A Destination Prediction Method Based on Behavioral Pattern Analysis of Nonperiodic Position Logs

Fumitaka Nakahara* and Takahiro Murakami**
*Service Platforms Research Laboratories, NEC Corporation
1753, Shimonumabe, Nakahara-Ku, Kawasaki-City, Kanagawa, 211-8666, Japan,
f-nakahara@aj.jp.nec.com
** New Business Development Division, NEC BIGLOBE, Ltd.
1-11-1, Osaki, Shinagawa-ku, Tokyo, 141-0032, Japan, t-murakami@biglobe.co.jp

ABSTRACT

This paper describes a method to identify the behavioral patterns of users from nonperiodic position logs recorded by GPS phones and to predict the users' destinations by using these patterns. Besides having high accuracy, this method reduces the battery consumption of GPS devices and alleviates privacy concerns. To create a user's behavioral pattern, the method plots position logs in latitude-longitude-time space in 24 hour segments. Next, it uses the plotted data to classify the locations the user has frequently spent time or has moved through. Then, it predicts the current destination from the current location by using pre-calculated reference data made for each path location by extracting the next destination from the position logs.

In an experimental evaluation, we identified users' behavioral patterns from position logs covering a period of one year. We then evaluated the prediction method by using one week's worth of logs. The results show that the prediction accuracy of our method is more than 66%.

Keywords: Location-based service, Spatio-temporal data mining

1 INTRODUCTION

Handheld devices containing GPS receivers have become popular, and the use of location-based services has increased significantly.

NTT DOCOMO's i-concier service [1] is well known in Japan as a service delivering information based on the user's current position. This service accumulates position logs from GPS at 5-minute intervals, and the user receives traffic data, weather information, etc., based on his or her current position and settings made in advance. Other services of this type, for example, Facebook Deals [2] and foursquare [3], provide functions to search for discount coupons at local retailers depending on the user's current position.

Moreover, there are location-based services using not only current positions but also position logs of the user's travel histories. For example, the Ministry of Economy, Trade and Industry of Japan (METI) conducted studies and experiments on next-generation location-based services. These examinations focused on providing recommendations and local advertising by analyzing position logs collected every 10 minutes. Users could receive information in a timely manner because the tested service predicted each user's next destination.

Service providers have developed an interest in such timely information delivery services. In particular, they can substantially reduce their advertising costs if they can deliver ads tailored to a particular area instead of one large region.

There are now many location-base services. However, they all face the following problems.

(1) Excessive battery consumption

Battery levels fall rapidly when a device records position logs. According to a study [4], when a cellular phone is used for GPS positioning at 5-minute intervals, the battery life is about two days. In contrast, a cellular phone not used for GPS positioning typically has a battery life of about 7 days. This means users might not want location-based services that have to record position logs frequently.

(2) Lack of privacy

There are privacy concerns regarding the user's place being known by the service provider. In a survey [5] done in November 2010 in Japan, about 20% of respondents said they had had some experience with mobile applications involving location-based services; however, about 40% of respondents felt reluctant to have their positions logged by GPS. This investigation demonstrates that strong concerns exist over user privacy.

Despite the above problems, service providers would like an effective means to deliver location-based information, such as advertisements. That is, they believe that information a user receives while on the move and that is related somehow to his/her current position and next destination would be especially useful.

The above discussion shows that service providers would like to learn the frequent locations that a user stays at or passes through. They could get such information by tracking position logs and then use that knowledge to

provide services. On the other hand, users worry about battery consumption and privacy, especially when their positions are frequently logged by the provider. We shall assume that such concerns about privacy could be alleviated if the service provider only collected position logs in accordance with the user's stated intention. Moreover, we think that the battery consumption issue could be resolved if the frequency of current position readings taken by GPS were relatively low.

There are several services in which users can publish their current position logs intentionally. One of the most popular services is Twitter [6]. Almost all Twitter clients provide functions to post tweets intentionally with the tweeter's current position. These functions do, however, have a problem when it comes to predicting destinations since many users tweet only a few times a day.

It is difficult to analyze the places that the user has stayed at and predict his/ her next destination since position logs intentionally released by a particular user would likely be few in number and sparse. For example, most users of services like Twitter send messages with their current positions only a few times a day. There have been many studies on user behavioral analysis of such tracking logs, but as yet there is no method that works well with a few and sparse position logs.

2 RELATED WORK

There have been a number of studies on predicting destinations on the basis of the person trip model [7]. This model involves simple movements from a point of origin to a destination. METI of Japan carried out the Information Grand Voyage Project in FY2007-2009 [8] [9]. In [9], position logs were separated into two categories, i.e., "STOP" and "MOVE". The "STOP" logs included the user's locations and the beginning and ending times at each location. Stops were determined to occur when the user had stayed within a 500 m radius for more than 30 minutes. When the user had left the "STOP" log's location, the next destination was predicted on the basis of the transition probability between the that stop and the other stops.

Monreale et al. [10] predicted the future movements of individuals by using trajectory pattern mining [11]. Trajectory patterns summarize the behaviors of moving objects as sequences of frequently visited regions together with typical travel times. Monreale et al. assumed that people tend to follow common paths. The prediction used a decision tree named the T-pattern tree. The tree is 'learned' from the trajectories going through a certain area, and the next location of a new trajectory is predicted by finding the best matching path in the tree. The prediction needs a large number of position logs taken at short intervals in order to extract sufficient movement patterns.

All of the above studies assume they can use short interval position logs to predict destinations. But it is difficult to apply these methods to predict destinations from long nonperiodic interval logs. In the following sections, we describe our method of predicting destinations from such nonperiodic historical position logs.

3 PRELIMINARIES

3.1 Problems with Nonperiodic Position Logs

There is a difference between periodic position logs and nonperiodic ones. Periodic position logs are automatically captured in the background by the terminals. On the other hand, nonperiodic position logs are intentionally gathered by users. A representative example of a service that uses nonperiodic position logs is Twitter; Users can post tweets together with position logs.

Figures 1 and 2 show examples of periodic and nonperiodic position logs collected over the course of half a day. Figure 1 shows an example of position logs measured every 30 minutes on a map. We can guess from Figure 1 that the user stayed at *Place 1* from 3:30 until 7:00 since the locations of consecutive logs are so close together. Similarly, we can guess that the user stayed at *Place 2* from 9:00 until 11:00.

On the other hand, the map of nonperiodic position logs in Figure 2 shows only three logs at 2:00, 8:30, and 9:30, despite the fact that the user's behavior is the same as in the previous example. In this case, we cannot easily guess that the user stayed at *Place 1* or whether the user stayed at *Place2* or just went through at 9:30. This shows that nonperiodic position logs are difficult to use to identify frequently stayed at locations and to predict destinations.

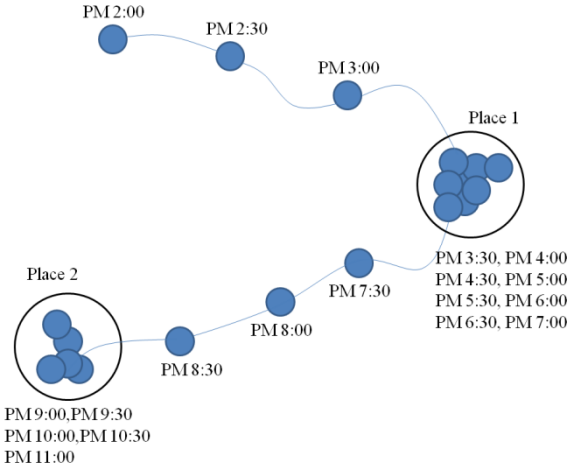

Figure 1: Trajectory of user evident from periodic position logs

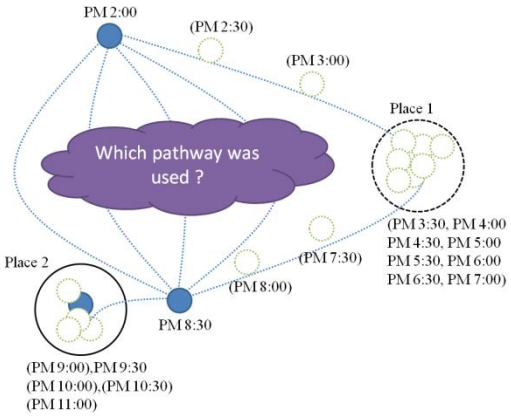

Figure 2: Trajectory of user not evident from nonperiodic position logs

3.2 Our Approach

The examples shown in Section 3.1 demonstrate that we cannot estimate the place where a user stayed from non-periodic position logs lasting only half a day. Therefore, we tried to make such estimates from nonperiodic historical position logs for a longer period. Figure 3 shows nonperiodic position logs of a user over the course of one day. Figure 4 shows those of the same user for 60 days. We can find the places where the user was but we cannot discriminate between the places s/he stayed at or went through.

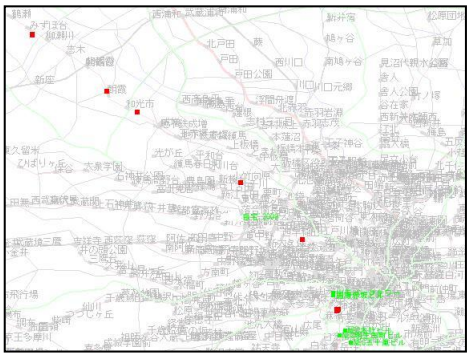

Figure 3: An example of a user's nonperiodic position logs for a day

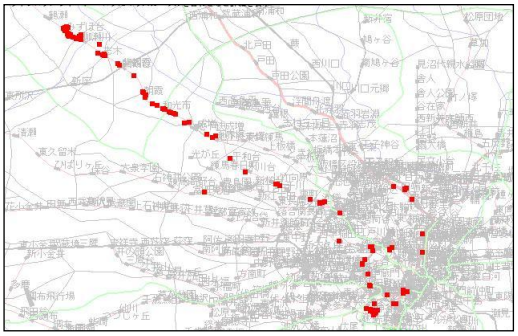

Figure 4: An example of a user's nonperiodic position logs for 60days.

Since human behavior tends to have periodic patterns (such as daily patterns), we can assume that position logs inherit such periodic patterns. Hence we made our method such that it maps nonperiodic historical position logs of a certain period accumulated by GPS to a period of 24 hours. This is simply accomplished by leaving the month and day of each log out of the accounting. From the longitude, latitude, and date data contained in the nonperiodic position logs, we retrieve longitude, latitude, and time data from the date data. Next, we plot these data on latitude, longitude, and time axes. However, we need to consider the scales of the axes, since their units are different.

Figure 5 shows the plotted position logs for one person. We can see that data sets enclosed by the solid line are parallel to the time axis and that the data sets enclosed by the dotted line are parallel to the latitude – longitude plane. Data sets enclosed by the solid line are considered to be frequently visited locations. We will call these locations "Places" in what follows. The sets enclosed by the dotted line are considered to be frequently traveled routes. We will call these "Routes".

In what follows, we describe how to extract Places by clustering the nonperiodic position logs of a certain period, and in the following section, we describe behavioral patterns by dividing them into two elements, as shown in Table 1.

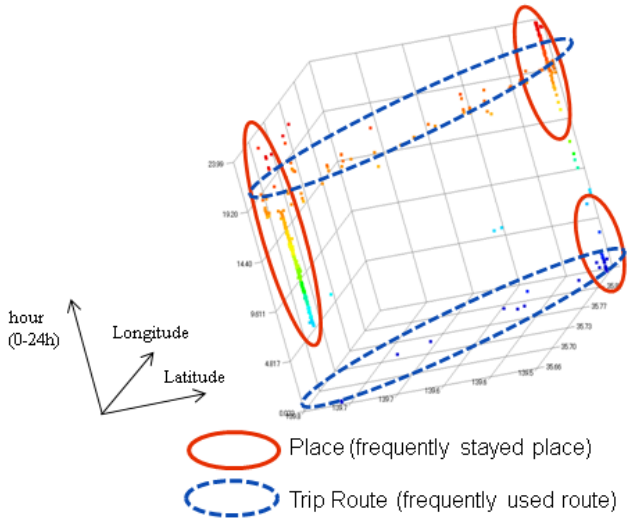

Figure 5: Plotting position logs on latitude, longitude, and time axes

Table 1: Components of Behavioral Patterns

Element	Definition
Stayed at Place	A place where a user frequently stayed
Trip Route	A route a user frequently used

4 ALGORITHM

4.1 Algorithm for Extracting "Places" from Spatio-Temporal Data

The algorithm for finding Places consists of two steps. The first step clusters historical position logs and extracts clusters of logs spatio-temporally. The second step counts the number of logs in each cluster and selects clusters whose logs exceed a certain threshold as the Places.

In the first step, we group the position logs neighboring each other. Each position log is expressed as a point in a three-dimensional Euclidian space of latitude, longitude, and time. This algorithm calculates the Euclidian distance between any pair of position logs and classifies these logs as the same cluster if the distance is less than a threshold. Note that the temporal dimension has a different property from that of the spatial dimensions. Therefore, we ought to extend the definition of distance in two ways.

One extension is to add a weight coefficient to the value in the temporal dimension. There are two reasons for doing so. The first is to convert the temporal unit into a spatial dimension. The second reason is to add a priority to the temporal dimension. If the logs are spatially close even though the logs are distributed in the temporal dimension, we should classify these logs into the same cluster. Therefore, the value in the temporal dimension is multiplied by a distance factor k not only to convert units but also to reduce the variance in the dimension (see Table 2).

The other extension is to consider that a value on the temporal dimension is a contiguous value. This means that 24 o'clock equals 0 o'clock. Therefore, when we calculate the difference between two logs, we select the smaller of the possible values. For example, when the time of position log A is 1 o'clock, and position log B is 23 o'clock, the difference is calculated as 2 hours, not 22 hours.

This system extracts clusters by using the nearest neighbor method. The variables of the position logs are latitude, longitude, and time, so the variable of position logs x_i can be defined by the following equation.

$$x_i = (x_{lat,i}, x_{lon,i}, x_{time,i}). \quad (1)$$

For example, position logs **a** and **b** can be represented as follows:

$$x_a = (x_{lat,a}, x_{lon,a}, x_{time,a}) \quad x_b = (x_{lat,b}, x_{lon,b}, x_{time,b}) . \quad (2)$$

The Euclidean distance between x_a and x_b is defined as follows:

$$d(x_a, x_b) = \min\left(\sqrt{(x_{lat,a} - x_{lat,b})^2 + (x_{lon,a} - x_{lon,b})^2 + k^2(x_{time,a} - x_{time,b})^2}, \sqrt{(x_{lat,a} - x_{lat,b})^2 + (x_{lon,a} - x_{lon,b})^2 + k^2(24 - x_{time,a} + x_{time,b})^2}\right) . \text{ where } x_{time,a} > x_{time,b} . \quad (3)$$

The cluster calculation terminates when the distance between two logs exceeds the predetermined threshold shown in Table 2.

Table 2: Weighting Values

Subject	Value
Distance factor $\boldsymbol{k}$	4.0×10^{-4}
threshold of the distance between the log $\boldsymbol{d}_{th}$	2.5×10^{-3}

In the second step, we determine the Place from the clusters. There are two kinds of clusters generated by the method: clusters indicating the places where the user frequently stays (Places), and clusters indicating where the user moves (Routes). Here, we focus on the number of logs that belong to these clusters.

Figure 6 plots the number of clusters versus the number of position logs of each cluster, as generated from 1288 real position logs of a user. The clustering procedure used the distance factor $\boldsymbol{k}$ and the distance threshold $\boldsymbol{d}_{th}$ shown in Table 2 and generated 246 clusters.

We can see from Figure 6 that many clusters are generated from a small number of position logs, and few clusters are generated from a large number of position logs. We shall consider the reasons for these features as follows. Clusters generated from a small number of position logs indicate the places through which the user moves. On the other hand, clusters generated from a large number of position logs indicate the places where the user frequently stays. The clusters where the user frequently stays are generated from the position logs indicating the user stayed at the location at a certain time. Thus, the number of logs in these clusters would be large. The clusters of routes are generated from the position logs indicating the user followed the same route at similar times. The logs for the same route are divided into several clusters. This is because the logs are not so close as they are for one cluster. If we decide on the Distance factor $\boldsymbol{k}$ to generate one cluster from all the logs in the same route, there is another problem that the logs of users who stayed at the location and the logs of users who moved near or through the location get merged into one cluster. To avoid this problem, we assume that the number of clusters versus the number of position logs of each cluster is normally distributed for classifying the clusters indicating Places the user frequently stays and the clusters indicating Routes the user moves through.

The above discussion leads us to define a cluster containing a larger number of logs than a predetermined threshold as a Place. We define the function to find Places as follows (see also Figure 7):

$$f(n) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(n-\mu)^2}{2\sigma^2}\right), \quad (4)$$

where n is the number of position logs, σ^2 is the variance of the number of logs in clusters, and μ is the average of the number of logs in clusters. The clusters are deemed to be Places when n satisfies the inequality,

$$\int_{-\infty}^{n} f(n) \geq \int_{-\infty}^{\mu+\sigma} f(n) \cong 0.8413 . \quad (5)$$

In the case of Figure 6, the method selects four high rank clusters as Places. Figure 8 plots the Places and position logs of a user on a map. The places enclosed by circles are Places.

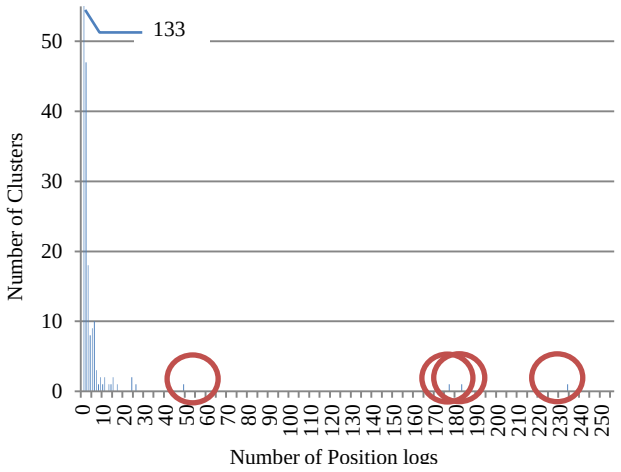

Figure 6: Number of Clusters versus the Number of Position logs

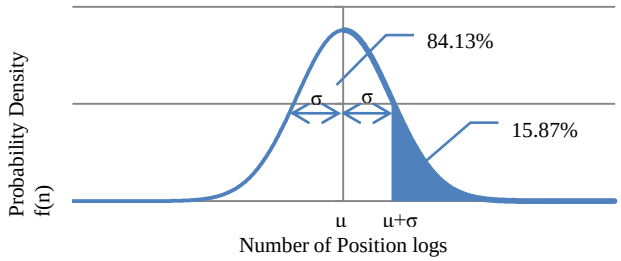

Figure 7: Function for determining Places

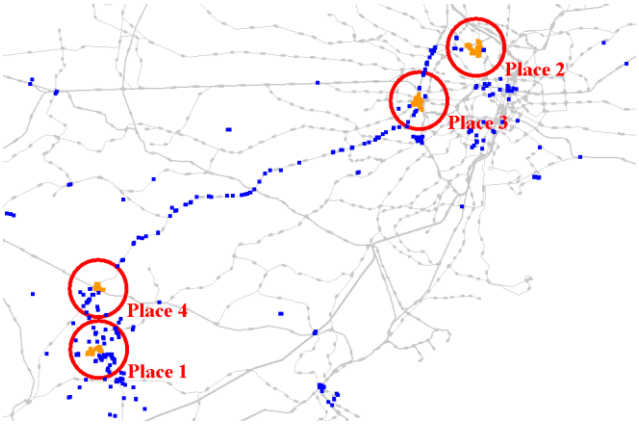

Figure 8: Example of a user's Places (red circles) on map

4.2 Destination Prediction Algorithm

The algorithm for predicting a user's destination selects the most probable Place where a user will go next from his/her current location. The destination is selected from the user's Places described in Section 4.1. The algorithm has two steps: generate a set of Destination Reference Data and calculate destination scores using the Destination Reference Data. The Destination Reference Data is composed of reliable behavioral track data generated from the user's historical position logs.

4.2.1 Generating the Destination Reference Data

The data consists of locations and Places that a user went to from an initial location. We define a Destination Reference Data, which consists of the latitude, the longitude, the time of one position log of a user and Place ID of the user's destination. The Data is as follows:

$$R = (R_{lat}, R_{lon}, R_{time}, R_{ID}). \quad (6)$$

Figure 9 shows an example of a user's position logs for a day. There are 8 sparse position logs in Figure 9. **P8** was measured at 15:00, June 24 and belongs to Place **Y**. **P2** was measured at 21:00, June 23 and belongs to Place **X**. We shall explain how to generate the Destination Reference Data with the help of Figure 9.

To determine that **P8** is Place **Y**, we check the times of previous logs. We select the previous log, **P7**; the time between **P8** and **P7** not exceeded the predetermined time threshold. If the time between two logs is long, there is a possibility that missing destinations exist between the logs. In this study, we set the threshold to 3 hours. The Destination Reference Data which includes the latitude, longitude, and time of **P7** and Place ID of **P8** is linked to **P7**. In this case, the Place ID of **P8** is **Y**, and the Destination Reference Data is as follows:

$$(R_{lat}, R_{lon}, R_{time}, R_{ID}) = (35.91, 139, 78, 12.5, \mathbf{Y}) . \quad (7)$$

Let us focus on position log, **P7**, and check the times of the previous logs. The previous log, **P6,** not exceeded the threshold time. The Destination Reference Data of **P6** and Place ID of **P8** are linked to **P6**. In this case, the Place ID of **P8** is **Y**, and Destination Reference Data is as follows:

$$(R_{lat}, R_{lon}, R_{time}, R_{ID}) = (35.92, 139, 72, 11, \mathbf{Y}). \quad (8)$$

Let us focus on position log, **P6**, and check the times of the previous logs. Looking at the previous log, **P5** we see that the time between **P6** and **P5** exceeded the predetermined threshold. In this case, we do not generate Destination Reference Data linked to **P5**.

Now let us look at position log, **P2**, and check the times of previous logs. Destination Reference Data including the latitude, longitude and time of **P1** and Place ID of **P2** are linked to **P1**. The above procedure is executed on all position logs.

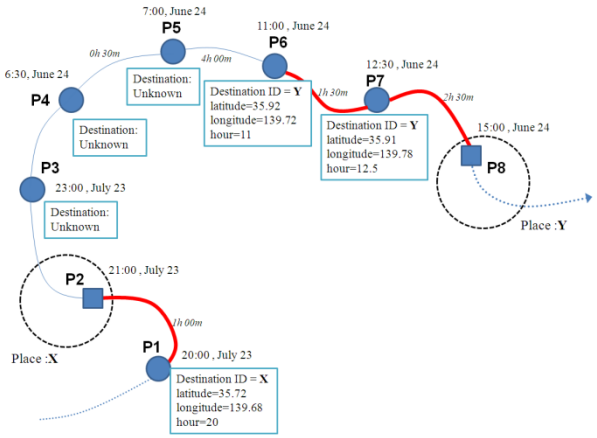

Figure 9: User's behavioral track for a day

4.2.2 Calculating the Destination Score

In this step, the algorithm selects the most probable Place by using the Destination Reference Data.

We predict user's destination by selecting among the extracted Places. Our method predicts the user's destination by comparing the similarity of the position log of a current location to each the Destination Reference Data. It calculates the probability scores of each Place. It selects all data with the same R_{ID} (Place ID) from the set of Destination Reference Data. Next, it calculates the proximity value of each selected Data and the current position log (the closer the proximity, the higher the proximity value). The proximity values of individual Place IDs are added up into a sum called the probability score. Last, the method decides predicted destination by selecting the Place ID that has the highest probability score.

We describe the steps of our method in detail below.

The current position logs contain the latitude, longitude, and time, so we define the variable **x** of the current position log as follows:

$$\mathbf{x} = (x_{lat}, x_{lon}, x_{time})^T. \quad (9)$$

The Destination Reference Data are also latitudes, longitudes, and times, so μ , the variable of the Destination Reference Data, is defined as shown below. *sp* represents the ID of the Place, and *n* is a sequential number applied to sets of Destination Reference Data with the same R_{ID}.

$$\mu_{n,sp} = \left(\mu_{lat,n,sp}, \mu_{lat,n,sp}, \mu_{time,n,sp}\right)^T. \quad (10)$$

First, we calculate the proximity value by using the current position log and one Destination Reference Data. We denote the proximity value by $f_{n,sp}(x)$. We define the score model of the proximity value beforehand. The model is that the value is large when the proximity is close. In this study, we used the normal distribution as the score model. As mentioned in Section 4.1, the times in **x** and μ are contiguous values. This means 24 o'clock equals 0 o'clock. Therefore, we should modify the normal distribution as follows:

$$f_{n,sp}(x) = \begin{cases} \frac{1}{(2\pi)^{3/2}\sqrt{\Sigma}} \exp\left(-\frac{1}{2}(\mathbf{x}-\mu_{n,sp})^T \Sigma^{-1}(\mathbf{x}-\mu_{n,sp})\right) \\ \left(\text{if } \left|x_{time}-\mu_{time,n,sp}\right| \le 12\right) \\ \frac{1}{(2\pi)^{3/2}\sqrt{\Sigma}} \exp\left(-\frac{1}{2}(\mathbf{x}-\mu_{n,sp}-t)^T \Sigma^{-1}(\mathbf{x}-\mu_{n,sp}-t)\right) \\ \left(\text{if } \left|x_{time}-\mu_{time,n,sp}\right| > 12, x_{time} > \mu_{time,n,sp}\right) \\ \frac{1}{(2\pi)^{3/2}\sqrt{\Sigma}} \exp\left(-\frac{1}{2}(\mathbf{x}-\mu_{n,sp}+t)^T \Sigma^{-1}(\mathbf{x}-\mu_{n,sp}+t)\right) \\ \left(\text{if } \left|x_{time}-\mu_{time,n,sp}\right| > 12, x_{time} \le \mu_{time,n,sp}\right) \end{cases} \quad (11)$$

Here, t and Σ are

$$t = (0,0,24)^T, \quad (12)$$

$$\Sigma = \begin{pmatrix} {\sigma_{lat}}^2 & 0 & 0 \\ 0 & {\sigma_{lon}}^2 & 0 \\ 0 & 0 & {\sigma_{time}}^2 \end{pmatrix}. \quad (13)$$

To simplify the calculation, we set the non-diagonal elements of Σ to 0. Moreover, we set σ_{time} so that one standard variation would be 2 hours. ${\sigma_{time}}^2$ is thus

$${\sigma_{time}}^2 = 4.0 \ . \quad (14)$$

Similarly, we design the other values of each diagonal element to fit one standard variation to 500 meters in the Tokyo area. The values give

$${\sigma_{lat}}^2 = 2.017 \times 10^{-5}, {\sigma_{lon}}^2 = 3.082 \times 10^{-5}. \quad (15)$$

Next, we calculate the probability score by summing the proximity values for each Place ID. The probability score is

$$\mathbf{score_{sp}}(\mathbf{x}) = \sum_{n=1}^{N} f_{n,sp}(x), \quad (16)$$

where N is the total number of the Destination Reference Data used in the calculation. Finally, we determine the destination from the highest $\mathbf{Score_{sp}}(\mathbf{x})$.

5 EVALUATION

We developed a prototype to evaluate the accuracy of the behavioral pattern analysis technique. In this section, we describe the evaluated data of the location information logs, the evaluation method, and the results.

5.1 Preparations

We used actual tweets from Twitter to evaluate this technique. As described in Section 1, Twitter users can add location information to their tweets. Such tweets include information on latitude, longitude, and dates. We collected tweets using the Twitter Streaming API from July 5, 2010 to July 4, 2011. We targeted users who had position logs in the Kanto region, including Tokyo and Yokohama, and these users made up 80% of the tweets.

Moreover, many tweets were posted automatically by bots. Tweets with location information posted by bots negatively affect the evaluations. Therefore, we needed to eliminate users who often used bots from our sample. We created a bot-filter based on white list of the Twitter client.

We analyzed the predicted destinations of 1041 users who each had more than 500 tweets containing location information. Figure 10 shows the number distribution of users versus position logs.

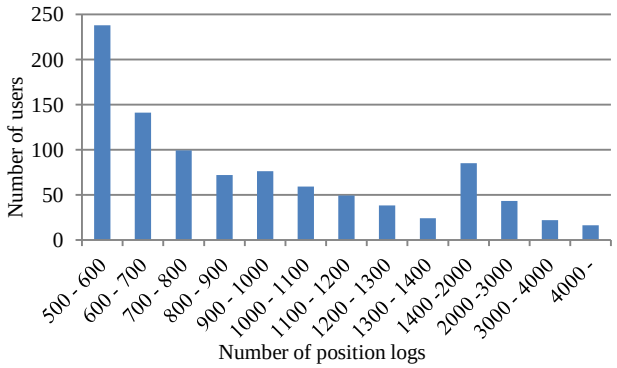

Figure 10: Number of users versus number of position logs

5.2 Evaluation of accuracy

We separated the last 7 days worth of tweets from the older tweets of individual users. We generated Destination Reference Data from the older tweets and tried to predict the destinations using the last 7 days' tweets on the basis of the Destination Reference Data. We then compared the predictions with the user's actual destinations in the last 7 days' tweets.

The actual destination is often not able to be determined from data based on nonperiodic position logs. Therefore, we assume that a correct destination is the first visited Place in a position log within 3 hours. If we don't find any Places within 3 hours from the position log, we do not use it to make the evaluation.

In addition, the number of predicted destinations depends on the user because of the different intervals between logs. If users send 10 tweets for the last 7 days, we can predict 10 destinations for each tweet. Therefore, we evaluated the accuracy with respect to the user instead of with respect to the number of predictions. However, users who are predicted only once or twice are not credible data for such an evaluation. Hence, we assessed the accuracy by looking at users who were predicted more than 3 times. In this evaluation data, it was predicted an average 6.44 times per user.

5.3 Results of the evaluation

We generated Destination Reference Data by using the Place information. As shown in Table 3, the number of users derived from older tweets (prepared as in Section 5.1) is 702. The number of users whose destinations could be predicted from the last 7 days' tweets is 541.

Of those 541, the users that could be predicted three or more times numbered 362. We evaluated the accuracy of our method on the basis of these 362 users.

Below, we analyze the data by using the evaluation method described in Section 5.1. We extracted the Places of the all users by using the algorithm described in Section 4.2.

Figure 11 shows Places for all users on a map. These Places tend to concentrate around the Yamanote Train Line in Tokyo. The gray lines indicate train routes in the Kanto area. The Places are around railway stations, and there were few logs recorded in areas far away from stations. This result is because many business districts are near stations in Japan. Therefore, using our method, we found that Places are locations where people gather.

Table 3: Results

Subject	Value
Number of users	1041
Number of users whose Destinations Reference Data were generated	702
Number of users whose destinations could be predicted	541
Number of users whose destinations could be predicted more than 3 times	362

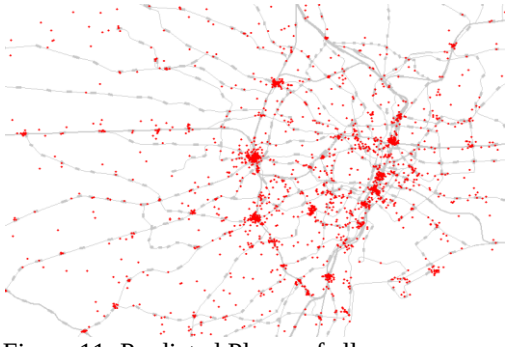

Figure 11: Predicted Places of all users on map

We evaluated the accuracy of the predicted destinations by using the same data. We assessed the accuracy of predicted destination of each 362 users individually. The results show that average accuracy of these users is 66.9%.

We compared accuracies obtained by our method and those of related work. The method described in [11] had an accuracy of 68%. Our result is comparable to theirs, despite that ours used a small number of sparse position logs.

We evaluated the accuracy in detail by plotting it against the number of Places of individual users. Figure 12 shows the users who have 2 or 3 Places have more than 80% accuracy. The accuracy increases as the Places decrease in number. The accuracy tends to fall as the number of Places increases. However, our algorithm gets an average of accuracy of 46.0% for users who have over 9 Places.

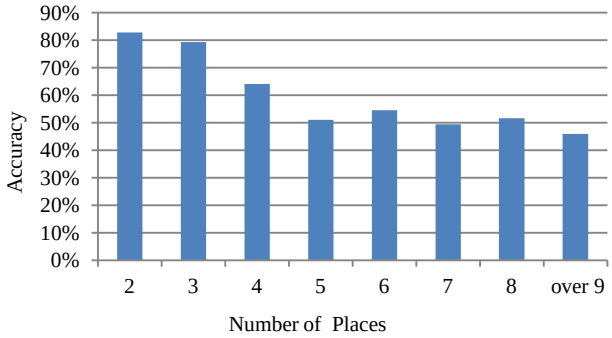

Figure 12: Accuracy versus number of Places per user

We examined the accuracy per user versus the number of logs (Figure 13 and Table 4). Although the overall average accuracy is 66.9%, according to Table 4, the accuracy for users with more than 1000 position logs is up to 70.2%.

Figure 13 reveals two major characteristics. The first characteristic is high variability. There are many users with 100% accuracy: the predictions were completely accurate for 113 users. On the other hand, accuracy was 0% for 28 users. The second characteristic involves the relation between number of logs and accuracy. Figure 13 shows that the number of users having many logs and low accuracy is small. Prediction accuracy has a large variation for users with a few position logs. According to [12], it is difficult to predict the locations of users with fewer than 1000 position logs. On the other hand, the Places of users with more position logs can be more accurately predicted according to [12], when a user has more than 1,000 position logs, their Places tend to be stably predicated.

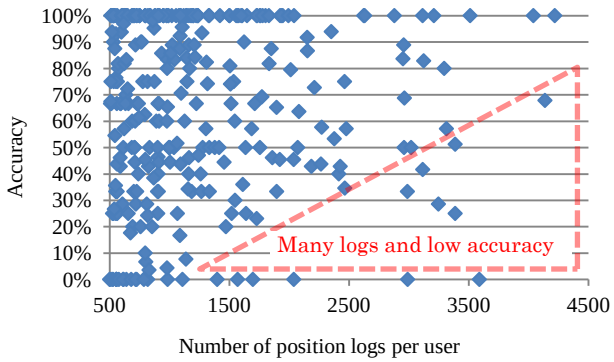

Figure 13: Accuracy versus number of logs per user

Table 4: Accuracy depending on the number of logs

Number of Logs	Accuracy Rate [%]
more than 500	66.9
more than 1000	70.2

6 CONCLUSION AND FUTURE WORK

We have shown that nonperiodic position logs are useful for behavioral pattern analysis to predict the destinations of users of location-based services. In an evaluation using real data, our method had an accuracy of 66.9% on average and 70.2% for users having more than 1000 position logs. This means our method based on nonperiodic position logs is comparable to methods based on periodically updated GPS position logs, for which there are problems such as lack of privacy and high battery consumption. Through our method, a service provider can obtain useful information about where users have stayed and their next destination.

In the future, we plan to increase accuracy by optimizing the parameters of the system. We also plan to research destination prediction using massive amounts of third-person position logs.

REFERENCES

[1] NTT DoCoMo, i-concier, http://www.nttdocomo.co.jp /english/service/imode/make/content/iconcier/

[2] Facebook, Deals, http://www.facebook.com/about /deals

[3] foursquare, https://foursquare.com/

[4] N.Yamada, Y.Isoda, M.Minami and H.Morikawa. Incremental Route Refinement for GPS-enabled Cellular Phones, The Fifth International Conference on Mobile Computing and Ubiquitous Networking (ICMU 2010), pages 87-93 (2010).

[5] MyVoice, Investigation concerning location-based service of cellular phone (in Japanese), http://myel. myvoice.jp/products/detail.php?product_id=14813, (2010).

[6] Twitter, http://twitter.com/

[7] T. Sasaki, Estimation of person trip patterns through Markov chains. Traffic Flow and Transportation, American Elsevier, pages 119–130 (1972).

[8] METI of Japan, Information Grand Voyage Project: My Life Assist Service, http://www.meti.go.jp/policy /it_policy/daikoukai/igvp/contents_en/activity09/ms09 /list/personal/ntt-docomo-inc-1.html

[9] METI of Japan, Information Grand Voyage Project: Demonstration of Model Services in Fiscal 2009 Business Report (in Japanese), http://www.meti.go.jp/policy/it_policy/daikoukai/igvp /index/h22_report/main/model01.pdf, (2010).

[10] A. Monreale, F. Pinelli, R. Trassarti, and F. Giannotti. WhereNext: a location predictor on trajectory pattern mining. In Proc. KDD'09, pages 637 – 645 (2009).

[11] F. Giannotti, N. Nanni, D. Pedreschi, and F. Pinelli, Trajectory pattern mining. In Proc. KDD'07, pages330 – 339 (2007).

[12] F. Nakahara and T. Murakami. 2011. Behavioral Pattern Analysis using noncontiguous position logs, IPSJ SIG Technical Report, Vol.2011-UBI-29 No.4 (2011) (in Japanese).