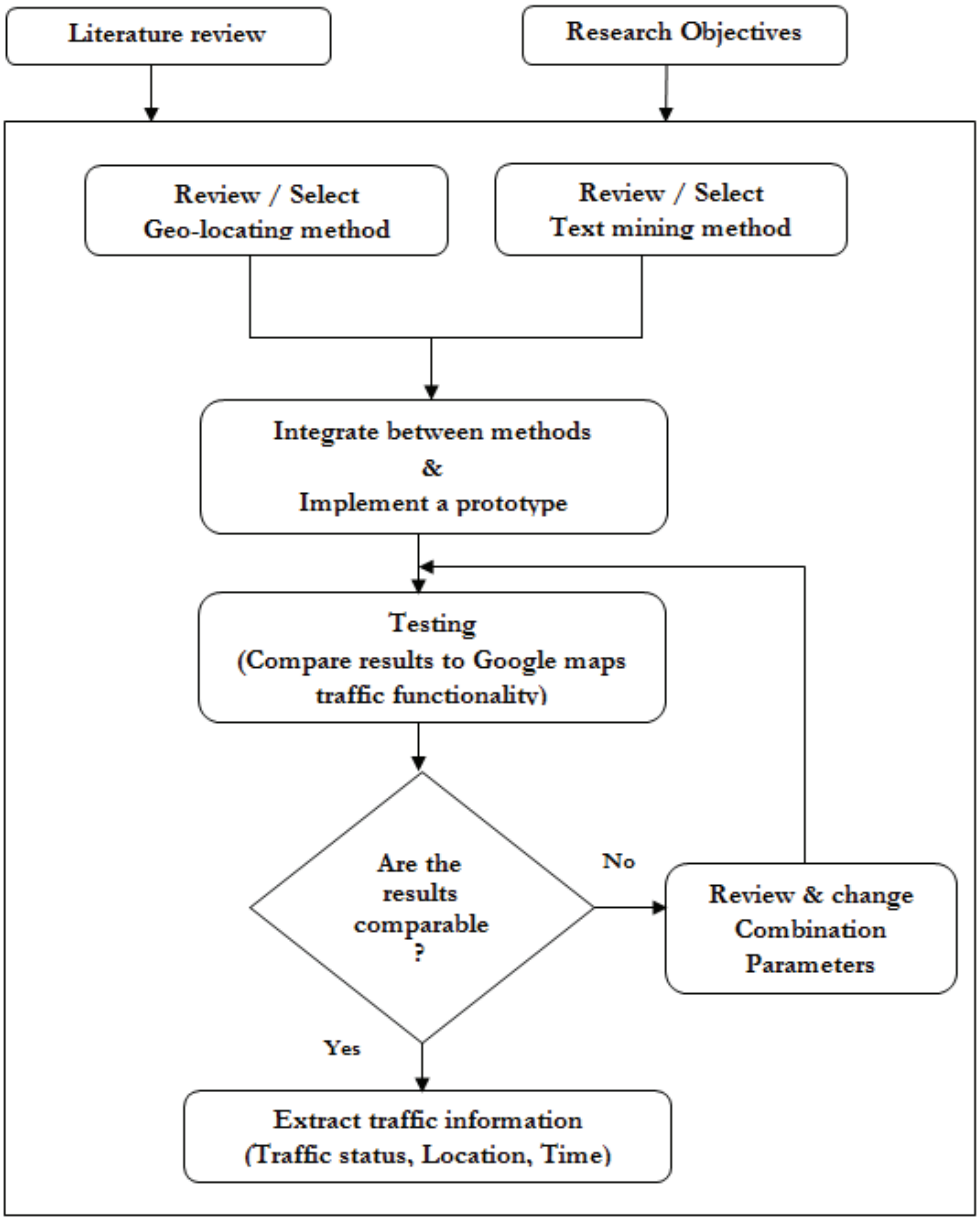


Figure 1: Research flowchart

1.6. Thesis Structure

This thesis consists of five chapters which are arranges as follows:

- **Introduction:** Chapter one introduces the research background and the problem statement. This chapter also contains research identification which includes research objectives, research questions and innovation of this research.

- **Literature Review:** Chapter two reviews the literature in extracting information from Twitter messages (tweets). It presents the studies that tried to use text mining for extracting information from tweets and to geo-locate the tweets based on their tweets. It also explains briefly the used methods in these literatures.

- **Proposed Methodology: C**hapter three presents a brief description of different methodologies for mining tweets and a more detailed description about the proposed methodology and justification for the selection of this proposed system.

- **Tweets Mining and Geo-locating:** Chapter four is divided into three parts. The first part presents a data collection step for better understanding which tweets to collect. The second part describes the research area. The third part elaborates the implementation of the tweets mining and geo-locating methodologies.

- **Workflow Design**: Chapter five is describing the implementation of the system workflow.

- **Testing and Results:** Chapter five presents the results of the proposed system and an evaluation of it. This evaluation is based on comparing the system against Google maps traffic functionality. It also shows how a reliable "Google maps" is.

- **Conclusion:** Chapter six provides answers about the research questions, some general achievements, limitations and recommendations for future work.

2. LITERATURE REVIEW

2.1. Information Extraction From Tweets

Using Twitter messages (tweets) for detecting and geo-locating events is a relatively new emerging research area. There are some studies started using Twitter for detecting different events like disasters, earthquakes and traffic jams. These studies used Text mining concept and Natural Language Processing (NLP) methodologiesto extract information from tweets(Ahmad, 2007).

Some studies use Named Entity Recognition system (NER)(Collobert et al., 2011) to extract information from tweets. NER's job is to classify words in the sentence into predefined categories like names of persons, organizations, locations and expressions of time. NER uses machine learning classifiers to train the system. These machine learning classifiers are like Maximum Entropy taggers(Robinson, 2009b), Support Vector Machine (SVM)(Guduru, 2006)and Conditional Radom Field (CRF)(Wallach, 2004). NER systems use CoNLL2003 (Sang & Meulder, 2003)which is a benchmark dataset for training and validating systems.

MacEachren et al. (2011)use Twitter for situational awareness system. They use NER for analysing the tweets. They developed a custom tool called ANNIE named entity extractor based on GATE (Bontcheva et al., 2004) .Abel et al.(2012) is another study tried to track and filter accidental information using Twitter messages. This study developed a system called Twitcident. This system also applies Named Entity Recognition (NER) (Mansouri et al., 2008)to extract information from tweets.

Some other studies use Part Of Speech (POS) tagging(Collobert et al., 2011)for information extraction. POS aims at labeling each word in the sentence with a tag indicating its syntactic role. For example, it labels the word as a noun, verb or adverb. POS method will be explained in more details later in the proposed methodology chapter.Endarnoto et al.(2011)use POS tagging to tag each word in the tweets. After that they apply rule based approach to extract the information from tweets. They use the system to analyse tweets that are written in a predefined format.

Tokenization(Guo, 1997) is one of the used methods for text mining. The tokenization method tokenizes (chop) the text to tokens (words). These tokens are categorized depending on custom dictionary into custom tags. Then these tags are used for information extraction.Wanichayaponget al. (2011)use tokenization to extract traffic information from tweets. They use tokenization with a custom built traffic word dictionary for Thai language. This dictionary tags the words into 4 categories place, verb, ban and preposition.

Some other studies start with further filtering the tweets to get topic related tweets. Then, they apply further text analysis method or they stop at the classification step. To do this classification job, they used Support Vector Machine (SVM)(Guduru, 2006). SVM is a (supervised) machine learning algorithm used to train the system to be able to automatically classify objects according to training dataset.

Sakaki et al(2010)present a study to use Twitter messages for real time earthquake detection in Japan. They use SVM to further filtering the tweets to get earthquakes related tweets. Then, they apply Morphology analysis(Ahmad, 2007) methods to extract information from tweets. Morphology analysis method used to identify the parts of speech in a sentence and how these parts interact together. It is another form of POS tagging. First, they filter the tweets using keywords like "earthquake" or "shake". Then, they apply SVM for classifying the tweets into positive (related) tweets and negative (Unrelated) tweets.

Carvalho(2012) also uses SVM to identify traffic related tweets. For creating a suitable training set, he used the tweets sent automatically by official news sources using robot users. The tweets of those robot users are easier to identify since they are written in a very strict format. On the other hand, the tweets written by human users are more difficult to identify. Because these tweets are full of grammatical mistakes spelling mistakes, non-standard punctuation, emotion icons and etc. So the generated training set is used to help the classifier to identify the positive (traffic related) human written tweets. After that, the positive tweets are added to the training set to enrich it and train the classifier again. This step is repeated till the classifier achieves a high precision in identifying the traffic related tweets posted by human users.

Yerva et al.(2010) is another study that uses SVM classifier. They aim at classifying tweets to check if they are mentioning as specific company or not. For example if the tweet contains the word 'apple' the study classifies the tweet and decides if it is related to the company Apple Inc.

The pre-mentioned literature demonstrates the importance of Twitter as a source of information. It also shows that Twitter is used in different areas like situational awareness, events extraction and traffic information extraction.

The innovation in this research is to use a specific POS algorithm for analyzing tweets. This research is also concerned with analyzing a more free text written tweets than analyzing tweets following a predefined format. This research also makes use of the Dictionary idea for English language combined with POS tagging.

2.2. Geo-locating Tweets

To detect events using Twitter messages, it is important to know the location where these tweets are being issued. This location is referring to where the events are taking place. On August, 2009 Twitter issued geo-tagging feature which associates the user's current location in the form of latitude and longitude values with each tweet. This feature will work only if the user enables it.

Some studies about event detection depend on Twitter geo-tagging feature for estimating the location of the event like(MacEachren et al., 2011). Other studies used the location information associated with the Twitter user's account like (Sakaki et al., 2010)and(Abel et al., 2012).

Some other studies propose an approach to geo-locate the tweets based only on their content. Paradesi(2011) presents a study to identify the locations references in a tweet and show relevant tweets to a user based on his location. This study developed "TwitterTagger" system for geo-tagging the tweets. "TwitterTagger" uses a POS tagger to tag the content of the tweets. Then it compares the resulted noun phrases to theU.S. Geological Survey (USGS) database("USGS," 2012) of locations. After that, the system performs two filters to filter the ambiguities out. These ambiguities could be that the noun phrase is not really a location name. Sometimes the ambiguity could be that the noun phrase refers to more than one location.

Wanichayapong, et al. (2011) also use the content of the tweets to estimate the tweets locations. Firstly, they use syntactic analysis on the tweets to extract locations names. Then, they look these locations names up a local place dictionary which is provided by ministry of transportation in Thailand. For the missed places they use Google geocoding API ("The Google Geocoding API," 2012) to retrieve the latitude and longitude values for these locations.

Cheng et al. (2010) also propose a framework for geo-locating Twitter user's city level based only on the content of his Twitter messages. They propose a classification component for automatically identify words in tweets with local geo-scope. After that, they propose a lattice-based neighborhood smoothing model for refining the estimated user location. The system they proposed estimates (n) possible locations for each user in descending order of accuracy. They developed a content-based user location estimation algorithm. This algorithm observes the actual distribution of some local words across cities. Then, further processing is done to differentiate between local and non- local words. Depending on the local words they estimate the current location of the user on the basis of city level.

These studies that propose methods for geo-locating tweets based only on their content will help in geo-locating the non-geo-tagged tweets to know where the traffic congestions are happening.

The innovation here is this research uses text mining method named POS to extract locations names. Like the pre-mentioned study (Wanichayapong et al., 2011) but with local dictionary for central London which is the research area of concern. This dictionary covers streets names, streets intersections and road segments.

3. PROPOSED METHODOLOGY

As mentioned in the literature review chapter, there are different methods used for text mining and tweets geo-location. In the next section these methods are described in more details.

3.1. Named Entity Recognition (NER)

Named entity recognition(Collobert et al., 2011)aims at classifying the words in a given text into the categories. The most common categories are Person e.g.: "Smith", Organization e.g. "Google" or Location e.g. London.
NER pipeline is composed of a list of tasks differ from a system to another but he main tasks are:

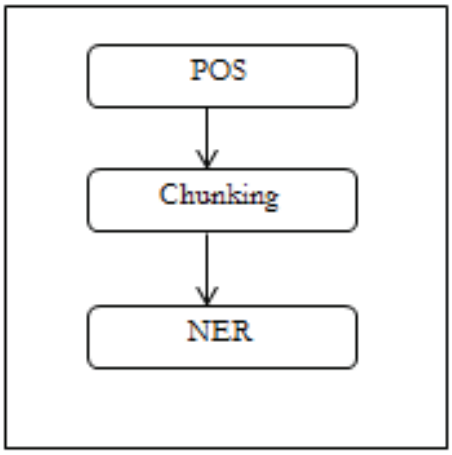


Figure 2:NER pipeline

1. **POS**

   This step is to label every word in the given sentence as noun, verb or adverb. It will be described in details later.

2. **Chunking**

   It is also called shallow parsing and it is responsible of labeling segments of a sentence syntactically into noun phrase (NP) or verb phrase (VP) (Patell, 2011).Chunking has two approaches :

   - **Rule-based approach:** this approach depends on written rules to classify the segments of the sentence.
   - **Supervised machine learning:** this approach use a training dataset which is a set of labeled data used to learn the system how to label the segments of the sentence.

3. Named Entity Recognition

Here, the actual classifications for the words (entities) are given. This classification is done using one of the following approaches:

1. Rule based/ Handcrafted approach

This approach depends on human made rules to recognize the entities. This system type has various methods:

- **List lookup**
  This NER system uses gazetteer to recognize entities. It only recognizes the entities listed in the system's lists in the gazetteer.

- **Linguistic**
  This approach allows the system to recognize the entities based on language based rules. It needs rich rules to recognize entities effectively.

2. Machine learning /Automated approach

This approach aims at classifying the entities more than recognizing them. It applies a classification statistical model. This approach has a variety of methods to handle the supervised learning:

- Hidden Markov Models (HMM)(Bikel et al., 1997).
- Maximum Entropy Markov Model (MEMM).
- Conditional Random Field (CRF).
- Support vector Machine (SVM).
- Decision Tree (DT)(Sekine, 1998).

3. Hybrid model approach

This approach mixes between rule based approach and machine learning approach to achieve higher accuracy in recognizing entities.

3.2. Support Vector Machine (SVM)

Support vector machine (SVM) is a supervised learning model used for classification and regression. Supervised learning technique is a technique that uses an (attribute, value) pair containing the (predictor, target) pair to learn the predictor and target value relation. SVM is a supervised learning technique that uses training dataset to be able to create a decision function for classification. SVM uses two different datasets for training and for testing. To use SVM for text mining purposes like text categorization, a feature extraction technique is needed(Guduru, 2006).

Feature extraction

A feature is a set of keywords that contains the main data characteristics. The function of feature extraction creates a new feature set similar to the original features set but with smaller size so it enhance the speed of supervised learning.

There are two forms of SVM:

1. **Linear classifier**
   It is used when the training data is linearly separable. Linear classification defines a hyperplane, which is a geometry concept that generalizes the plane into a number of dimensions, in the input space.

2. **Non-linear classifier**
   It is used when the classes are non-separable in the input space. The classes take the polynomial shaped surface rather than a hyperplane.

3.3. POS

POS aims at labeling each word with a tag indicating its syntactic role in the sentence like if the word is noun, verb or adverb. POS tagger is a computer program that does this task. Tag-sets are used by the taggers to tag the sentence word. Taggers use a large amount of annotated training corpus to tag (label) properly.POS tagger architecture has three main steps(Robinson, 2009c):

1. **Tokenization**
   It also called Pre-processing. It divides the given text into separated words and punctuation called Tokens.

2. **Ambiguity look-up**
   The aim of this step is to find the most suitable tags for the unknown words using lexicon and guesser.

3. **Ambiguity resolution (Disambiguation)**
   This step also called disambiguation. It uses information about the probability of the word to be a noun or a verb. It also uses information like the sequence of the previously tagged words.

There are three main approaches used for POS taggers. These types are rule-based, stochastic or transformation-based learning approaches. The difference between these approaches is how they assign the tags to the words.

- **Rule-based approach** uses dictionary or lexicon to assign tags to words. Hand write Rules are used to select the most suitable tag when there are more than one suggested tags. This approach needs a direct human interaction to check the written rules. TAGGIT is an example of the rule based tagger (Brill, 1992).

- **The stochastic approach** also called **probabilistic**, uses a training corpus to assign the most suitable tag for a word. Stochastic taggers use Hidden Markov Model (HMM)(Robinson, 2009a). Markov model is a machine learning method based on probabilistic models. HMM is used to find the optimal tags sequence T= {t1, t2… tn} for the given words sequence W= {w1, w2… wn}(Merialdo, 1994) .

- **The transformed-based approach**is a mixture between the rule-based approach and the stochastic approach. It tags the given text automatically but based on rule based algorithm. The transformation-based approach picks up the most probable tag based on a training corpus then applies a set of written rules to see how suitable the chosen tag is (Robinson, 2009d).

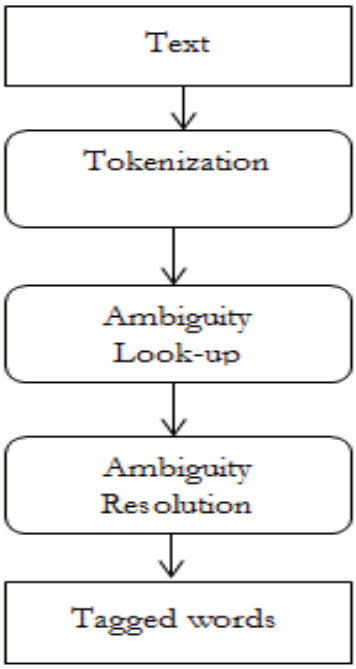


Figure 3: POS tagger architecture

As mentioned in the beginning of this chapter, there are different methods for text analysis and information extraction from tweets like NER, POS and SVM. These methods had limitations when considered for extracting traffic information from tweets. The following paragraph emphasizes what these limitations are.

To use NER method, T-NER tool (Ritter et al., 2011) is one of the high accurate tools for recognizing entities in tweets. This tool is used to recognize a lot of entities like "Person", "Geo-Location", "Company" and "Facility". The problem with this tool is that it is not re-trainable for recognizing new entities like traffic status.

SVM is concerned with classifying tweets to relevant or irrelevant more than extracting information from tweets. This research concerned with following tweets sent by an official Twitter account for reporting traffic information so using further filtering for the tweets is out of scope if this research.

For geo-locating tweets, as mentioned in literature review chapter, most of studies use text analysis methods to detect events use the geo-tagging feature of Twitter to geo-locate the tweet. The drawback of this feature is that according to (Cheng et al., 2010)the tweets sample they collected showed that only 0.42% of all tweets use this feature. Other studies depend on the location information associated with the Twitter user's account like which is not always updated to current location.

Depending on this POS tagging is the method that was most applicable for this research for text analysisfor both information extraction and geo-location.

3.4. Proposed Methodology

3.4.1. T-POS

This research uses the T-POS algorithm developed by(Ritter et al., 2011). Ritter and his team followed the prior experiments of developed POS taggers and how accurate they are. According to Ritter and his team, the baseline POS tagger achieved accuracy of 0.97 on the Brown corpus when it achieved accuracy of 0.76 on tweets. The Stanford POS tagger obtained accuracy of 0.97 using the Penn Treebank WSJ (PTB) as a training dataset. When it applied to tweets it obtained accuracy of 0.8. The main reason for this drop is that tweets contain more OOV (Out Of Vocabulary) like "2morrow" than proper grammatical sentences.

To overcome this problem, Ritter and his team collected and annotated 800 tweets using PTB tag-set. These annotated tweets were used as a training dataset for the T-POS tagger. They added new tags like "retweet", '#hashtag' and '@username'. They also perform clustering to the words that have the same distribution using Jcluster(Sekine, 1998). These clusters are helpful to obtain the lexical variations which are effective for OOV problem. The following example describes the lexical variations for the word "tomorrow" from one cluster:

"2morrow", "2mor", "2moro", "2moro", "2mrw"

Tag	Description	Tag	Description	Tag	Description
CC	Coordination conjunction	JJR	Adjective, comparative	NNPS	Proper noun, plural
CD	Cardinal number	JJS	Adjective, superlative	PDT	Pre-determiner
DT	Determiner	LS	List item marker	POS	Possessive ending
EX	Existential *there*	MD	Model	PRP	Personal pronoun
FW	Foreign word	NN	Noun, singular or mass	PRP$	Possessive pronoun
IN	Preposition or subordinating conjunction	NNS	Noun, plural	RB	Adverb
JJ	Adjective	NNP	Proper noun, singular	RBS	Adverb, superlative
RP	Particle	SYM	Symbol	TO	to

UH	Interjection	VB	Verb, base form	VBD	Verb, past tense
VBG	Verb, gerund or present participle	VBN	Verb, past participle	VBP	Verb, non-3rd person singular present
VBZ	Verb, 3rd person singular present	WDT	Wh-determiner	WP	Wh-pronoun
WP$	Possessive wh-pronoun	WRB	Wh-adverb		

Table 1: PTB tag-set ("The Penn Treebank Tag Set," 1998)

T-POS used the Stanford POS tagger with enhancements to overcome its limitations on tweets. The result of these enhancements was an achievement of 41% error reduction and accuracy of 0.883. The following table shows the errors that Stanford POS tagger made in tagging while T-POS reduce these errors. The table shows the fraction of times the Gold (the right tag) tag is misclassified as the predicted tag (the proposed tag by the taggers) for the two taggers.

Gold	Predicted	Stanford POS tagger	T-POS tagger	Error reduction
NN	NNP	0.0102	0.072	29%
UH	NN	0.387	0.047	88%
VB	NN	0.071	0.032	55%
NNP	NN	0.130	0.125	4%
UH	NNP	0.200	0.036	82%

Table 2: Comparison between the performance of Stanford tagger and T-POS tagger

3.4.2. Information Extraction from Tweets

The role of T-POS tagging is to generate the tagged set of the input tweets. Then, to extract information the tagged set are compared against a pre-defined dictionaries for:

- Traffic status.
- Cause.

For extracting traffic status from tweets, the proposed system extracts the tagged set for the input tweets. Then the tagged set is filtered out to get all the nouns, verbs, adjectives and adverbs from the tagged set. Then, these words are compared against the traffic status dictionaries. These dictionaries are three dictionaries for normal, crowded and jammed statuses.

The same methodology is used for extracting the cause. The proposed system extracts all nouns from the tagged set. Then these nouns are compared against cause dictionary to know the reason behind the traffic status.

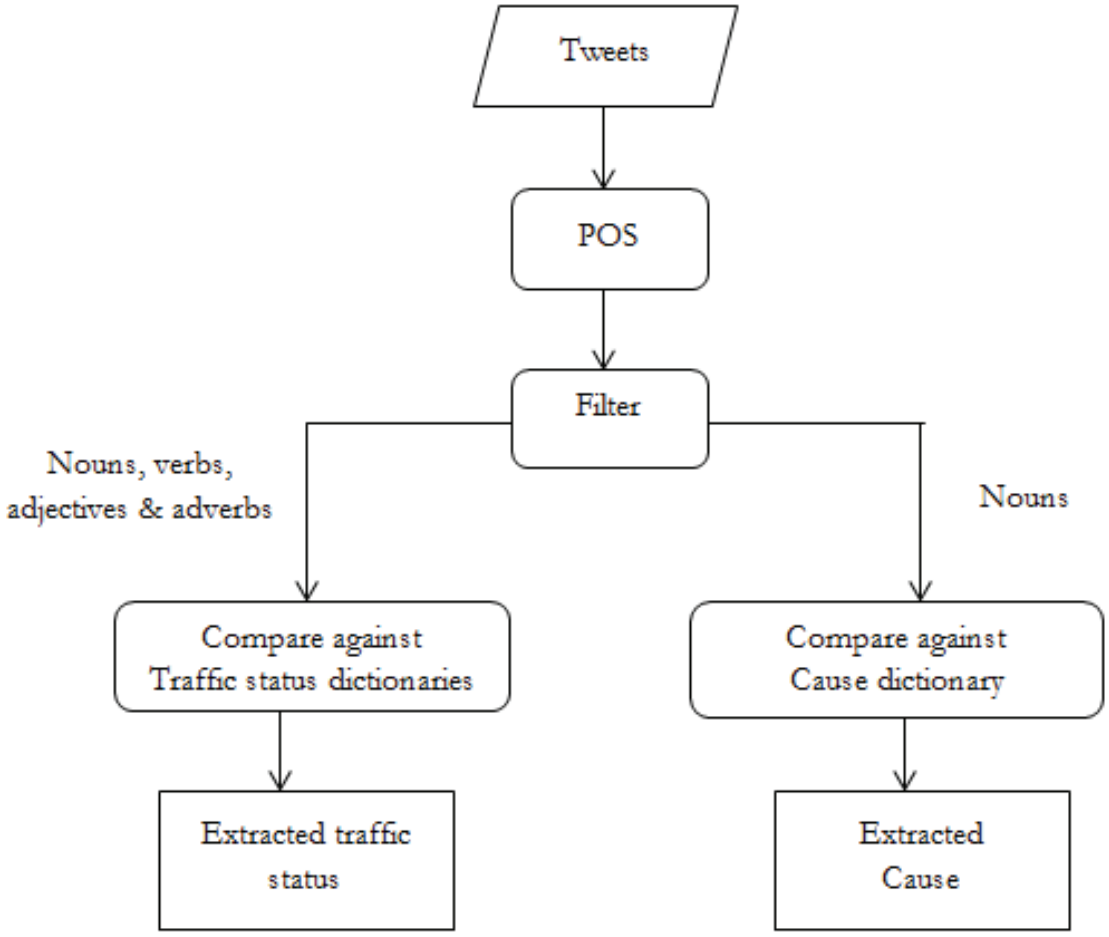


Figure 4: Information extraction

3.4.3. Geo-Locating Tweets

The T-POS tagger also used for extracting locations using custom dictionaries for the following:

- Local streets in central London.
- Road segments that have the same name.
- Roads intersections.

The T-POS role here is to extract all the nouns from the text. Then these nouns are compared against the local dictionaries. If they have streets names these names are extracted. These streets names are sent after that to Google Geo-Code API web service to get the latitude (lat) and longitude(long) for these addresses.

There are some issues to discuss here:

- The official traffic reporting Twitter account reports traffic at roads intersections and express it in the tweet text using "at" keyword.
- Google Geo-Code API can't geo-locate intersections.
- Some roads in London have the same name. For example, the main streets like A4 have small roads segments like "Ellesmere Road" and "Cedars Road" and there are a small road also called "Ellesmere Road". To differentiate between the two roads, the main street is referred to as "A4 Ellesmere Road". This composite name also can't be geo-located by Google Geo-Code API.

To address this problem, two dictionaries were created:

- One for main roads intersections in my research area and their (lat) and (long).
- The other dictionary for the main roads segments and their (lat) and (long).

So after extracted the locations form the text, the extracted locations are compared against the road intersections dictionary and against road segments dictionary. If the locations found at the roads intersections then the location of the road segment or the intersection is extracted from the dictionary in the form of (lat) and (long).

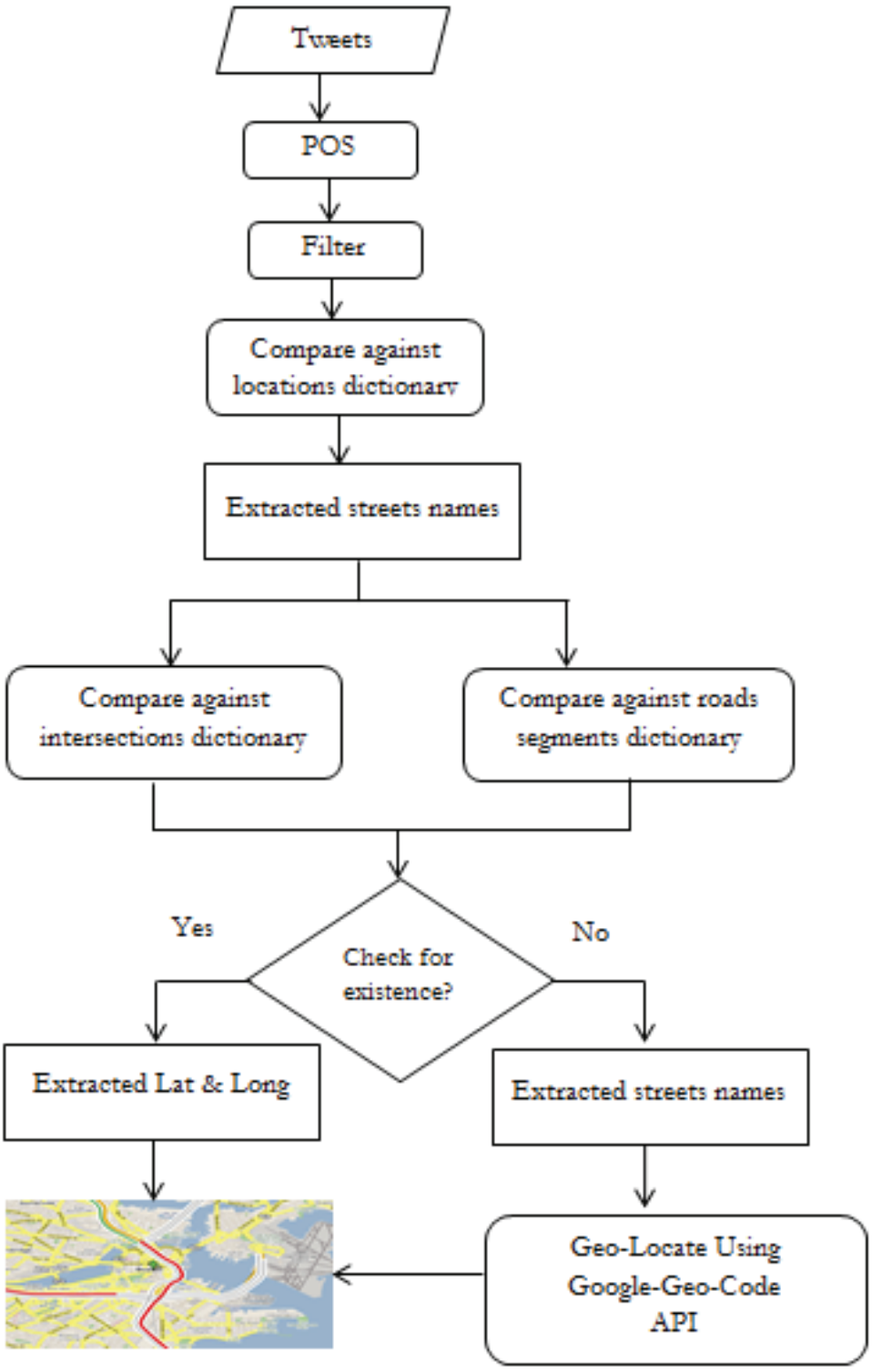


Figure 6: Geo-locating tweets

4. TWEETS MINING & GEO-LOCATING

4.1. Research Area

London city in UK is the area that this research is concerned with. London is one of the most congested cities in Europe (Bryant, 2010).Traffic for London (TfL) ("Transport for London," 2013) is the official website for the local government organization responsible of transport system in Grater London. TfL is responsible of managing and maintaining traffic on major London's road corridors. These road corridors are the main roads into and around London. TfL divides the roads corridors into 18 main corridors each corridor covers a set of streets. These corridors are Central London, North Circular, South Circular, Blackwall Tunnel, A1, A10, A12, A13, A2, A20, A21, A23, A24, A3, A316, A4, A40, and A41.

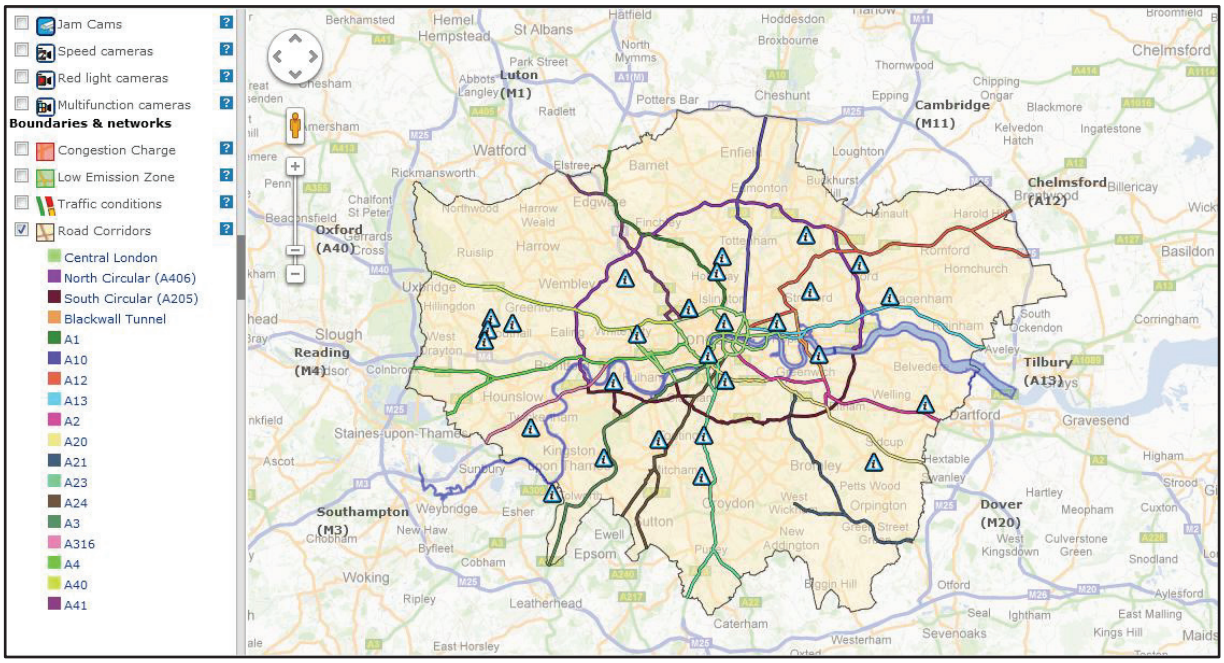


Figure 7: TfL Road Corridors ("live travek news," 2013)

This research focuses on "Central London" corridor as a research area because it covers most of streets in the central part of London. The Central London corridors comprise the Inner Ring, Southern River Route, Bishopsgate Cross Route, City Route, Farringdon Cross Route, and Western Cross Route.

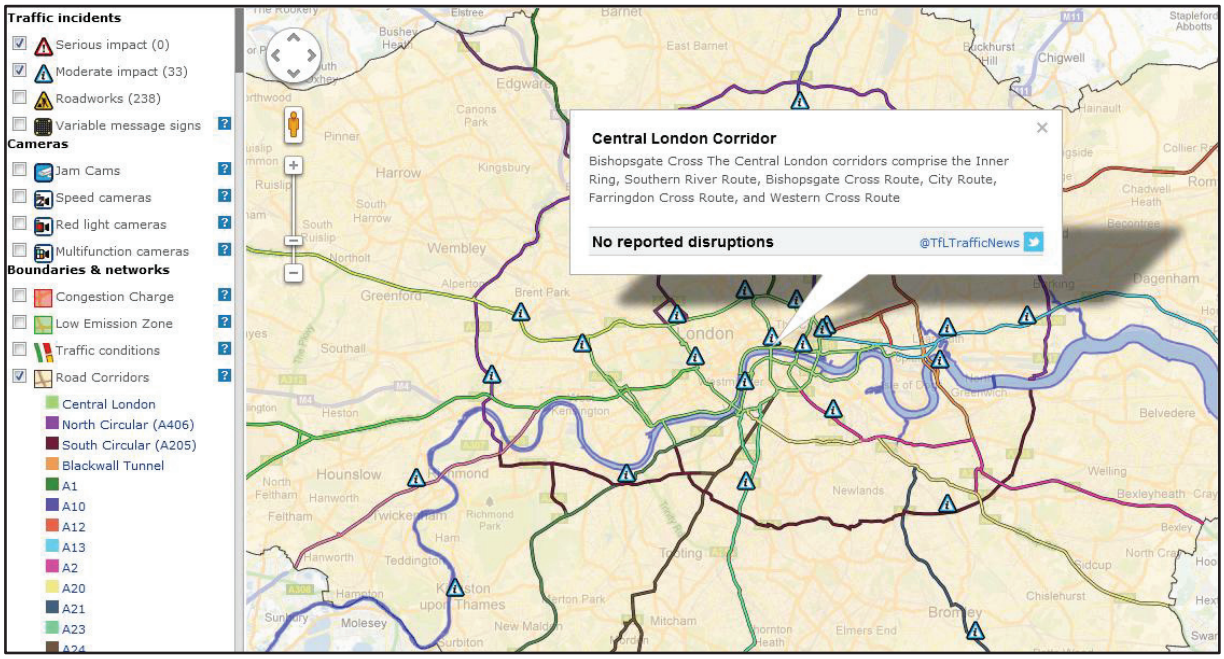


Figure 8: Central London Corridor ("live travek news," 2013)

4.2. Data Collection

In the start of this research, a set of tweets talking about traffic were collected manually. Keywords like "traffic" and "London" were used. The main reason behind these manually collected tweets was to collect information about:

- The number of tweets really related to traffic status in London.
- The number of tweets that are helpful in terms of describing where the traffic problem is, what the problem is.
- How many tweets are geo-located using geo-tagging feature of Twitter.

The output of this small research was as following:

- The number of results for the query "Traffic in London" was 116 tweets
  - 57 tweets contained the words "traffic" and "London" but are irrelevant to traffic status in London.
  - 59 tweets were related to traffic status in London.
  - 25 of the related tweets were descriptive in terms of describing what the traffic status is and where.
  - 12 of the descriptive tweets were sent by official news Twitter accounts.
  - 7 tweets of the all 116 tweets were geo-tagged using geo-tagging feature of Twitter.

According to this output, the research is concerned mainly with the officially sent tweets as they are more reliable than the tweets sent by individuals. @TfLTrafficNews is an official Twitter account for real time road traffic updates in London. This account is managed and updated by Transport for London (TfL). It operates from 06:30 till 21:00. The research will collect tweets sent by this Twitter account.

4.3. System Design & Implementation

The system implementation is composed of four main parts. The first part is to build a framework to collect the tweets in a real time manner and save theminto a local database. The second part is the implementation of the customized T-POS tagger. The Third one is to extract the locations from the second step and geo-locate approximately the traffic location(s). The fourth step is to plot on Google maps the geo-located street with different symbols according to the extracted traffic status.

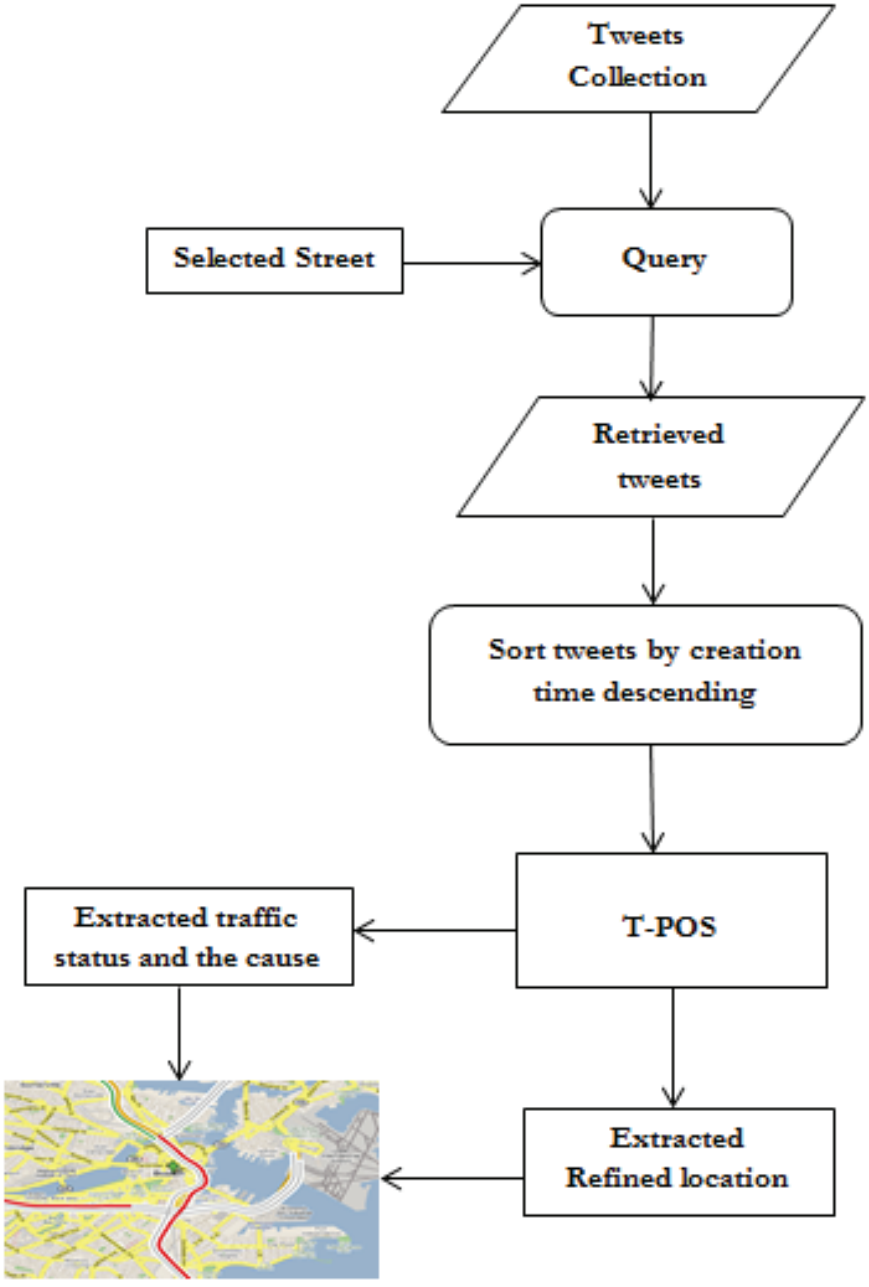


Figure 9: Implementation flowchart

4.1. T-POS Implementation

This algorithm is implemented using Python scripting language on Ubuntu operating system. The output of the algorithm is a list of pairs each pair is composed of the word and its POS tag as an example:

"@paulwalk: It's the view from where I'm living for two weeks"

Word	Tag	Word	Tag	Word	Tag
@paulwalk	USR	where	WRB	weeks	NNS
It	PRP	I	PRP		
's	VBZ	'm	VBP		
the	DT	living	VBG		
view	NN	for	IN		
from	IN	two	CD		

Table 3: Example of "POS tagging"

In this research, the system needs four main entities of information to be extracted from tweets. These entities are:

- **Location**

To extract the locations entity (address), all the nouns (NN, NNP, and NNS) were extracted from the tweets and, then these nouns were compared to a list of London's locations dictionary whichwas saved in a local text file. This list contains all the streets covered by "Central London" corridor. If there are different locations list in the processed tweets, then these locations are taken and separated by "/". For example, the following two tweets are talking about the same incident in the same location.

"The A4 Ellesmere Rd has reopened at Sutton Court Rd following the earlier collision. Residual Qs remain back to junction 2 on the M4"

The output for this step is like this:

"a4ellesmererd/ Sutton court rd"

The extracted words *"a4 ellesmererd"* and "*Sutton court rd"*are roads included in "Central London" corridor. The *"M4"* is not extracted here because it is not in the locations dictionary.

- **Roads segments**

If the extracted streets names has a composite name like *"a4 ellesmererd"*
This location is compared against "Road_Segmnts" dictionary. If it exists then street name will be replaced in the output with its lat and long. So the output of this step is:

"51.487570, -0.264304/ Sutton court rd"

- **Estimation locations**

Estimation locations are locations, for example cross roads intersection name, mentioned in the tweets to report more accurate incident location. These estimation locations or distances usually come after preposition words like "near", "near to", "close to", "about", "at", "in front of", "behind" or "from". Firstly, all the preposition words were extracted from the tweets (IN and TO). Then, the words were extracted after these prepositions. If there are more than one estimated location, they will be extracted and separated by "/".

If the preposition "at" exists in the extracted prepositions tweet text,then the extracted locations from the first step will be compared against "Roads_Intersections" dictionary. If the locations exist in the dictionary, then the (lat) and (long) of the intersection will be retrieved. The output of this step is:

"51.487596,-0.267255"

- **Traffic status**

  To extract the traffic status, all nouns (NN, NNP, and NNS), verbs (VBD, VBG and VBN), adjectives (JJ) and adverbs (RB) were extracted from tweets. Then, it was compared to traffic keywords saved in three text files " Jammed_Traffic" text file, "Crowded_Traffic" text file and "Normal_Traffic" text file. The "Jammed_Traffic" file contains keywords like "closed", "blocked", "blocking" and "jam". The "Crowded_Traffic" file contains words like "delays", "crowd", "crowded" and "slow". The "Normal_Traffic" file contains keywords like "normal", "clear", "smooth" and "open".

  The output of this step is:

  "normal"

- **Cause**

  To extract the causes from the tweets all nouns (NN, NNP, and NNS) were extracted from the tweets. Then compare it to pre-defined causes keywords saved in a local text file "causes". This file contains keywords like "accident", "crash", "roadworks" and "flood". For the pre-mentioned tweets example, the output of the cause is:

  "collision".

The full output of the algorithm for the pre-mentioned tweets is as following:

"51.487570, -0.264304/ Sutton court rd
51.487596,-0.267255
normal
collision".

5. WORKFLOW DESIGN

5.1. System Architecture

The system architecture is composed of client side and server side like most of the web applications architecture. The client side is represented by the web browser. Web browser represents the web application to the user. The server side includes the system server, the tweets server and the Google Geo-Coding server.

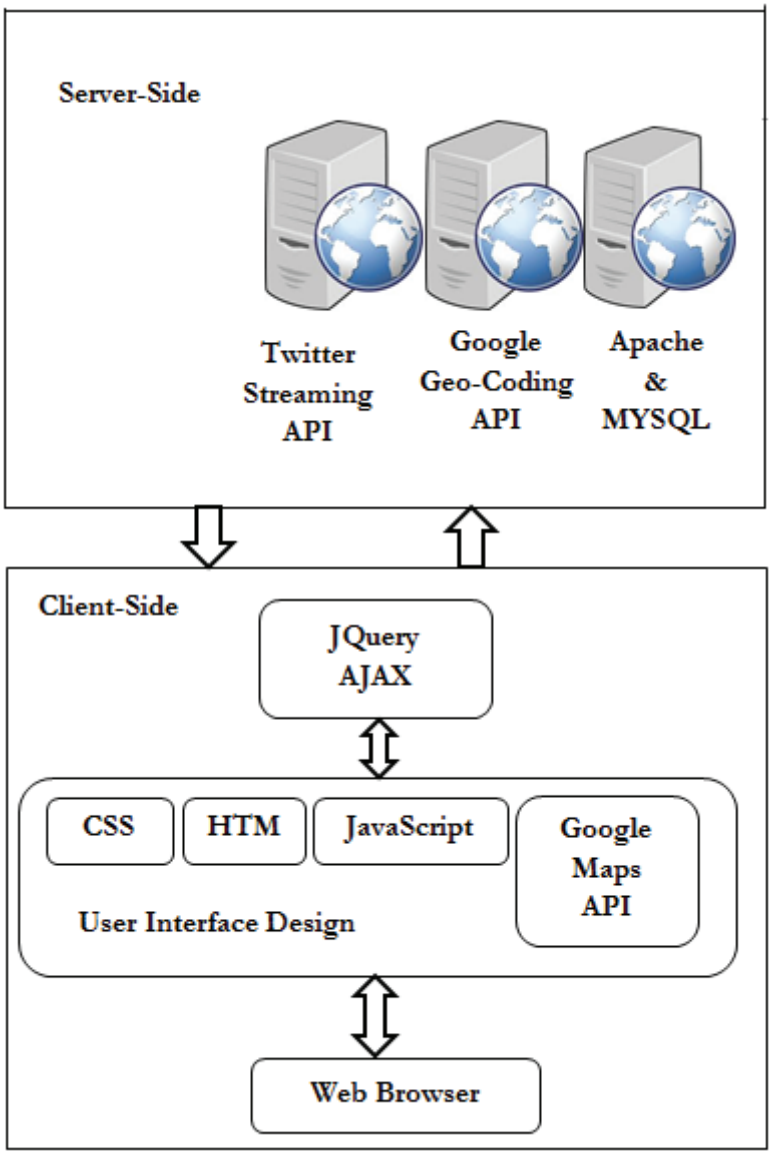


Figure 10: System architecture

5.2. Tweets Data collection

5.2.1. Twitter Streaming API

This study uses Twitter Streaming API ("The Streaming APIs," 2012) for collecting tweets in a real time manner . A PHP script ("PHP," 2013)is used to collect the tweets in a real time manner. Then it saves the returned tweets into the system server using the database layer.

5.2.2. Server's database layer

The database layer is managed by MYSQL Database Management System (DBMS)("MYSQL," 2013). "twitterdatabase" database was created to save the tweets. PHP and MYSQL are running on Apache webserver ("The Apache Software Foundation," 2012).

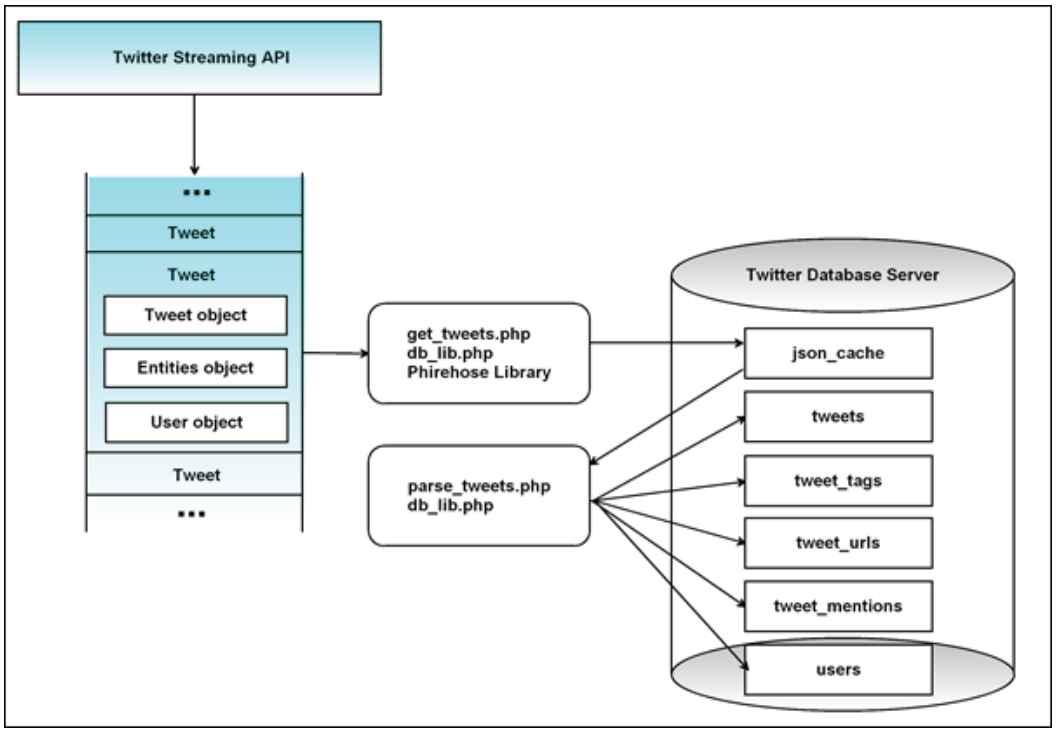


Figure 11: Tweets collection architecture (Green, 2013b)

This framework was originally created by (Green, 2013a) for collecting tweets in a real time manner . Some changes were made on it to fit my requirement(s). The tweets are collected based on a set of keywords using PHP script. These keywords are the street names for the research area and some traffic expressions like "traffic", "jammed" and "congestion".

Every new tweet is saved to "json_cache" table. Then it is parsed by PHP code to extract all the information included in the tweet like the tweet text, the sender, if it has annotations, if it has a hash tag or if it has URLs. The tweet text and some other information are saved to "tweets" table. If the tweet has mentions; these mentions are saved to "tweet_mentions" table. Tweet tags, if included, are saved to "tweet_tags" table. The same if the tweet has URLs; these URLs are saved to "tweet_urls" table. The sender information is saved to "users" table.

The "tweets" table is the most important table for the system as it has the tweets text and when they are sent. The tweets are also classified before being saved to the "tweets" table into four main categories:

- **Official_geoTagged**
  This class is for the tweets sent by official news Twitter account(s) and the tweets are geo-tagged using geo-tagging feature of Twitter.

- **Official**
  This class is for the tweets sent by official news Twitter accounts but the tweets are not geo-tagged.

- **Individual_geoTagged**
  This class is for the tweets sent by individual Twitter accounts and the tweets are geo-tagged using geo-tagging feature of Twitter.

- **Individual**
  This class is for the tweets sent by individual Twitter accounts but the tweets are not geo-tagged.

Tweet_text	Created_at	Lat	Long	Official	Official_ Geo-tagged	Individual	Individual_ Geo-tagged	Screen_name
One lane open both ways past the collision on A23 Streatham High Rd (north of Norbury Station) mainly northbound traffic delays on approach	2/12/2013 11:52:49 AM	0.0	0.0	1	0	0	0	TfLTrafficNews
The traffic signals on A10 Kingsland Rd at Balls Pond Rd are currently out of order; please approach the junction with caution.	2/12/2013 2:51:16 PM	0.0	0.0	1	0	0	0	TfLTrafficNews

Table 4:"tweets" table dataset

This part of the system is running in a standalone mode, which means that it runs as a background service all the time to receive the tweets in a real time format. To run it as a background service a terminal command of Ubuntu is used.

5.3. System Workflow

The system starts with calling the home page "index.php" in the browser.

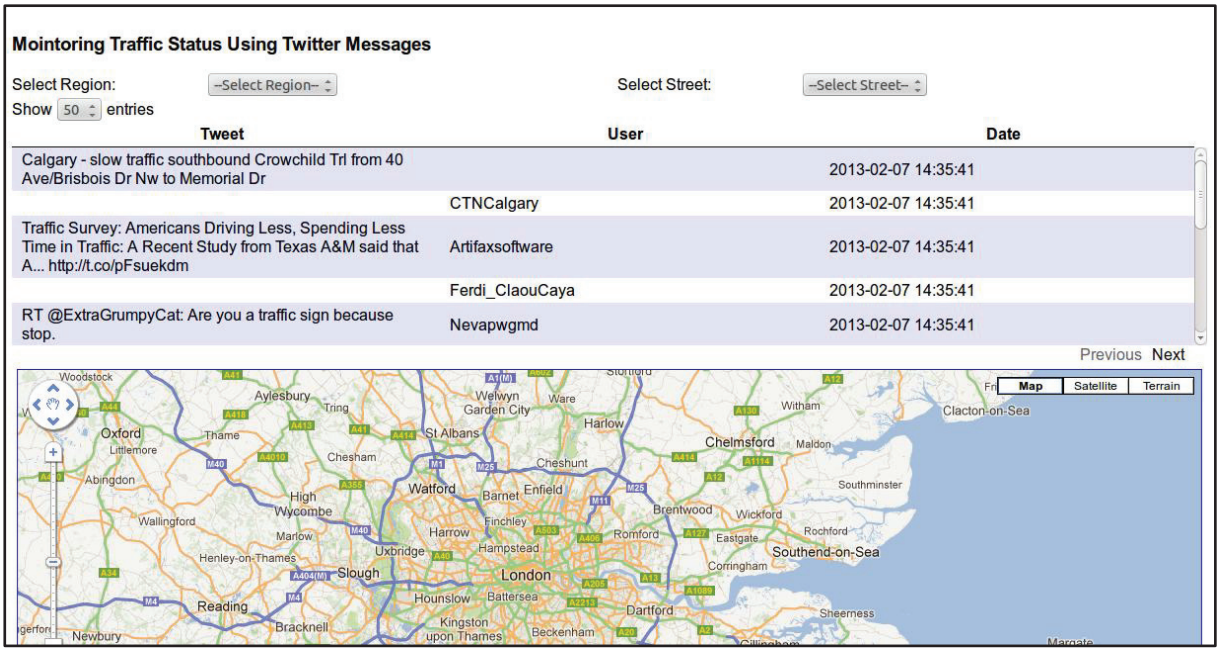


Figure 12: System interface "index.php"

The home page contains all the tweets collected in "twitterdatabase" database shown in the data table. It also has an overview map for London. It has also two drop down boxes to allow the user to select which region and which street in London he wants to check the traffic status for through the sent tweets. The user has to select one region from the region drop down box. In this research, "Central London" is the listed region as it is the research area of concern.

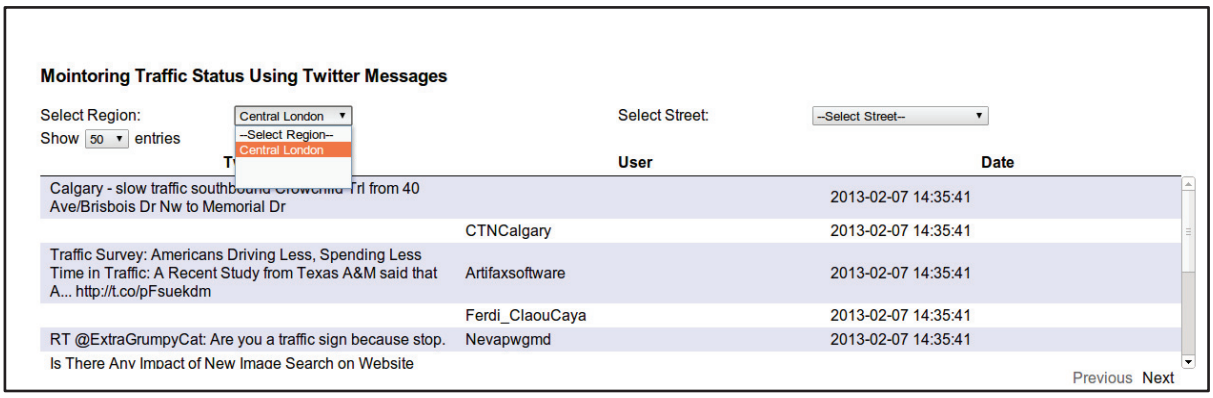


Figure 13: Region selection

Depending on the user's first selection, the second drop down box will be filled with the streets inside the selected region.

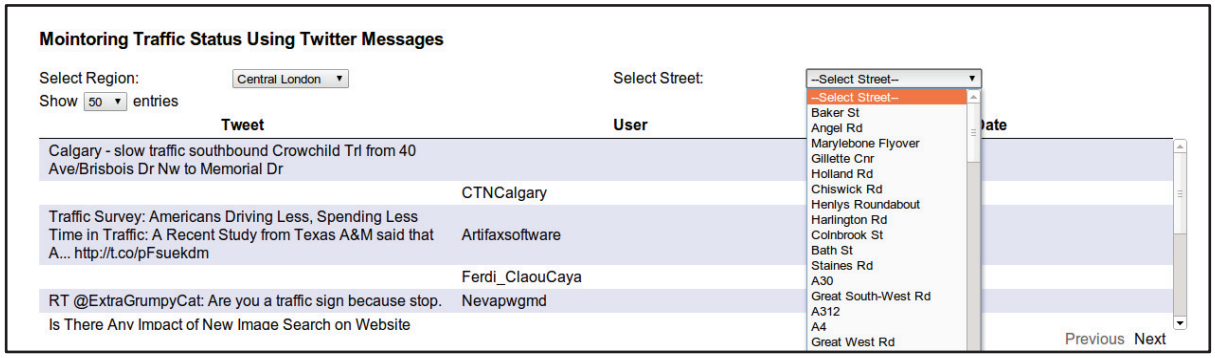


Figure 14: Street selection

The selected street name will be using to query the "tweets" table in the database. The retrieved tweets are shown in the data table.

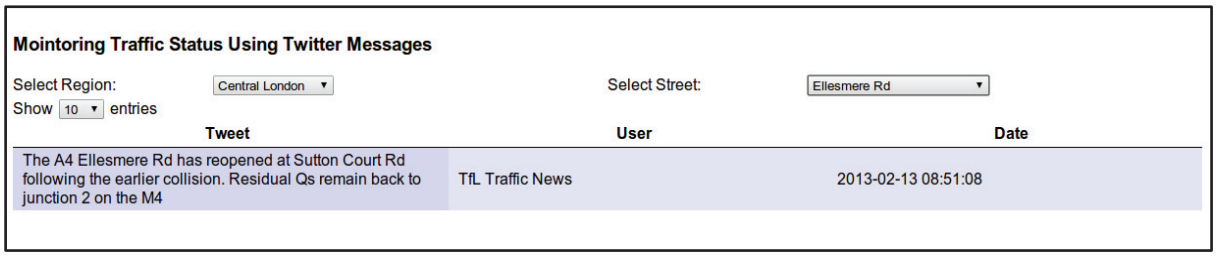


Figure 15: Retrieved tweets from database

These retrieved tweets are being processing using the NLP tool. The results must contain a location and a traffic status at least. If the system couldn't extract this information from the tweets, it will notify the user that the system can't identify traffic status for the given street from the tweets. Otherwise, the extracted locations (locations and estimated locations) will be sent using PHP web service to Google Geo-Code web service which returns latitude (lat) and longitude (lng) for each listed location.

If at least two (lat)s and (lng)s for two locations are found, this information are sent with the traffic status and the incident information to Google Maps API V3. The role of Google Maps API is to plot the (lat) s and (lng) s on the map and to highlight the route between these two locations. The highlight colour will change according to the traffic Status. If the traffic status is "normal" then the route will be highlighted in green colour. If the traffic status is "corwded" the route colour will be yellow. For the "jammed" traffic status, the route colour will be red. An information window will be shown indicating the traffic status written and the cause of the traffic problem.

While all these processes are working in the background, a loading image will be shown to the user.

Figure 16: Loading map

Then, the result will be shown on the map.

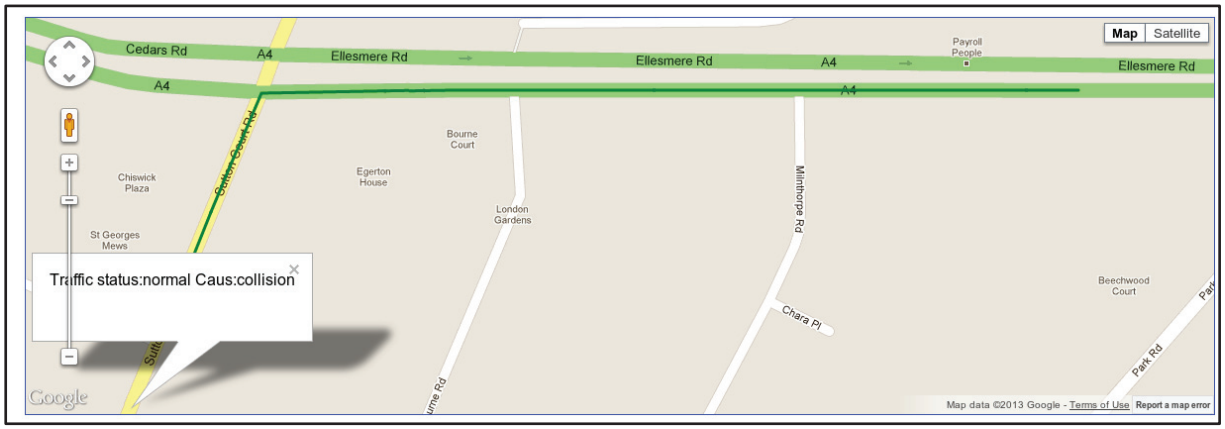


Figure 17: Highlighted Street

6. TESTING AND RESULTS

As stated in the research objectives, the output of the prototype was compared against Google Maps traffic feature to test its accuracy.

6.1. Google Maps Traffic feature

Traffic feature of Google Maps shows the current traffic conditions. The used colours correspond to the speed of the traffic. Red colour means heavy traffic congestion, yellow means a medium congestion and green means free flowing traffic.

Google uses the traffic cameras and the crowdsourcing information. When users use Google Maps on their phones with GPS, Google Maps tracks how fast the users are moving in which street("Traffic conditions on Google Maps," 2009).

Google Maps traffic feature is relatively new and there are relatively few studies that evaluated its accuracy. As a result, a small experiment was executed to collect the traffic conditions for two areas in London (Wandsworth and Bedford). The experiment took one week covering different day times. At the same

time, the same information was tracked for the same times on TomTom("TomTom," 2013)website. A comparison was held between the two systems and the results were almost the same. This experiment was taken into account to know how reliable Google Maps traffic feature is.

6.2. Testing

To test the prototype, screen shots of Google Maps traffic map were taken and saved with date and time. These screen shots were compared to the results with the system prototype.

Date: 18/02/2013
Time: 07:02
Tweet: "Matt here at the LSTCC, just a quick 1, A217 Mitcham Rd closed e/b at A24 Tooting High St due to a building fire, seek alternative route."

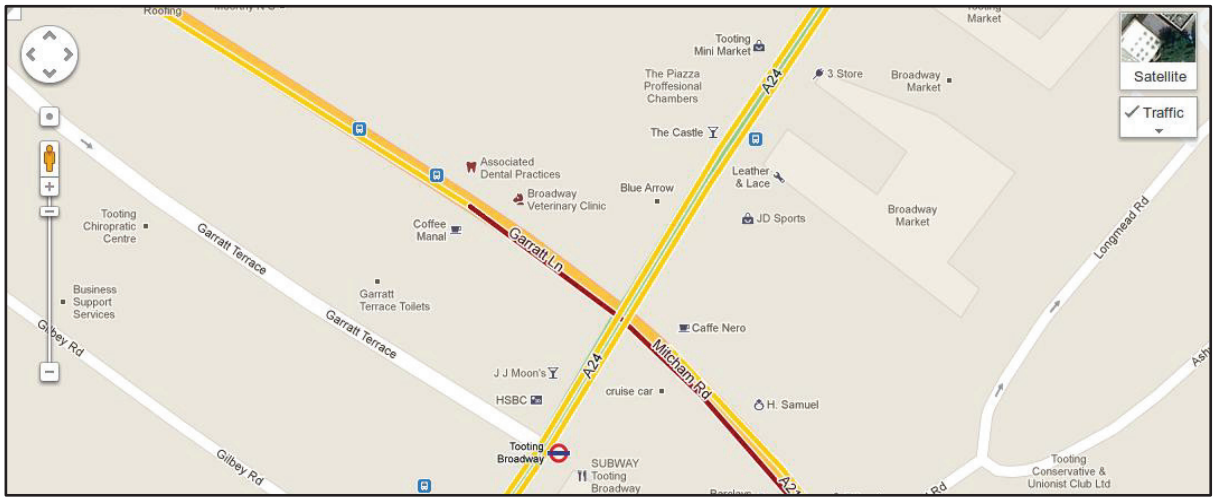


Figure 18: Google maps (traffic status)

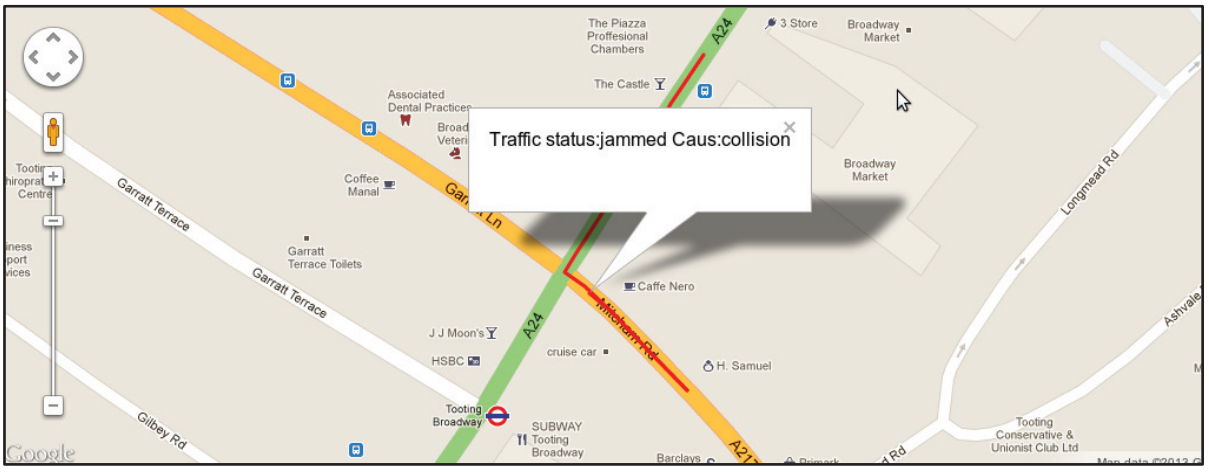


Figure 19: System prototype result

For this example we can see that the same lane of A217 Mitcham Road in Google maps and the system prototype is the same. It is jammed in both outputs when it is not the same for A24 Tooting High street road. The prototype draws the road segment till Tooting High Street when it should have stopped at the intersection point.

Date: 18/02/2013
Time: 12:28 PM
Tweet: "A3218 Old Brompton Rd is closed in both directions at Gloucester Rd due to a collision."

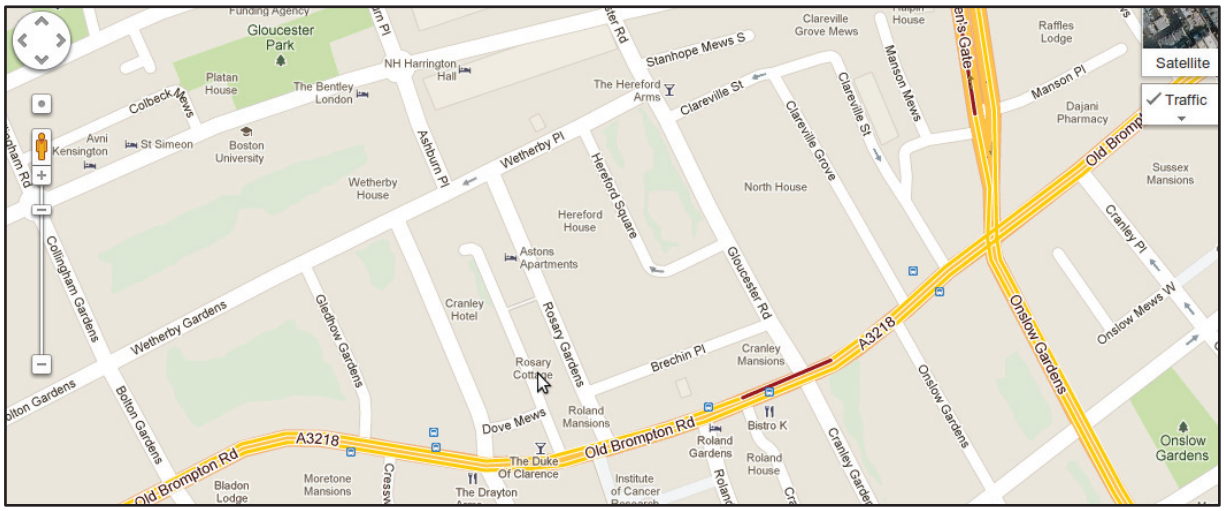


Figure 20: Google traffic map

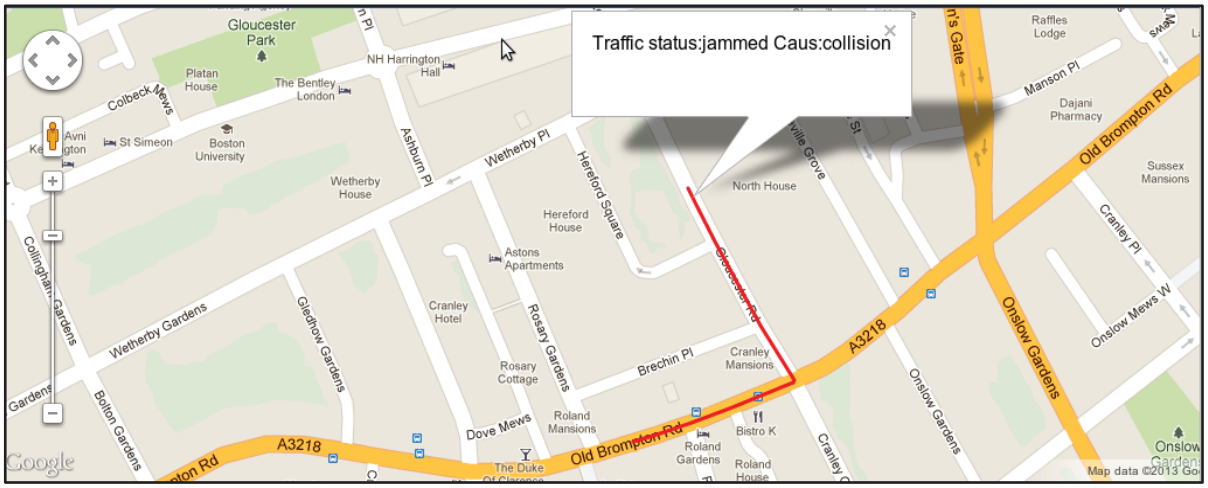


Figure 21: System result

In this example, you can find that a part of A3218 Old Brompton road is Google map is jammed when in the system's map a much longer road portion is highlighted as jammed. The same problem is that the system's prototype doesn't stop drawing till the intersection point but it highlights also the other road in the intersection point.

Date: 18/02/2013
Time: 17:20
Tweet: "A11 Whitechapel High St at Commercial St eastbound the earlier restrictions for gas works have now been removed with all lanes open"

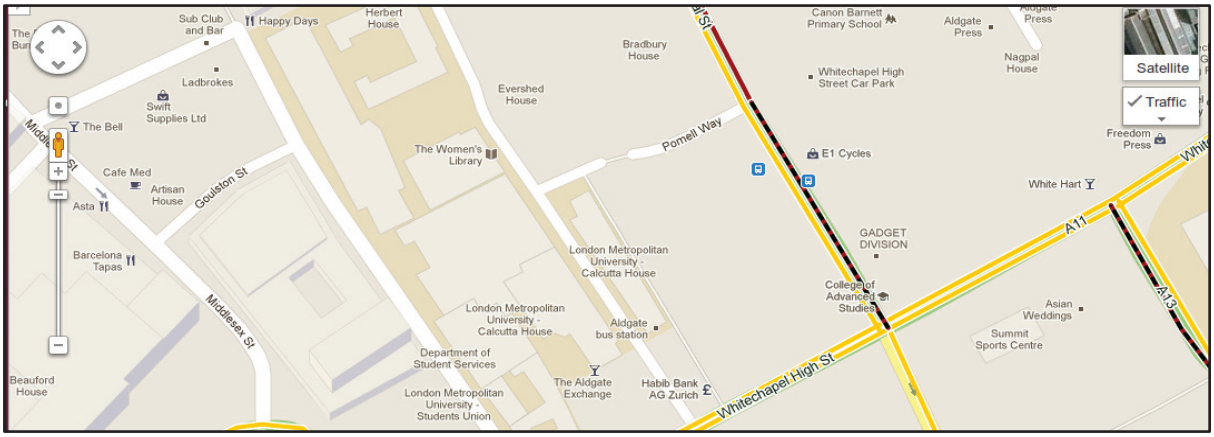


Figure 21: Google traffic map

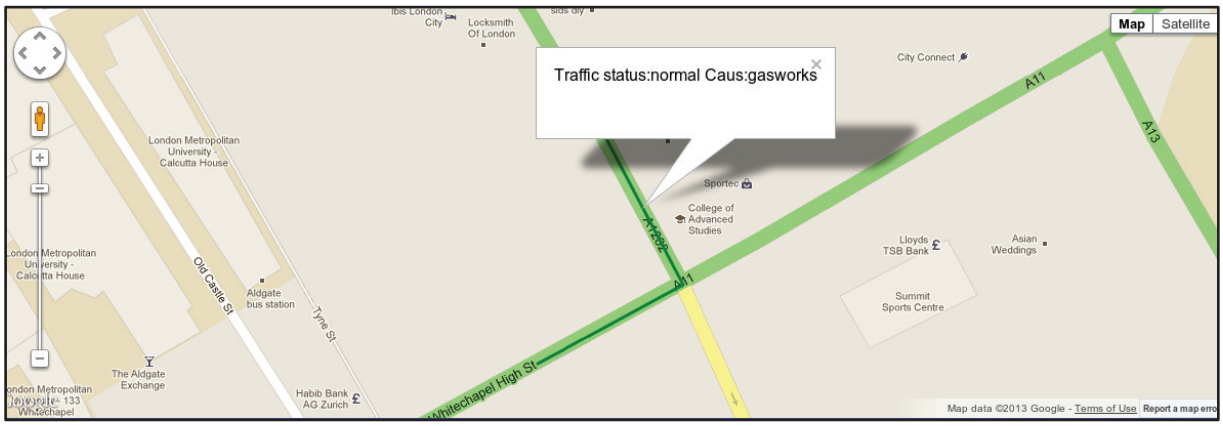


Figure 22: System result

In the previous example, we can see that the tweet that mentioned Whitechapel High street is now open, was interpreted by the system prototype as normal (green highlight). At the same time, Google Maps highlighted the same street as crowded (yellow highlight). Therefore, the same problem of highlighting the intersection street was found.

Date: 18/02/2013
Time: 21:20
Tweet: "A316 Chertsey Road is blocked E/B, towards Richmond at Whitton Rd (Twickenham Stadium) due to collision, very slow traffic on approach"

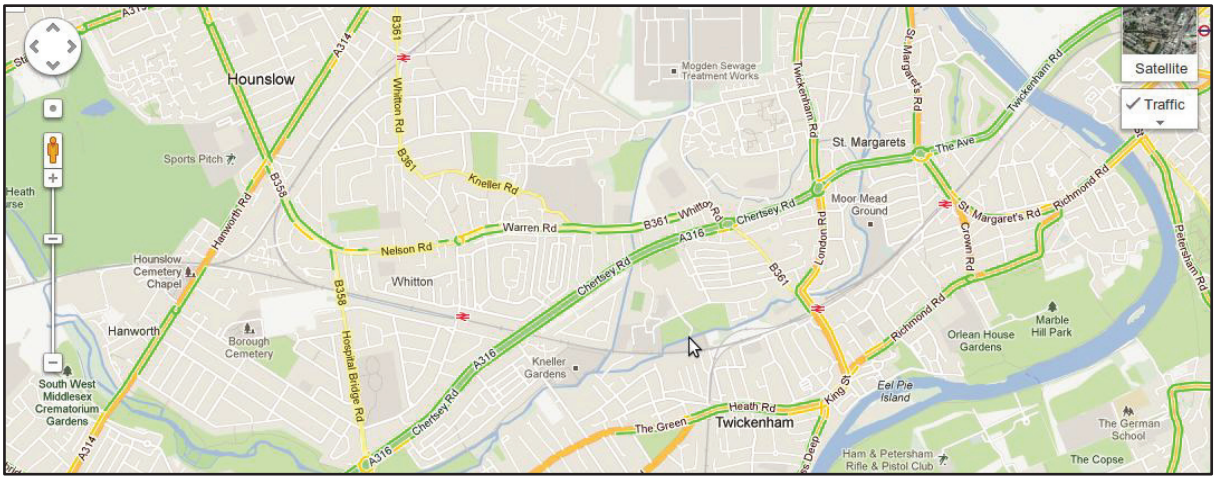


Figure 22: Google traffic map

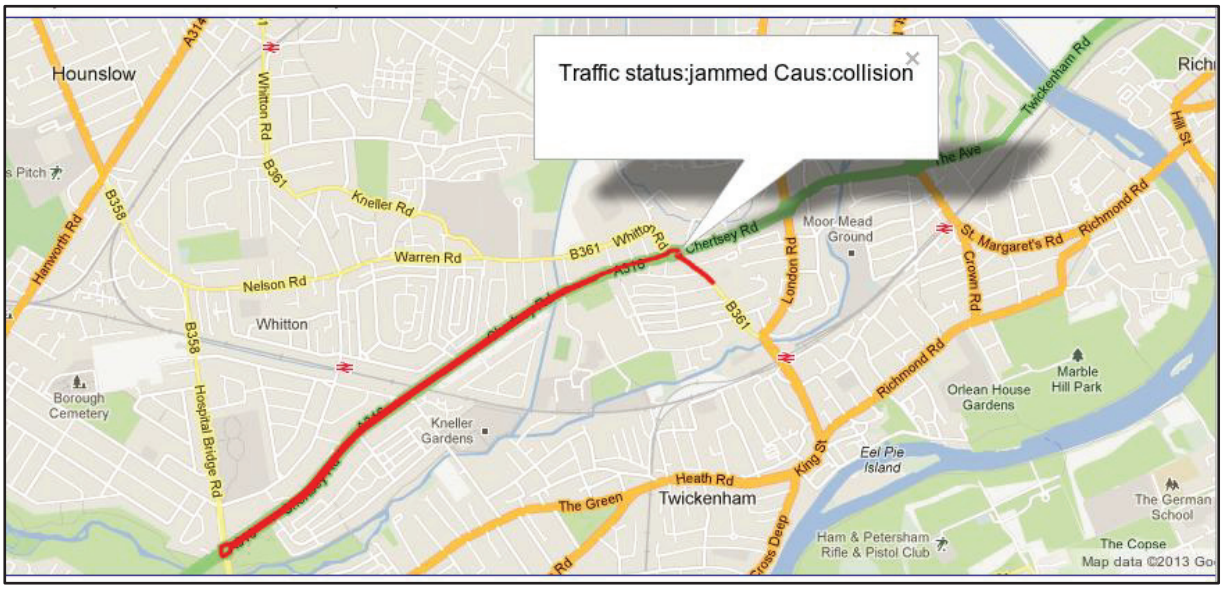


Figure 23: System result

In this example the result are not comparable. Google traffic map shows that A316 Chertsey road is free while the tweet stated that the road hasa slow traffic which is interpreted as jammed traffic in the system.

.

6.3. Results

While selecting these test cases, different times were covered during the day. The testing considered two main features for accuracy:

- **Traffic status accuracy**

From test cases we can infer that, the system uses the traffic status mentioned in the tweets and highlights the route depending on how accurate the information is. As the sender of the tweets is an official Twitter account, so the information is trusted and up to date.

- **The Geo-location accuracy**

The system geo-locate the tweet properly in terms of highlighting the right road in the right area. The main drawbacks in the system's prototype are:

1. It doesn't always highlight the right portion of the road.
2. It highlights a part of the intersection street while it is mentioned in the tweet for its traffic status.

7. CONCLUSION AND RECOMMENDATION

The testing and results chapter describes the evaluation of the system's prototype. The testing based on comparing the system's resulted maps against Google traffic maps. The comparison criteria are the accuracy of traffic status and the accuracy of the location. This chapter describes the conclusion of this research in terms of the fulfilled objectives, the limitations, recommendations and future work.

7.1. Conclusion

The main objectives of this research are to propose a methodology and a system framework to extract traffic information from tweets and the geo-locating of these tweets. Answering the research questions helps in evaluating to what extent this research fulfilled its main objectives.

1. **How to retrieve the tweets in a real time manner?**
   The Twitter Streaming API was used for this task to retrieve the tweets in real time and save it to my DB.

2. **How to filter the tweets using specific query?**
   The filtering of the tweets was to use general traffic keywords such as: "congestion", "stuck" or "jammed".

3. **How to classify the tweets according to the sender into official and individual classes?**
   The system classified the retrieved tweets depending on the name of the sender Twitter account. Only one Twitter account was considered as official and all the rest were considered as individuals.

4. **Which official news Twitter account to follow?**
   The @TfLTrafficNews was the followed account to receive up to date traffic information about London's roads.

5. **Which text mining method to use or modify?**
   Customized Part of speech (T-POS) tagging methods for tagging tweets were used.

6. **How to define the problem of the traffic stated in the tweets?**
   A custom developed dictionaries where used to define the traffic status and the reason behind this problem.

7. **Which method to use or modify to geo-locate the tweets?**
   Part of speech tagging (POS) was also used for this task with special dictionaries for roads, roads segments and intersections in central London.

8. **How to combine between geo-locating step and information extraction step?**
   As the same Method is used then it was easy to combine between the two steps. I use the same algorithm but with different dictionaries. First I use the location dictionaries to geo-locate the tweets. Then the status and causes dictionaries were used with the algorithm to extract traffic status.

9. **Which system architecture to use?**
   It was more reasonable to create a web application. The Client-Server web architecture was used.

10. **How to connect between the system work flow and the implemented algorithm?**
    Web services were like the messengers to move easily between the used algorithm, the server and the client side.

11. **How to plot the extracted locations on the map?**
    Google Maps API V3 was used to plot the extracted location and information on a map.

12. **How to use different line segments' colours and labels to show the traffic status?**
    The Style attributes of Google Maps API were very helpful in highlighting the roads segments with different colours according to the traffic status.

13. **How reliable Google Maps is?**
    A small experiment was executed to compare between Google Maps traffic feature and TomTom traffic information and it was almost the same.

14. **What are the comparison criteria?**
    The criteria to compare between the proposed system's results and Google Maps traffic feature are:
    - The traffic status accuracy.
    - The geo-location status.

15. **How to test the system results?**
    To test the system results, the system processed different tweets during different time of the same day. At the same time, Screen shots were captured for the same locations on Google Maps traffic results for the same times of issuing and processing the tweets.

As a result of answering the previous set of questions, this research has fulfilled its main objectives.

7.2. System's prototype Limitations

The system's prototype has a number of limitations in the used T-POS algorithm and the geo-location job. Google Maps Geo-coding API also has some limitations. The system work flow also has its limitations.

- **T-POS algorithm**

    1. The algorithm has limitations in analysing the tweets, as the tweets are free text written by humans. The word road is written in different forms such as: "Road" or "Rd". Small keywords like "roads", "lanes" and "street" are considered but cases of unexpected text such as: "Great West Road" or "Gt W Rd", are neither controllable nor coverable.

    2. The T-POS is limited to the defined dictionaries. So if the cause of the traffic problem is not listed in the dictionary, it will not be defined by the system.

    3. To consider an address as a location, the T-POS algorithm depend on the key words "road", "street" or "lane" in the end on the address statement. If none of these words were found, then the algorithm will not consider it as a location.

    4. The T-POS algorithm can't process the locations listed with the preposition "between".

- **Google Geo-Code API**

    Two main problems in Google Geo-Code API were experienced:

    1. It can't geo-locate roads intersections.

    2. It can't geo-locate a road segments with composite names which was previously described in methodology chapter.

- **System Workflow**

  The system workflow has some limitations in plotting the route on map:

  1. It can't highlight a route with no more than three locations. If it has more than three locations, it will get confused and mis-plot the route. It also can't highlight a route with less than two locations. These two limitations waste a lot of information listed in the tweet that has more than three locations or it has only one location.
  2. As mentioned in the testing and results chapter, it doesn't highlight the route very properly and it highlights a part of the intersection street which is not right to do.

7.3. Recommendations and future work

The main research objectives were fulfilled. Nevertheless, this research has some limitations. In the next section some recommendations will be shown to get better performance and more accurate results. The recommendations are:

- Develop a suitable Named Entity Recognition (NER) system re-trainable to traffic data and locations data. This will be more dynamic than using pre-defined dictionaries. Also the tweets sent by individual (accounts) are a wasted great source of information. So using a good NER system could make use of it.

- A more robust geo-coding web service will be better for geo-coding intersections and roads with composite names.

Finally, as mentioned before, extracting traffic information from tweets is a relatively new emerging research area. Geo-locating task is the most important component. Accurate definition of which road segment and which lane has the problem will be very helpful for people. Further researches should focus more on this issue.

LIST OF REFERENCES

Abel, F., Hauff, C., Houben, G. J., Stronkman, R., & Tao, K. (2012). *Twitcident: fighting fire with information from social web streams*.

Ahmad, S. (2007). Tutorial on Natural Language Processing. *Artificial Intelligence, 810*(161).

. The Apache Software Foundation. (2012) Retrieved 15-12, 2013, from http://www.apache.org/

Arnott, R., & Small, K. (1994). The economics of traffic congestion. *American Scientist*, 446-455.

Bikel, D. M., Miller, S., Schwartz, R., & Weischedel, R. (1997). *Nymble: a high-performance learning name-finder*. Paper presented at the Proceedings of the fifth conference on Applied natural language processing.

Bontcheva, K., Tablan, V., Maynard, D., & Cunningham, H. (2004). Evolving GATE to meet new challenges in language engineering. *Natural Language Engineering, 10*(3-4), 349-373. doi: doi:10.1017/S1351324904003468

Brill, E. (1992). *A simple rule-based part of speech tagger*.

Bruns, A., & Burgess, J. E. (2011). # Ausvotes: how Twitter covered the 2010 Australian federal election. *Communication, Politics and Culture, 44*(2), 37-56.

Bryant, M. a. B., M. (2010). Britain's most congested street is in south London Retrieved 10-02, 2013, from http://www.standard.co.uk/news/britains-most-congested-street-is-in-south-london-6519888.html

Carvalho, S. F. L. (2012). Real-time sensing of traffic information in twitter messages.

Cheng, Z., Caverlee, J., & Lee, K. (2010). *You are where you tweet: a content-based approach to geo-locating twitter users*. Paper presented at the Proceedings of the 19th ACM international conference on Information and knowledge management, Toronto, ON, Canada.

Collobert, R., Weston, J., #233, Bottou, o., Karlen, M., Kavukcuoglu, K., & Kuksa, P. (2011). Natural Language Processing (Almost) from Scratch. *J. Mach. Learn. Res., 12*, 2493-2537.

Endarnoto, S. K., Pradipta, S., Nugroho, A. S., & Purnama, J. (2011). *Traffic Condition Information Extraction & Visualization from Social Media Twitter for Android Mobile Application*.

. The Google Geocoding API. (2012) Retrieved 08-01-2013, 2013

Green, A. (2013a). 140dev Twitter Framework Retrieved 15-12, 2013, from http://140dev.com/free-twitter-api-source-code-library/

Green, A. (2013b). Free Source Code – Twitter Database Server: Code Architecture Retrieved 15-12, 2013, from http://140dev.com/free-twitter-api-source-code-library/twitter-database-server/code-architecture/

Guduru, N. (2006). *Text Mining with Support Vector Machines and Non-negative Matrix Factorization Algorithms*. University of Rhode Island.

Guo, J. (1997). Critical tokenization and its properties. *Computational Linguistics, 23*(4), 569-596.

Java, A., Song, X., Finin, T., & Tseng, B. (2007). *Why we twitter: understanding microblogging usage and communities*.

. live travek news. (2013), from http://www.tfl.gov.uk/tfl/livetravelnews/realtime/road/?showCorridors=true&showModerate=true&showSevere=true&corridor=Select+a+road+corridor

MacEachren, A. M., Jaiswal, A., Robinson, A. C., Pezanowski, S., Savelyev, A., Mitra, P., . . . Blanford, J. (2011). *SensePlace2: GeoTwitter analytics support for situational awareness*.

Mansouri, A., Affendey, L. S., & Mamat, A. (2008). Named entity recognition approaches. *International Journal of Computer Science and Network Security, 8*, 339-344.

Merialdo, B. (1994). Tagging English text with a probabilistic model. *Comput. Linguist., 20*(2), 155-171.

. MYSQL. (2013) Retrieved 15-12, 2013, from http://www.mysql.com/

Paradesi, S. (2011). Geotagging tweets using their content. *Proceedings of the Twenty-fourth International Florida Artificial Intelligence Research Society Conference*, 355-356.

Patell, P. (2011). EXPERIMENTS ON PHRASAL CHUNKING IN NLP USING EXPONENTIATED GRADIENT FOR STRUCTURED PREDICTION.

. The Penn Treebank Tag Set. (1998) Retrieved 15-02, 2013, from http://www.ims.uni-stuttgart.de/projekte/CorpusWorkbench/CQP-HTMLDemo/PennTreebankTS.html

. PHP. (2013) Retrieved 15-12, 2013, from http://php.net/

Ritter, A., Clark, S., Mausam, & Etzioni, O. (2011). *Named Entity Recognition in Tweets: An Experimental Study*.

Robinson, A. C. (2009a). MARKOV MODELS Retrieved 12-01, 2013, from http://language.worldofcomputing.net/pos-tagging/markov-models.html
Robinson, A. C. (2009b). MAXIMUM ENTROPY Retrieved 12-01-2013, 2013, from http://language.worldofcomputing.net/pos-tagging/maximum-entropy.html
Robinson, A. C. (2009c). PARTS-OF-SPEECH TAGGING Retrieved 12 january, 2013, from http://language.worldofcomputing.net/pos-tagging/parts-of-speech-tagging.html
Robinson, A. C. (2009d). Transformation Based Learning Retrieved 15-01, 2013, from http://language.worldofcomputing.net/pos-tagging/transformation-based-learning.html#
Sakaki, T., Okazaki, M., & Matsuo, Y. (2010). *Earthquake shakes Twitter users: real-time event detection by social sensors*.
Sang, E. F. T. K., & Meulder, F. D. (2003). *Introduction to the CoNLL-2003 shared task: language-independent named entity recognition*. Paper presented at the Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003 - Volume 4, Edmonton, Canada.
Sekine, S. (1998). *NYU: Description of the Japanese NE System used for MET-2*. Paper presented at the Proc. of the Seventh Message Understanding Conference (MUC-7).
. The Streaming APIs. (2012) Retrieved 15-09, 2013, from https://dev.twitter.com/docs/streaming-apis
. TomTom. (2013) Retrieved 15-01, 2013, from http://www.tomtom.com/nl_nl/
. Traffic conditions on Google Maps. (2009) Retrieved 15-01, 2012, from http://www.visualcomplexity.com/vc/project.cfm?id=697
. Transport for London. (2013), from http://www.tfl.gov.uk/
. USGS. (2012) Retrieved 08-01-2013, 2013
Wallach, H. M. (2004). Conditional random fields: An introduction. *Technical Reports (CIS)*, 22.
Wanichayapong, N., Pruthipunyaskul, W., Pattara-Atikom, W., & Chaovalit, P. (2011). *Social-based traffic information extraction and classification*. Paper presented at the ITS Telecommunications (ITST), 2011 11th International Conference on.
Yerva, S. R., Miklós, Z., & Aberer, K. (2010). *It was easy, when apples and blackberries were only fruits*. Paper presented at the Third Web People Search Evaluation Forum (WePS-3), CLEF.
Zook, M., Graham, M., Shelton, T., & Gorman, S. (2010). Volunteered geographic information and crowdsourcing disaster relief: a case study of the Haitian earthquake. *World Medical & Health Policy, 2*(2), 7.