

**DATA EXPLORATION AND ANOMALY DETECTION
ON ROAD NETWORK WITH UNSUPERVISED
OUTLIER DETECTION ON LARGE-SCALE TAXIS GPS
DATA ASSISTING WITH SOCIAL DATA**

BY

DEEPROM SOMKIADCHAROEN

**A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF MASTER OF
ENGINEERING (INFORMATION AND COMMUNICATION
TECHNOLOGY FOR EMBEDDED SYSTEMS)
SIRINDHORN INTERNATIONAL INSTITUTE OF TECHNOLOGY
THAMMASAT UNIVERSITY
ACADEMIC YEAR 2016**

**DATA EXPLORATION AND ANOMALY DETECTION
ON ROAD NETWORK WITH UNSUPERVISED
OUTLIER DETECTION ON LARGE-SCALE TAXIS GPS
DATA ASSISTING WITH SOCIAL DATA**

BY

DEEPROM SOMKIADCHAROEN

**A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF MASTER OF
ENGINEERING (INFORMATION AND COMMUNICATION
TECHNOLOGY FOR EMBEDDED SYSTEMS)
SIRINDHORN INTERNATIONAL INSTITUTE OF TECHNOLOGY
THAMMASAT UNIVERSITY
ACADEMIC YEAR 2016**

DATA EXPLORATION AND ANOMALY DETECTION ON ROAD NETWORK WITH
UNSUPERVISED OUTLIER DETECTION ON LARGE-SCALE TAXIS GPS DATA
ASSISTING WITH SOCIAL DATA

A Thesis Presented

By

DEEPROM SOMKIADCHAROEN

Submitted to

Sirindhorn International Institute of Technology

Thammasat University

In partial fulfillment of the requirements for the degree of
MASTER OF ENGINEERING (INFORMATION AND COMMUNICATION
TECHNOLOGY FOR EMBEDDED SYSTEMS)

Approved as to style and content by

Advisor and Chairperson of Thesis Committee



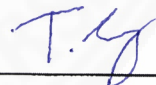
(Asst. Prof. Dr. Teerayut Horanont)

Committee Member and
Chairperson of Examination Committee



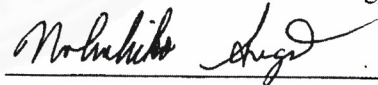
(Dr. Wasan Pattara-Atikom)

Committee Member



(Prof. Dr. Thanaruk Teeramunkong)

Committee Member



(Assoc. Prof. Dr. Nobuhiko Sugino)

AUGUST 2017

Abstract

DATA EXPLORATION AND ANOMALY DETECTION ON ROAD NETWORK WITH UNSUPERVISED OUTLIER DETECTION ON LARGE-SCALE TAXIS GPS DATA ASSISTING WITH SOCIAL DATA

by

DEEPROM SOMKIADCHAROEN

Bachelor of Engineering in Computer Engineering, Mahidol University, 2014

Master of Engineering (Information and Communication Technology for Embedded Systems), Sirindhorn International Institute of Technology, Thammasat University, 2017

Flows of traffic on road is a complex phenomenon. Even a small event can cause massive change on road network as cars can alter their paths or dramatically drop in overall speed. Traffic anomalies can be caused by various factors, for example, accidents, control, protests, sport events, celebrations, and natural disasters. However, as drivers on the road, we cannot know what cause the change in traffic. Thus, we called this anomaly on road network. With advancement in mobile computing and social networking services, and cheaper internet service, data are flooded from various kinds of sensors and user generated data. Combining two or more data sources to confirm to one another would yield significant results. Anomaly detection on taxi mobility data and inferring its cause from Twitter can demonstrate how can we combined sensors and social data to gain information. As a result, we tested our anomaly and inferring method on a Muang Thong Thani area. We are able to detect anomalies on the road and infer their causes via Twitter. From 20 alerted anomalies, we are able to infer 16 of their causes from hashtags. Two of the anomalies are found from cleaned twitter data. We have one false anomaly and the last one we can confirmed on the Twitter website.

Keywords: Anomaly Detection, Data Mining, GPS, Taxi, Twitter

Acknowledgements

This research is financially supported by Thailand Advanced Institute of Science and Technology (TAIST), National Science and Technology Development Agency (NSTDA), Tokyo Institute of Technology, Sirindhorn International Institute of Technology (SIIT), Thammasat University (TU) under the TAIST Tokyo Tech Program.

I would like to express my deepest appreciation to Dr. Teerayut Horanont for his continuous guidance and generous help throughout this research. I also would like to extend my gratitude to committee members for suggestions and serving time as committee members.

Massive thanks to friends, colleagues, and ex-colleagues who share both good and bad time. It is an honor to meet these fantastic people.

I would like to express my gratitude to my family and my girlfriend for the endless love, and ridiculously and continuously support me for every decision I made.

Lastly,

"You can't connect the dots looking forward; you can only connect them looking backward. So you have to trust that the dots will somehow connect in your future. You have to trust in something--your gut, destiny, life and karma, whatever. This approach has never let me down, and it has made all the difference in my life."

Steven Paul Jobs

Table of Contents

Chapter	Title	Page
	Signature Page	i
	Abstract	ii
	Acknowledgements	iii
	Table of Contents	iv
	List of Figures	vi
	List of Tables	vii
1	Introduction	1
	1.1 Background	1
	1.1.1 Intelligent Transportation Systems	1
	1.1.2 Emerging of Massive Data	1
	1.1.2.1 Global Positioning System	1
	1.1.3 Data Analysis	3
	1.1.3.1 Machine Learning	3
	1.2 Objectives	8
	1.4 Outline	9
2	Literature Review	10
	2.1 Spatial Data Set	10
	2.2 Data Exploration	10
	2.2 Anomaly Detection and Verification	11
3	Architectures and Methodology	13
	3.1 Systems and Architecture	13
	3.2 Dataset	16
	3.2.1 Taxi Data	16
	3.2.2 Social Data	18
	3.2.3 Map Data	21
	3.2.3.1 Bangkok Grid Data	21
	3.2.3.2 Road Network Data	22
	3.2.5 Anomaly Detection	23

3.2.6	Infering Root Cause	23
4	Data Exploration on Protesting Period	24
4.1	Overview	24
4.1.1	Dataset	24
4.2	Limitations	25
4.3	Data Cleaning and Exploration	25
5	Anomaly Detection and Inferring	34
5.1	Overview	34
5.2	Data Cleaning	35
5.2.1	Taxi Data	36
5.2.2	Twitter Data	37
5.3	Limitation	37
5.4	Feature Extraction	37
5.4.1	Taxi Data	37
5.4.2	Twitter Data	38
5.5	Data Modeling	39
5.6	Anomaly Events	40
5.7	Verification	41
6	Discussions and Conclusions	43
6.1	Anomaly Detection on Road Network	43
6.2	Problem with Hashtag and Informal Thai	43
6.3	Social Media User Target	43
6.4	Improvements	44
	References	45

List of Figures

Figures	Page
1.1 Example of GPS data.	2
1.2 Example of social data with location based.	3
1.3 Example of decision tree.	6
1.4 Flowchart of assembling decision trees in random forest.	7
1.5 Random forest visualized.	8
3.1 Apache Hadoop 2.0 on Hortonworks Data Platform.	13
3.2 Apache Ambari.	15
3.3 Implemented stack.	16
3.4 Sample of data.	18
3.5 Our data and Google traffic	18
3.6 One record of Tweet in JSON format	21
3.7 Bangkok grid.	22
3.8 Road network in Bangkok.	23
4.1 Closed intersections.	24
4.2 Average numbers of taxis in protesting area.	26
4.3 Average numbers of taxis in non-protesting area.	26
4.4 Average speed in the protesting area.	27
4.5 Average speed in the non-protesting area.	27
4.6 Number of trips from outside to outside without passengers.	28
4.7 Number of trips from outside to outside with passengers.	28
4.8 Number of trips from protesting area to outside without passengers.	29
4.9 Number of trips from protesting area to outside with passengers.	29
4.10 Number of trips from outside to protesting area without passengers.	30
4.11 Number of trips from outside to protesting area with passengers.	30
4.12 Occupy ratio from outside to protesting area	32
4.13 Occupy ratio from protesting area to outside	32
5.1 Application overview	34
5.2 Area of Muang Thong Thani on map	35
5.3 Overview of Muang Thong exhibition halls and resident area	35

5.4 R-trees	36
5.5 Left is one record, right is extracted records	39
5.6 Example of anomaly on 2016-03-19 at tf=73	41



List of Tables

Tables	Page
3.1 Computer specification in Hadoop cluster	14
3.2 Attributes of taxi data	17
5.1 Extracted features	38
5.2 Extracted attributes	40



Chapter 1

Introduction

1.1 Background

Flows of traffic on road is a complex phenomenon. Even a small event can cause massive change on road network as cars can alter their paths or dramatically drop in overall speed. Traffic anomalies can be caused by various factors, for example, accidents, control, protests, sport events, celebrations, and natural disasters. However, as drivers on the road, we cannot know what cause the change in traffic. Thus, we called this anomaly on road network. With advancement in mobile computing and social networking services, and cheaper internet service, data are flooded from various kinds of sensors and user generated data. Combining two or more data sources to confirm to one another would yield significant results. In modern cities, transportation is an essential part in everyday life. Therefore, there is emerging of intelligent transportation systems.

1.1.1 Intelligent Transportation Systems

Transportation has major impact in everyday life ranging from sea to ground to air. By make use of plenty of data and data analysis, a lot of researchers try to come up with better solution to improve transportation efficiency. Many researchers working on this research field with various applications. For example, analyzing movements of people in a city [3, 5, 6, 18], giving better mobility of public transportation [2, 25]. Many researchers working on how traffic flow in a city based on time and events. Some work with how to protect privacy of the massive dataset [4]. One of the major topic in ITS is finding anomaly on road network.

1.1.2 Emerging of Massive Data

1.1.2.1 Global Positioning System

Global Positioning System or GPS provides geolocation and time information to GPS receivers anywhere on earth within line of sight to four or more GPS satellites. GPS itself does not require user to transmit any data to satellites thus make it independent to radio and mobile signals. There are various kinds of applications in GPS integrated systems for example navigation systems, disaster control, and agriculture.

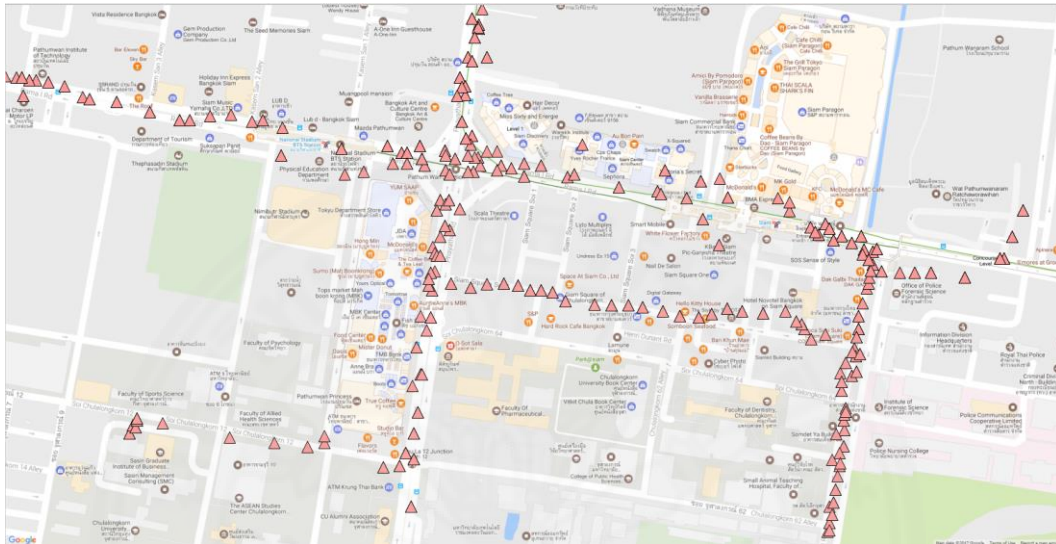


Figure 1.1 Example of GPS data.

1.1.2.2 Social Media Data

Social media data is a user generated data on social media websites such as Twitter, Instagram, and Facebook. There are two kinds of data from user-generated data which are semi-structured and unstructured data. Semi-structured data means there is partial predefined manner of data. It is a text-heavy data that may attached with locations, dates, numbers, and facts. Semi-structured data possibly can be mined with natural language processing (NLP) which is a part of data analysis. Unstructured data have been massively generated by users on the social sites. It is in a form that cannot fit in traditional databases for example videos and images. With advancement in data analysis and hardware. Unstructured data also can be analyzed with various kinds of

techniques in images and video processing.

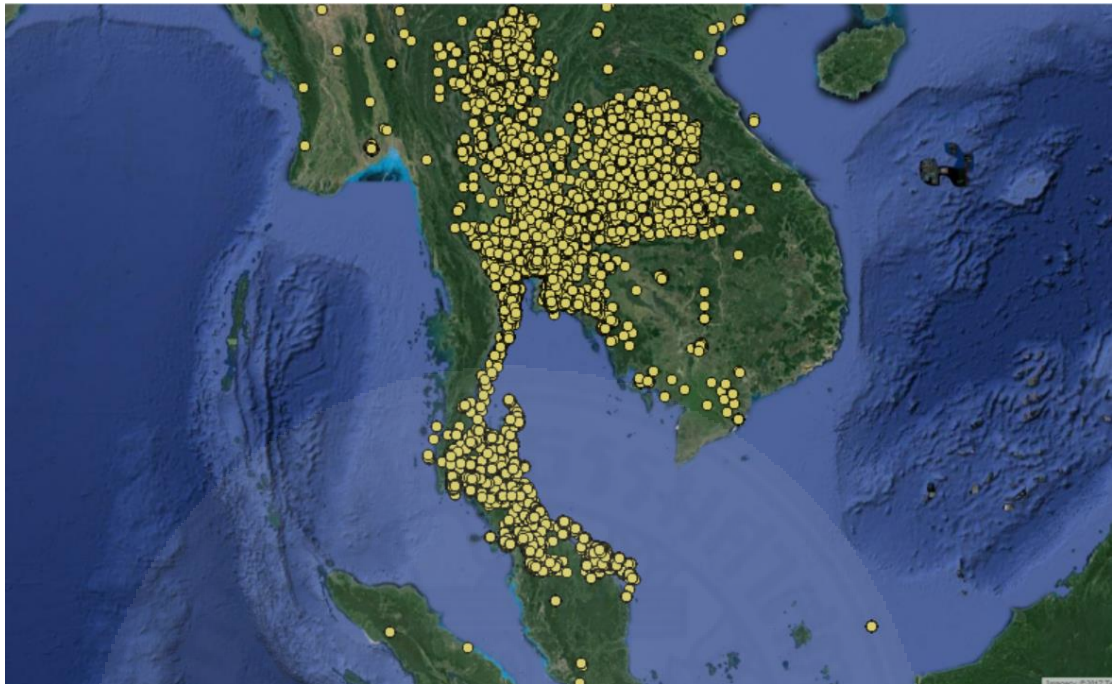


Figure 1.2 Example of social data with location based.

1.1.3 Data Analysis

Data analysis is a process of cleaning, transforming, modeling, and visualizing data to extract information and gain deeper understanding of the data. These information would support decision making, and suggesting conclusion. It is widely use in business, science, and social science domains. When it comes to data analysis, it has various names and approaches. One of the most famous tool is machine learning.

1.1.3.1 Machine Learning

Machine learning is a study that gives computers the ability to learn without explicitly programmed. The assumption of machine learning is to build algorithms that receive input data and use statistical analysis to predict the output. Machine learning can be divided into two categories which are supervised and unsupervised. The supervised algorithms require both input and desired output from human to train the data model. Once the model is made, it can apply what was learned from the training data to new data. Training data for supervised algorithms come with pairs of input and

desired output. For example, vehicle pictures might be labelled as cars and trucks. After some training time and sufficient amount of pictures to train the model, it can classified cars and trucks without labelling the pictures. Unsupervised algorithms, on the other hands, do not require classified output. The algorithms may group unsorted data according to similarities and differences even though there are no categories provided. Therefore, no prior training required to use unsupervised algorithms. We have reviewed some algorithms that are benefits to this research.

1.1.3.1.1 Principal Components Analysis

Principal components analysis or PCA is an algorithm to solve Eigen problem. The algorithm is made to find maximize variance and mutually orthogonal between data regarding on its plane. It is a way to find patterns in data to find similarity and differences. Since patterns in data can be hard to discover in multi dimension which is difficult to visualize, PCA is a recommended tool for analyze ones. Another advantage of PCA is that you can reduce numbers of dimension while losing less information.

There are few simple steps to perform PCA on a set of data which we can demonstrate with a data set with 2 dimensions. From a data set with 2 dimensions, we subtract the mean from each of the data dimensions. The subtracted mean is the average across each dimension. Therefore, each x value has mean \bar{X} subtracted, and each y value has mean \bar{Y} subtracted. Then, we calculate the covariance matrix from what we had computed. Since the data is 2 dimensional, the covariance matrix will be 2×2 . After we obtained the matrix, we can find eigenvectors and eigenvalues of the covariance matrix. From this step we can reduce dimension of the data as the eigenvector with the highest eigenvalue is the principle component of the dataset. It describes most significant relationship between data dimensions. Normally, once we found eigenvectors from covariance matrix, we order them from highest to lowest regarding to eigenvalues. As a result, we get components in order of significance, and we can decide to omit the components that have lesser significance. The omitted components will result in loss few information but it is less significant as it has less eigenvalue.

To sum up, if we have n dimensions data, we calculate n -eigenvectors and eigenvalues, then choose only first p eigenvectors. We get the final data with p dimensions. Once we have preferred eigenvectors we can create a new data set by multiply the eigenvectors with mean-adjusted data. As a result, we have a final data set with data items in columns and dimension along rows

1.1.3.1.2 Decision Tree

Decision tree is one of techniques in predictive modelling in statistics, data mining, and machine learning. Decision tree classifier is constructed from a finite set of attributes where leaves represent class labels and trees represent conjunctions of features lead to the class labels.

The goal of decision tree is to create a classification from multivariable inputs. The tree can be formed by splitting the class-labeled dataset into subsets.

Decision trees consist of three types of node which are root node, internal nodes, and leaf or terminal node. The root node has no incoming edge and zero or more outgoing edges. Internal nodes has one or more incoming nodes and two or more outgoing edges. Leaf or terminal nodes has one incoming node and zero outgoing edges.

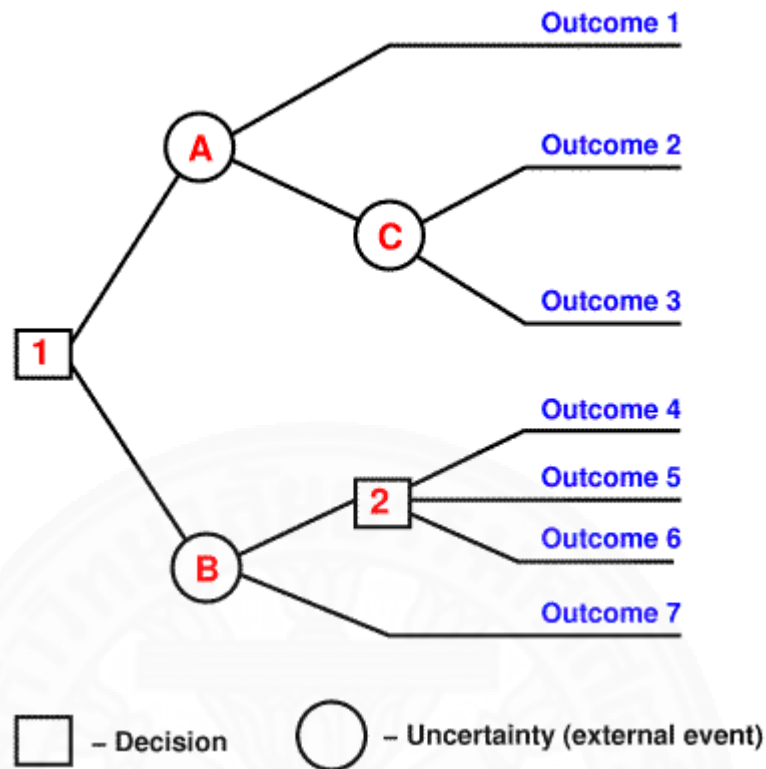


Figure 1.3 Example of decision tree.

1.1.3.2 Random Forest Algorithm

Random forest is assemble of multiple decision trees. To classify a new object based on attributes, each decision tree classifies features based on the inputs and votes for the class. It has property of averaging features to improve the predictive accuracy and avoid overfitting.

$$I_G(p) = \sum_{i=1}^J p_i(1 - p_i) = \sum_{i=1}^J (p_i - p_i^2) = \sum_{i=1}^J p_i - \sum_{i=1}^J p_i^2 = 1 - \sum_{i=1}^J p_i^2 = \sum_{i \neq k} p_i p_k \quad (1.1)$$

The algorithm works as following steps as shown in figure 1.4. First, the algorithm will create N tree of bootstrap samples from the data. Then, each bootstrap sample will grow an unpruned classification tree with randomly sample M try of the predictors and choose the best split among variables. After that, it predicts new data by aggregating the prediction of N trees (majority voting or average for regression). Error estimation can be computed by two methods. The first one is computed at each bootstrap iteration. Data that are not in the bootstrap sample (out-of-bag data) will be

tested against the grown tree with bootstrap sample. The second error estimation is aggregated the out-of-bag predictions and calculate the error rate. We call this the out-of-bag estimate of error rate. From equation 1.1 random forest classifier that we use has Gini impurity which means if any randomly picked features are mislabeled from, Gini impurity will have higher value.

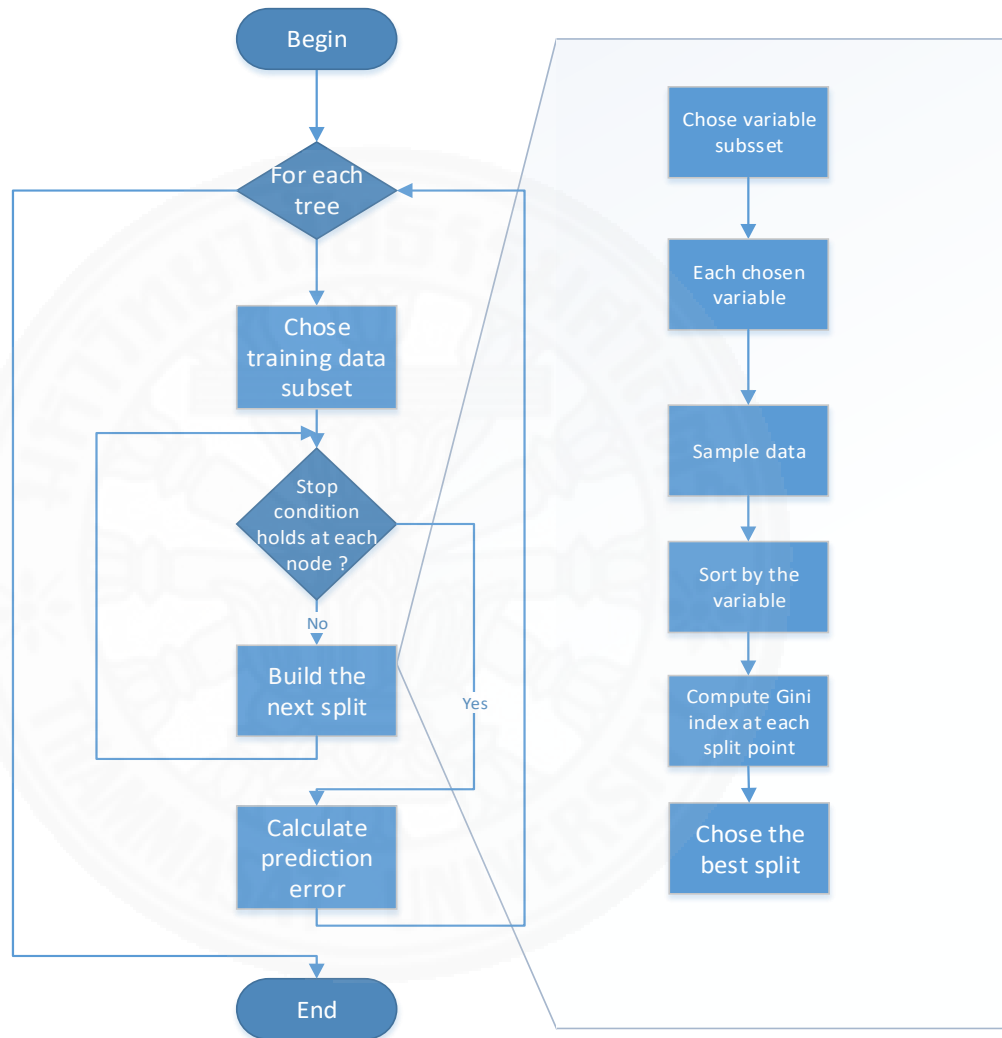


Figure 1.4 Flowchart of assembling decision trees in random forest.

Creating random forest classification and regression yield two additional information which are a variable importance and internal structure of the data. The variable importance is calculated from how much prediction errors increases when out-of-bag data for a specific variable is permuted while other variables are unchanged. The proximity measure is produced by calculating fraction of trees which elements I and J

fall in the same terminal node. The proximity matrix can be used to detect structure of the data too.

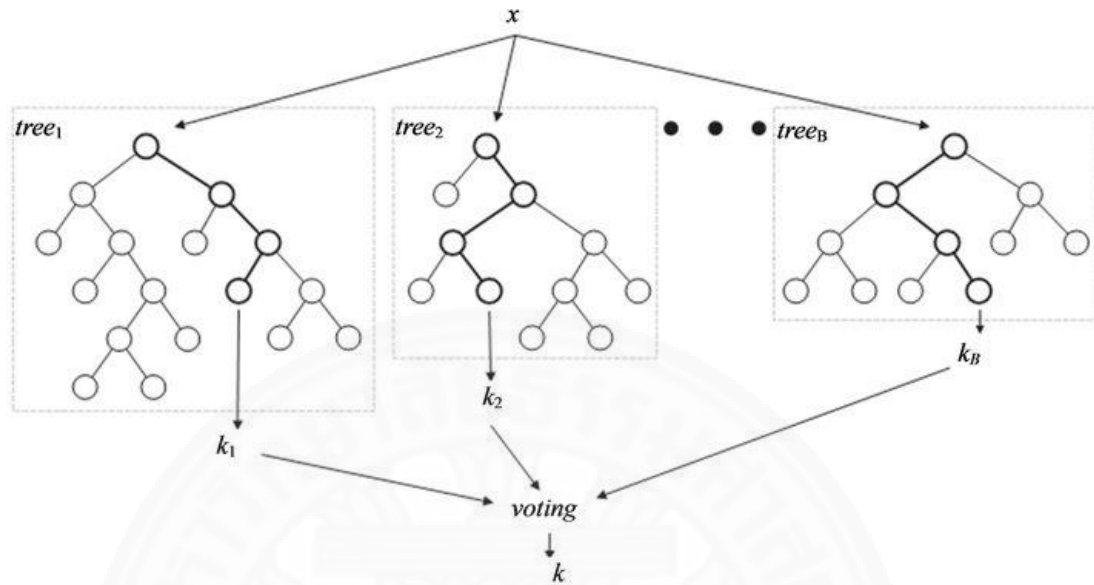


Figure 1.5 Random forest visualized.

1.2 Objectives

On this research, we proposed a platform to achieve the following goals.

- Detect anomaly on road network with massive probed taxi data
- Infer the root cause via Twitter data.

1.3 Contribution

On this research, we made the following contributions.

First, we demonstrate a solution to manage massive dataset analysis of geospatial data effectively.

Second, we present a way to compute spatial operations effectively, as a byproduct of this research, on Apache Hive which would take months of doing in every conventional database.

Third, we proposed a platform that combines two sources of spatial-temporal data into accomplish one purpose, to detect anomaly events and infer possible causes with interval of every 15 minutes.

The platform will detect anomaly from enhanced data of road network combined with GPS data, and will be inferred the cause by collections of hashtags from Twitter data that contain spatial and temporal values.

1.4 Outline

The rest of this thesis is organized in the following manner:

- **Chapter 1** introduces general terms, motivations, and limitation of this thesis.
- **Chapter 2** reviews works by other researchers related to anomaly detection and verification on road networks.
- **Chapter 3** presents systems and architectures to manage massive scale datasets, and describe data sets that we are going to analyze on the next chapter.
- **Chapter 4** devoted to data exploration specifically on protesting period in Bangkok, Thailand.
- **Chapter 5** from chapter 4 we have some improvement on the method from extracted features and perform anomaly detection and inferring the cause.
- **Chapter 6** we discuss the result and what can we make this one better.

CHAPTER 2 LITERATURE REVIEW

As we have two related works on this thesis which are data exploration and anomaly detection, we divided into 3 section which are spatial data set, data exploration, and anomaly detection and verification.

2.1 Spatial Data Set

On city-scale social event detection and evaluation with taxi traces, there are two set of data which are GPS data and event data [32]. The first data set is GPS data which are gathered from 19 September 2009 to 31 December 2011 in Shanghai, China. It consists of 10 billion records of GPS from over 10,000 taxi operated at the time period. The second dataset is records of events from 1st May 2009 to 20th April 2010. The method to find the event is by Google search. If the result of such events appears in the first 10 rows on the website, then it's a credible event.

Looking at another research, inferring the root cause in road traffic anomalies [3], has only one dataset which is GPS data. The data they have 800 million records from 30,000 taxi cars within just 3 months in Beijing, China. In this research, they tried to find anomaly and the root cause path. The data modeling and event detection will be discussed in the next section.

From what we learned so far, finding anomalies on road network requires massive dataset.

2.2 Data Exploration

By reviewing “Extracting Descriptive Life Profiles from Mobile GPS Data” and “Uncovering cab drivers’ behavior patterns from their digital traces”, we adapt some methods from life profiles to taxi profiles because both datasets have a lot of similarities [3, 4]. Zhang D., et al described behavior of taxis that they work on two shifts in China which has similarity to Thailand. Therefore, the same IMEI number of taxi may behave differently when the shift was changed. As they try to uncover most efficient strategies based on large scale of data, they came up with three interested methods which are the

way drivers search for passengers, delivering method, and preferred driving region. This leads us to make one assumption that there are some taxi drivers who prefer to work in protesting area as they see the event as opportunity, not struggles. Pan G., et al used pickup and set-down numbers which were counted in small block 10 x 10 square meters in Hangzhou, China with IDBSCAN algorithm to cluster large scale of data to observe what we call in this research as origin-destination of taxi drivers [18].

2.2 Anomaly Detection and Verification

Anomaly detection is one of the major topics in finding odd patterns in the data. This topic can be found from signal processing such as acoustic anomaly scene by Komatsu to anomalies on road networks by various researchers [7, 11, 13, 20].

On city-scale social event detection and evaluation with taxi traces, the objective of the research is to detect social events and evaluate its impact via taxi GPS. The feature that they used on the research was pick-up and drop-down which we would like to refer it as origin-destination (OD) numbers over regions and quantify impact on transportation systems [5, 31]. Then to detect such events, they use probabilistic model to detect by creating 3D matrix of probability of events. After that, they consider this as an image stacking on top of each other. With watershed algorithm, an image processing technique, they are able to find events that stand out from others.

Chawla proposed 2-step approach to detect anomaly on road network. All of this were done with historical GPS data [33]. The first step is to identify anomaly from historical traffic. To find the anomaly, the algorithm that they implemented was PCA. It searched anomaly on connected links of road network between two regions. The second step is from the feature that they extracted from GPS data which is OD. They converted into OD matrix and apply L1 regularization on the matrix. Solving L1 inverse lead to inferring the route that alters the travelling path which is considered anomaly.

Anomaly detection is not only applied to road network, but also works on actual computer network. On a research called anomaly based network intrusion detection with unsupervised outlier detection [5], they proposed unsupervised method to detect anomalies on network traffic. They implemented unsupervised random forest algorithm to detect anomalies as they did not have attack-free data. To do so, they used 40 features

from traffic data and classified services on the network into 3 classes which are HTTP, Telnet, and FTP and then trained the algorithm with such data. Finally, they got a model that can predict anomalies based on two assumptions that majorities of network traffic are normal and the attacks. If any services pass this predictor and have false labels, it is likely to be anomaly.



Chapter 3

Architectures and Methodology

3.1 Systems and Architecture

To manipulate massive dataset on this research, we use Apache Hadoop stack as a foundation of our system. Apache Hadoop is an open source software that be able to distribute files and process the data via MapReduce model. It is capable to use cluster of commodity hardware because Hadoop is made on assumption that hardware failure is expected and will be handled by the framework. The core system of the Hadoop is known as Hadoop Distributed File System or HDFS, and the MapReduce is the processing part of it. The way Hadoop storing files is to distribute the small chunks of files to all nodes in the cluster. When the processing time comes, the nodes will read data from small chunks and process quickly. This is an advantage of data locality by keeping the data to local system before the need of processing, and it also reduces internal network load too. Apache Hadoop, since version 2.0, contains varieties of additional software and features to facilitate users to work faster than before as shown in figure 3.1.

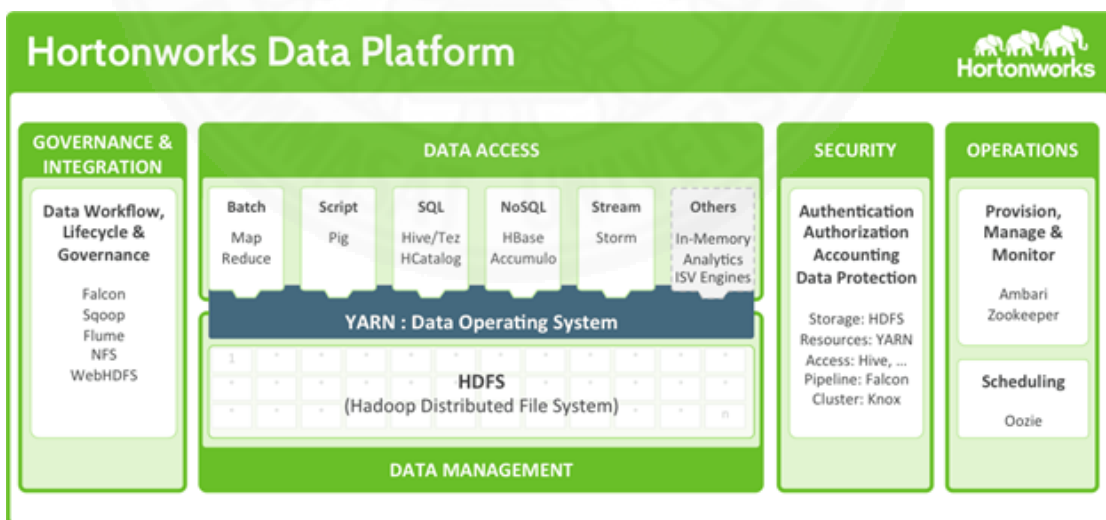


Figure 3.1 Apache Hadoop 2.0 on Hortonworks Data Platform

Because Hadoop is an open source software, there are many companies adopt Hadoop into their data platform technologies, for example, Cloudera, Hortonworks, and Oracle. Implementing the whole system that we prefer requires a lot of tasks and deep

understanding how Linux system work, so we decided to use one of the big company working on big data platform, Hortonworks. The main reason we selected Hortonworks over other brands is that Hortonworks provides the whole system free of charge while other competitors collect royalty fee.

Our cluster consists of 8 computers, 3 of them contain commodity hardware. The specification of the servers has shown in the table 3.1. The more numbers of storage improves performance in reading and writing performance by utilize the available resources, thus spending lesser time in computing. To implement such framework into heterogeneous environment, we use Apache Ambari as a provisioning and installing to simplify implementation. Implementation and provisioning are not only effective with Ambari, but also it works well with performance tuning. Ambari can have multiple versions of tuning for performance tracking and different tuning specifications for heterogeneous cluster. The overview of Apache Ambari can be seen in figure 3.2.

Table 3.1 Computer specification in Hadoop cluster

Components	Dedicated	Commodity
CPU	Xeon 4 Cores 8 Threads	Xeon 4 Cores 4 Threads
Memory	32 GB RAM	16 GB RAM
Storage	8TB HDD	6TB HDD
No. Storage	4	3

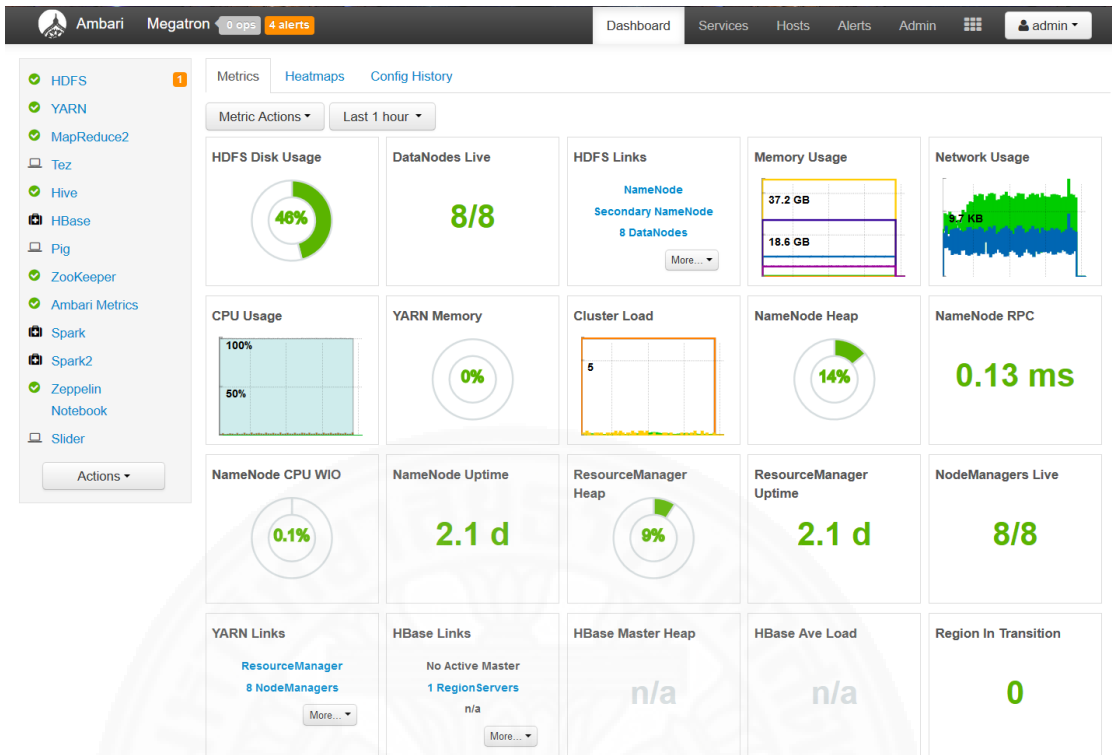


Figure 3.2 Apache Ambari.

The software and services that we use to analyze in this research are Apache Hive and Apache Spark which are built on top of YARN and HDFS. Apache Hive is an SQL like engine which translates SQL command into MapReduce tasks. We will use this tool to clean data and extract features which will be described in other section. Another software that we use for machine learning and visualization is Apache Spark. Spark is an in-memory computing engine that be able to connect to HDFS and compute engine. It will utilize allocated memory in the cluster to work on MapReduce task. As in-memory perspective, data will be loaded into memory once required and will be held in memory for latter computing which makes Spark become user's' favorite. Also, Spark is shipped with native machine learning library, data manipulation tools in various languages which are Java, Scala, R, and Python. The whole stack that we are implemented can be shown in the figure 3.3.

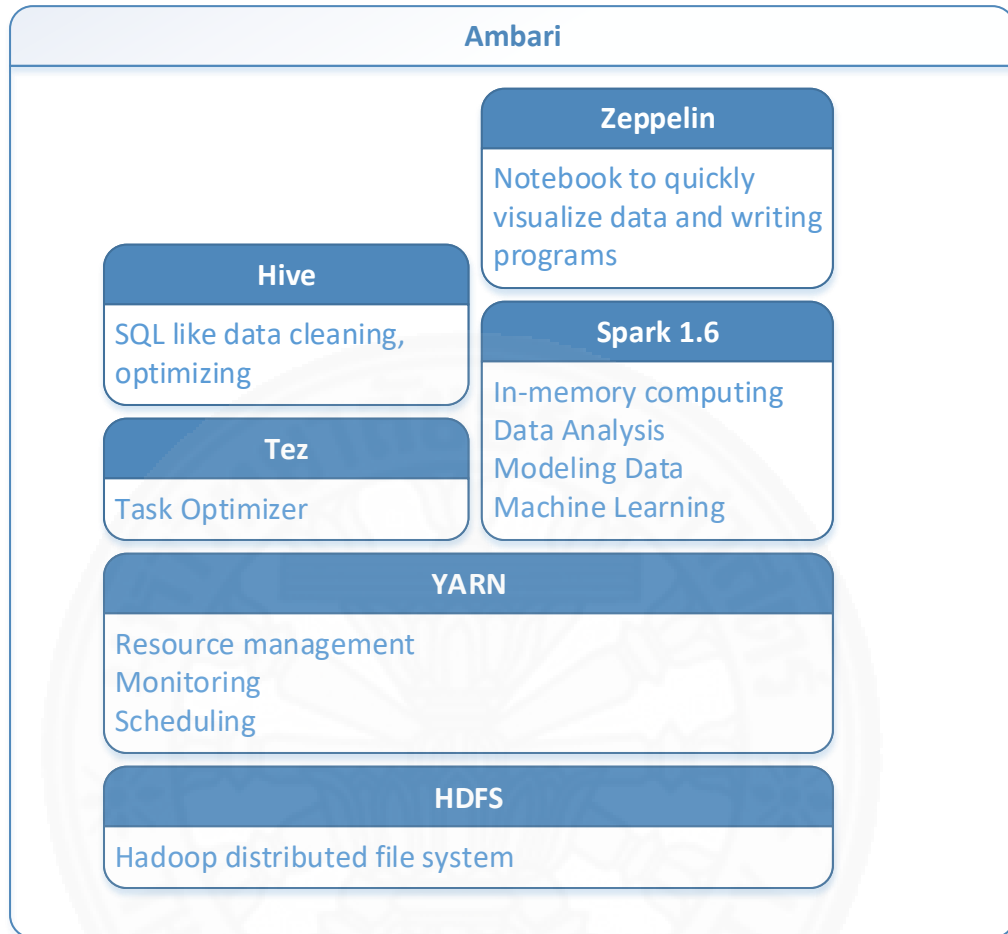


Figure 3.3 Implemented stack.

3.2 Dataset

There are four kinds of data that we have in our hands. Some are requested from a private company, Toyota Tsusho Electronics. Another we gathered from social media sites by ourselves.

3.2.1 Taxi Data

First, we have taxi GPS data that comes with not only their location, time, and current speed, but also meter status which means they have passengers on board or not. We have this data for two separated period of time. First dataset we obtained has a period of one month from 15th December 2013 to 15th January 2014. The GPS data had been collected every 5 seconds on

every car. This period of data is very unique in terms of taxi drivers' behaviors because in Bangkok Thailand we had protesting the government going on and they decided to close 7 major junctions in Bangkok. This data has been explored and published to KICSS 2015 conference. There are roughly 12 billion raw records, 120GB in size, consisting of roughly 8,000 taxi on the dataset. Attributes are shown in the table 3.2. This data is already hide drivers' identities by having only IMEI attached each car.

Table 3.2 Attributes of taxi data

Field	Instance	Memo
IMEI	353419036164759	Identification of the taxi
Latitude	13.74992	Degree
Longitude	100.55402	Degree
Speed	12.0	Speed (km/h)
Direction	116.1	Degree
Error	1.7	Floating point
Acceleration	0 or 1	0 no acceleration 1 an acceleration
Meter	0 or 1	0 no passengers 1 with passengers
Date Time	1387040401	Unix Time
Data source	46,8,9,50	Kinds of source 8 and 9 are for taxi

Second dataset is identical to the first dataset except that this time we have more recent data and longer period of time. The length is 5 months during 1st January 2016 to 31st May 2016 on the same area of Bangkok and perimeters. During this period of time, there is no protesting nor road blockades, so we expected most of the GPS data

points to be normal. Also, there are roughly 60 billion records, 600GB in size, for 8,000 taxis. This dataset is the main part of our research for anomaly detection on the road network.

T00T0S00	T3'04013	T00'14810	T00'0	85'0	0'0	0	0	S0T3-IS-T2	S3:21:21
T00T021T	T3'18T82	T00'00ST2	00'0	00'0	0'0	0	0	S0T3-IS-T2	S3:21:21
T0002430	T3'1T110	T00'41302	0'0	0'0	0'0	T	T	S0T3-IS-T2	S3:21:20
3234T0030T0032	T3'8S00T	T00'20440	01'0	S0'8	T'1	0	T	S0T3-IS-T2	S3:21:30
T00T0001	T3'04005	T00'20305	0'0	T00'0	0'0	0	0	S0T3-IS-T2	S3:21:20
T00T0121	T3'00S03	T00'12080	0'0	S08'0	0'0	0	T	S0T3-IS-T2	S3:21:20
T00T0433	T3'00ST1	T00'12022	0'0	S0'0	0'0	0	T	S0T3-IS-T2	S3:21:24
T00T0040	T3'11003	T00'25508	3S'0	T00'0	0'0	0	0	S0T3-IS-T2	S3:21:20
3234T0030T018140T	T3'18003	T00'22212	5S'0	T11'1	T'2	0	0	S0T3-IS-T2	S3:21:25
T00T0280	T3'13283	T00'10351	0'0	0'0	0'0	T	T	S0T3-IS-T2	S3:21:20

Figure 3.4 Sample of data

We try to ensure that our data is trust worthy by comparing with trusted source, Google map traffic, on the same period of time. As a result we obtain similar average speed.

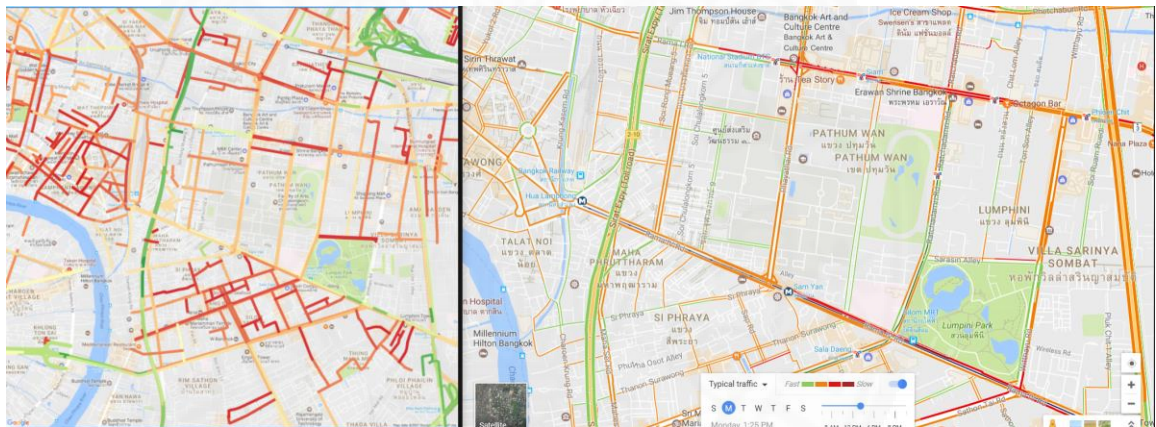


Figure 3.5 left our data, right Google traffic

3.2.2 Social Data

The third dataset is social data. We gathered Twitter's tweets from available application programming interfaces (APIs) for developers. We gathered tweets by writing a crawler that listen to tweet streaming service. The API comes with various options to retrieve data, for example, we can get the data from part of the words, hashtags, or locations. We selected only tweets that contain geolocation and store them in text files. To select data with locations, we have to select bounding box of latitude and longitude format. In our case, we bound the whole Thailand and some parts of neighbor countries. However, bounding countrywide has some downsides too. Twitter only gives partial of data if our bounding size is too large, we receive lesser data in the

area too. The gathered data came in JSON format containing 40-50 key-value attributes with User-related data attached. An example data is shown in the figure 3.4. We are able to crawl at least 100,000 to 200,000 tweets per day depending on day of week and events. It has the size of 300 - 500 MB for each day.



```

{
  'favorited': "False",
  'id': 723919365087744000,
  'in_reply_to_status_id_str': None,
  'retweeted': False,
  'created_at': 'Sat Apr 23 17:00:01 +0000 2016',
  'place': {
    'country_code': 'TH',
    'attributes': {},
    'bounding_box': {
      'coordinates': [
        [
          [
            99.616626,
            13.746851
          ],
          [
            99.616626,
            13.944927
          ],
          [
            99.970718,
            13.944927
          ],
          [
            99.970718,
            13.746851
          ]
        ]
      ],
      'type': 'Polygon'
    },
    'url': 'https://api.twitter.com/1.1/geo/id/01834fee4a54f87d.json',
    'id': '01834fee4a54f87d',
    'country': 'ประเทศไทย',
    'entities': {
      'symbols': [],
      'user_mentions': [
        {
          'id_str': '2920384321',
          'name': 'if',
          'indices': [
            62,
            71
          ],
          'id': 2920384321,
          'screen_name': 'ifworada'
        }
      ],
      'urls': [
      ],
      'hashtags': [
      ]
    },
    'in_reply_to_status_id': None,
    'in_reply_to_screen_name': None,
    'contributors': None,
    'id_str': '723919365087744000',

```

```

'user': {
  'verified': False,
  'geo_enabled': True,
  'profile_sidebar_fill_color': '000000',
  'id': 2153278388,
  'following': None,
  'description': 'happ happ ifworada~',
  'time_zone': 'Bangkok',
  'created_at': 'Thu Oct 24 16:34:03 +0000 2013',
  'name': 'px',
  'friends_count': 130,
  'profile_image_url_https': 'https://pbs.twimg.com/profile_images/710893286861316096/L1FBV0h_normal.jpg',
  'is_translator': False,
  'profile_banner_url': 'https://pbs.twimg.com/profile_banners/2153278388/1458559445',
  'profile_image_url': 'http://pbs.twimg.com/profile_images/710893286861316096/L1FBV0h_normal.jpg',
  'contributors_enabled': False,
  'profile_background_image_url': 'http://abs.twimg.com/images/themes/theme18/bg.gif',
  'statuses_count': 27709,
  'utc_offset': 25200,
  'profile_background_color': 'ACDED6',
  'profile_background_image_url_https': 'https://abs.twimg.com/images/themes/theme18/bg.gif',
  'listed_count': 1,
  'lang': 'th',
  'notifications': None,
  'default_profile_image': False,
  'protected': False,
  'location': None,
  'profile_link_color': '038543',
  'id_str': '2153278388',
  'default_profile': False,
  'follow_request_sent': None,
  'url': None,
  'profile_sidebar_border_color': '000000',
  'screen_name': 'prprouda',
  'favourites_count': 290,
  'profile_background_tile': False,
  'profile_text_color': '000000',
  'followers_count': 200,
  'profile_use_background_image': True
},
'geo': None,
'lang': 'th',
'in_reply_to_user_id_str': None,
'timestamp_ms': '1461430801065',
'is_quote_status': False,
'filter_level': 'low',
'truncated': False,
'in_reply_to_user_id': None,
'retweet_count': 0,
'text': 'ก็จะแฮปปี้เป็นคนแรกของเมืองนี้ได้ออกกับเมืองเลยนะฮะฮะฮะแฮปปี้แฮปปี้❤️ @ifworada',
'coordinates': None,
'favorite_count': 0,
'source': '<a href="http://twitter.com/download/iphone" rel="nofollow">Twitter for iPhone</a>'
}

```

Figure 3.6 One record of Tweet in JSON format

3.2.3 Map Data

To explore relationships among massive spatial temporal data, maps are needed to visualized extracted information. However, maps can come in various forms such as grid or road networks. In this research, we have used both grid and road networks map which will be described in the next sub topic.

3.2.3.1 Bangkok Grid Data

Bangkok grid data is made from bounding Bangkok area and then divided into grid of 1 km² to map GPS points into each grid to analyze mobility of taxis. Bangkok

grid can estimate a coarse location of each taxi to inspect its mobility. Also, grid data is easier to manipulate programmatically. The grid is shown in figure YYY.

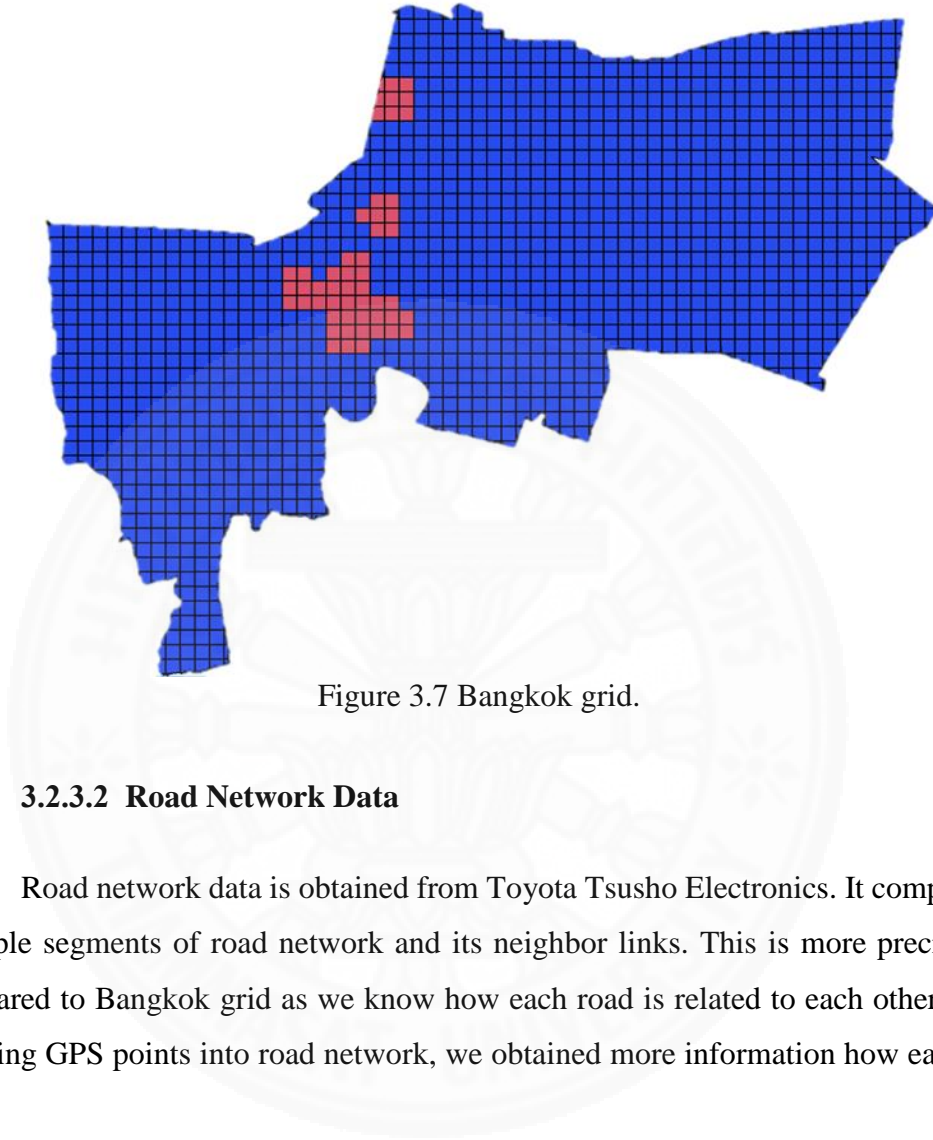


Figure 3.7 Bangkok grid.

3.2.3.2 Road Network Data

Road network data is obtained from Toyota Tsusho Electronics. It composes of multiple segments of road network and its neighbor links. This is more precise data compared to Bangkok grid as we know how each road is related to each other. When mapping GPS points into road network, we obtained more information how each road

link change according to space and time.



Figure 3.8 Road network in Bangkok.

3.2.5 Anomaly Detection

We define anomaly as something that alter relationship among attributes on road network. If the relationship among attributes in the time frame and day of week is corresponded to their historical data, we consider this to be normal. However, if the relationship is incorrect, we consider this part of road network to be anomalous on the time period which is needed to be later inferred by social media data source, in our case, tweets from Twitter.

3.2.6 Inferring Root Cause

We inter the cause of the anomaly by checking our historical crawled location-attach tweets from the social media site API. By aggregating term frequency of hashtags according to 3.1 where tf is term frequency, $n_{i,j}$ is number of a hashtag, and $\sum_k n_{k,j}$ is the summation of total hashtag.

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}} \quad (3.1)$$

Chapter 4

Data Exploration on Protesting Period

4.1 Overview

Data exploration of taxi drivers on unusual political situation in Bangkok, Thailand leads to impressive adaptation. There are 7 major junctions closed during protesting period which shown in figure 4.1. With nearly 8,000 GPS tracking on taxi, the gathered data is analyzed by extracting features which are average speed, origin-destination of trips, and number of active cars on the road in Bangkok in areas. We can uncover anomalies lying on known protesting area with described criteria.

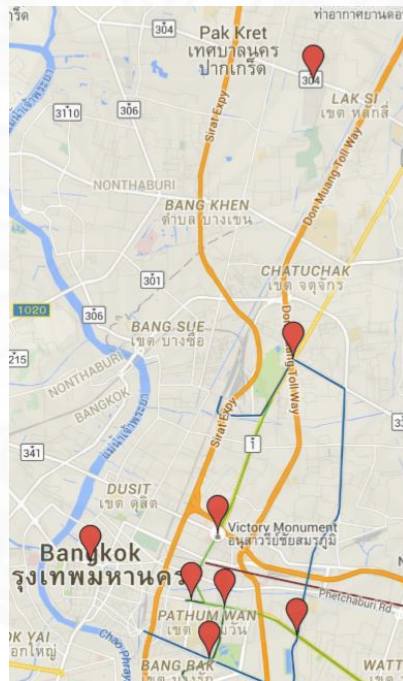


Figure 4.1 Closed intersections.

4.1.1 Dataset

The GPS dataset had collected continuously from 15th December 2015 to 15th January 2014 every 5 seconds interval for every car that was operated resulting in 12 billion raw records with 120GB in size. Gathered attributes are IMEI as identification of each taxi, location as latitudes and longitudes, spotted speed, spotted acceleration, status of taxi meter, and UNIX timestamp. Taxis are operated 24 hours on each day

with two shifts which mean two drivers share the same vehicle. Driving patterns in daytime and nighttime are expected to be different. Also, drivers can cruise for passenger in any area around Bangkok and perimeters, or stop at any preferred spots.

4.2 Limitations

Although 8,000 taxis seems to be a huge number, there are 120,000 taxis registered in the system while 80,000 are active. This dataset is only 10% of the whole system which may lead to decrease in accuracy due to lack of cars. Moreover, some of the drivers operate in the perimeters of Bangkok which have less or no effects from protesting.

4.3 Data Cleaning and Exploration

First, we clean up unwanted data by bounding interested area, Bangkok. Then we divided Bangkok into grids. Each grid has the size of 1 square kilometer resulting in 1392 blocks in total. We mapped 8 points that have protesting area with road blockages. There are 52 blocks are affected by the protest. Then we clean the data further by removing too high speed off the records, then we classify data further by date, hour, and protesting area. Finally, extracted 3 features which are average speed, origin-destination, and numbers of taxis.

4.3.1 Average Speed

Average speed came from individual on hourly manner which can imply overall mobility of Bangkok.

4.3.2 Origin-Destination

Origin-destination describes how taxis travel from places to places on the defined grid area. This facilitates us to look at the flow of cars in Bangkok. We have both origin-destination from taxi with and without passengers.

4.3.3 Number of Taxi on Grid

Number of taxis operates on the given date and hour on the Bangkok grids. It affects the chance of getting passengers. If the number is lower on the grid, chances are higher to get the passengers.

4.4 Result

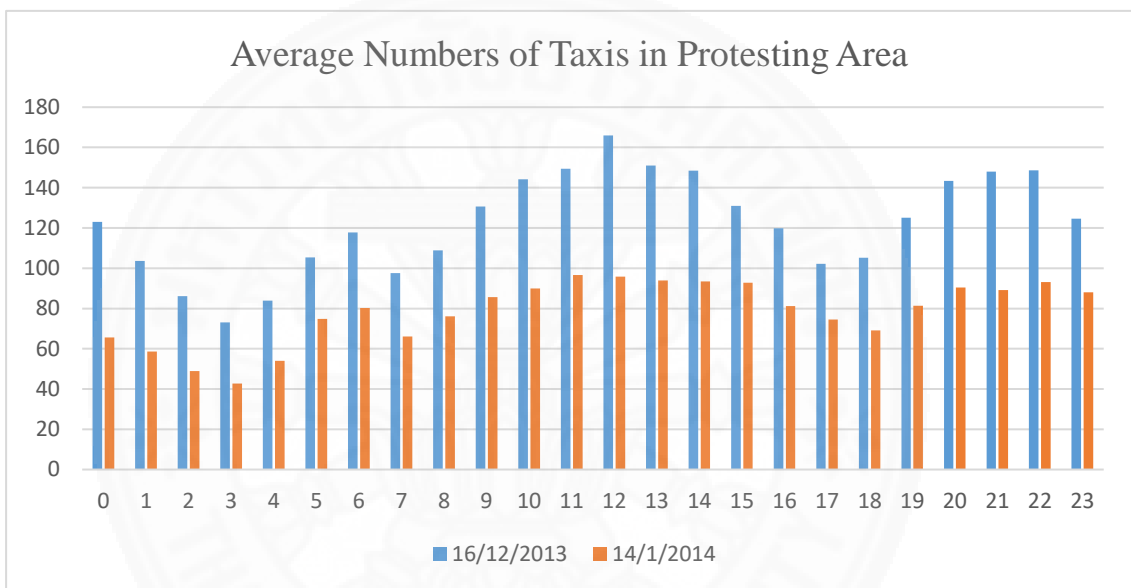


Figure 4.2 Average numbers of taxis in protesting area.

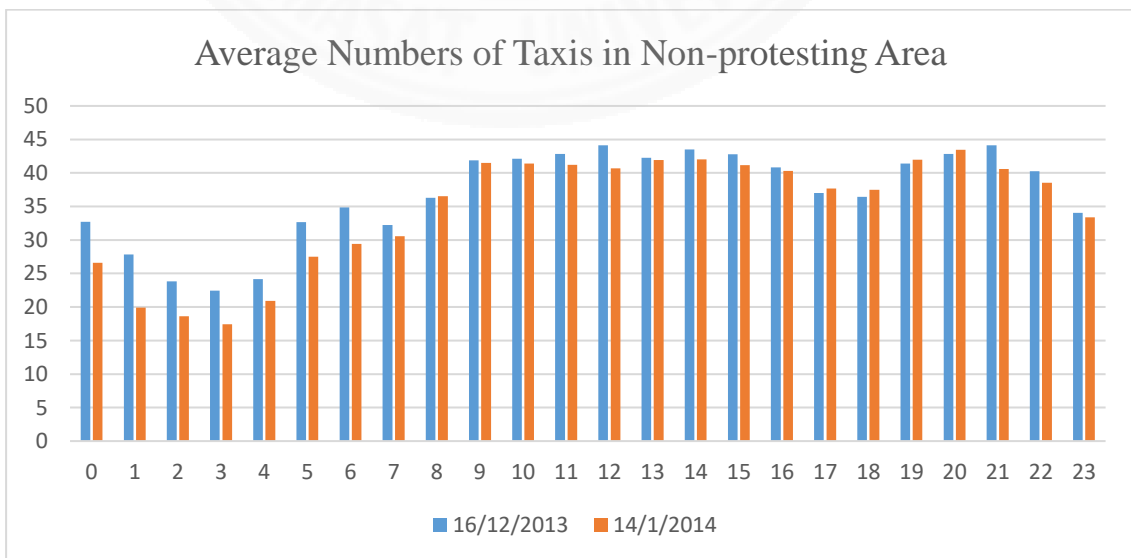


Figure 4.3 Average numbers of taxis in non-protesting area.

Figure 4.2 and 4.3 demonstrate numbers of taxis in protesting and non-protesting area based on the selected dates and the grid that we made as shown in figure 3.4. Each row represents time in 24 hour basis (0-23). The numbers of taxis in protesting area compared on both dates drop significantly on Bangkok Shutdown day.

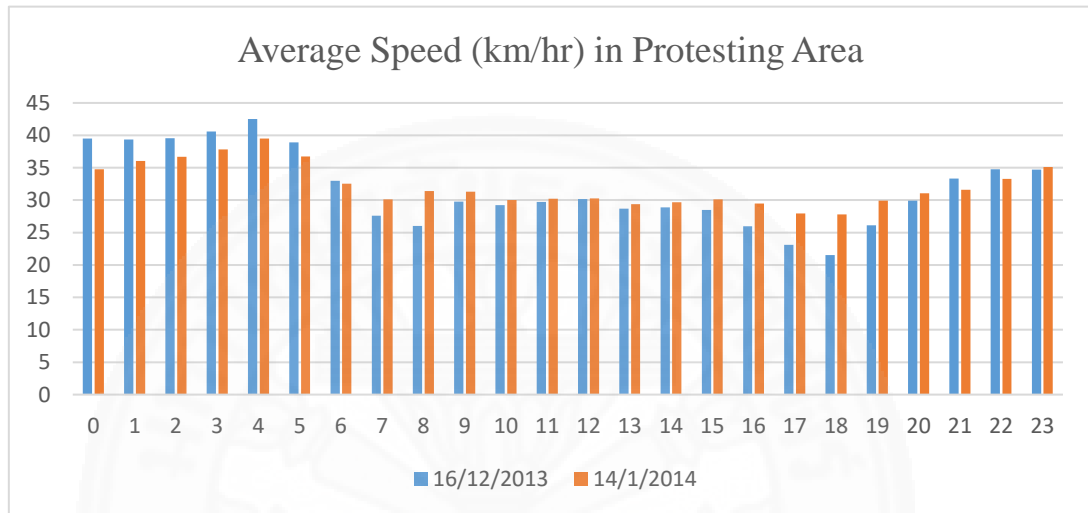


Figure 4.4 Average speed in the protesting area.

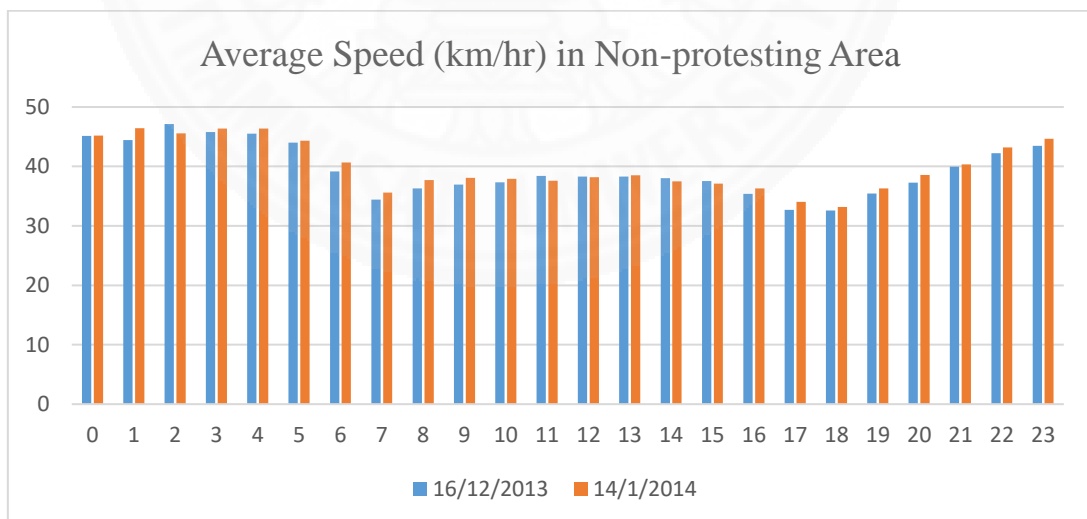


Figure 4.5 Average speed in the non-protesting area.

Figure 4.4 and 4.5 show average speed of taxis in protesting and non-protesting area based on the selected dates and the grid that we made as shown in figure 3.4. Each

row represents time in 24 hour basis (0-23). Both days show similar in average speed in protesting and non-protesting area.

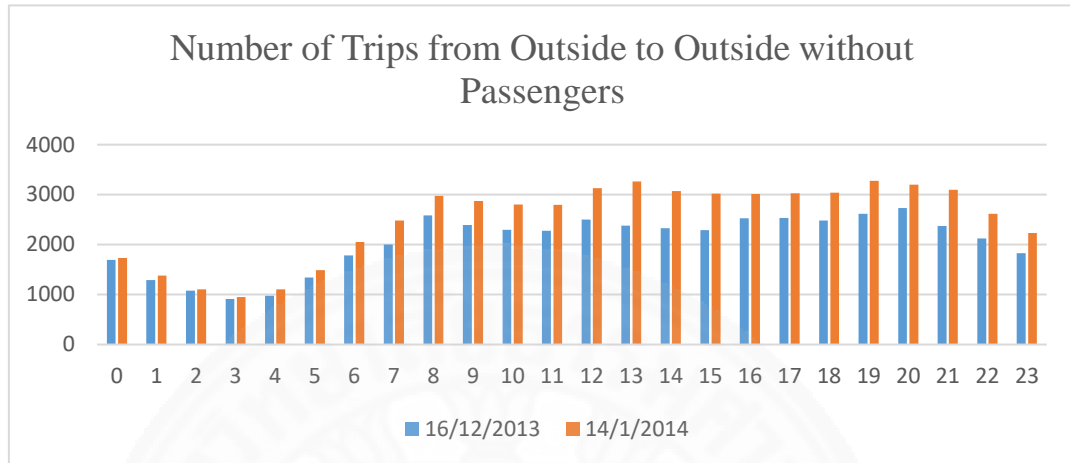


Figure 4.6 Number of trips from outside to outside without passengers.

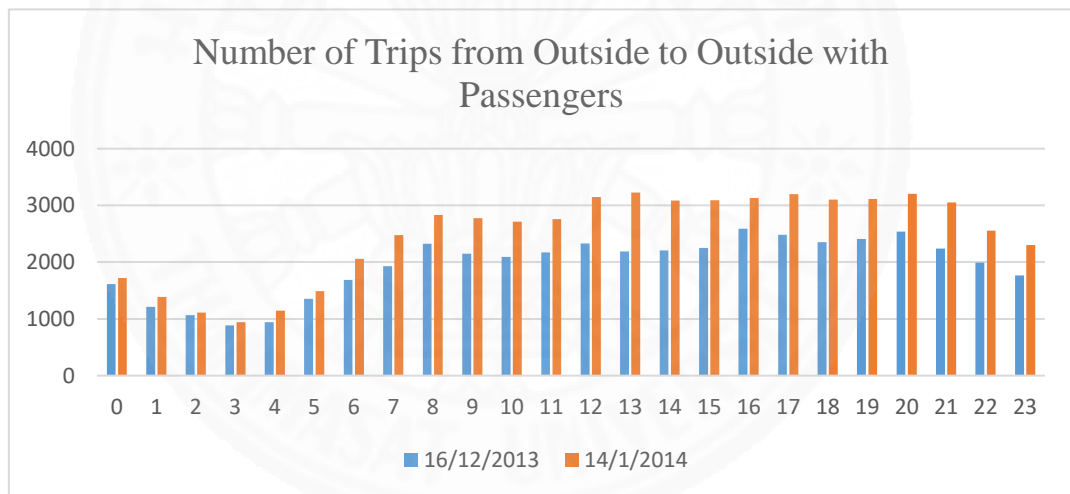


Figure 4.7 Number of trips from outside to outside with passengers.

Figures 4.6 and 4.7 show numbers of taxis travelled from non-protesting to non-protesting area based on the selected dates and the grid that we made as shown in figure 3.4. Each row represents time in 24 hour basis (0-23). We can see that the use of taxis outside protesting area increases significantly on Bangkok Shutdown event.

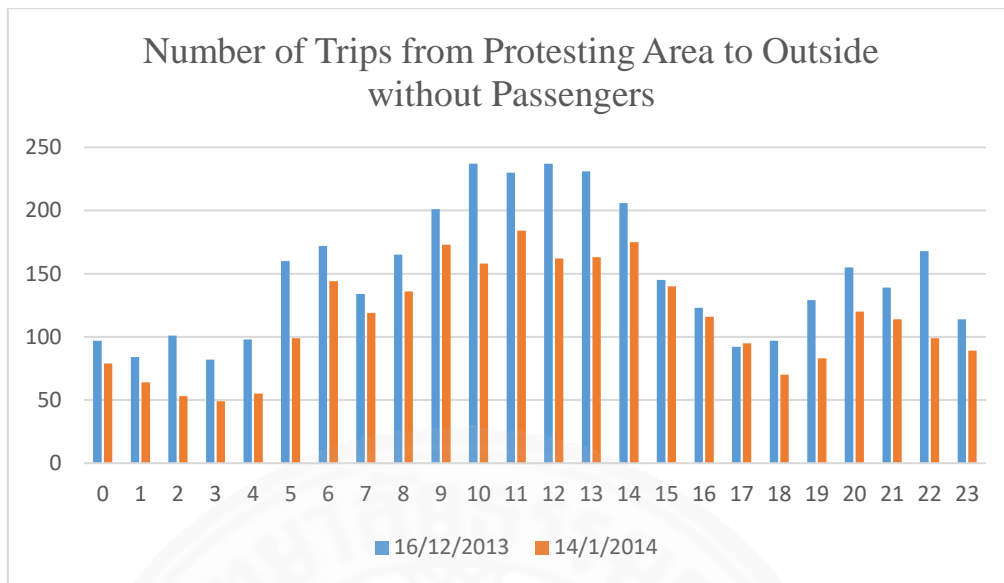


Figure 4.8 Number of trips from protesting area to outside without passengers.

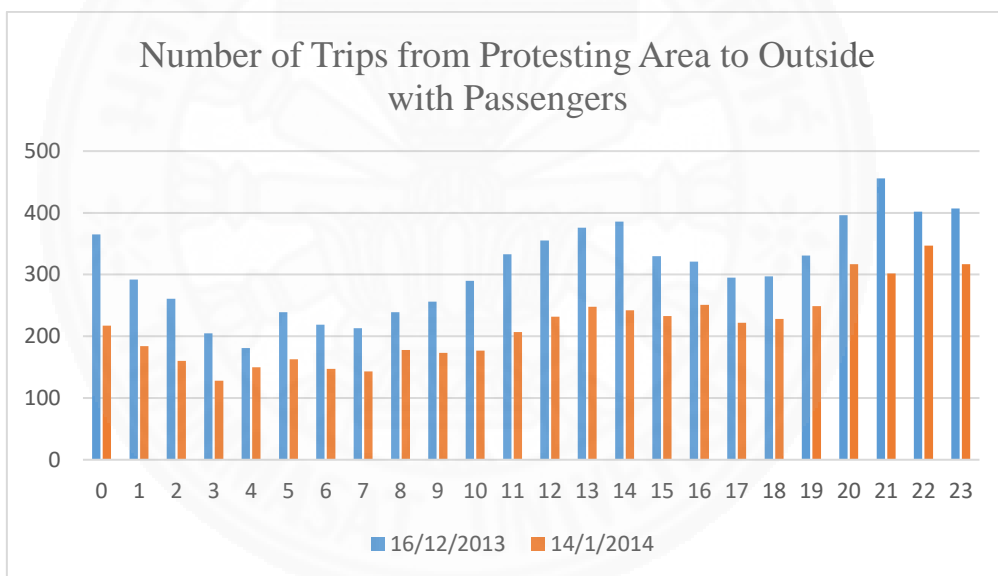


Figure 4.9 Number of trips from protesting area to outside with passengers.

Figures 4.8 and 4.9 show numbers of taxis travelled from protesting to non-protesting area based on the selected dates and the grid that we made. Each row represents time in 24 hour basis (0-23). We can see that the use of taxis from protesting area to outside protesting area decreases significantly on Bangkok Shutdown event.

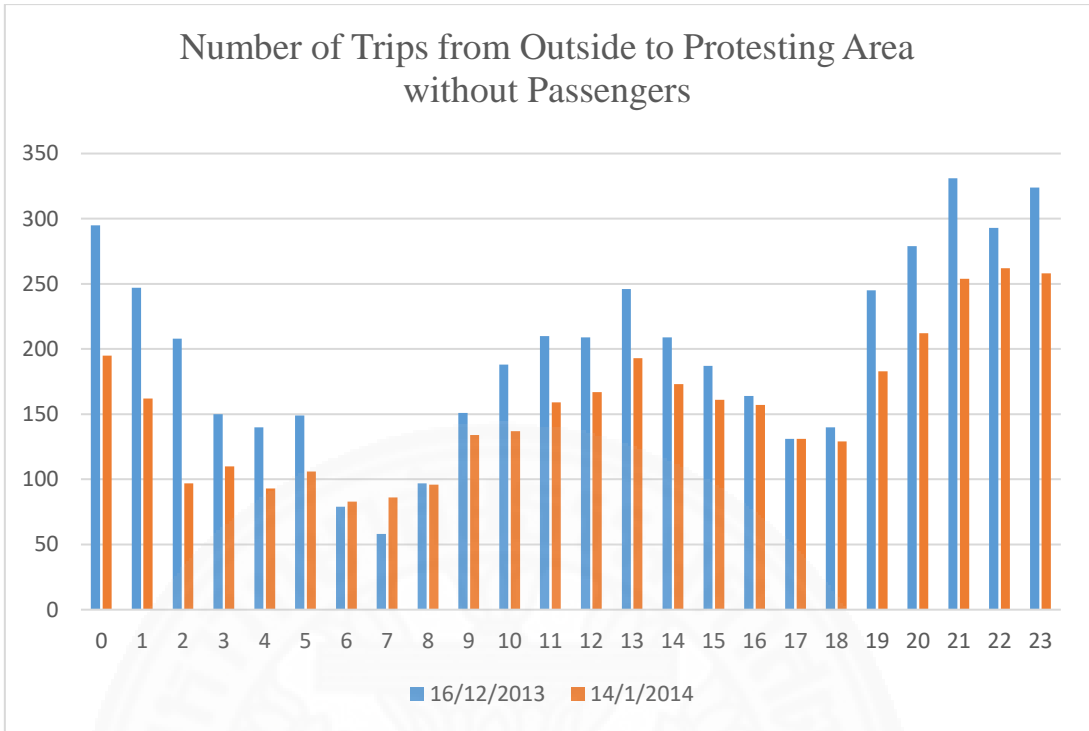


Figure 4.10 Number of trips from outside to protesting area without passengers.

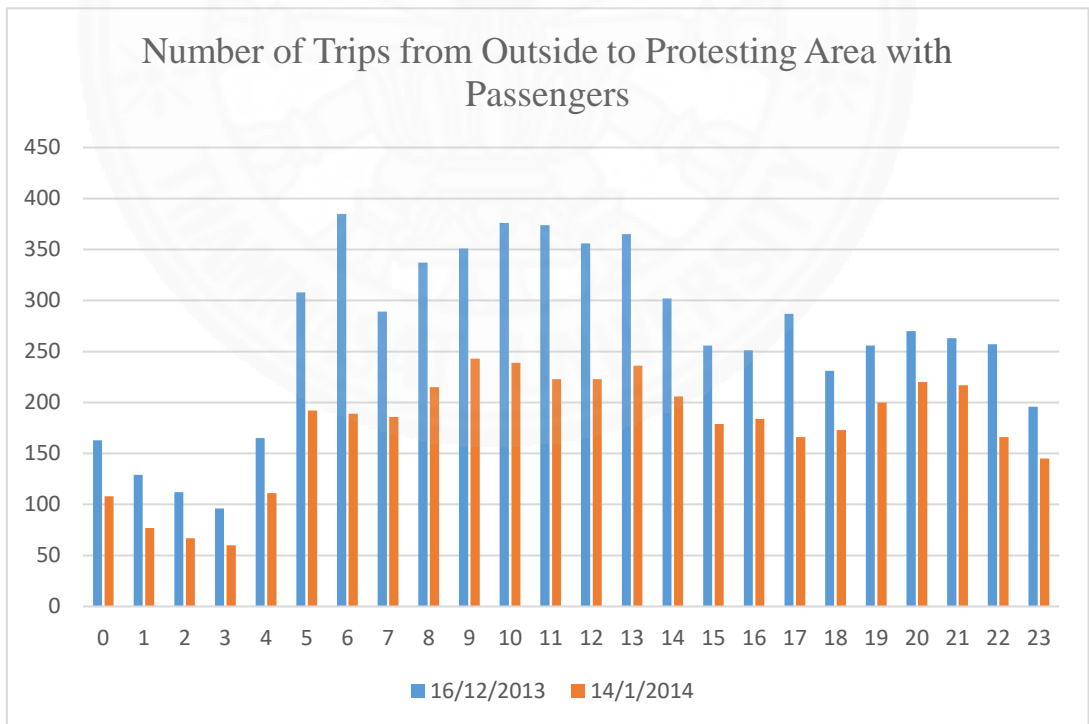


Figure 4.11 Number of trips from outside to protesting area with passengers.

Figures 4.10 and 4.11 show numbers of taxis travelled from non-protesting to protesting area based on the selected dates and the grid that we made as shown in figure 3.4. Each row represents time in 24 hour basis (0-23). We can see that the use of taxis from outside protesting area to protesting area decreases significantly on Bangkok Shutdown event.

4.5 Discussion

From figures above, we have some explanations based on observation and interpretation which are divided into sections.

Firstly, from figure 4.2 and 4.3, average numbers of active cars in protesting area significantly drop compared to regular day on 16th December 2013 declined by 50% as taxi drivers tend to avoid road blocks and bad traffic. However, numbers outside protesting area tends to be the same on both days.

Secondly, as we analyze the graph in the protesting area and non-protesting area on both days, the average speed does have a slightly difference in the protesting area. On 14th January 2014, the average speed in the morning in protesting area has 10 km/hr. more than the other day. This may cause by decline in numbers of taxis on the protesting area. Therefore, the ones in area can drive faster. In non-protesting area, both days show the same trend of incline and decline of average speed. We can summarize that protesting does not significantly affect traffic outside of their 1 km protesting grid. This is because there are alternative ways to commute through protesting area and are not affected by the road blockades such as Bangkok Mass Transit System (BTS) and Metropolitan Rapid Transit (MRT).

Thirdly, drivers are unlikely to drive around protesting area, as we analyze origin-destination graph. We noticed that the numbers of trips they travel from non-protesting area to protesting area decreased. Also, we like to point out that from 7pm to 10pm as the protesting leaders had given speech daily. We could see that the traffic from outside protesting area to inside protesting area from 5pm to 8pm are more than 9pm to 11pm.

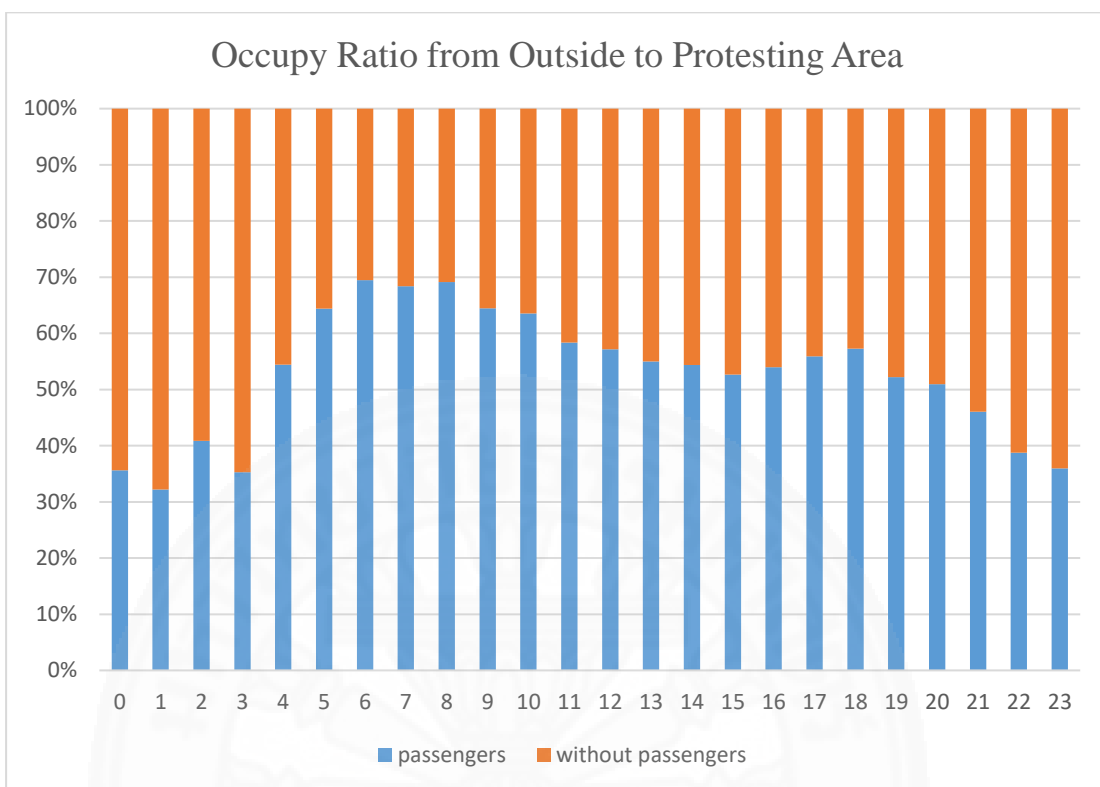


Figure 4.12 Occupy ratio from outside to protesting area

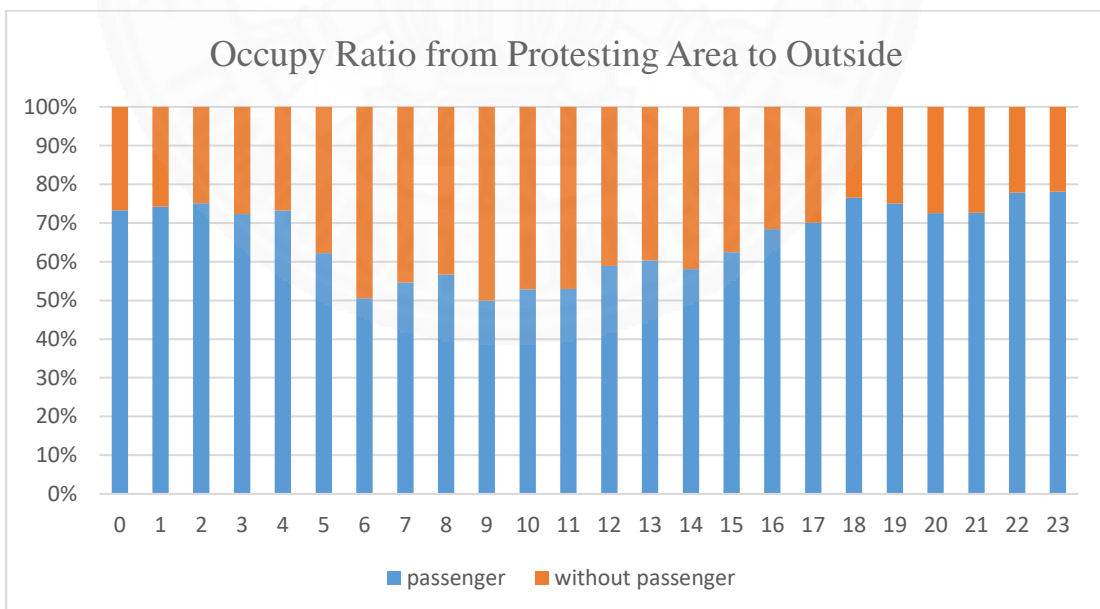


Figure 4.13 Occupy ratio from protesting area to outside

From figure 4.12, in the morning rush hours, taxis driving from outside protesting area to protesting area have around 70% chance to have passengers on board because in the area, there are major business centers and government sectors. Also, it

can be applied to the afternoon rush hours where people commute back from their workplace to their homes because there are around 70% chance that taxis from protesting area to have passengers.

As we uncover the dataset, we could see some taxis preferred to pick and drop passengers from and to protesting area because there are taxi stops arranged by protesters waiting to pick up passengers too.

4.5 Conclusion

In conclusion, we can use two from three extracted features which are origin-destination and number of taxi on grid. The average speed cannot differentiate normal area and protesting area as the traffic is generally bad.

Chapter 5

Anomaly Detection and Inferring

5.1 Overview

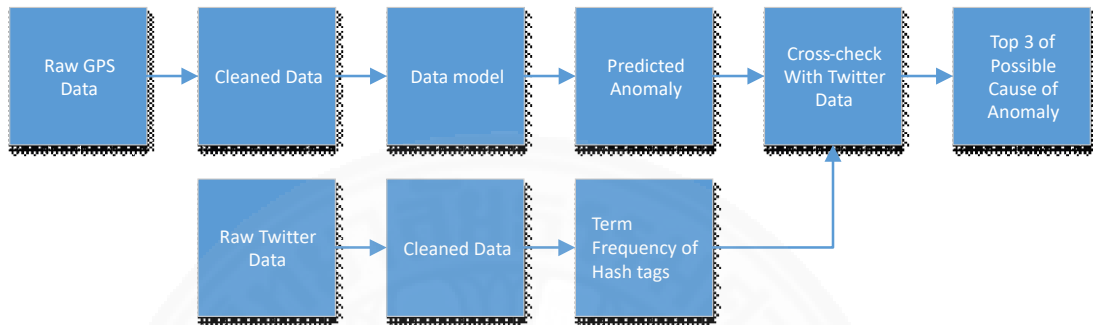


Figure 5.1 Application overview

From figure 5.1, first we clean up unwanted data and then compare with trusted source. Data cleaning is a very important step in our data analysis. Without properly clean up the data, it would lead to inaccurate feature extraction and then lead to inaccurate prediction. There are two dataset which are taxi and Twitter that we need to clean. Also we would like to give some general idea on the testing area.

5.2 Testing Area

Muang Thong Thani has a unique characteristic as it is fairly remoted from city center with a cluster of exhibition center. Only way to travel to the place is by road because there is no other public transportation like BTS and MRT. Also, there are usually events on daily basis, but not every event affects much on road network around the place. Only major events like concerts or famous exhibitions where people gather in large group for certain period of time before the concerts start will cause anomaly on the road network where our algorithm can detect.

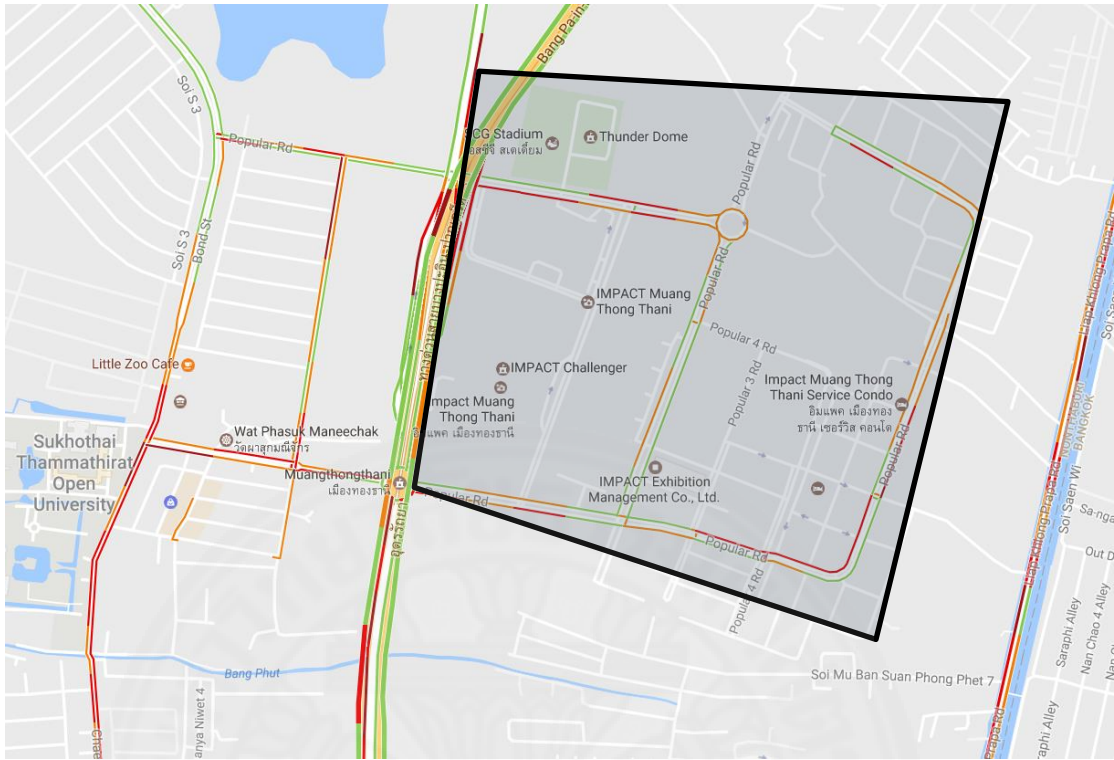


Figure 5.2 Area of Muang Thong Thani on map



Figure 5.3 Overview of Muang Thong exhibition halls and resident area

5.2 Data Cleaning

5.2.1 Taxi Data

The taxi data for anomaly detection is on the period of 5 months from 1st January 2016 to 31st May 2016. This dataset has 60 billion records for 600GB in size. First we clean up unwanted data by looking at each attribute on the records. We get rid of unrelated data with the data source attribute. We select only data source with value of 8 and 9 because the rest of the data are not taxi. Then we filter out the records that have unrealistic speed, for example 200 km/hr.

Next, we find the nearest road network for each point with road network file. These kinds of spatial operations require writing user-defined function (UDF) in Apache Hive because any standard relational database would take weeks or months to accomplish this task. To build quality functions, we follow Java Topology Suite (JTS) standard and library. First, we implement the road network into R-trees, data structures for indexing spatial objects, then we buffer each point for 50 meters radius. We do spatial intersect on the road network and the buffered point, so we yield some possible nearest road links that this point belongs to. Next, we find nearest distance from this point to the intersected road network. As a result, we have a record belong to a road link on the road network. These operations not only clean up unwanted data but also mapping points to road network. Looking at performance of this computing, for 60 billion record of this dataset, we achieve these spatial operations just within one hour.

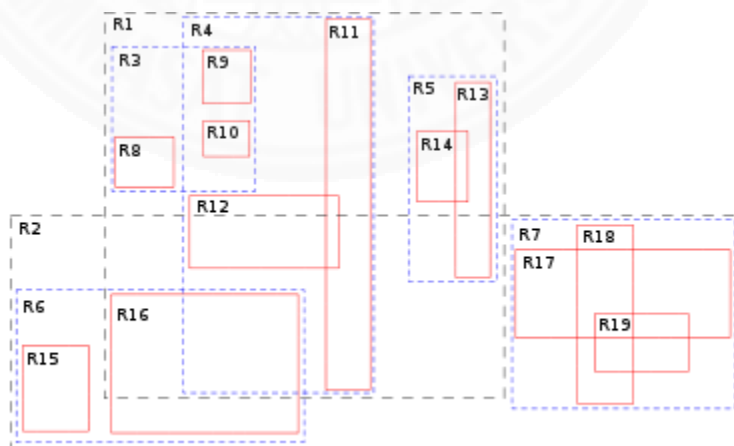


Figure 5.4 R-trees

5.2.2 Twitter Data

As we explored Twitter data from streaming API, we found that the location data have various size of geolocations. They vary from one single point location to few hundreds kilometers of bounding box, which means the exact location can be anywhere inside the bounding box. In this research, we selected locations that are points and locations with bounding box that has less than 100 m² in size. This is because the larger the bounding size it has less meaning to our interested locations. After we clean unwanted data by locations, there are around 40 to 50 key values pairs that we have to deal with. They are mostly user information and how they interact with other people on the website, which in this research it is not relevant to us. What we need to use is texts, location and timestamp.

5.3 Limitation

For taxi data, most of the drivers operate closer to Bangkok while our testing location at Muang Thong Thani is fairly remote from the city center. Lesser taxis operate around this area which lead to lesser data to make a prediction model. For Twitter data, even though we have collected 300MB to 500MB per day, it is a small data set when we consider records per area. Also, the quality of fine location data is even lesser when we cleaned data.

5.4 Feature Extraction

5.4.1 Taxi Data

For taxi data, we can extract more features on each record from date. We can differentiate date into the following types.

- Weekday
- Weekend
- Holiday
- Events on weekday but not holiday - Valentine's day

Next, we aggregated each record into road link attribute, so we gain more features. The new obtained features when we aggregated data for 15 minutes interval are variation of the speed which we statistically separated to be the following attributes.

- First quartile
- Average or second quartile
- Third quartile
- Number of GPS location on the road link
- Numbers of taxi on particular road network

These features will be used for data modeling and later, for anomaly detection.

Table 5.1 Extracted features

Entity	Value	Description
linkid	20944	Unique road link
speed	30.24	Average speed
no_point	10	Number of GPS points
no_car	3	Number of taxis
first quartile	10.1	First quartile of speed
third quartile	25.2	Third quartile of speed
date	25-01-16	Date
time frame	30	4-100
day of week	2	"1 - 14"

5.4.2 Twitter Data

In this research, we filter out most of the data for simplicity as doing natural language processing in Thai is difficult especially word segmentation for informal Thai

in social media sites. Moreover, many tweets come in multi-language. Sometimes, it is a mixture of Thai and English, or Thai with Korean. To simplify the process, we extract text and hashtag (#) separately from each Twitter record and store them with original date time and location. As for the locations, we simplified bounding boxes by using only centroids. As a result, we have set of records as follow.

- Time frame
- Text and Hashtag
- Cumulative latitude
- Cumulative longitude
- Bounding box size

The image shows a code editor with two panels. The left panel displays a full JSON record for a tweet, including fields like 'id', 'created_at', 'geo', 'user', and 'text'. The right panel shows the extracted features from this record, such as 'tf', 'text', 'clat', 'clon', and 'boundingsize', along with the original text and location information in Thai and English.

Figure 5.5 Left is one record, right is extracted records

5.5 Data Modeling

As we extracted features from taxi records, now it is time to make use of them by making a model from the data. The algorithm we used to predict anomaly is unsupervised random forest. To make unsupervised learning from supervised algorithm, we classify by label data into two types which are real and unreal. We make unreal data by using real data and we swap values within each column, so values of data on each record are still correct but relationship among records are broken. For example, we take one real record and swap values. The first quartile might has value higher than the average which makes the data unusual. Then we combine both data into one dataset and make a prediction model from them. We use only 3 months data from

1st January 2016 to 31st March 2016 because there are too many holidays on April, so it affects model accuracy.

Table 5.2 Extracted attributes

Attribute	Example
Linkid	20944
speed	30.24
no_point	10
no_car	3
first quartile	10.1
third quartile	25.2
time frame	30
day of week	2

The features that we use for prediction model are average speed, first quartile of speed, third quartile of speed, numbers of cars, numbers of GPS points, day of week, and last but not least data label. Random forest classifier that we use has Gini impurity which means if any randomly picked features are mislabeled from, Gini impurity will have higher value. As Apache Spark offer pipelines, we can select multiple depth of our trees, so we put the range from 2 to 7 levels. Also, Spark offers how many folds we prefer to use for cross validation, in our research we use 3 folds. The pipeline will select the best model that has least error rate for us, so we have the one with max depth of 7.

5.6 Anomaly Events

We have the testing area which is located in the perimeters of Bangkok. Muang Thong Thani is a place in the perimeters with various size of convention centers. Generally, they will have events almost every day according to its website. We pick up some Thai and international concerts that held in one of the convention centers in total of 6 events on 6 separate days as a reference because we assume that these kinds of events have impact on the road link around the area. Therefore, we bound the 3km of area around the convention centers as a testing area for this data model.

As we created an unsupervised random forest classifier, now we use them to predict anomaly events. We declared anomaly events by the label of the given features. 6 days of real data inside 3km bounding will be passed into the prediction model. If any record is predicted as unreal, we keep it into arrays to verify it with our filtered Twitter data. We have some known events from what we known above. As a result, we have 20 alerted.

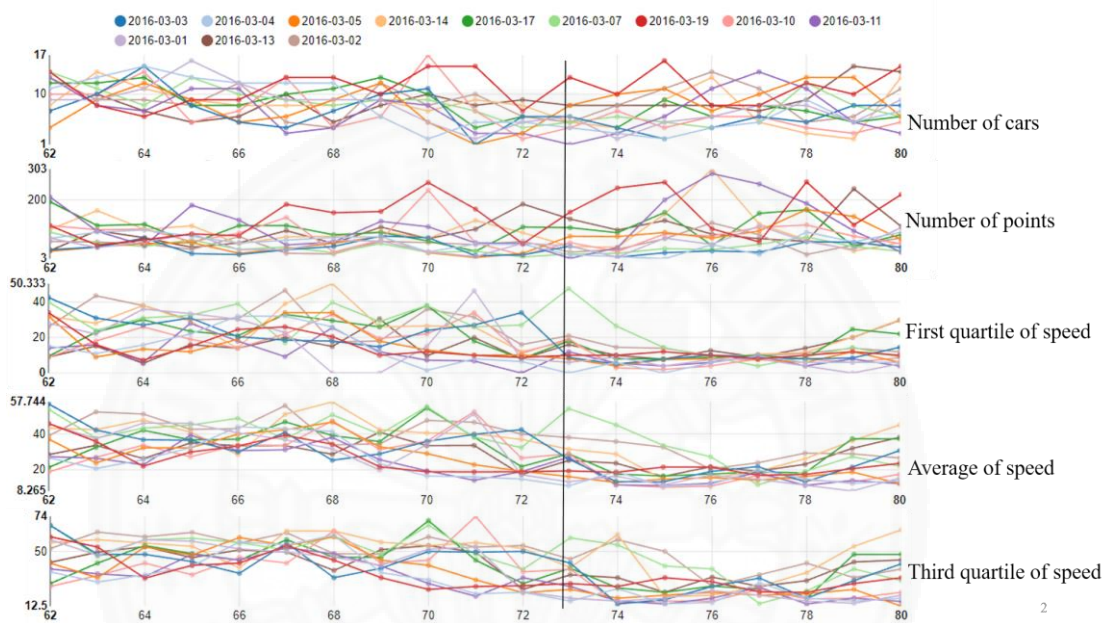


Figure 5.6 Example of anomaly on 2016-03-19 at $tf=73$

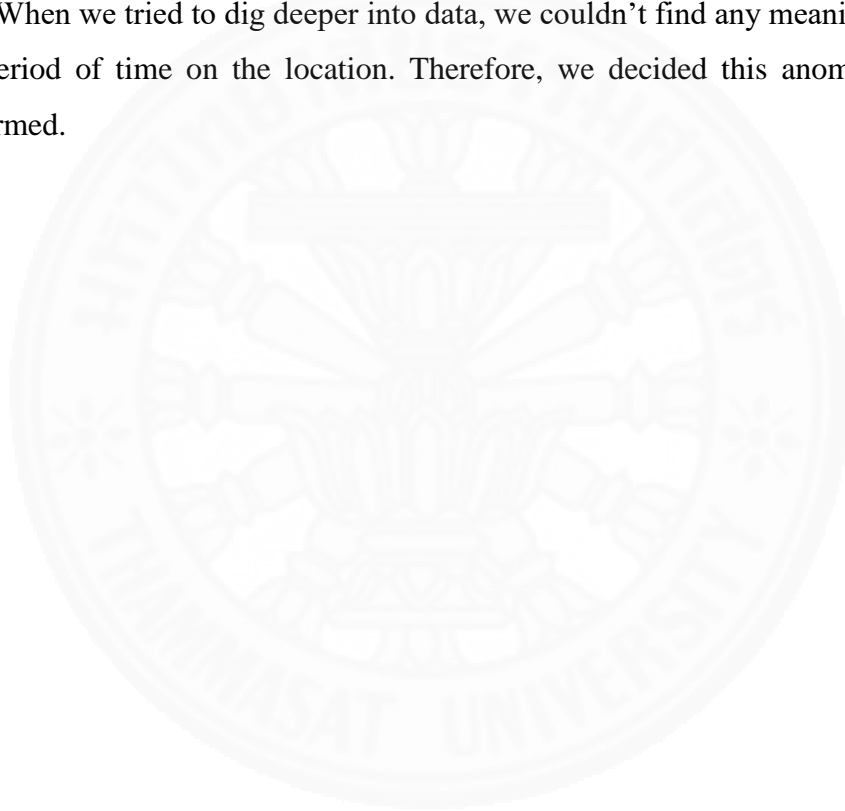
5.7 Verification

We verified the anomaly by cross validation from cleaned Twitter data. We found that most of the time when anomaly occurs, the Twitter data also has some activity in it. With 20 anomaly event alerted, we have checked with Twitter hashtag (#) with occurrence more than 3 times in the same interval and found at least 3 potential cause of the anomaly around Muang Thong Thani.

16 out of 20 anomalies can be confirm immediately by the hashtag frequency with has magnitude more than 5. While 1 of 3 is alerted but we couldn't find the solid frequency of the hashtags, we decided to dig deeper into our cleaned Twitter data. The data show singer and concert name during the anomaly is alerted, but it has no hashtags in those tweets which is the reason why we couldn't infer the cause with anomaly. However, we can still infer the root cause of anomaly events.

Another anomaly cannot be confirmed because when the anomaly occurred, we have no data on our kept Twitter records. This is because our Twitter crawler were down on the period. However, we still can manage to confirm the event by looking at the hashtag of the Tweets. We found multiple of related Tweets on the social site on the period and at the exact venue they were live. As a result, we could infer one of the unknown events.

One last anomaly that we cannot confirm is occur around 2 in the morning. When we tried to infer the cause, it shows only not meaningful hash tags with frequency of 1. When we tried to dig deeper into data, we couldn't find any meaningful words in the period of time on the location. Therefore, we decided this anomaly cannot be confirmed.



Chapter 6

Discussions and Conclusions

6.1 Anomaly Detection on Road Network

Our model can detect anomalies in the same way as network intrusion detection systems (NIDS).

Our anomaly detection can detect more anomalies than only concerts because when we built the model, we only give 2 labels on the algorithm which is real and unreal. With more data sources to confirm the anomalies, we can detect more than concerts around the area. Data sources can be news or other social media sites where location and time are with user data, for example, Instagram and Facebook.

To continue using unsupervised random forest classifier, more features can be inserted into the model to increase accuracy.

GPS points on road network around perimeters are a lot lesser than urban area. Combining other kinds of spatial-temporal data, for example cell detail record (CDR) to gain more information would lead to more accurate prediction.

6.2 Problem with Hashtag and Informal Thai

Twitter hashtags, at a certain level, can gain some information on particular events. However, some events cannot be determined with hashtags because users do not use it to identify. Therefore, implementing NLP is preferred over hashtags as we can have more insights from the text in case hashtags cannot determine. Informal Thai is also troublesome in word segmentation. Some of the informal words are meant to be the same as formal words, but they are composed of different or repetitive alphabets that cause errors in word segmentation. Making dictionary of informal Thai separately from formal Thai would increase accuracy of word segmentation, however this is a hard task to labeling each word and it requires huge dataset to make it really accurate. Also, as Thai is an alive language, the informal words are changing overtime. Following new trend of words requires a lot of resources from both human and dataset.

6.3 Social Media User Target

Twitter users in Thailand is fairly limited to certain groups of users. While comparing to Facebook has wide range of generations, Twitter, in contrast, has a majority of young people who follow international concerts. Therefore, it is difficult to use only Twitter to infer cause of anomaly.

6.4 Improvements

There are plenty of room for improvements on this research. First, this research is done with historical GPS and Twitter data in batch processing. Making anomaly detection in near real time will be a huge advancement on this research. As J. Raiyn did the real time anomaly detection, it is does not get accurate result from moving average algorithm [8]. Furthermore, implementing other algorithms for anomaly detection is recommended because there are plenty of room for newer method to identify anomalies, for example, deep learning algorithms. Implement this anomaly on different place is also an improvement.

Other crowdsourcing data other than one social media website would be more accurate. Also, using data from social media sites where locals prefer to use can gain a lot of data which increase chance to gain more information as well. As for Twitter in Thailand, most of the locals are not using the website much compare to other countries, for example Japan. While Thais generate data with location around 400-500MB per day, Japanese generates around 3GB to 4GB a day which is around 1000 times more than Thais.

Also, implement word segmentation on Thai tweets is also recommended, however the tweets have to be cleaned first by removing symbols and repetitive letters. Preposition words are meant to be excluded from the tweets too.

For anomaly to be alerted, we could change this process by making social media data alerts first when frequency of tweets rises first, then we check anomaly with road network.

REFERENCES

- [1] L. Vinet and A. Zhedanov, *RANDOM FORESTS*, vol. 58, no. 12. Cambridge: Cambridge University Press, 2010.
- [2] L. I. Smith, "A tutorial on Principal Components Analysis Introduction," *Statistics (Ber)*, vol. 51, p. 52, 2002.
- [3] L. Liu, C. Andris, and C. Ratti, "Uncovering cabdrivers' behavior patterns from their digital traces," *Comput. Environ. Urban Syst.*, vol. 34, no. 6, pp. 541–548, 2010.
- [4] T. Horanont and A. Witayangkurn, "Extracting Descriptive Life Profiles from Mobile GPS Data ‡ Cennter for Spatial Information Science , the University of Tokyo."
- [5] J. Z. J. Zhang and M. Z. M. Zulkernine, "Anomaly Based Network Intrusion Detection with Unsupervised Outlier Detection," *2006 IEEE Int. Conf. Commun.*, vol. 5, pp. 2388–2393, 2006.
- [6] V. Nikulin, "Driving style identification with unsupervised learning," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2016, vol. 9729, pp. 155–169.
- [7] Y. L. Hsueh, W. L. Lai, C. C. Lin, and P. P. Lindenberg, "Traffic anomalous region detection model," in *Proceedings - 2016 5th IIAI International Congress on Advanced Applied Informatics, IIAI-AAI 2016*, 2016, pp. 647–650.
- [8] J. Raiyn and T. Toledo, "Real-Time Road Traffic Anomaly Detection," *J. Transp. Technol.*, no. July, pp. 256–266, 2014.
- [9] Y. Shavitt and N. Zilberman, "A geolocation databases study," *IEEE J. Sel. Areas Commun.*, vol. 29, no. 10, pp. 2044–2056, 2011.
- [10] A. Gr, M. Weber, M. Guggisberg, and H. Burkhart, "Traffic Flow Measurement of a Public Transport System through automated Web Observation," 2017.
- [11] B. Pan, Y. Zheng, D. Wilkie, and C. Shahabi, "Crowd Sensing of Traffic Anomalies Based on Human Mobility and Social Media," *Proc. 21st ACM SIGSPATIAL Int. Conf. Adv. Geogr. Inf. Syst.*, pp. 344–353, 2013.
- [12] T. Komatsu and R. Kondo, "DETECTION OF ANOMALY ACOUSTIC SCENES," pp. 376–380, 2017.
- [13] X. Meng, S. Zhao, H. Mo, and J. Li, "Application of Anomaly Detection for Detecting Anomalous Records of Terroris Attacks," *2nd IEEE Int. Conf. Cloud Comput. Big Data Anal. Appl.*, pp. 70–75, 2017.

- [14] C. Zar, "MARITIME ANOMALY DETECTION IN FERRY TRACKS," *IEEE Int. Conf. Acoust. Speech Signal Process.*, pp. 2647–2651, 2017.
- [15] R. S. Fanhas, "Discovering Frequent Origin-Destination Flow from Taxi GPS Data," 2016.
- [16] J. Zhang, Q. Liu, C. Yuan, H. Shi, and L. Cui, "EasiTMC : Transportation Mode Classification With A High Accuracy Trajectory Detection Method," 2016.
- [17] Z. Liao and B. Chen, "Anomaly Detection in GPS Data Based on Visual Analytics," pp. 51–58, 2010.
- [18] Z. Zhang, A. Tong, L. Zhu, M. Chen, and P. Su, "An Anonymous Scheme for Current Taxi Applications," in *2016 IEEE 14th Intl Conf on Dependable, Autonomic and Secure Computing, 14th Intl Conf on Pervasive Intelligence and Computing, 2nd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress*, 2016, pp. 168–172.
- [19] M. Douriez, H. Doraiswamy, and J. Freire, "Anonymizing NYC Taxi Data : Does It Matter ?," in *2016 IEEE International Conference on Data Science and Advanced Analytics Anonymizing*, 2016, pp. 140–148.
- [20] J. A. Deri, F. Franchetti, and M. F. Moura, "Big Data Computation of Taxi Movement in New York City," in *2016 IEEE International Conference on Big Data (Big Data) Big*, 2016, pp. 2616–2625.
- [21] W. Yong-dong, X. Dong-wei, H. De-feng, G. Hai-feng, and Z. Gui-jun, "The design of the operation monitoring and statistics analysis system for taxi based on the GPS information," pp. 466–469, 2017.
- [22] G. Dai, J. Huang, S. Wambura, and H. Sun, "A Balanced Assignment Mechanism for Online Taxi Recommendation," in *2017 IEEE 18th International Conference on Mobile Data Management*, 2017, pp. 102–111.
- [23] J. Kim and P. Montague, "An Efficient Semi-Supervised SVM for Anomaly Detection," pp. 2843–2850, 2017.
- [24] Z. Hasani, "Robust Anomaly Detection Algorithms for Real-time Big Data," in *2017 6th MEDITERRANEAN CONFERENCE ON EMBEDDED COMPUTING (MECO)*, 2017, no. June, pp. 11–15.
- [25] L. Yin, J. Hu, L. Huang, F. Zhang, and P. Ren, "Detecting Illegal Pickups of Intercity Buses from Their GPS Traces *," *2014 IEEE 17th Int. Conf. Intell. Transp. Syst.*, pp. 2162–2167, 2014.

- [26] J. La-inchua, S. Chivapreecha, and S. Thajchayapong, "A New System for Traffic Incident Detection Using Fuzzy Logic and Majority Voting," no. 1, pp. 0–4, 2013.
- [27] Z. Ning, F. Xia, N. Ullah, X. Kong, and X. Hu, "Vehicular Social Networks: Enabling Smart Mobility," in *IEEE Communications Magazine*, 2017, vol. 55, no. 5, pp. 16–55.
- [28] S. Thajchayapong, E. S. Garcia-Trevino, and J. A. Barria, "Distributed Classification of Traffic Anomalies Using Microscopic Traffic Variables," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 1, pp. 448–458, Mar. 2013.
- [29] X. Xing, X. Zhou, H. Hong, W. Huang, K. Bian, and K. Xie, "Traffic Flow Decomposition and Prediction Based on Robust Principal Component Analysis," in *Proc. 2015 IEEE 18th International Conference on Intelligent Transportation Systems*, 2015, pp. 2219–2224.
- [30] X. Wang and X. Zhao, "The Detection Algorithm of Anomalous Traffic Congestion Based on Massive Historical Data."
- [31] W. Kuang, S. An, and H. Jiang, "Detecting Traffic Anomalies in Urban Areas Using Taxi GPS Data," *Math. Probl. Eng.*, vol. 2015, 2015.
- [32] W. Zhang, G. Qi, G. Pan, H. Lu, S. Li, and Z. Wu, "City-Scale Social Event Detection and Evaluation with Taxi Traces," *ACM Trans. Intell. Syst. Technol. - Surv. Pap. Regul. Pap. Spec. Sect. Particip. Sens. Crowd Intell.*, vol. 6, no. 3, pp. 1–20, 2015.
- [33] S. Chawla, Y. Zheng, and J. Hu, "Inferring the root cause in road traffic anomalies," *Proc. - IEEE Int. Conf. Data Mining, ICDM*, pp. 141–150, 2012.
- [34] G. Pan, G. Qi, Z. Wu, D. Zhang, and S. Li, "Land-Use Classification Using Taxi GPS Traces," *Intell. Transp. Syst. IEEE Trans.*, vol. 14, no. 1, pp. 113–123, 2013.
- [35] S. Qian, Y. Zhu, and M. Li, "Smart recommendation by mining large-scale GPS traces," *IEEE Wirel. Commun. Netw. Conf. WCNC*, no. June 2015, pp. 3267–3272, 2012.
- [36] D. . ZHANG ET AL., "UNDERSTANDING TAXI SERVICE STRATEGIES FROM TAXI GPS TRACES," *IEEE TRANS. INTELL. TRANSP. SYST.*, VOL. 16, NO. 1, PP. 123–135, 2015.