Topic-aware Scheme for Collecting Local Tweets

Carina MIWA YOSHIMURA† and Hiroyuki KITAGAWA††

† School of Systems and Information Engineering, University of Tsukuba
†† Center for Computational Sciences, University of Tsukuba
1 Chome-1 – 1 Tennodai, Tsukuba, Ibaraki 305-857
E-mail: †miwayoshimura@kde.cs.tsukuba.ac.jp, ††kitagawa@cs.tsukuba.ac.jp

Abstract Twitter, one of the world's most popular social media platforms, hosts a large and diverse amount of information that makes up a corpus of data valuable to a wide range of institutions from marketing firms, to governments. The collection of tweets can enable analysis like surveys of public opinions, marketing analysis or target analysis to users who live in a specific area. To collect useful data for a given task, the ability to capture tweets related to a specific topic sent from a specific area is needed. However, performing this kind of task on significantly sizable data sources such as the twitter stream data using just the Twitter API is a big challenge because of limitation relating to usage restrictions and lack of geotags. There is some research to collect tweets sent from a specific area using bandit algorithm, but this data collection is done massively without taking into consideration the topic of the tweets. In this work, we propose "TLV-Bandit", that collect topic-related tweets sent from a specific area based on the bandit algorithm and analyze its performance. The experimental results show that our proposed method can collect efficiently the target tweets in comparison to other methods when considering the three aspects of collecting requirements: Locality (sent from the target area), Similarity (topic-related) and Volume (number of tweets).

Key words Twitter, Social Media, Bandit Algorithm, Crawling, Location Estimation

1 Introduction

Many people regularly access a variety of social media and it has become an important part of people's everyday lives.

Twitter is one of the most used social media in the world. Since Twitter can easily publish information as a tweet from a mobile device, the tweet may include content linked to the user's location and information related to the user's local area. Therefore, it is possible to use it for disaster analysis [1], detection of events [2] and even marketing. To achieve this, collecting local tweets (target tweets) would be an interesting task.

As an approach to collecting tweets, it is conceivable to use the existing collection API(注1) provided by Twitter. However, due to Twitter API usage restrictions, the time and computer resources are limited to collect target tweets. The Twitter Streaming API, which is used to collect tweets in real time, can only acquire 1% of randomly selected tweets from all tweets. Although Twitter REST API is used to acquire past tweets, there is also a limitation in the number of times of use and the number of tweets that can be acquired at one time.

To collect tweets sent from a specific area, geotags with location information will be collected, but according to related research [3], the percentage of geotagged tweets is as small as about 2% of the total. Thus, there would be a significant probability that enough tweets cannot be collected.

Also, in order to collect the tweets of residents of a specific area it is necessary to specify the home location of the user profile. But due to privacy issues, users choose not to disclose this information or provide misleading information. In addition, Twitter has recently announced the removal of precise location tagging in tweets like latitude and longitude [4]. All together, these factors make target tweet collection more challenging.

In previous research [5], they proposed a method to collect tweets sent from a specific area using bandit algorithm. In this method, tweet collection is achieved by following local users who are likely to tweet from the target area. Also, in the work [6], the search for newly followed users was improved by using the friend relationship between users on Twitter.

Twitter is a good source to analyze individual interests and opinions. It hosts a large and diverse amount of information that makes up a corpus of data valuable to a wide range of institutions from marketing firms, to governments.

(注1) : https://developer.twitter.com/

For example, in a survey of public opinion about a news in a specific area, the extraction of the local tweets related to that news is needed.

However, the previous method has taken into account only the quantity of the collected local tweets. In other words, the tweets collection is done massively, but without taking into consideration the topic of them.

In this work, we propose "TLV-Bandit", that collect topic-related tweets sent from a specific area based on the bandit algorithm and analyze its performance.

2 Related Works

2 1 Bandit Problem and Its Algorithms

The Bandit Problem [7] is applied when there are multiple options and the distribution of data is unknown. It is a problem to look for the best option sequentially maximizing the expected reward. In order to maximize the reward, two types of actions must be used, exploration and exploitation. Exploration examines the distribution of rewards in multiple trials. At this time, the number of trials is limited and the rewards obtained from the options is unknown. In such situation, it is not convenient to choose option with less reward many times, and it is required to choose option that can get more rewards while searching for the better option. It is called exploitation. The bandit algorithm aims to maximize the accumulated reward under such a exploration-exploitation trade-off.

Bandit algorithms include ϵ-greedy algorithm [8], UCB (Upper Confidence Bound) algorithm [7], Thompson sampling algorithm [9], among others.

The ϵ-greedy algorithm is an algorithm that makes a random selection with probability ϵ, and makes a selection with the highest expectation of the past reward with probability $1 - \epsilon$.

The UCB algorithm is an algorithm that balances exploration and exploitation by selecting options with high rewards and the least selected up to the present because of its uncertainty. In this way, it is possible to decrease the exploration for options with low reward along with the number of selections.

The Thompson sampling algorithm assumes a probability distribution in advance for the rewards of each option, and updates the posterior probability distribution with the obtained rewards. It is an algorithm to select according to this a posteriori probability.

2 2 Focused data capture using Bandit Algorithm

Gisselbrecht et al. [10] proposed a method to collect tweets on a specific topic. This method uses a bandit algorithm to follow users who post many tweets on a topic. The relevancy with a specific topic is calculated from the tweets collected, and this is used as the reward. The follow user is switched according to the obtained reward. However, in our case, it is not possible to get the reward directly from the collected tweets because the problem setting in this research often does not know where the tweet originates.

Therefore, in this research, the reward is estimated by performing the source location estimation and also the computation of the similarity between the tweet and a specific topic. Then, the next follow user selection is performed using bandit algorithm according to the estimated reward.

3 Previous work

Ueda et al. [5] proposed a method to collect tweets sent from a specific area using bandit algorithm. Also, Nakagawa et al. [6] proposed a method based in the previous work using friend relationship information between users. The method proposed by Ueda et al. follows users who are likely to send tweets from the target area and collects their tweets. The ϵ-greedy algorithm is used to select the next users to follow in an environment where it is unknown which users are tweeting from the target area. The flow of processing of the ϵ-greedy method in time window t is shown below.

The given time span is split into T time windows and for each time window t, the next processes are repeated:

（1） Select users: the set of K users U_t to follow is selected via the ϵ-greedy bandit algorithm.

（2） Collect tweets: the set of tweets X_u^t from each followed user u is collected.

（3） Estimate rewards: proxy rewards $g_{u,t}$ for each user u is estimated by the estimator.

In the process 1, K users are selected at random for the initial time window. For the remaining time windows, the set of users to follow is decided by selecting one user at random with probability ϵ and retaining the user with the maximum cumulative proxy reward with probability $1 - \epsilon$. This selection is repeated K times. The cumulative proxy reward Q_u, t_{f-1} for user u from time window t_1 to t_{f-1} is calculated as:

$$Q_{u,t_{f-1}} = \begin{cases} 0 & (F_{u,t_{f-1}} = 0) \\ \frac{1}{F_{u,t_f-1}} \sum_{t=1}^{t_f-1} g_{u,t} & (otherwise) \end{cases} \quad (1)$$

where $F_{u,t_{f-1}}$ is the number of times that user u was selected and followed from time window t_1 and t_f, and $g_{u,t}$ is the proxy reward of user u in time window t, which is estimated in the process 3.

In the process 3, the reward of user u is the expected value of the number of tweets sent by u from the target area. Because of the extremely low number of tweets with geotags, the calculation of the reward using geotags is difficult. For this reason, Ueda et al. calculated the reward by estimating

the origin of the tweet without considering the geotags of the tweet. The probability $Loc(x, l)$ that tweet x is posted from the target area l is estimated. The estimation of the probability is done by the classifier and the CMN (Class Mass Normalization) [11]. Finally, the proxy reward of user u in time window t is calculated:

$$g_{u,t} = \sum_{x \in X_u^t} Loc(x, l) \tag{2}$$

4 Proposed Scheme

4.1 Description of the scheme

The objective of this work is to collect specific topic related tweets sent from the target area. We call these tweets "local similar tweets".

In the proposed scheme, we use bandit algorithm to select users to follow, considering the local similar tweets of these users as rewards. Then, the selection process is repeated every fixed time window. It is based on the assumption that users who frequently post tweets from the target area in the past can usually visit the target area and post tweets.

Using the bandit algorithm, it is capable to deal with user behavior and the limitation of the Twitter API. Furthermore, due to the lack of tweets with geotags, it is not possible to calculate the reward based on the location information of the user. Instead, we use tweet text to estimate the source location. Consequently, we can calculate the probability that the tweet is sent from the target area and also how similar it is with a specific topic.

The general process of the proposed scheme is shown in Fig 1.

In the proposed scheme, the set of follow-up candidate users, the target area, and the text related to a topic are given as input. The text could be, for example, news articles or other tweets related to the topic of interest. The process can be divided into three steps, and the final results can be obtained by repeating these steps in each time window.

(1) Select users: the set of K users U_t to follow is selected via the ϵ-greedy bandit algorithm.

(2) Collect tweets: the set of tweets X_u^t from each followed user u is collected during the time window t.

(3) Estimate rewards: proxy rewards $q_{u,t}$ for each user u es estimated based on the collected tweets.

4.2 Selection of the user

In this step, we select the users to follow in the time window t using bandit algorithm. There are many types of the bandit algorithm but in this work, we will use the ϵ-greedy algorithm [8].

In this proposed scheme, we select users randomly with probability ϵ and select users with the highest expectation of the rewards with probability $1 - \epsilon$.

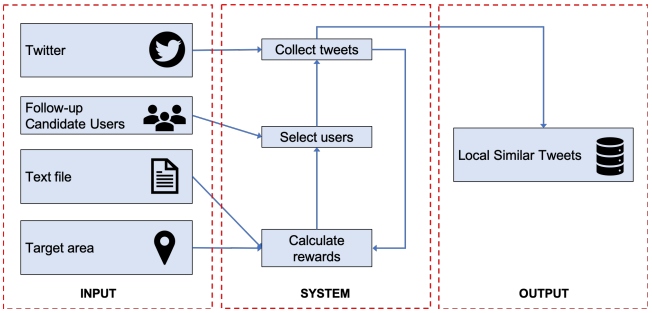


Figure 1: General process of the collecting scheme

The reward of user u in time window T could be calculated as follows:

$$Q(u, T) = \begin{cases} 0 & (F_{u,T-1} = 0) \\ \frac{1}{F_{u,T-1}} \sum_{t=0}^{T-1} q(u, t) & (F_{u,T-1} > 0) \end{cases} \tag{3}$$

where $q(u, t)$ is the proxy reward of the user u in time window t, $F_{u,T-1}$ is the number of times the user u was selected and followed.

In the proposed scheme, K users are selected at random for the initial time window. For the remaining time windows, the set of users to follow is decided by selecting one user at random with probability ϵ and retaining the user with the maximum cumulative proxy reward with probability $1 - \epsilon$. This selection is repeated K times.

4.3 Collection of the tweets

In this step, we continuously collect the tweets from the followed users in the range of time window t. The proposed scheme uses the Streaming API by setting the follow parameter and collect the tweets posted in real time from the set of users.

4.4 Estimation of the reward

When the time window t finishes, the rewards of all the followed users are estimated as follows. The reward of each user is estimated by the Locality reward $Loc(x, location)$ which is calculated by the probability that x is posted from $location$ and the Similarity reward $Sim(x, text)$ which is calculated by the cosine similarity between x and the $text$. The combination of these both rewards is defined as the reward for each user (4).

$$q(u, t) = \sum_{x \in X_u^t} Loc(x, location) \times Sim(x, text) \tag{4}$$

where X_u^t is the tweet set of user u at time window t, and x represents each tweet. This reward is used in expression 3 to select users in the step 1.

4.4.1 Estimation of the Locality Reward

In this work, Naive Bayes Classifier is used to calculate the

probability.

First, we construct a feature vector for each tweet of the tweet set X_u^t. we employ tweets with geotags to extract nouns as feature words and then construct the feature vector for each tweet according to the bag of words. Each element of the vector contains the frequency of corresponding nouns.

We use set of tweets with geotags which were previously collected as training data and we establish two classes: positive class c_x^p (tweet sent from the target area) and negative class c_x^n (tweet sent from outside of the target area).

However, when the task of classification is binomial, the distribution of the classes in the real data is considered to be largely biased toward the tweets sent from outside the target area. To fix the effect of this class imbalance, the Class Mass Normalization (CMN) [11] is used.

If there are two classes: label 0 and label 1, and the probability of classification of the node a as label 1 is $f(a)$, then in a normal classification, the node a is classified into class l_a as follows:

$$l_a = \begin{cases} \text{label 1} & (f(a) > 1 - f(a)) \\ \text{label 0} & (otherwise) \end{cases} \tag{5}$$

If the class labels 0 and 1 are highly imbalanced, CMN is used. The classification result is adjusted using CMN according to the following equation.

$$l_a = \begin{cases} \text{label 1} & (q\frac{f(a)}{\sum_a f(a)} > (1-q)\frac{1-f(a)}{\sum_a (1-f(a))}) \\ \text{label 0} & (otherwise) \end{cases} \tag{6}$$

where q is the desired proportion of the label 1.

In our context, first, at the end of time window t, for each collected tweet x, the probability p_x that x is posted from l is estimated by the Naive Bayes Classifier. Then, the CMN is used to calculate the following c_x^p and c_x^n.

$$\begin{cases} c_x^p = q\frac{p_x}{\sum_{\{x \in X^t\}} p_x} \\ c_x^n = (1-q)\frac{1-p_x}{\sum_{\{x \in X^t\}} (1-p_x)} \end{cases} \tag{7}$$

where X^t is the set of tweets collected in the time window t.

Finally, the proxy Locality Reward of the tweet x is calculated as follows:

$$Loc(x, location) = \frac{c_x^p}{c_x^p + c_x^n} \tag{8}$$

4 4. 2 Estimation of the Similarity Reward

The similarity value between the tweet and the text is computed as the cosine similarity of their vectors.

$$\text{Sim(x, text)} = \cos(\text{vec(x)}, \text{vec(text)})$$

First, we extract nouns as feature words from the text and then construct feature vector for the text and for each tweet of the tweet set X_u^t according to the bag of words. Each element of the vector contains the frequency of corresponding nouns. In this way, we can judge the topic of the text by the words it contains.

5 Experimental Evaluation

In this experiment, we evaluate if the proposed method can collect local similar tweets in comparison to other methods.

5 1 Dataset and Settings

5 1. 1 Dataset

In order to verify the effectiveness of the proposed method, we collected tweets sent from specific regions using Twitter Streaming API.

First, we collected geotagged tweets from Japan in a time range from May 26th, 2017 to June 4th, 2017. Then, we picked up the top 20,000 users who tend to tweet many tweets and we continuously collected their tweets in a time range from June 16th, 2017 to July 31th, 2017. The total number of users tweeted in that time range was 18,466 users.

Additionally, we collected 10,000 tweets with geotags from each target area and from outside the target area in Japan to use as training dataset for the location estimator. The classifier employed is Naive Bayes as explained in section 4 4. 1.

5 1. 2 Settings

We performed the same experiments in four different location in Japan: Tokyo's 23-wards, Tsukuba city, Yokohama-city and Kyoto-city. Because of the relatively high population in these areas, the performance of the scheme is affected. We estimated said effect on the popularity (shown in Table 1) by doing the experiment with another set of 100,000 geotagged tweets. These estimated values are used as the desired class proportion q.

Table 1: The relative population of target areas

Target area	Tokyo	Kyoto	Yokohama	Tsukuba
q	0.13	0.01927	0.02356	0.00251

We divided the dataset into T=267 time windows, where the duration of each time window is 4 hours. It is equivalent to 44 days.

The number of users to follow in each time window was set to $K = 1000$ and $K = 100$.

We did 20 repeated trials and determined the average metric values. The ϵ value we used were 0.3, 0.5 and 0.7.

As the topic of the text could be judged by the words it contains, we picked up two web articles in Japanese from the news archive:

- Relocation of Tsukiji Market: published in June 21th, 2017. It talks about the relocation of Tsukiji Market to

Toyosu.

- Congestion of tourist in Kyoto: published in June 14th, 2017. The news is about the concern around the overcrowded busses in Kyoto as a result of increasing tourist.

The feature vectors constructed for the articles and tweets are bag of words, where each element is a number of occurrences of the nouns in the articles. We use MeCab Library(注2) and mecab-ipadic-NEologd(注3) for the Japanese morphological analysis.

5 2 Evaluation metrics

In this experiment, we consider four evaluation metrics to evaluate the performance:

- nSimLoc: The Volume, Similarity and Locality are the weighted criteria. When this metric is high, it means we can collect more number of local similar tweets.
- nLocal: The weighted criteria are Volume and Locality. When this metric is high, we can collect more number of local tweets.
- nVol: The weighted criteria is the Volume. When this metric is high, it means we can collect more number of tweets.
- nSim: The weighted criteria are the Volume and Similarity. When this metric is high, we can collect more number of similar tweets.

Table 2: Evaluation Metric

Metric	Weighted criteria			Collection Target
	Volume	Similarity	Locality	
nSimLoc	✓	✓	✓	number of local similar tweets
nLocal	✓	-	✓	number of local tweets
nVol	✓	-	-	number of tweets
nSim	✓	✓	-	number of similar

5 2. 1 Total number of local tweets

Since many collected tweets are not geotagged, we estimate the number of tweets sent from a specific region using the collected geotagged tweets. This evaluation metric was proposed in the previous work [5]. The calculation procedure of this evaluation index is shown below.

（1） Identify the type of followed user $\hat{u}$ in the time window t. There are three types of users:

- $U_{t,A}$: users who tweeted at least one tweet with geotags in time window t.
- $U_{t,B}$: users who tweeted but his tweets do not have any geotag in time window t
- $U_{t,C}$: users who did not tweet any tweet in time window t.

（2） According to the users' type, estimate the number of local tweets collected from a followed user $\hat{u}$ in the time window t.

- $U_{t,A}$: estimate by the ratio between the number of tweets of user $\hat{u}$ with geotags of the target area $\hat{l}$ and the total number of tweets of the user $\hat{u}$.

$$nLocal(\hat{u},t) = \frac{\mid \{x \in X^t_{u,geo} \mid u = \hat{u}, l = \hat{l}\} \mid}{\mid \{x \in X^t_{u,geo} \mid u = \hat{u}\} \mid} \times \mid \{x \in X^t_u \mid u = \hat{u}\} \mid$$

where $X^t_{u,geo}$ is the set of tweets with geotags of u in time window t and X^t_u is the set of tweets of u in time window t.

- $U_{t,B}$: Since there is no geotagged tweet posted by the user $\hat{u}$, estimate using the information of $U_{t,A}$. Calculate the ratio between the total $nLocal(u,t)$ of all $U_{t,A}$ and the total number of tweets in the time window t

$$nLocal(\hat{u},t) = \frac{\sum_{u \in U_{t,A}} nLocal(u,t)}{\mid \{x \in X^t_u \mid u \in U_{t,A}\} \mid} \times \mid \{x \in X^t_u \mid u = \hat{u}\} \mid$$

- $U_{t,C}$: The number of tweet collected is 0 because no tweets have been posted within the time window t.

$$nLocal(\hat{u},t) = 0 \tag{9}$$

（3） Summarize all the estimated number of target tweets collected from all the followed users during the time window t. It is considered as the total number of local tweets.

5 2. 2 Total number of tweets

To estimate this evaluation index, we will get into account the volume of number of tweets and the following procedure is performed:

（1） Count the plain number of the collected tweet of user $\hat{u}$ in a time window t

$$nVol(\hat{u},t) = \mid \{x \in X^t_u \mid u = \hat{u}\} \mid \tag{10}$$

where X^t_u is the set of tweets of user u in time window t

（2） Summarize all the counted number of tweets collected from all the followed users during the time window t. It is considered as the total number of tweets.

5 2. 3 Total number of similar tweets

To estimate this evaluation index, the following procedure is performed:

（1） Calculate the cosine similarity between the collected tweet of user $\hat{u}$ in a time window t and the *text*

$$nSim(\hat{u},t,text) = \sum_{x \in X^t_{\hat{u}}} Sim(x,text) \tag{11}$$

where $X^t_{\hat{u}}$ is the set of tweets of $\hat{u}$ in time window t.

（2） Summarize all the estimated number of similar tweets collected from all the followed users during the time window t. It is considered as the total number of similar tweets.

(注2) : https://taku910.github.io/mecab/

(注3) : https://github.com/neologd/mecab-ipadic-neologd

5 2. 4 Total number of local similar tweets

Based on the evaluation of *nLocal* to estimate the final number of local similar tweets, the calculation procedure is shown below.

（1） Identify the case of followed user $\hat{u}$ in the time window t to calculate $PLoc(x, \hat{l})$, which is the probability that x has tweeted from $\hat{l}$ and then multiply by the similarity index of their tweets.

$$nSimLoc(\hat{u}, t, text) = \sum_{x \in X^t_{\hat{u}}} PLoc(x, \hat{l}) \times Sim(x, text) \tag{12}$$

where $PLoc(x, \hat{l})$ of user u varies according to two types of users:

（a） $U_{t,A}$: users who tweeted at least one tweet with geotags in time window t.

- If x has geotag of the target area $\hat{l}$:

$$PLoc(x, \hat{l}) = 1 \tag{13}$$

- If x has geotag of outside of target area $\hat{l}$:

$$PLoc(x, \hat{l}) = 0 \tag{14}$$

- If x has no geotag: $PLoc(x, \hat{l})$ is estimated by the ratio between the number of tweets from the target area and the number of tweets with geotags.

$$PLoc(x, \hat{l}) = \frac{\mid \{x \in X^t_{u,geo} \mid u = \hat{u}, l = \hat{l}\} \mid}{\mid \{x \in X^t_{u,geo} \mid u = \hat{u}\} \mid} \tag{15}$$

where $X^t_{u,geo}$ is the set of tweets with geotags from user u in time window t

（b） $U_{t,B}$: users who tweeted but his tweets do not have any geotag in time window t. It is estimated using the information of $U_{t,A}$. Calculate the ratio between the total $PLoc$ of $U_{t,A}$ and the total number of tweets of all the $U_{t,A}$.

$$PLoc(x, \hat{l}) = \frac{\sum_{u \in U_{t,A}} \sum_{x \in X^t_u} PLoc(x, \hat{l})}{\mid \{x \in X^t_u \mid u \in U_{t,A}\} \mid} \tag{16}$$

where X^t_u is the set of tweets of u in time window t.

（2） Summarize all the estimated number of local similar tweets collected from all the followed users during the time window t. It is considered as the total number of local similar tweets.

5 2. 5 Comparisons with Baselines

In this experiment, we compare our proposed method and the following approaches (Table 3):

- LV-Bandit : proposed method of the previous work. Uses the $\epsilon - greedy$ algorithm. The reward here is the Locality and Volume of the collected tweets.
- TV-Bandit : uses the $\epsilon - greedy$ algorithm. The reward here is the Similarity between the topic and the collected tweets and its volume.
- V-Bandit : uses the $\epsilon - greedy$ algorithm. The reward here is the plain number of tweets posted by each user.
- 1DStatistics and 3DStatistics: estimate-then-collect approaches. These approaches follow randomly Kusers in the first D days to estimate the rewards of them (Similarity, Locality and Volume) and keep following the top K users with the highest rewards until the end. We set $D = 1$ days and $D = 3$ days respectively.
- Random: follows randomly selected K users, and collects tweets from them.

Table 3: Comparison with baselines

Method	Volume	Similarity	Locality
TLV-Bandit (proposed method)	✓	✓	✓
LV-Bandit (previous method)	✓	-	✓
TV-Bandit	✓	✓	-
V-Bandit	✓	-	-
3DStatistics	✓	✓	✓
1DStatistics	✓	✓	✓
Random	-	-	-

5 3 Results and Discussions

In this section, we explain the results of the experiments by each evaluation metrics.

5 3. 1 Parameter ϵ

We analyze the effect of parameter ϵ by varying between these three values [0.3, 0.5, 0.7] setting the target area to Kyoto and K = 100 users. The news' topic is Kyoto's worry regarding the overcrowded buses resulting from the increasing number of tourists. So, it is considered as local news mainly discussed in Kyoto.

According to the results, all the schemes that use bandit algorithm collected the greatest number of local tweets when ϵ was set to 0.3. Thus, the bandit algorithm for collecting tweets from a target area, the exploitation and exploration can be balanced with lower value of ϵ.

5 3. 2 Total number of local similar tweets collected *nSimLoc*

In this experiment, we considered the Locality and the Similarity with the news as the objective of the tweets' collecting scheme.

According to the result shown in Figure 2, the method that deals with location estimation, such as “LV-Bandit”, “TLV-Bandit” can collect more tweets than others.

This News topic is about the relocation of Tsukiji market to Toyosu, which means that it is a local topic mainly discussed in Tokyo area. In Tokyo, the method “TLV-Bandit” achieves best collecting performance than others.

In areas other than Tokyo, the “LV-Bandit” has achieved the best collection of tweets. From this clue we can assume

that since there is low quantity of users who tweet about the news, it was not possible to collect tweets related to the aforementioned news stories with the "TLV-Bandit" method.

The same situation can be confirmed in the experiment done with the local news of Kyoto. The figures 3 show that the "TLV-Bandit" method has contributed to collect more target tweets in Kyoto area, while in the other areas, the methods that consider "Similarity" in the collecting scheme, are below than the methods which contemplate "Locality". We can see that from about July 15th, our proposed method exceeds the previous method. It seems that the Kyoto's biggest annual festival Gion Matsuri starts from July 1th, and the main events parade, held on July 17th and July 24th. We can assume that due to the highest amount of users who are tweeting about the news in Kyoto, the above method has contributed to increase the number of collected tweets.

6 Conclusion and Future Works

In this work, we proposed "TLV-Bandit", that collects topic-related tweets sent from a specific area based on the bandit algorithm and analyzed its performance.

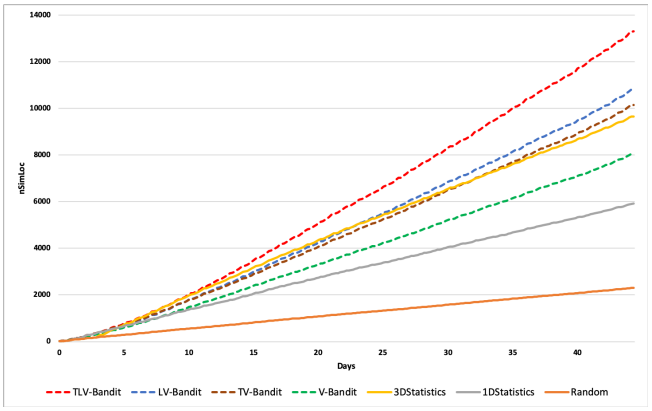
The objective of this work was to collect specific topic related tweets sent from the target area. We call these tweets "local similar tweets".

However, performing this kind of task on significantly sizable data sources such as the twitter stream data using just the Twitter API is a big challenge because of limitations relating to usage restrictions and lack of geotags.

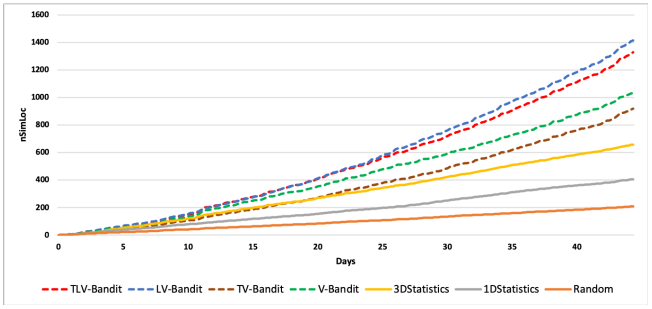
We considered performing similarity analysis and location estimation in parallel. In that way, Locality and Similarity proxy rewards are calculated and used in the bandit algorithm.

We conducted experiments with news articles of specific areas and compared proposed method and other possible approaches. The experimental results showed the advantage of using bandit algorithm on given task and our proposed method "TLV-Bandit" can collect efficiently "local similar tweets" in comparison with other methods when considering the three aspects of collecting requirements: Locality (sent from the target area), Similarity (topic-related) and Volume (number of tweets).

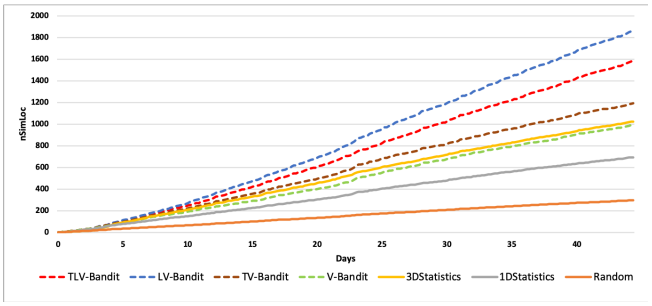
As future works, it would be interesting to do the same analysis in other areas of Japan, where the population size is different. Also, we intend to use other similarity measure algorithms in addition to cosine similarity to measure the similarity between tweets and the specific topic of interest.



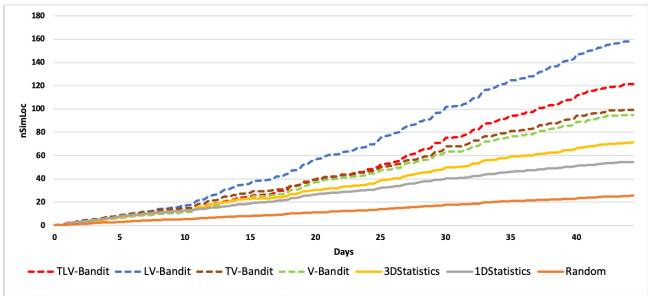
(a) Tokyo



(b) Kyoto



(c) Yokohama



(d) Tsukuba

Figure 2: News: Tsukiji Market. Tweets collected from target areas when considering Similarity and Locality; K=1000; ϵ=0.3

Acknowledgments

This work was partly supported by JSPS KAKENHI Grant Number JP19H04114.

References

[1] Sarah Vieweg, Amanda L Hughes, Kate Starbird, and Leysia Palen. Microblogging during two natural hazards events: what twitter may contribute to situational aware-

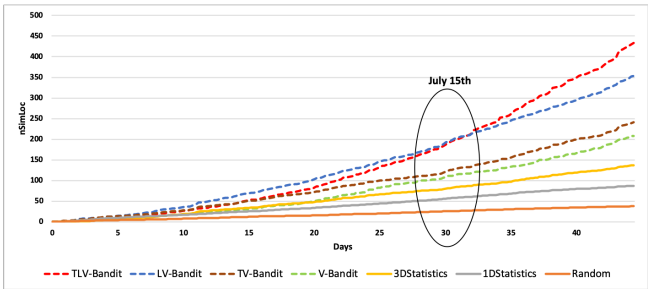


Figure 3: News: Kyoto Traffic. Tweets collected from Kyoto when considering Similarity and Locality; K=100; ϵ=0.3

ness. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 1079–1088. ACM, 2010.

[2] Takeshi Sakaki, Makoto Okazaki, and Yutaka Matsuo. Earthquake shakes twitter users: real-time event detection by social sensors. In *Proceedings of the 19th international conference on World wide web*, pages 851–860. ACM, 2010.

[3] Kalev Leetaru, Shaowen Wang, Guofeng Cao, Anand Padmanabhan, and Eric Shook. Mapping the global twitter heartbeat: The geography of twitter. *First Monday*, 18(5), 2013.

[4] Twitter Support, 2019 (accessed July 3, 2019).

[5] Saki Ueda, Yuto Yamaguchi, and Hiroyuki Kitagawa. Collecting non-geotagged local tweets via bandit algorithms. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, CIKM '17, pages 2331–2334, New York, NY, USA, 2017. ACM.

[6] Hiroyuki Kitagawa Masashi Nakagawa, Yuto Yamaguchi. Efficient collection of tweets sent from specific areas using follow relationship. Master thesis, Department of Computer Science, Graduate School of Systems and Information Engineering, 2018.

[7] Peter Auer, Nicolò Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Mach. Learn.*, 47(2-3):235–256, May 2002.

[8] Christopher John Cornish Hellaby Watkins. *Learning from Delayed Rewards*. PhD thesis, King's College, Cambridge, UK, May 1989.

[9] Shipra Agrawal and Navin Goyal. Analysis of thompson sampling for the multi-armed bandit problem. In *COLT*, pages 39–1, 2012.

[10] Thibault Gisselbrecht, Ludovic Denoyer, Patrick Gallinari, and Sylvain Lamprier. Whichstreams: A dynamic approach for focused data capture from large social media. In *Proceedings of the Ninth International Conference on Web and Social Media, ICWSM 2015, University of Oxford, Oxford, UK, May 26-29, 2015*, pages 130–139, 2015.

[11] Xiaojin Zhu, Zoubin Ghahramani, John Lafferty, et al. Semi-supervised learning using gaussian fields and harmonic functions. In *ICML*, volume 3, pages 912–919, 2003.