東京大学大学院新領域創成科学研究科

社会文化環境学専攻

2018 年度

修　士　論　文

Location Estimation of Event Based on Twitter

Twitter によるイベントの位置推定

2019 年 1 月 21 日提出

指導教員　瀬崎　薫　教授

楊　　珂為

Yang, Kewei

Abstract

As we known the popular micro-blogging tool Twitter has been used in multiple fields due to the high value of its data. This Master's Thesis presents a study of using twitter data to estimate the event location. To resolve the problem that it is difficult to get a location in Twitter directly now, I studied on the researches of using specific word as the tweeting search condition, and used the "event", in this research it means the event like earthquake or fire disaster which has geographical locality, as the specific word in the data crawling. The method of estimating location is based on the common thought of almost tweets associated with event would be tweeted by the people who are near to the event place, if we can analyze the distribution of the geographical words in these tweets, we can get an event location. In the experiment I acquired some tweets about the earthquakes that happened on June at Japan and used the MeCab to extract the geographic words from these tweets to estimate the event location. After observing the collected data, I used a machines learning-based analysis – K-means clustering to obtain more precise location of earthquake. After experiment, I gathered the estimate locations and compared them with the actual locations to verify if this method is available. At last I did a discussion about the factors could influence the results and described the future work.

Contents

Figures

Tables

Chapter 1 Introduction

1.1 Motivation

In recent years there has seen a rapid growth in micro-blogging such as Twitter, it has been an important tool for data mining because Twitter could get amount of real-time information. As of 2016, Twitter had more than 319 million monthly active users. [1] On the day of the 2016 U.S. presidential election, Twitter proved to be the largest source of breaking news, with 40 million election-related tweets sent by 10 p.m. (Eastern Time) that day. [2] Twitter information could be used in sentiment analysis, early warning system to the natural disasters such as earthquake and the crime prediction etc. In the twitter analysis, location data is useful in these systems developing and is benefit in the extended research. For example, in an earthquake, the rescue could get the SOS message from Twitter and they could arrive the accident scene quickly. However, it is difficult to get the user's location directly, almost users choose to shut off the Geotag function for protecting their privacy. To the researchers it has been a challenge to speculate users' location from Tweets.

For resolving this problem, researchers began to find the method of estimating user's location. In the survey of user location estimation researches I found some of researchers tended to collect the users information by search a unique word and use it to speculate users location. This idea inspired me that it is may be possible to select a special keyword in the location estimation of event.

This research will focus on analyzing event location based on tweets' contents. The main question of this thesis is: "How can I do an event location estimation by using Twitter data without Geotag". The purpose of this paper is to serve as the basis for user's location speculation. And I will introduce the method I found in this paper in chapter 3.

1.2 Overview

Following chapters describe respective parts of the thesis:

Chapter 1 provides the thesis's motivation and overview.

Chapter 2 introduces the background of Twitter and the related work of predicting event location and user's location by Twitter.

Chapter 3 describes the method of event location estimation in this research.

Chapter 4 mainly introduces the tools used in data acquiring and clustering analysis.

Chapter 5 shows the data acquired by the Twitter API and achieve location estimation by doing a clustering analysis.

Chapter 6 compares the result with previous researches and describes the future work.

Chapter 7 provides final conclusion over the thesis.

Chapter 2 Background

This chapter introduces the background of Twitter and discusses the reason of difficult to get the location from Twitter directly. Then I summarize the related work of location estimation based on Twitter for solving the problem of lack of Geotag.

2.1 Twitter and location

Twitter is an online micro-blogging and social-networking platform, which allows users to write short status, updates of maximum length 140 characters known as “tweets”. Users can tweet by the Twitter website and applications at almost countries. A tweet can be forwarded by other users to their own feed and this action calls “retweet”. Users can group posts together by topic or type by use of hash tags – words or phrases prefixed with a “#” sign. Similarly, the “@” sign followed by a username is used for mentioning or replying to other users [3]. Those signs usually influence the accuracy of result.

With the rapid increase of GPS-equipped smartphones, users were concerned about the problem of leaking personal information because they can post tweets with location information easily. Some extended applications could extract the specific tweets such as the tweets include “come back now”, and if the number of similar tweets is large, everyday life habits and behavior styles could be light easily. It is concerned about the possibility of getting involved in crimes.

Fig. 2-1 The privacy and security page of twitter web site

Based on above problems, if the information is acquired in the large cities, the official applications would

change the location information to the roughly place. It is possible to post the location with GPS coordinates by setting “add the position information” for each tweet. Also, it is possible to remove the position information from the selection of “posted tweets with position information” in the settings page. Above these mentioned information, it is difficult to get user’s position directly now.

This limitation has affected many researches of Twitter, one of them is the event location estimation. Geotag usually be an important analysis factor in researching and verification, the lack of Geotag means we should find other factor to instead it. At the same time, researchers still explore the approaches of location estimation. The huge real-time data could be used in many areas and contains a wealth of research value.

2.2 Related works

(1) The researches about detecting local event

There have many researches about using the information posted on social media to detect events in the real world.

In Rattenburys’[4] research they proposed a method to discover temporal or geographical burst by using photos with location tag posted on Flickr. The method of [4] focus on analyzing the temporal and geographical distribution of these tags (New York, World Cup, dog, etc.) and assumed the tags with signify bias as the events.

Lappas [5] proposed a method to detect the word with temporally or geographically burst in the case of data following from multiple streams what have geographical distribution.

Walther [6] constructed a system to detect events on the geospatial space. They discussed what kind of feature can be useful in accurately detecting events, and reported that it can obtain good results by analyzing the number of users and topics of the post at a certain place. At the same time this research pointed that if people tweeting from the same place use the same words, it is likely that they talk about the same thing, which probably is some noteworthy event.

Lee[7] used the DBSCAN to cluster tweets and estimate the event location by analyze the time zone of tweets the users posted.

In Watanabe’s [13] research, they proposed a method to detect events in a small geographical grain size by using the tweets with GPS position information tags. But this approach is not applicable to the current Twitter environment.

Be different form other researches, the research [21] was no need to select keywords nor use clustering algorithms for geographic location grouping. This work discovered events based on location over the

Twitter stream, using time series analysis. This research was helpfully with the detection of events with local relevance.

The researches [29][30] also provided the approaches of detecting events.

(2) The researches about estimating user's location

According to [8] there are two main approaches to predict users' locations, one is graph-based method and the other is contents-based method. The graph-based method uses closeness assumption, which is based on the assumption of users (friends, etc.) contacted on the social graph would be close to each other. For example Clodovue [9] gathered information of users who are following each other and set the location where the most talked in their tweets as the residence of the user. And in the contents-based method, it estimates the user's place using the words contained in the content submitted by users.

Most studies are based on the research [17], it used a probabilistic framework to estimate city-level location based on the contents of tweets without considering other geospatial clues. Their approach achieved the accuracy on estimating user locations within 100 miles of error margin (at best) varying from 0.101 (baseline) to 0.498 (with local word filtering).

(3) The value of keyword in location estimation

In the past research [10] demonstrated that users in a particular location tend to query some search keyword more often than users in other locations, especially for some topic words such as sport teams, city names, or newspaper. The Fig.2-2 shows this situation.

Fig. 2-2 The relationship between Keywords and users

As [6] discussed, choosing the right keywords could improve the accuracy of the estimation. And in the

event location estimation is also in choosing the species of event. I would introduce the event selection in the next chapter and show the method of acquiring Twitter.

Chapter 3 Method of location estimation

This chapter mainly discusses the method of location estimation of event and the meaning of it.

3.1 The assume of solving non-Geotag problem

As the survey in chapter 2.2, Geotag is useful in the location estimation because it have the most directly geographic location. But most users shut off the Geotag function of Twitter currently, contained the Geotag in the photo. Therefore it is necessary to find the method of estimating location without Geotag.

Above the researches, the usual method of estimating location is selecting a set of words that show strong locality (termed as local words) instead of using entire corpus to improve the accuracy of predicting. In the research [11] it mainly discussed to predict users' locations by fixed factors. In the contents-based method people should first select a unique factor to extract the useful tweets and in the past researches always select the fixed factor from the area. Follow this thought we can also use the selection of keyword in location estimation.

As the main research target of this paper, I mainly discuss the event location estimation. An event usually contains several factors: event name, time and location. When people tweeted an event related tweet, some of them would like to tweet the event location. For example after an earthquake happened, the people who lived near to the event would tweet in the first time with their locations, and these locations could pointed out the event location. Considering the locations might not be a same location, if we can analyze the geographical distribution of these locations the event location might be estimated. In this approach, it is important to extract the geographical words from the tweets, so I selected "event", the event with geographic locality, which only happened at a limit area, as the keyword of researching in this paper. Using this approach we can get more samples than only using Geotag. After that, the content-based approach described in chapter 2.2 could be used in the next step.

As discussed above, this research mainly divided to 3 parts:

(1) Crawling related tweets;

(2) Extracting the geographic words;

(3) Analyzing the geographical distribution of these words and get a estimated location of event.

In the part of (1) and (2), I used usual Twitter analysis approaches to achieve them, and in the part of (3), I choose clustering as the analysis method.

3.2 Clustering

By showing the collected places of the crawling results on the map, we can visually observe the possible locations of event. According to the common sense, the event could more likely happen at where the locations are denser. However, the results of this step are not accurately enough, it only shows the approximate location on the map and do not have a certain point. For solving this problem, I used the cluster analysis in this research.

Cluster analysis is the task of grouping a set of objects in such a way that objects in the same group are more similar to each other than to those in other groups. It is a method of unsupervised learning and is a common technique for statistical data analysis used in many fields. [31] The table 3-1 shows the main clustering algorithms.

Method name	**Parameters**	**Scalability**	**Usecase**	**Geometry (metric used)**
K-Means	number of clusters	Very large Medium with MiniBatch code	General-purpose, even cluster size, flat geometry, not too many clusters	Distances between points
Mean-shift	bandwidth	Not scalable	Many clusters, uneven cluster size, non-flat geometry	Distances between points
Spectral clustering	number of clusters	Medium small	Few clusters, even cluster size, non-flat geometry	Graph distance (e.g. nearest-neighbor graph)
DBSCAN	neighborhood size	Very large medium	Non-flat geometry, uneven cluster sizes	Distances between nearest points
Gaussian mixtures	many	Not scalable	Flat geometry, good for density estimation	Mahalanobis distances to centers

Table 3-1 Main clustering algorithms[1]

Through the clustering we can analysis the data, as the result similar data would be divided into one

1 This table is derived from skit-learn website [18]

group. For the goal of finding the event location, I choose the centroid-based clustering as analysis method. In this paper the main research object is the geographical word extracted from the related tweets. After crawling I found most tweets do not contained the geographic words, so the analyze objects would not be too much. Considering these factors I choose the K-means clustering as the analysis method in the analyzing part.

K-means clustering [20] is a type of unsupervised learning, which is used when I have unlabeled data (i.e., data without defined categories or groups). The goal of this algorithm is to find groups in the data, with the number of groups represented by the variable K. The algorithm works iteratively to assign each data point to one of K groups based on the features that are provided. Data points are clustered based on feature similarity.

The results of the K-means clustering algorithm are:

(1) The centroids of the K clusters, which can be used to label new data.

(2) Labels for the training data (each data point is assigned to a single cluster).

The K-means clustering algorithm is used in many areas and mainly for finding groups which have not been explicitly labeled in the data. This can be used to confirm business assumptions about what types of groups exist or to identify unknown groups in complex data sets. Once the algorithm has been run and the groups are defined, any new data can be easily assigned to the correct group. And in this research I use the K-means to speculate where the event might be occurred. And in this paper I defined the centroid of group as the event location. The specific steps of K-means clustering would be introduced in chapter 4.

3.3 The meaning of event location estimation

Comparing with the past work of event location estimation, this approach could be useful in speculating user's location. Before discussing the reason, as we know Tweet have many elements, like time, contents, retweet, location and others. In this research I mainly discuss the time, contents and location. The Figure 3-1 shows the structure of tweet. As I discussed in chapter 2, users always close the Geo tag function therefore one of remaining methods to obtain geographic location is analyzing tweet content. However it is difficult to judge whether the tweet was tweeted in the place event happened. For example, Figure 3-2 shows the result of searching the word "event" after a random event named "event" happened. The "event" may be some shopping events, the Christmas event, disasters, alarms or news. In the timeline the tweets a, b, c, d, e have the same search word "event", but in fact they tweeted at different time and from different places, event some of these do not have a geographic location.

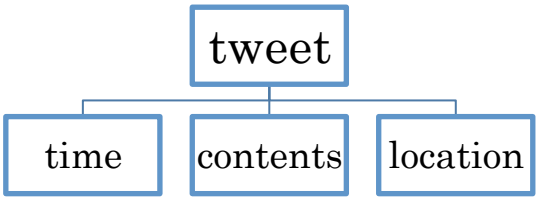


Fig. 3-1 The structure of tweet in this research

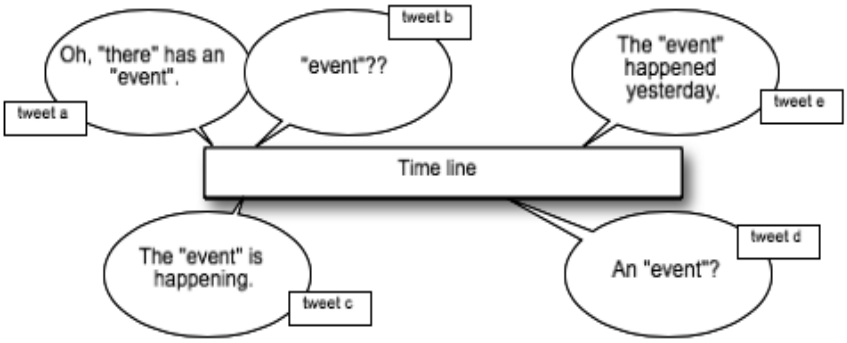


Fig. 3-2 The situation of search the word "event"

In order to avoid this situation, in this paper I mainly set the local events with geographical locality as main object, like earthquake, typhoon, an accident happened suddenly. Using these words to extract related tweets and extract the geographic factors from these tweets can be helpful in estimating location.

Except the event location estimation, this approach could be helpful for estimating the user's location. In the analysis of those events, I hypothesize that users' tweeting behavior are following a certain pattern. For example, after an earthquake happened, the users closer to the earthquake may have more tweeting activity than others. In this case I should consider a specific location pattern of the users' location. For example, an earthquake occurred at Tokyo, for the directly shock to the people who live in Tokyo, they might tweet a lot of related tweets after earthquake; at the same time the people who lived around Tokyo also felt shock and they might tweeted about earthquake. However the number of twitter could be less than the people lived in Tokyo posted.

As an example, consider an earthquake happened at 7pm in the place P. Fig.3-3 shows the tweets posted by the people lived in P1, and Fig.3-4 shows the tweets posted by the P2, where 200 kilometers away from the P1. People at P2 could not feel so strong shock as P1, therefore the amount of tweets could be less.

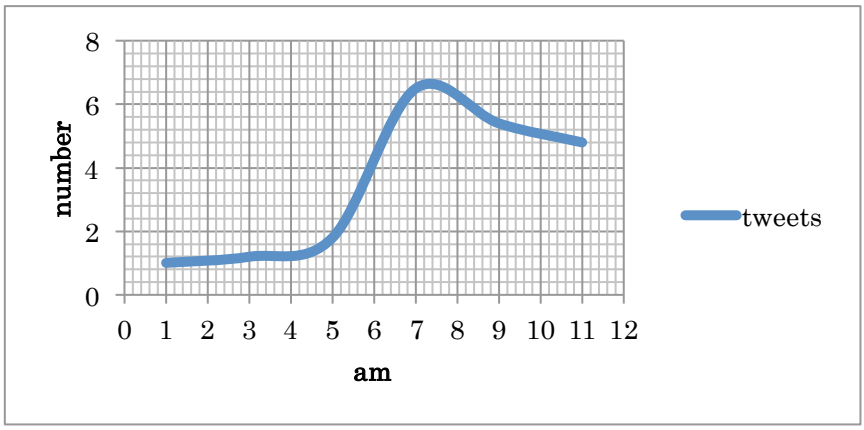


Fig. 3-3 The status of tweets when an event occurred

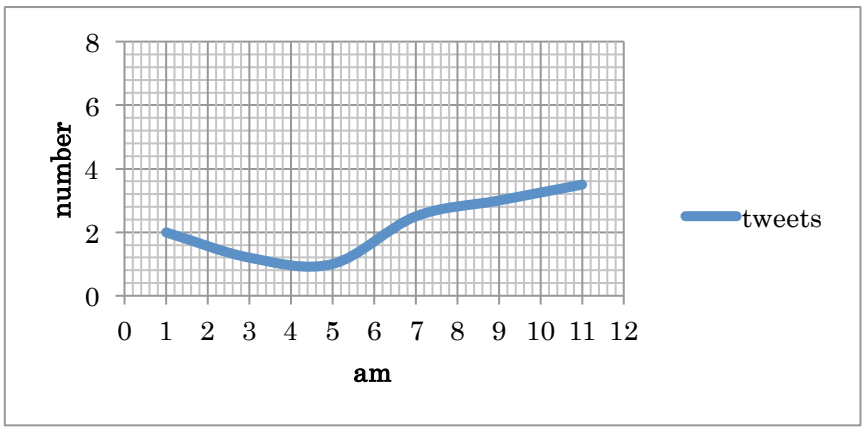


Fig. 3-4 The status of tweets in normal situation

By analyzing the heat of tweets it is easy to extract the special time zone of tweets, then we can set the time as the event time. The users who tweeted event at the same time zone could be the user who lived near the event. Combined with the method in chapter 3.1, we can also speculate the user's location at the same time. But in this situation there has a question: if the earthquake happened at the place between Tokyo and Chiba, both of people would like to post a lot about the earthquake, and we could speculate that the earthquake may occurred somewhere between Tokyo and Chiba. After that, when an event happened at the same time in two places, these should be solved in the future work.

After all, this paper mainly discusses the part of event location estimation. For testing the assumption of solving non-Geotag problem in chapter 3.1, I did a experiment of it in next chapter.

Chapter 4 Interface development

This chapter is based on the assumption discussed at chapter 3. To resolve the non-location problem I use "event" as the research object and design the location estimation system for getting event location. According to the discussion in chapter 3, the process is divided into three parts – crawling data, changing data and clustering. This chapter introduces the structure of the system and main tools used in this paper.

4.1 Twitter collect

Firstly I designed a system to collect Twitter data. The method of collecting data is using python through the Twitter API. The data I scrawled was stored in the MongoDB, and Fig.4-1 shows the process of data collecting.

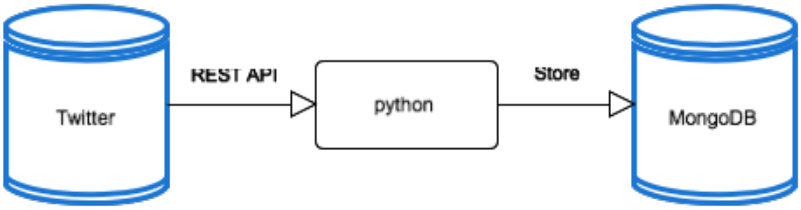


Fig. 4-1 The process of crawling

Next parts I will introduce technologies that make up this system.

4.1.1 Twitter API

Titter provides the APIs (application programming interfaces) for the companies, developers, and users with programmatic access to Twitter data. Twitter data is unique from data shared by most other social platforms because it reflects information that users choose to share publicly. It supports APIs that allow users to manage their own non-public Twitter information and provide this information to developers whom they have authorized to do so. [15]

Rest API

In this research I mainly use the REST API to acquire the data. It allows access to Twitter's core data,

update timelines, status data, and user information. The REST API identifies Twitter applications and users using OAuth; responses are in JSON format.

At the latest version 1.1 of the Twitter API, the data I used the “GET search/tweets” I crawled is divided into two kinds: TwitterListResponse and metadata. I would show one of the results I searched the tweets include the event word “地震” in appendix A. And the appendix B shows the dictionary of tweet data[16]. In this research I mainly discuss the id, place, coordinates and text.

Rate limiting

The Twitter API makes a limited number of calls in a given time interval, and in the version 1.1 the rate limits becomes 15 minute intervals by different types of request. To resolve this question, I design the catch data as a loop structure. If the amount of data reaches the limit, the program should sleep for several minutes.

4.1.2 MongoDB

MongoDB is a free and open-source cross-platform document-oriented database program. Classified as a NoSQL database program, MongoDB uses JSON-like documents with schemas. The document model maps to the objects in application code, making data easy to work with Ad hoc queries, indexing, and real time aggregation provide powerful ways to access and analyze data. At the same time, it is a distributed database at its core, so high availability, horizontal scaling, and geographic distribution are built in and easy to use. MongoDB is developed by MongoDB Inc[18].

4.1.3 Mecab

Mecab is an open source text segmentation library for use with text written in the Japanese language originally developed by the Nara Institute of Science and Technology and currently maintained by Taku Kudou now. It can analyze and segment a sentence in to its parts of speech. The latest version is 0.996[14].

In the contents analyze part I use the Mecab to extract the geographic elements contained in the tweets. Figure shows the main part of contents analyze. When the content in the tweet matches the “固有名詞” and “地域” dictionaries of mecab, I would extract the word into the database and treat this tweet as valid in the research. Fig.4-3 shows the code of extracting the word of “地域”.

```
def location_name_mecab(sentence):
#
   t = MeCab.Tagger('-Ochasen')
sentence = sentence.replace('¥n', ' ')
    text = sentence.encode('utf-8')
    node = t.parseToNode(text)
    result_dict = defaultdict(list)
    for i in range(140):
        if node.surface != "":  #
            #
            if (node.feature.split(",")[1] == "固
有名詞") and (node.feature.split(",")[2] == "地域
"):
                plain_word =
node.feature.split(",")[6]
                if plain_word !="*":
                    result_dict[u'地域名称
'].append(plain_word.decode('utf-8'))
        node = node.next
        if node is None:
            break
    return result_dict
```

Fig. 4-2 The code of extracting the word of “地域”

4.1.3 Latitude and longitude transformation

Geocoding is the process of converting addresses (like “1600 Amphitheatre Parkway, Mountain View, CA”) into geographic coordinates (like latitude 37.423021 and longitude -122.083739), which I can use to place markers or position the map. In this step I use Geocoder to retrieve google’s geocoded data from Google Geocoding API.

At last, the data transformation process is as Fig.4-4.

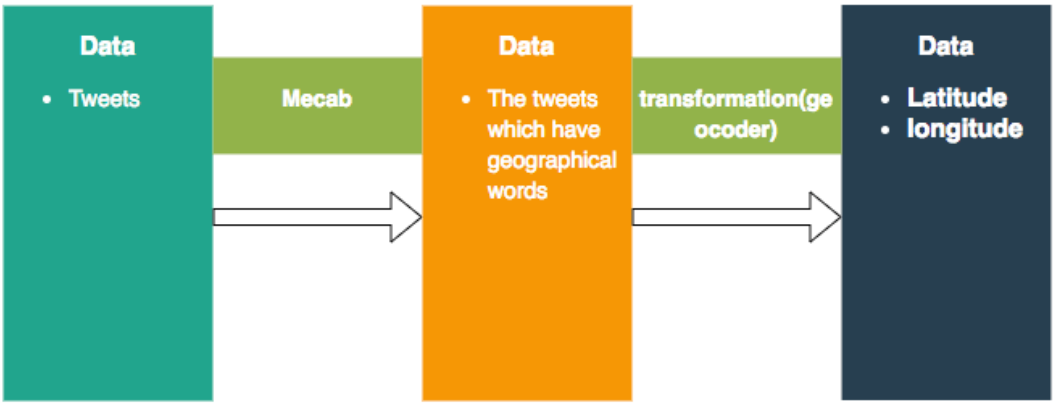


Fig. 4-3 The process of obtaining coordinates of the tweets

4.2 K-means clustering algorithm

The K-means clustering algorithm uses iterative refinement to produce a final result. The algorithm inputs are the number of clusters K and the data set. The data set is a collection of features for each data point. The algorithms starts with initial estimates for the K centroids, which can either be randomly generated or randomly selected from the data set. The algorithm then iterates between two steps:

1. Data assignment:

Each centroid defines one of the clusters. In this step, each data point is assigned to its nearest centroid, based on the squared Euclidean distance. More formally, if c_i is the collection of centroids in set C, then each data point x is assigned to a cluster based on

$$\underset{c_i \in c}{\operatorname{argmin}}\, dist(c_i, x)^2$$

where $dist(.)$ is the standard (L_2) Euclidean distance. Let the set of data point assignments for each i^{th} cluster centroid be S_i.

2. Centroid update:

In this step, the centroids are recomputed. This is done by taking the mean of all data points assigned to that centroid's cluster.

$$c_i = \frac{1}{|S_i|}\sum_{x_{i\in S_I}} x_i$$

The algorithm iterates between steps one and two until a stopping criterion is met (i.e., no data points change clusters, the sum of the distances is minimized, or some maximum number of iterations is reached).

This algorithm is guaranteed to converge to a result. The result may be a local optimum (i.e. not necessarily the best possible outcome), meaning that assessing more than one run of the algorithm with randomized starting centroids may give a better outcome.

In this algorithm should find the clusters and data set labels for a particular pre-chosen K. In general there is no method for determining exact value of K, so I choose the commonly method of comparing results across different values of K by calculating the average distance between the data point and its cluster centroid.

Combing with I discussed in 4.1, the three steps help us to get an approximate location of event, and in the next chapter I will do the experiment to evaluate this idea.

Chapter 5 Experiment

5.1 Experiment object

In order to estimate the event geographical location, I used the twitter API to collect the tweets and extracted the useful tweets from them, then plotted the results on the map to observe where the event happened directly. After that, I used the K-means clustering algorithm to improve the accuracy of location estimation. Through the experiment I want to achieve 2 goals:

(1) Verifying whether there were more tweets posted near the place where the event occurs.

(2) The final result is close to the actual location.

5.2 Data set

Considering the define of the event in the chapter 3.2, I choose "地震" as the "event". Earthquake is easy to observe and has detailed data in the web, and in almost case it happened only at one place what help us to verify the results. I collected the data between 2018.6.21 to 2018.6.26 when the earthquakes frequently happened, and divided it into three datasets with the automatic stop of the API. All of these were crawled by the keyword "地震".

Their statistics overview can be seen in Table5-1.

Stats	Set A	Set B	Set C
Tweets	43600	248599	137600
Geo tag	11	91	88
The tweet contained word of "地震"	2066	4786	1893

Table 5-1 The status of crawling result

We can find only few tweets contained Geotag, this result proves that the necessary of we should find other methods to estimate user's location without Geotag analyzing. Compare with tweet contained word of "地震", the number of available data increase significantly.

5.3 Crawling and changing results

After crawling the data, I extract the tweets contained geographic location and put the locations in the map. The follow table shows the twitter active states of the three datasets. As we known Twitter has a tweet called “Retweet”, and it might bother result. Considering that RT contents also mention event-related content, I kept this part of data and examined “RT” and “NotRT” contents together. In order to increase the availability of data, I removed the “spam”, one of the tweets what can be generally described as unsolicited, repeated actions that negatively impact other people.

The follow 3 tables and graphs show the status of 3 data sets.

(1)

Date	#ALL	#NotRT	#RT
2018 06/21 21 Thu	15700	4189	11511
2018 06/21 22 Thu	3613	1145	2468
2018 06/21 23 Thu	18699	5908	12791
2018 06/22 00 Fri	5588	2089	3499

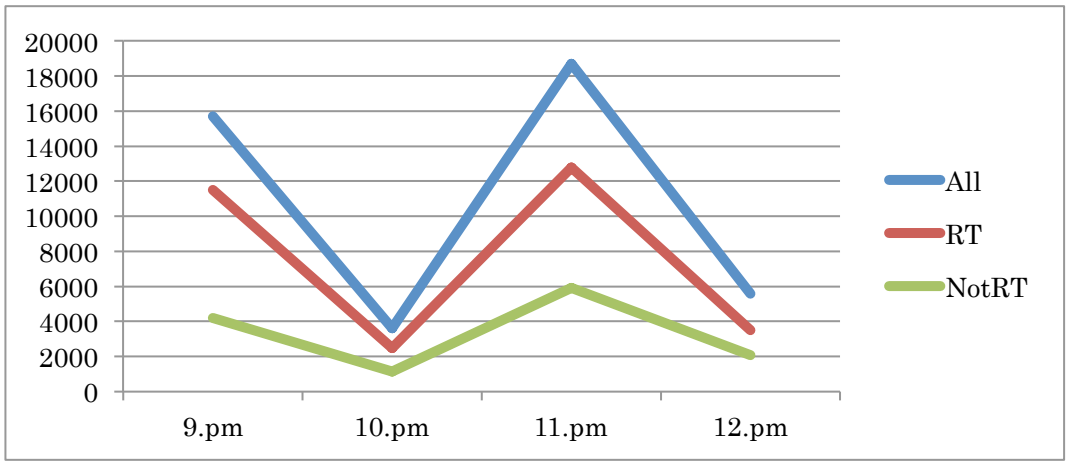


Table 5-2 Data of set1

(2)

Date	#ALL	#NotRT	#RT
2018 06/22 01 Fri	4000	1232	2768
2018 06/22 03 Fri	300	81	219
2018 06/22 08 Fri	3537	891	2646
2018 06/22 09 Fri	2063	537	1526
2018 06/22 12 Fri	6793	1590	5203
2018 06/22 13 Fri	2707	715	1992
2018 06/23 15 Sat	2800	707	2093
2018 06/23 17 Sat	3600	1102	2498
2018 06/23 18 Sat	100	30	70
2018 06/23 19 Sat	1903	543	1360
2018 06/23 20 Sat	9810	2935	6875
2018 06/23 21 Sat	11456	3577	7879
2018 06/23 22 Sat	4519	1346	3173
2018 06/23 23 Sat	191011	137906	53105
2018 06/24 12 Sun	700	197	503
2018 06/24 13 Sun	1000	222	778
2018 06/24 14 Sun	300	70	230
2018 06/26 00 Tue	300	86	214
2018 06/26 14 Tue	1600	508	1092
2018 06/26 19 Tue	100	96	4

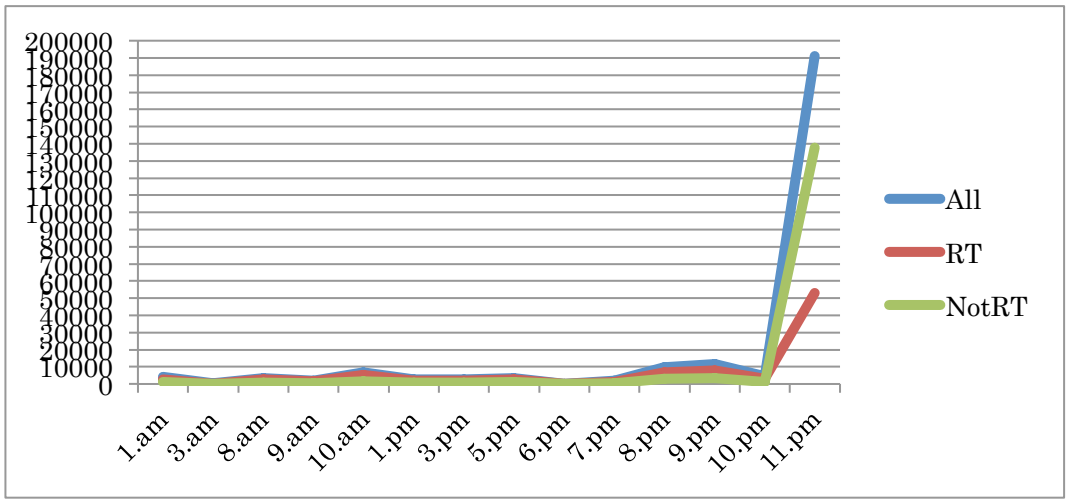


Table 5-3 Data of set2

(3)

Date	#ALL	#NotRT	#RT
2018 06/26 19 Tue	38400	28784	9616
2018 06/26 20 Tue	87762	61316	26446
2018 06/26 21 Tue	5338	2741	2597
2018 06/26 22 Tue	6100	2762	3338

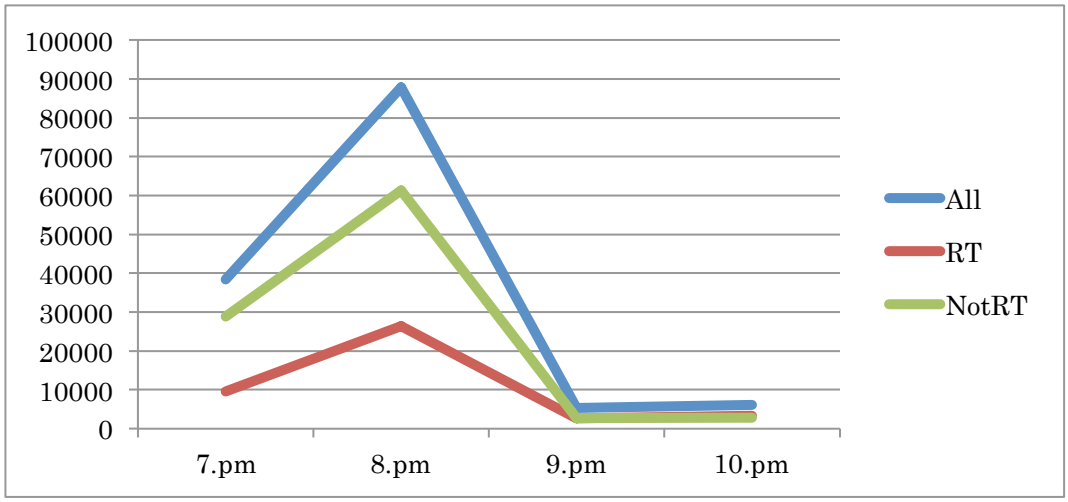


Table 5-4 Data of set3

We can observe that there are several high frequencies of tweeting at certain times, considering all the tweets contained the word “地震”, the increase related tweets may mean the occurrence of earthquakes. Except it, we can intuitively find that when the twitter be hot, the quanity of “NotRT” tweets could be more than the “RT” tweets, and in usual case, it would be opposite. This phenomnon may be caused by the people who were at the near place of earthquake and tweeted soon, other people who can’t feel the hapenning of earthquake would retweeted after the earthquake happened. I will analize the posibility eflect of this phenomnon in the chapter 6.

At the next step, I extract the geographic locations of the data and plotted them on the map for comparing with the actual earthquake location (from the Meteorological Bureau website). For comparing the results, I extract the result of Japan result. The result was as follow figures.

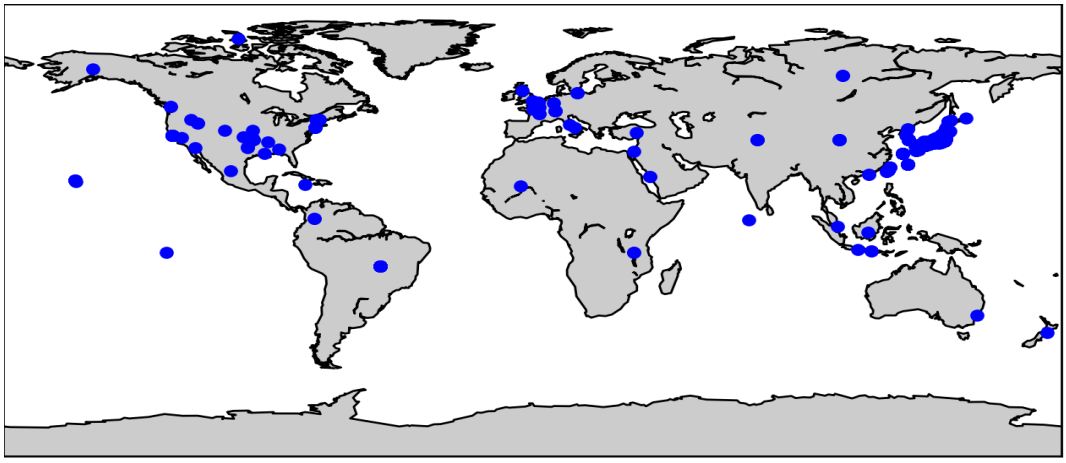
The structure of figures as follows:

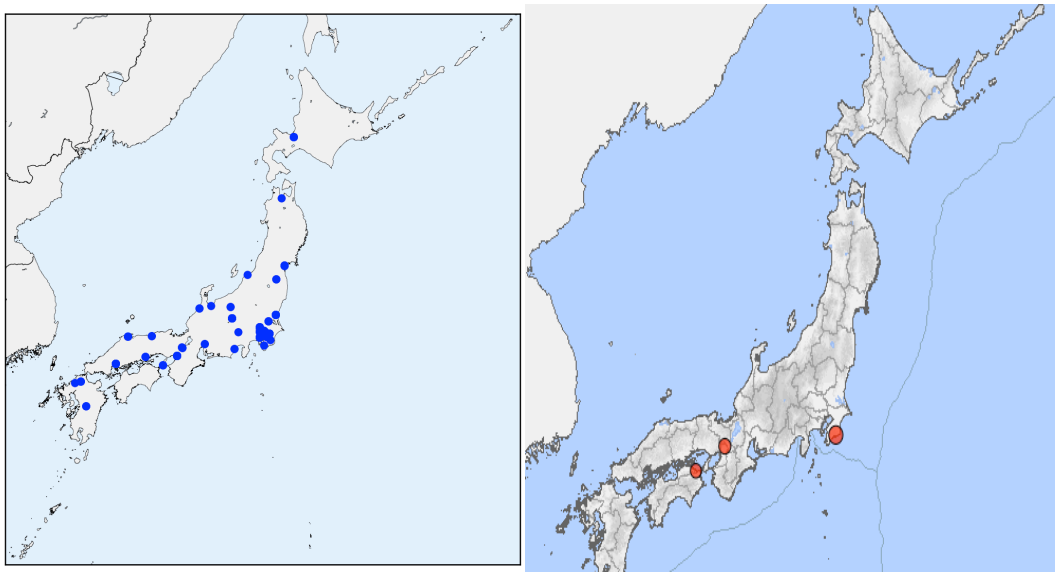
(1) The result of plotting crawling data(have been changed into Latitude and longitude) on the world map

(2) (Left) The plot result of Twitter data in Japan area (Right) The earthquake depth [22]

(3) The data of earthquake happened time, place, longitudes and latitudes of the center, magnitude.

(1) Set 1



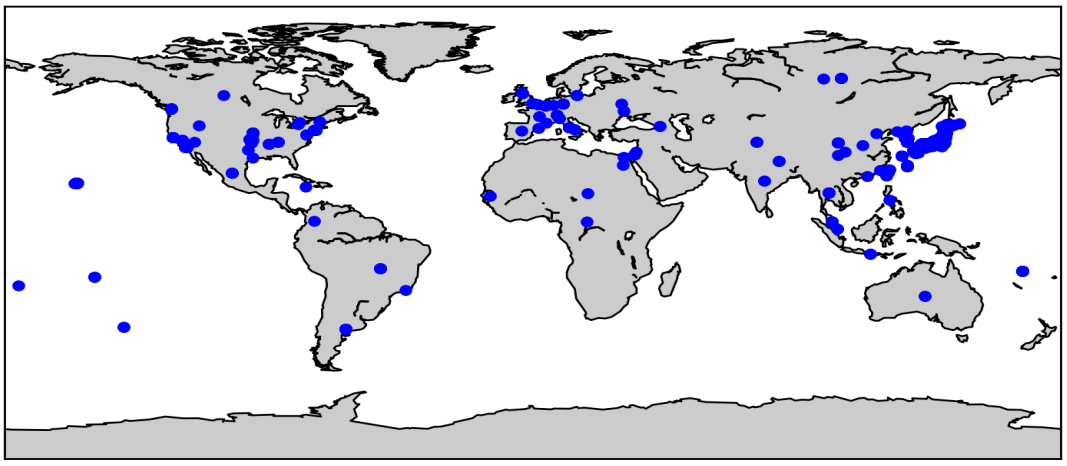


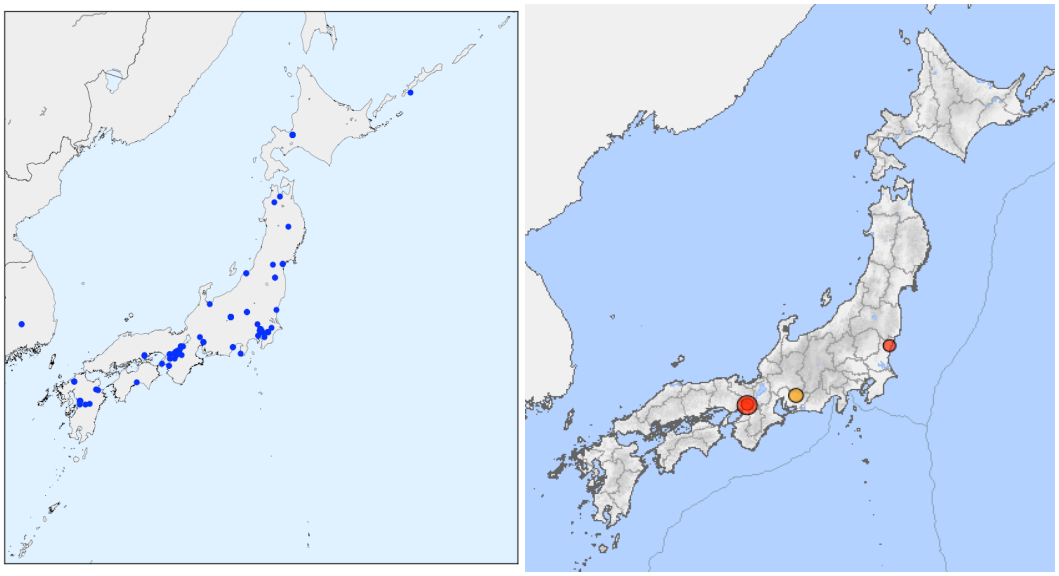
Time	Location	Longitude	Latitude	Depth	M
2018/06/21 19:18:36.5	千葉県南部 (South Chiba)	35°07.9'N	140°14.9'E	11km	M3.3

Fig. 5-1 Set 1

Different from the other two sets, the amount of data in set 1 is the smallest. Although the magnitude of earthquake that happened in the periods of set 1 was belonged to small earthquake, the seismic coefficient was not enough to make most people feel it. Considering that this result may inflect the accuracy of the location estimation, I specifically proposed this situation to investigate.

(2) Set 2



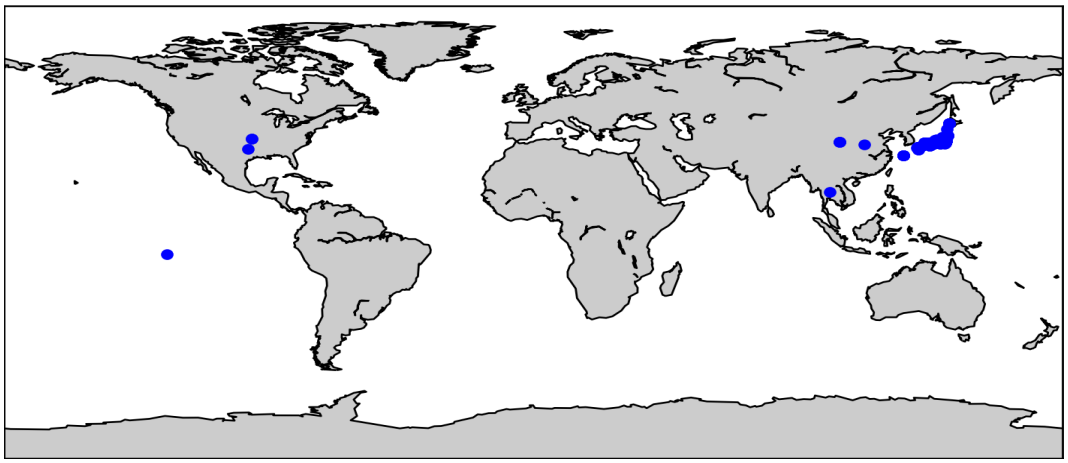


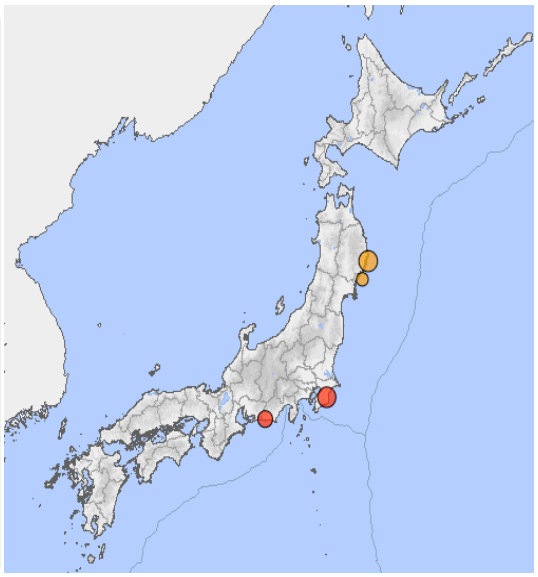
Time	Location	Longitude	Latitude	Depth	M
2018/06/23 23:08:45.7	大阪府南部 (South Osaka)	34°49.9'N	135°37.3'E	11km	M4.0

Fig. 5-2 Set 2

In the set 2 I collected the data from 6.22 to 6.26, to distinguish it from the set 3 I mainly discussed the tweets between 6.22 and 6.23, there are very few tweets about earthquakes tweeted at other time. As we seen the tweets were tweeted from all over the country, and these points mainly concentrated to Osaka, but the crawling result also showed that some tweets pointed out Tokyo. Then this situation happened at every set, I guess because the speed of intelligence transmission now allows people to get news at the first time, and the high population of Tokyo effects the tweet place which could bother the results when we do a directly observation.

(3) Set 3





Time	Location	Longitude	Latitude	Depth	M
2018/06/26 19:46:22.3	千葉県南部 (South Chiba)	35°20.9'N	140°20.7'E	26km	M4.3

Fig. 5-3 Set 3

Set 3 contains the highest level of earthquakes, which is easy to observe by native people. But regardless of the usefulness of the data, the points on the map are still focus on Tokyo.

The common feature of these sets is that only one major earthquake occurred at the time zone, and this paper is based on this special case.

5.4 Clustering results and evaluation

The above observations are based on the geographic location contained in the Twitter, because the results are transformed by the Mecab directly, even if almost results are pointing to one place that would not be showed in the map, what means the result does not reflect the quantity of data. In the observation result of chapter 5.3, we also know the result tend to be close to Tokyo what means the result is not accurate.
Therefore I choose the K-means clustering to estimate the dense tweeted location and treat the centroid of group as the event location.
As I introduced K-means clustering in chapter 4, it selects random k point as the means and by associating every observation with the nearest mean creates k clusters, then the centroid of each of the k clusters becomes the new mean. After repeating the data assignment and centroid update steps until convergence has been reached. The result of clustering would be different when select different k. For choosing the best result, we should find other method to do evaluation. In this paper, I choose an usual method of clustering – elbow method.

5.4.1 elbow method

The elbow method is a method of interpretation and validation of consistency within cluster analysis designed to help finding the appropriate number of clusters in a dataset. The idea of the elbow method is to run k-means clustering on the dataset for a range of values of k, and for each value of k calculate the sum of squared errors (SSE). SSE tends to decrease toward 0 as we increase k, if the line chart looks like an arm, then the "elbow" on the arm is the value of k that is the best. The method can be traced to speculation by Rpbert L.Thorndike.
The algorithm of SSE is as follows:

$$SSE = \sum_{i=1}^{k} \sum_{p \in c_i} | p = m_i |^2$$

In this algorithm, p is the point belonged ith cluster c_i, m_i is the centroid of c_i.
I used this method to calculate the best value of k in every set.

5.4.2 clustering results

As usual rule, the value of K could not be too big, so in this step I set the K between 2 and 10 and do a comparison of these results. The result of clustering shows the world range results like the Fig.5-4, which shows the clustering result of set 2 when k = 2.

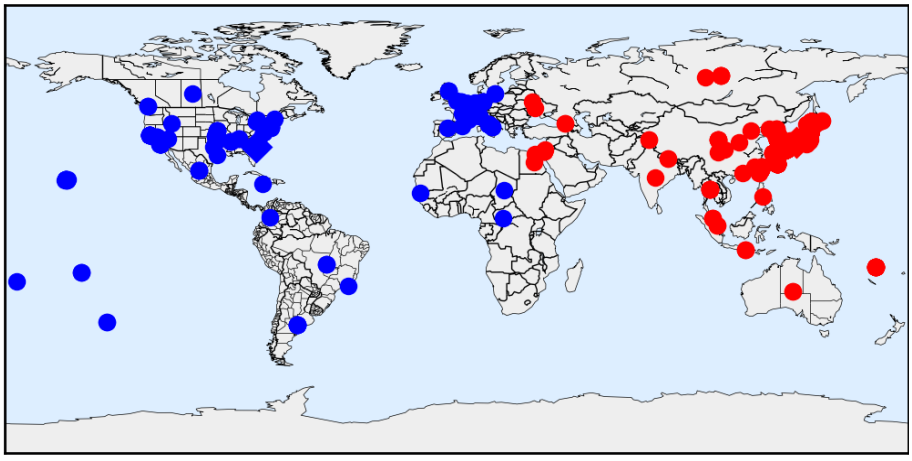
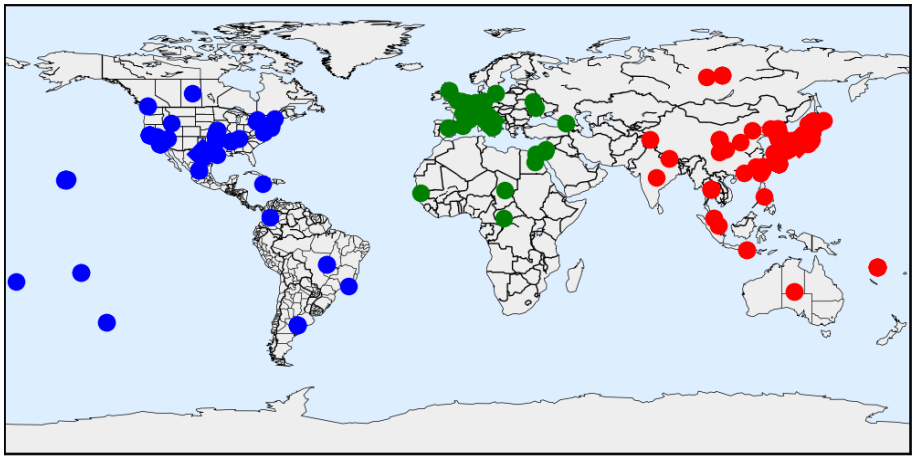


Fig. 5-4 The clustering result of set 1 when k = 2

As we can see the group divided into two groups, one of centroids is (32.988, -79.714), the other is (34.727. 135.280), what means one of the centroid is just in Japan. In the same time I also observed other results of choosing different k, Fig.5-5 shows the result of k = 3 and k = 4, we can find no matter how the value of k changed, there still has an centroids just at Japan area (the red group on the map).



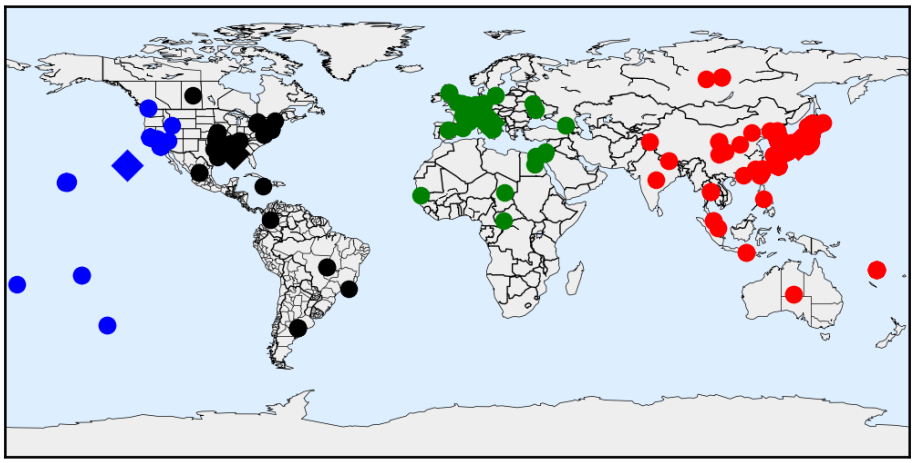


Fig. 5-5 The clustering results of set 2 when k = 3 and 4

Table 5-5 shows the centroids in these results, we can find the first row in every k group point out the centroid is just in Japan. This situation also happened in other set.

K	Centroids
2	34.781,135.266
	34.854,135.695
3	34.854,135.695
	38.292,21.892
	29.910,-98.739
4	34.855,135.603
	36.659,-105.839
	38.570,15.589
	-11.548,-55.121

Table 5-5 The centroids of set 2 when k = 2,3 and 4

After that, I used the "地震(earthquake)" in Japanese language as the search word in this paper. According to common sense the centroids should be concentrated in Japan, but some of bother twitter could be found in the set. At the same the word of "地震" is also used in Chinese language. Considering these cases, I mainly discussed the centroids in Japan and treat them as the analyze object in this research. The following figures showed the results of clustering data sets. The order of these figures is:

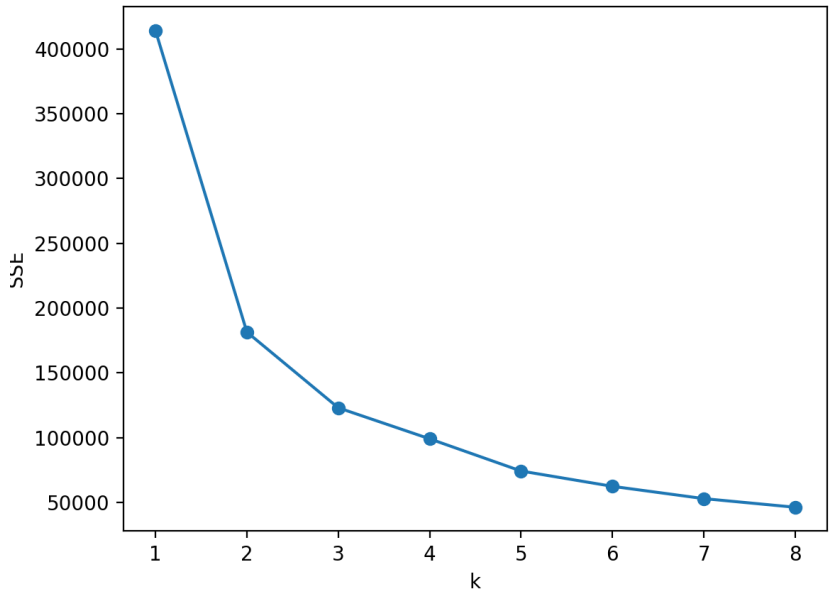
a. The result of centroids in every set (Japan area).

b. Elbow method result in every set.

(1) Set 1

K (set1)	Centroids
2	34.781,135.266
3	34.854,135.695
4	34.855,135,603
5	34.839,135.571
6	35.130,135.933
7	35.160,136.049
8	35.188,136.016
9	34.481,134.744

(a) The result of centroids in set 1



(b) The result of elbow method in set 1

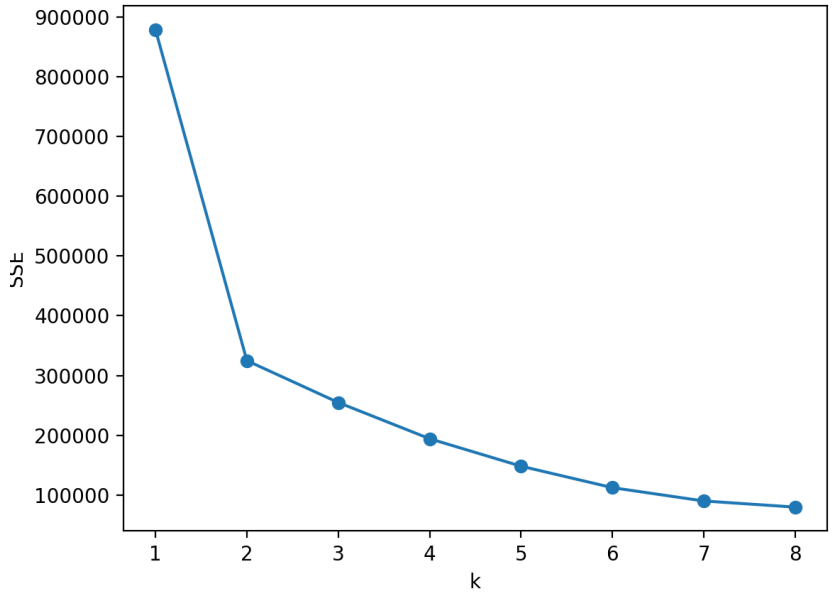
Fig. 5-6 Clustering result of set 1

When k equals 3 the decreasing of SSE tends to be stable. In this case, the appropriate value of k is 3.

(2) Set 2

K (set2)	Centroids
2	34.727,135.27
3	34.724,135.545
4	34.724,135,545
5	34.724,135.545
6	34.861,135.875
7	34.962,135.812
8	34.962,135.816
9	34.962,135.812

(a) The result of centroids in set 2



(b) The result of elbow method in set 2

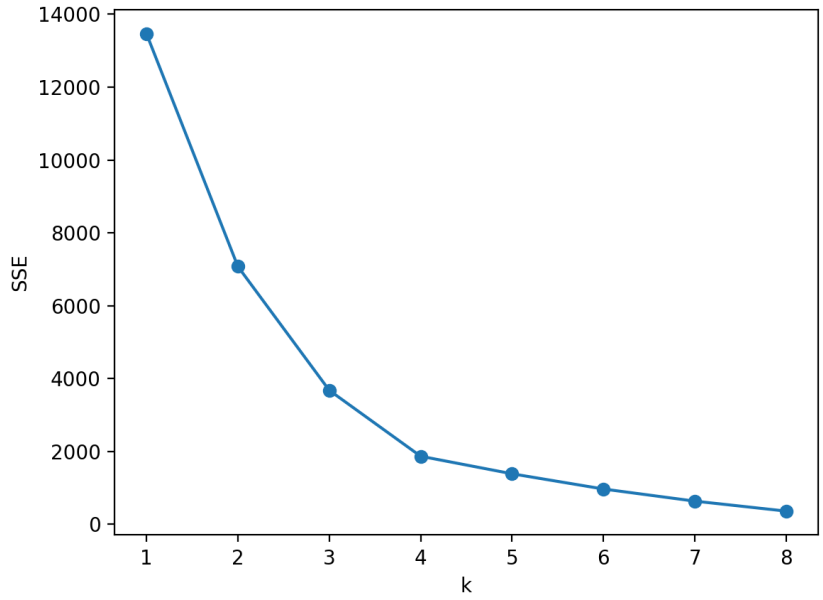
Fig. 5-7 Clustering result of set 2

When k equals 2 the decreasing of SSE tends to be stable. In this case, the appropriate value of k is 3.

(3) Set 3

K (set2)	Centroids
2	35.605,139.672
3	35.663,139.969
4	35.676,140.040
5	35.677,140.040
6	35.677,140.040
7	35.645,140.035
8	35.654,140.035
9	35.624,140.110

(a) The result of centroids in set 3



(b) The result of elbow method in set 3

Fig. 5-8 Clustering result of set 3

When k equals 4 the decreasing of SSE tends to be stable. In this case, the appropriate value of k is 3.

As result we get every appropriate value of k in set 1, 2 and 3. In order to observe the accuracy of estimation, I compared the estimate earthquake location and actual earthquake location, and defined the distance between two locations as error distance. And the result is shown in Table 5-6.

Set	K	Centroid	Actual	Error Distance(km)
1	3	34.854,135.695	35.131,140.248	432.689
2	2	34.727,135.279	34.832,135.622	33.469
3	4	35.676,140.040	35.348,140.345	45.793

Table 5-6 The error distance between the results and actual locations

As I discussed in chapter 5.3, the distance in set 1 shows the unsuitable observe object will engender fairly great error. In fact the earthquake happened in that period only level 2 and most people could not feel it directly, so most collected tweets were told about the big earthquake happened at Osaka two days ago. When we cannot guarantee the people feel the event happened actually, those tweets without directly association would bother the location estimation. In the set 2 the error distance just be 33.469 km. The set 3 also has a small error distance of 45.793 km.

In general this result proves that it is possible to use this system to estimate event location within an small error distance. And this result based on searching the event with geographic locality and the event only happened once at the same time.

Chapter 6 discussion

6.1 Discussion

Comparing with previous researches, the work in this paper detected event location without using Geotag. From the approaches of predicting event location, I choose a common method of searching keywords, and used the event name as search condition directly. In the location estimation part, I selected the clustering based on the usual rule more tweets about the location of event would be posted by the users. The table 6-1 shows the settings of research.

	Geotag	Twitter stream	Clustering
This work	X	O	O

Table 6-1 The status of this research

As the result I obtained the estimate location of earthquake in the chapter 5.4.These results were divided into three situations: (1) The small scale of earthquake that cannot feel directly; (2) There are no interference conditions; (3) There just occurred onece earthquake in the periods.

According to the result, I summarized the condition of applying this method:

(1) It based on using the keyword of “event”, the event that has a geographical limitation.

(2) The event is best only happening at one place.

(3) Because of using the Mecab to extract the location and the dictionary of Mecab is limited, this method is suitable in city level estimation.

In this part I compare the location estimation with the past work.

In the chapter2.2 I discussed the contents-based method, and I would also discuss the other location estimation method -- graph-based method. Clodoveus’ method set random users as testing users, and collected information from users who are following each other, what is called in followers in the Twitter. In this case they assumed the location where the most users residence is as the residence of the test users. This method had a high accuracy of location estimation but it is less practical with the less using of Geotag.

Chengs’ method should prepare two sets of data, one is learning data and another is testing data. They defined the users who have a known location as testing user, and none of learning user was the same with

the test user. Then they collected a fixed quantity (like 100) tweets that the learning users tweeted recently. To take advantage of these data the used these tweets to study the location distribution of words. After that they collected recent tweets from the testing users, and predict the user location by using the learning model made in the previous step.

In the method combined with graph-based method and contents-based method, Lis [23] also used the Mecab in their research. Since it is necessary to learn geographical dispersion of place names, they collected the recent tweets from the testing users and extracted the “location” word from these tweets. Different from my research they changed the location to coordinates by the API of OpenStreetMap. In the next step Lis implemented the same method as Clodoveus’ because of the need of followers information. Other researches [24][25][26][27][28] also provided me with ideas for improving my method.

As the result of the comparison, I found several factors may inflect the accuracy of results, and I extract these possible problem and discussed at here.

Problem1 The effectiveness of the keyword

Although I used the earthquakes what happened in a limited range and easy to test, some events do not have this feature. If we search the word “交通事故”, it may be happened nationally at the same time. Even the people can see the event at the TV and tweet some feelings about the event. In this case it is difficult to estimate an accurate event location. Apart from this problem, the keyword also could be some other words that do not relate to event, like lyrics, a game name and unique word. If we do not choose a unique word, we cannot get a high accuracy result at last. How to correspond to this event with multiple geographic attributes is an important future task.

Problem2 Population density

Because of the convenience of information circulation, when an event happened people can get the news at the first time. But in these cases we should consider the population density problem. As we known the city like Tokyo and Osaka are high-density city with a very high usage rate of twitter (Facebook etc.). Whenever an event occurs, there always have a fair amount of event-related tweets would be tweeted. In the test of chapter 5.3 and 5.4, we can observe there is always a certain amount of data sent from the Tokyo.

Problem3 Twitter classification

In my work I mainly removed spam from the twitter and only used the tweets contained the geographic

words. But this is not enough in the classification. Some bot and news tweet would like to report the repeat contents, and in the crawling results there have a lot of those tweets what may inflects the calculate of tweeted frequency. It is necessary to build a classification to remove the useless tweets and increase the accuracy of estimation.

6.2 Future work

As I discussed in chapter 6.1, the works I should do are:

(1) To build a twitter classification and extract useful information.

(2) Considering the people density problem, find a method that could separate such users from the crawling data.

(3) In this system the events are all manually entered, and only tested the “earthquake”. It should build detect methods to define the event and separate the kind of event from the crawling. In this case we can test other event and verify the available of my method.

(4) I mainly use the K-means clustering in this paper, and do not test precisions of other algorithms for comparing. It is possible to get a higher accuracy by using other algorithms.

(5) Using the system in the user location estimation. This research is based on the content-based method in location estimation of users, and in the case we can also estimate the users location by the rule that people near the event would tweet the related contents about event.

Chapter 7. Conclusions

In the approaches of estimating location, I choose a contents-based method in my research. In the experiment, I mainly collected the tweets contained the keyword "地震"—the event which has a geographic locality. By observing these data crawling by "地震", I found that these results could prove the idea that people would tweet a lot after an event happened. Then I extracted the geographic locations from the data sets and found it is difficult to get the location directly by observing these locations on the map. Considering the high frequency of the geographic words may point out where the event occurred, I decided to use the clustering to analyze data sets. As result I got the estimate locations of earthquake location by using the K-means clustering to analyze the data set. The result of experiment showed two of data sets get accurate event locations with small error distance -- the distance between the estimate location and actual location. It proved the availability of my method under limited conditions: (1) it based on using the keyword "event", the event that has a geographical limitation; (2) the event only happened at one place at the same time. At last, I discussed the methods of increasing the accuracy of estimating the location of events based on Twitter.

Reference

[1] "Twitter over counted active users since 2014, shares surge on profit hopes". USA Today.

[2] Isaac, Mike; Ember, Sydney (November 8, 2016). "For Election Day Influence, Twitter Ruled Social Media". The New York Times. Retrieved November 20, 2016.

[3] “Twitter,” https://twitter.com/.

[4] Rattenbury, T., Good, N. and Naaman, M.: Towards Automatic Extraction of Event and Place Semantics from Flickr Tags, SIGIR, pp.103–110 (2007).

[5] Lappas, T., Vieira, M.R., Gunopulos, D. and Tsotras, V.J.: On the Spatiotemporal Burstiness of Terms, PVLDB, Vol.5, No.9, pp.836–847 (2012).

[6] Walther, M. and Kaisser, M.: Geo-spatial Event Detection in the Twitter Stream, ECIR, pp.356–367 (2013).

[7] Lee, C.-H.: Mining Spatiotemporal Information on Microblogging Streams Using a Density-based Online Clustering Method, Expert Syst. Appl., Vol.39, No.10, pp.9623–9641 (2012).

[8] Ajao, O.; Hong, J.; Liu, W. A survey of location inference techniques on twitter. J. Inf. Sci. 2015, 41, pp.855–864.

[9] Clodoveu, Jr., A.D., Pappa, G.L., de Oliveira, D.R.R. and de Lima Arcanjo, F.: Inferring the Location of Twitter Messages Based on User Relationships, T. GIS, Vol.15, No.6, pp.735–751 (2011).

[10] L. Backstrom, J. Kleinberg, R. Kumar, and J. Novak, “Spatial variation in search engine queries,” in WWW, Beijing, China, pp. 357–366 (2008).

[11] Chang, H.-W., Lee, D., Eltaher, M. and Lee, J.: @Phillies Tweeting from Philly? Predicting

Twitter User Locations with Spatial Word Usage, Proceedings of the 2012 International Conference on Advances in Social Networks Analysis and Mining (ASONAM2012) IEEE Computer Society, pp. 111–118 (2012).

[12] 山口祐人, 伊川洋平, 天笠俊之, 北川博之, “ソーシャルメディアにおけるローカルイベントを用いたユーザ位置推定手法”, 情報処理学会論文誌データベース Vol.6 No.5 23–37 (Dec. 2013).

[13] Walther, M. and Kaisser, M.: Geo-spatial Event Detection in the Twitter Stream, ECIR, pp.356–367 (2013).

[14] Mecab, https://taku910.github.io/mecab/

[15] About Twitter’s APIs, https://help.twitter.com/en/rules-and-policies/twitter-api

[16] Tweet objects, https://developer.twitter.com/en/docs/tweets/data-dictionary/overview/tweet-object

[17] Z.Cheng, J.Caverlee,and K.Lee, “You are where you tweet: a content-based approach to geo-locating twitter users,” in ACM CIKM, Toronto, ON, Canada, 2010, pp. 759–768.

[18] Clustering, https://scikit-learn.org/stable/modules/clustering.html#clustering

[19] MongoDB, https://www.mongodb.com/

[20] Introduction to K-means Clustering, https://www.datascience.com/blog/k-means-clustering

[21] Dos Santos ADP, Wives LK, Alvares LO (2012), Location-based events detection on micro-blogs. CoRR abs/1210.4008

[22] 気象庁震度データベース検索, https://www.data.jma.go.jp/svd/eqdb/data/shindo/index.php

[23] Li, C., Sun, A. and Datta, A.: Twevent: Segment- based Event Detection from Tweets, CIKM, pp.155–164 (2012).

[24] Song, S., Li, Q., & Zheng, N. (2010). A spatiotemporal framework for related topic search in micro-blogging. In Proceedings of the 6th international conference on active media technology, Toronto, Canada.

[25] 三木 翔平,新田 直子,馬場口 登, “単語の地理的局所性の経時変化を考慮したツイートの発信位置推定”, DEIM Forum 2014, B3-1.

[26] Sadilek, A., Kautz, H.A. and Bigham, J.P.: Finding Your Friends and Following Them to Where You Are, WSDM, pp.723–732 (2012).

[27] Sakaki, T., Okazaki, M. and Matsuo, Y.: Earthquake Shakes Twitter Users: Real-time Event Detection by So- cial Sensors, WWW, pp.851–860 (2010).

[28] Yardi, S. and Boyd, D.: Tweeting from the Town Square: Measuring Geographic Local Networks, ICWSM (2010).

[29] Petrovic, S., Osborne, M., Lavrenko, V.: Streaming first story detection with application to Twitter. In: Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, NAACL-HLT 2010 (2010)

[30] Petrovic, S., Osborne, M., Lavrenko, V.: Using paraphrases for improving first story detection in news and Twitter. In: Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2012 (2012)

[31] The 5 Clustering Algorithms Data Scientists Need to Know, https://towardsdatascience.com/

[32] See, e.g., David J. Ketchen, Jr; Christopher L. Shook.: The application of cluster analysis in Strategic Management Research: An analysis and critique. Strategic Management Journal, pp.441-458 (1996).

Acknowledgement

Foremost, I would like to sincerely thank my advisor professor Prof. Sezaki Kaoru, who had taken care of me a lot since I entered school. Especially when I hesitated to choose the topic, Sezaki sensei still encourage me to select the research I want to do and give me many suggestion in writing this paper.

I also want to thank sub advisor Professor Hiroki Kobayashi, for his encouragement and insightful comments. Prof. Kobayashi usually helped me to through the examination.

At the same time, I wan to thank my sub advisor professor Prof.Sadahiro Yukio for his insightful comments and hard questions.

My sincere thanks also goes to assistant professor Masaki Ito, he provided me many ideas when I hesitated to choose my topic.

I also want to thank the secretaries of Sezaki Lab, Matsumoto Kaho and Naito Jun, who always help me in dealing with various procedures.

Last but not least, I want to thank my lab mates in Sezaki lab: Jiang Tiantian, Sun Yao, Song Chenwei, Zhang Ruichao, Yuichi Nakamura, Koki Ishida, and the OB member Umezawa Keisuke, Ben Ruktanttichoke, the people who spent with me two years and had many happy memories during my Master's course.

Appendix

A The structure of the twitter data crawled


```
{
        "_id" : ObjectId("5b3219bb6e1f9fb892e19c22"),
        "contributors" : null,
        "truncated" : false,
        "text" : "また地震だ",
        "is_quote_status" : false,
        "in_reply_to_status_id" : null,
        "id" : NumberLong("1011561492091506688"),
        "favorite_count" : 0,
        "entities" : {
                "symbols" : [ ],
                "user_mentions" : [ ],
                "hashtags" : [ ],
                "urls" : [ ]
        },
        "retweeted" : false,
        "coordinates" : null,
        "source" : "<a href=\"http://tapbots.com/tweetbot\" rel=\"nofollow\">Tweetbot for iOS</a>",
        "in_reply_to_screen_name" : null,
        "in_reply_to_user_id" : null,
        "retweet_count" : 0,
        "id_str" : "1011561492091506688",
        "favorited" : false,
        "user" : {
                "follow_request_sent" : false,
                "has_extended_profile" : true,
                "profile_use_background_image" : true,
                "default_profile_image" : false,
                "id" : NumberLong("92771546265321472O"),
                "profile_background_image_url_https" : null,
                "verified" : false,
                "translator_type" : "none",
                "profile_text_color" : "333333",
                "profile_image_url_https" : "https://pbs.twimg.com/profile_images/1004914331111194624/36urj6va_normal.jpg",
                "profile_sidebar_fill_color" : "DDEEF6",
                "entities" : {
                        "description" : {
                                "urls" : [ ]
                        }
                },
                "followers_count" : 84,
                "profile_sidebar_border_color" : "C0DEED",
                "id_str" : "92771546265321472O",
                "profile_background_color" : "F5F8FA",
                "listed_count" : 0,
                "is_translation_enabled" : false,
                "utc_offset" : null,
                "statuses_count" : 4137,

                "description" : "PUBG.MHW.OW.etc\nお墓†@wachi_ika†",
                "friends_count" : 89,
                "location" : "",
                "profile_link_color" : "1DA1F2",
                "profile_image_url" : "http://pbs.twimg.com/profile_images/1004914331111194624/36urj6va_normal.jpg",
                "following" : false,
                "geo_enabled" : false,
                "profile_banner_url" : "https://pbs.twimg.com/profile_banners/927715462653214720/1513177674",
                "profile_background_image_url" : null,
                "screen_name" : "wat_channel",
                "lang" : "ja",
                "profile_background_tile" : false,
                "favourites_count" : 171,
                "name" : "わっちゃん",
                "notifications" : false,
                "url" : null,
                "created_at" : "Tue Nov 07 01:52:57 +0000 2017",
                "contributors_enabled" : false,
                "time_zone" : null,
                "protected" : false,
                "default_profile" : true,
                "is_translator" : false
        },
        "geo" : null,
        "in_reply_to_user_id_str" : null,
        "lang" : "ja",
        "created_at" : "Tue Jun 26 10:47:08 +0000 2018",
        "in_reply_to_status_id_str" : null,
        "place" : null,
        "metadata" : {
                "iso_language_code" : "ja",
                "result_type" : "recent"
        },
        "created_datetime" : ISODate("2018-06-26T10:47:08Z")
}
```

B Tweet data dictionary

Name		Description
Id		The integer representation of the unique identifier for this Tweet.
id_str		The string representation of the unique identifier for this Tweet.
user		The user who posted this Tweet. See User data dictionary for complete list of attributes.
	id	
	name	
	screen_name	
	decsription	
	friends_count	
	followers_count	
	statuses_count	
	favourites_count	
	location	
	created_at	
text		The actual UTF-8 text of the status update.
retweeted_status		This attribute contains a representation of the original Tweet that was retweeted.
retweeted		indicates whether this Tweet has been Retweeted by the authenticating user.
retweet_count		Number of times this Tweet has been retweeted.
favorited		Indicates whether this Tweet has been liked by the authenticating user.
favorite_count		Indicates approximately how many times this Tweet has been liked by Twitter users.
coordinates		Represents the geographic location of this

		Tweet as reported by the user or client application.
entities		Entities which have been parsed out of the text of the Tweet.
	symbols	
	user_mentions	
	hashtags	
	urls	
source		Utility used to post the Tweet, as an HTML-formatted string.
lang		When present, indicates a BCP 47 language identifier corresponding to the machine-detected language of the Tweet text, or und if no language could be detected.
create_At		UTC time when this Tweet was created.
place		When present, indicates that the tweet is associated (but not necessarily originating from) a Place.
in_reply_to_screen_name		If the represented Tweet is a reply, this field will contain the screen name of the original Tweet's author.
in_reply_to_status_id		If the represented Tweet is a reply, this field will contain the integer representation of the original Tweet's author ID.
in_reply_to_status_id_str		If the represented Tweet is a reply, this field will contain the string representation of the original Tweet's author ID.