Abstract

Tourism recommender systems suggest suitable tourist spots by matching the characteristics of the tourist spots with those of the user. In this paper, we focus on an essential source of these characteristics—geotagged tweets. To solve the problem of associating geotagged tweets to tourist spots, we propose a mapping method that infers the region of a target spot on the basis of two geotagged items. The first is a geotagged tweet, which demonstrates that the tweeter was indeed at the target spot at the time the tweet was posted. We call this a “now-tweet.” The second item is a geotagged photo of the target spot, which we call a “spot-photo.” We regard these now-tweets and spot-photos as training data, and then determine the region of the tourist spot by inferring the geographical distribution of the training data. Next, we map geotagged tweets from the extracted region to the target spot. To improve the accuracy with which the tourist spot is inferred, we apply a clustering algorithm to the training data. Experimental results indicate that photo-based mapping with sophisticated training data produces the most improved performance over baseline methods. When applied to 4,559,643 geotagged tweets, our method maps them to tourist spots with an average granularity of 144.85 m.

Keywords: Geotagged tweets; Geotagged photos; Tourist spot analysis

1. Introduction

Tourism recommender systems suggest suitable tourist spots by matching the characteristics of the tourist spots with those of the user. Content-based tourism recommender systems generate recommendations based on feature vectors that describe the similarity between tourist spots and user profiles. In contrast, collaborative-based tourism recommender systems generate recommendations according to similarities between users. This user similarity is calculated in a user–spot rating matrix. To enhance the accuracy of recommender systems, it is important to improve the extraction of tourist spot characteristics and the user-spot rating matrix.
We focus on geotagged tweets, which are an essential source of such characteristics. Geotagged tweets are a textual representation of what people are thinking and doing at a given time and place. Although these live texts are very useful for directly characterizing tourist spots, it can be a problem to map them to a precise location.

A study by Lee et al. analyzed the characteristics of regions based on geotagged tweets. They divided Japan into 300 clusters based on the geographical distribution of the tweets, but the granularity of these clusters was around city-level at best. The geolocalization of tweets (i.e., estimating the location of tweets that do not contain geotags) is a popular topic of research. Although estimating the location of non-geotagged tweets is an important problem, most studies have only achieved city-level granularity. To differentiate between the subject of geotagged tweets, we attempt to map to tourist spots at a finer granularity level.

However, based solely on the text of a tweet and the location from which it was sent, it can be difficult to infer which tweets are associated with which spots. If a tweet includes the name of a tourist spot, such as “Kiyomizu-dera Temple, now,” it is easy to map the tweet to the POI, but most tweets do not explicitly include the name of a tourist spot. We could attempt to map a tweet to a POI if it originates from within some arbitrary radius of the spot. This presents the problem of defining an appropriate region around a tourist spot, because the extent and shape of such a region is not usually clear, and will obviously depend on the POI.

To solve the above-mentioned problem, we propose a mapping method that infers the region of a target spot on the basis of two geotagged items. The first is a geotagged tweet, which demonstrates that the tweeter was actually at the target spot at the time of the tweet, such as “Kiyomizu-dera Temple, now.” We call this a “now-tweet.” The second item is a geotagged photo whose object is the target spot, i.e., its title includes the name of the target spot. We call this a “spot-photo.” First, we regard the now-tweets and spot-photos associated with the target spot as training data, and then extract the region of the spot by inferring the geographical distribution of the training data. Finally, we map geotagged tweets from unknown locations to the extracted region with the target spot. In a previous study, a simple mapping method with qualitative evaluation was proposed, but quantitative evaluation has not been conducted.

The contributions of this paper are as follows:

- We propose a method that learns the region of POIs using training data from now-tweets and spot-photos. The method is based on an analysis of the geographical distribution of geotagged tweets and photos, and uses a one-class support vector machine (SVM), Graham’s scan, and density-based spatial clustering of applications with noise (DBSCAN). This allows us to map geotagged tweets from unknown locations to tourist spots based on the learned regions of POIs.
- To improve the accuracy with which we infer the POI region, we apply the DBSCAN algorithm to the training data. Our experimental results show that this process improves the inference accuracy.
- We compare the results produced using our method with those from baseline methods based on ground-truth data obtained by crowd sourcing. This comparison shows that our proposed method outperforms the baseline methods. In particular, the photo-based method exhibits the best accuracy.
- Applying our method to 4,559,643 geotagged tweets shows that the tweets can be mapped to tourist spots at an average granularity of 144.85 m.

2. Related work

This paper is related to the geolocalization of tweets and POI extractions. We introduce related work in each field, and discuss the differences between these studies and our approach.

2.1. Geolocalization of tweets

Whereas the task of tweet geolocalization aims to estimate the location of tweets that lack geotags, our task is to map geotagged tweets to tourist spots. Even when a tweet was posted from around the target spot,
it is not always clear whether the tweeter actually visited the POI. Thus, our aim is to estimate the target spot referred to by ambiguous tweets.

Schulz et al.\textsuperscript{5} used a multi-indicator approach to estimate the tweet location, and managed to infer a median accuracy of 29.6 km for 92\% of the tweets. However, finer granularity is needed to map tweets to tourist spots. Even for a relatively large POI, e.g., Tofukuji Temple in Kyoto, Japan, the extent of the region has only a 600 m radius. Kinsella et al.\textsuperscript{3} generated a geographical language model using coordinates extracted from geotagged tweets. They then modeled locations with varying granularity from zip code-level to country-level. Cheng et al.\textsuperscript{7} proposed a probabilistic framework to infer city-level locations of Twitter users based on the contents of their tweets. This approach located 51\% of Twitter users to within a granularity of 100 miles. Eisenstein et al.\textsuperscript{8} developed a method for predicting the location of users posting text content by modeling geographical areas of linguistic consistency. Their method can predict locations with an average granularity of 900 km. However, these methods have not been evaluated with finer granularity.

As stated above, most studies on tweet geolocalization have a granularity that is city-level, or zip code-level at best. To map tweets to tourist spots, finer granularity (600 m at worst) is needed. Although “only 0.7\% of tweets contained structured geolocation information”\textsuperscript{9}, we believe that geotagged tweets represent an important source of information for the direct characterization of tourist spots.

2.2. POI extraction

Many studies have attempted to extract POIs from user-generated content such as GPS trajectories, Foursquare check-ins, geotagged photos, and geotagged tweets.

Zheng et al.\textsuperscript{10} determined locations from GPS trajectories, and used a tree-based hierarchical graph to extract POIs\textsuperscript{11}. However, live texts are not always included in such data. Hence, we use geotagged tweets to map the location of tourist spots, because tweets include more live text.

Crandall et al.\textsuperscript{12} used the mean-shift clustering method to extract landmarks at which many people had taken photos from geotagged photographs posted on Flickr\textsuperscript{2}. They considered a metropolis-area scale with a granularity of 100 km and a landmark scale of 100 m. Serdyukov et al.\textsuperscript{13} also used photos posted on Flickr as the dataset for location estimation. Their method predicted the location of 7\% of photos at a granularity of 1 km, and could locate 14\% of photos to within 5 km. Hays et al.\textsuperscript{14} used features within the images themselves to predict the location of 16\% of photos with a granularity of 200 km. However, none of these studies utilize tweets.

Ye et al.\textsuperscript{15} considered check-in data obtained from Foursquare and Whrl to denote POIs, and Gao et al.\textsuperscript{16} used the check-ins posted on Foursquare to determine locations. Because check-ins are posted for pre-defined locations (latitude/longitude), one spot is associated with a unique location. Therefore, a spot is represented as a point. However, we believe that a POI should be represented as a region, because these spots have a physical extent.

Hu et al.\textsuperscript{17} proposed a spatial topic model that extracts relations among users’ activities, interests, and intended locations. They used geotagged tweets and Yelp data, but still only represented locations as a point, namely the average coordinates related to the location. Li et al.\textsuperscript{18} used check-in tweets posted via Foursquare, which include the text “I’m at [location name].” That is, a check-in tweet is equivalent to a check-in from Foursquare, and is represented as a point. In this paper, we consider tourist spots to be a region rather than a point.

3. Data analysis

Our data analysis is based on tourist spots collected from Foursquare, geotagged tweets collected from Twitter, and geotagged photos collected from Panoramio\textsuperscript{3}. In this section, we investigate the geographical distribution of tourist spots, geotagged tweets, and geotagged photos.

\textsuperscript{2}https://www.flickr.com/
\textsuperscript{3}http://www.panoramio.com/
3.1. Data set

In this analysis, we take data from within the rectangular region enclosed by the latitude/longitude coordinates (34.87069N, 135.566713E) and (35.12967N, 135.935152E). This is the Kyoto city area of Japan. In this study, we define a tourist spot as a venue visited by many people for sightseeing. Following this definition, we take venues belonging to the categories, such as Temple, and Shrine.

We have collected 148 tourist spots, 4,559,643 geotagged tweets, and 12,480 geotagged photos using the Foursquare API, Twitter Streaming API, and Panoramio API, respectively.

In this paper, $s_i \in S$ denotes spot $i$, $t_j \in T$ denotes tweet $j$, and $p_k \in P$ denotes photo $k$. The attributes of spot $s_i$ are its latitude/longitude and name. Each attribute is denoted by $s_i$.lat, $s_i$.lng, and $s_i$.name, respectively. The attributes of tweet $t_j$ are its latitude/longitude and text. These attributes are denoted by $t_j$.lat, $t_j$.lng, and $t_j$.text, respectively. The attributes of photo $p_k$ are its latitude/longitude and photo title. These attributes are denoted by $p_k$.lat, $p_k$.lng, and $p_k$.title, respectively.

3.2. Selection of target tweets for analysis

In this section, we take the Kinkaku-ji tourist spot as an example, and analyze the distribution of tweets around this location.

First, we selected tweets from within 500 m of Kinkaku-ji, which is located at (35.039273N, 135.729990E). This gave 17,368 tweets, but we excluded those with a fixed latitude/longitude published via specific location-based applications, such as Foursquare and Loctouch. These check-in tweets include a message like “I’m at [s_i.name]” or “Touch [s_i.name].” In these check-in functions, a venue is represented by a unique latitude/longitude. These were excluded because we believe a spot should be represented as a region rather than a point.

In Japan, people often tweet “[s_i.name], now” when they visit spot $s_i$. These tweets are very useful for inferring the region of a tourist spot, because users explicitly state the spot they are visiting. These now-tweets are used as training data for inferring the region of a spot in Section 4. We explain how to extract now-tweets in Section 4.2.1. In this analysis, we split the collected tweets into now-tweets and others in advance.

Further, tweets with “@username” are called mention tweets. Users publish such tweets for private users. We removed such tweets for privacy reasons.

Using the above process, we obtained 270 now-tweets referring to Kinkaku-ji, and a further 6,512 tweets from around Kinkaku-ji.

3.3. Annotation

Some tweets are actually related to the target spot, whereas others are simply sent from around the target spot. To investigate the distribution of these tweets, 100 tweets were selected at random from the 6,512 tweets and manually annotated.

For annotation, we use the five labels suggested in previous work. These are $POI_p$, which denotes “the user had already visited the target spot before the time of the post,” $POI_f$, which denotes “the user was visiting the target spot at the time of the post,” $POI_i$, which denotes “the user planned to visit the target spot after the time of the post,” $NPOI$, which denotes “this tweet is not related to the target spot,” and $Unknown$, which denotes that “this tweet cannot be determined.”

Annotation was conducted by crowd sourcing. A worker selected one of the five labels based on the target spot name and location, tweet text and location, and the time of the post. The locations of the target spot and tweet are shown on the map. Three different workers labeled each tweet.

---

4https://developer.foursquare.com/
5https://dev.twitter.com/streaming/overview
6http://www.panoramio.com/api/data/api.html
7http://tou.ch/
Fig. 1. Distribution of different tweets and photos around Kinkaku-ji.

$POI_z$ denotes that the user actually visited the target spot at the time of the post. We consider a positive tweet to be one that is labeled $POI_z$ by at least two out of three workers. A negative tweet is one labeled either $POI_p$, $POI_f$, or $NPOI$ by at least two workers. We excluded the other tweets. In the case of Kinkaku-ji, we obtained 21 positive tweets and 65 negative tweets.

3.4. Tweet distribution analysis

Figure 1(a) shows the distribution of positive and negative tweets about Kinkaku-ji. The red marker denotes the location of Kinkaku-ji, blue circles denote positive tweets, and green circles denote negative tweets. Positive tweets are distributed in and around the Kinkaku-ji region, whereas negative tweets are distributed in other regions.

Figure 1(a) exhibits a circular region of arbitrary size around Kinkaku-ji. As shown in Figure 1(a), the larger the circular region, the greater the improvement in recall, but the lower the precision. Thus, a circular region centered on the spot cannot appropriately represent the spot region.

Figure 1(b) shows the distribution of now-tweets around Kinkaku-ji. Although there are some outliers, the distribution of positive tweets seems to be predicted by extracting the high-density region of now-tweets. Furthermore, Figure 1(c) shows the distribution of spot-photos whose title includes the text “Kinkaku-ji” or “Rokuon-ji” (the official name of Kinkaku-ji). This distribution is closer to that of positive tweets than the distribution of now-tweets. We explain how to obtain the spot-photos in Section 4.2.2.

From the above observation, we believe that the region of a spot can be estimated by learning the high-density region from now-tweets or spot-photos related to the target spot. Section 4 explains how to estimate the region of a spot using now-tweets and spot-photos as training data.

4. Proposed method

This section describes our proposed method. Our approach includes the acquisition and clustering of training data, spot region extraction, and tweet prediction. We now describe the problem setting and explain each component of our method.

4.1. Problem setting

Given a spot $s_i \in S$ and a tweet $t_j \in T$, our method predicts whether $t_j$ is related to $s_i$. We take now-tweets and spot-photos of $s_i$ as training data, and estimate the region of $s_i$. Based on the estimated region, our method maps tweet $t_j$ to spot $s_i$. 
4.2. Acquisition of training data

4.2.1. Now-tweets related to the target spot

Now-tweets of the form “[si.name], now” are very useful for inferring the region of a spot, because users explicitly state the spot they are visiting. We obtain the now-tweet set \( NT_i \) for each spot \( s_i \), and use this set as training data.

The spot name \( s_i.name \) consists of plural phrases that represent spot names, such as “Rokuon-ji (Kinkaku-ji)\(^8\).” Therefore, we extract phrase sets by applying a morphological parser to the spot name \( s_i.name \). The parts of speech extracted are nouns. We extract compound phrases that connect continuing nouns. For instance, in the case of “Ninomaru Garden,” there are two nouns, Ninomaru and garden; thus, we extract “Ninomaru Garden” as a single compound phrase. The extracted phrases are represented by \( s_i.subname_l \). In the case of “Rokuon-ji (Kinkaku-ji),” we obtain \( s_i.subname_1 = “Rokuon-ji” \) and \( s_i.subname_2 = “Kinkaku-ji.” \)

Secondly, we obtain a now-tweet \( ntij \in NT_i \). This is a tweet whose text \( t_j.text \) includes “\( s_i.subname_l \), now.” For example, tweets with “Kinkaku-ji, now” or “Rokuon-ji, now” are extracted as now-tweets.

Figure 1(b) shows the location of Kinkaku-ji and its now-tweets. This can be used to find the geographical distribution of people who visited Kinkaku-ji.

4.2.2. Spot-photos of a target spot

Most geotagged photos on Panoramio included a tourist spot as their object. Such photos have titles that include the target spot names. We extract spot-photos \( p_k \) whose title \( p_k.title \) includes the target spot name \( s_i.subname_l \) extracted as described in Section 4.2.1. In the case of Kinkaku-ji, photos whose title includes “Kinkaku-ji” or “Rokuon-ji” can be extracted as spot-photos of Kinkaku-ji.

4.3. Modifying the training data using DBSCAN

Based on the distribution of the now-tweets and spot-photos obtained as described in Sections 4.2.1 and 4.2.2, we can estimate the target spot region.

However, as shown in Figures 1(b) and 1(c), some regions include many now-tweets or spot-photos, whereas others have few. The latter regions are too far from Kinkaku-ji. It is not always the case that the users had actually visited Kinkaku-ji at the time of the post. Regarding regions with many now-tweets, we can be highly confident that the region represents Kinkaku-ji, because many users tweeted in this region.

Thus, based on the notion of the wisdom of crowds, we try to improve the accuracy of region estimation by selecting training data from regions of many now-tweets. We use DBSCAN\(^{19}\), which is a density-based clustering method, to extract high-density regions.

Using DBSCAN, we extract high-density regions of now-tweets as follows:

1. Select any now-tweet \( nt_{ij} \) and mark it as visited.
2. Search the now-tweet set \( \epsilon \)-neighborhood\((nt_{ij}) \) taken from within a distance of \( \epsilon \) from tweet \( nt_{ij} \).
3. If the number of found tweets is MinPts or above, then add \( nt_{ij} \) to cluster \( c \); otherwise, label \( nt_{ij} \) as noise and repeat step 1 for other unvisited tweets.
4. Add tweets that are directly density-reachable from \( nt_{ij} \) to the same cluster \( c \).
5. If all directly density-reachable tweets have been found, repeat step 1 for all remaining unvisited tweets.

Here, \( y \) is said to be directly density-reachable from \( x \) if the following conditions are satisfied: \( y \in \epsilon \)-neighborhood\((x) \) and \( |\epsilon \)-neighborhood\((x)| \geq \text{MinPts} \), where \( \epsilon \) and \( \text{MinPts} \) are experimentally set parameters. The above process is also applied to spot-photos using DBSCAN.

Figure 1(d) shows an example of the application of DBSCAN to now-tweets from Kinkaku-ji. We set \( \epsilon = 50\text{m} \) and \( \text{MinPts} = 10 \). One cluster is formed in the region with multiple now-tweets. However, tweets that are too far from Kinkaku-ji are regarded as noise, and are excluded from the training data. In this

\(^{8}\text{Kinkaku-ji is an alias of Rokuon-ji}\)
manner, our modification of the training data improves the accuracy of spot region estimation. Note that, depending on the spots, two or more clusters can sometimes be formed.

4.4. Spot region extraction and tweet mapping

We extract a region of the target spot \( s_i \) based on the modified training data. Tweets within the extracted region are mapped to the target spot. In this study, we consider the following three methods of extracting spot regions: (a) One-class SVM, (b) Graham’s scan, (c) Direct density-reachability. We now explain how to extract spot regions and map tweets to the spot in each method.

4.4.1. One-Class SVM

SVMs\(^{20}\) are two-class classifiers. The one-class SVM (OC-SVM)\(^{21}\) is an extension to one-class problems, and is often used to estimate high-density regions using training data.

Given training data \( x_1, x_2, \ldots, x_N \), the decision plane can be represented by:

\[
f(x) = \text{sgn} \left( \sum_{i=1}^{N} \alpha_i K(x_i, x) - \rho \right)
\]

(1)

When \( f(x) = +1 \), \( x \) is classed as positive data, and when \( f(x) = -1 \), \( x \) is considered to be an outlier. Here, \( \alpha_i \) and \( \rho \) are determined through the training process. \( K(x_i, x) \) is a kernel function. We employ the radial basis function (RBF) kernel.

Based on the latitude/longitude set of the given training data, the OC-SVM learns the above decision plane. Given a tweet location \( (t_j, \text{lat}, t_j, \text{lng}) \), tweets \( t_j \) with \( f(t_j) = +1 \) are mapped to the target spot \( s_i \).

4.4.2. Graham’s scan

Graham’s scan\(^{22}\) is a method for finding the convex hull of a given point set. The convex hull is the smallest convex polygon for which each point is either on the boundary of or inside the polygon. We find the convex hull for the latitude/longitude set of the given training data using Graham’s scan. A tweet \( t_j \) existing in the polygon is mapped to the target spot \( s_i \).

4.4.3. Direct density-reachability

We apply the notion of direct density-reachability, described in Section 4.3, to the training data. If a tweet \( t_j \) is directly density-reachable from any of the training data, then \( t_j \) is mapped to the target spot \( s_i \).

5. Experiments

We evaluate our proposed method by comparing its performance with baseline methods.

5.1. Dataset

We used data on tourist spots obtained from geotagged tweets and photos. The 34 spots selected for the experiments were those with 20 or more now-tweets. Following the approach described in Section 3.2, we selected training data for each spot.

We used test datasets of tweets that had been labeled as positive or negative, following the method discussed in Section 3.3.

5.2. Metrics

We calculated the precision, recall, and F1-measure, which are widely used as evaluation metrics. These metrics can be calculated as follows: Precision = \( \frac{|C_i|}{|N_i|} \), Recall = \( \frac{|C_i|}{|C_i|} \), and F1-measure = \( \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \), where \( C_i \) denotes the positively labeled tweet set for spot \( s_i \), and \( N_i \) denotes the tweet set mapped to spot \( s_i \) by the method being evaluated.
Table 1. Comparative methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Notation</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Circular Region</td>
<td>CIRCLE</td>
<td>This method outputs the tweet set from within 500 m of the target spot.</td>
</tr>
<tr>
<td>Spot Name</td>
<td>NAME</td>
<td>This method outputs the tweet set whose text includes the target spot name.</td>
</tr>
<tr>
<td>Spot Name within Circular Region</td>
<td>CIRCLE + NAME</td>
<td>This method outputs the tweet set whose text includes the target spot name and is within a radius of 500 m of the target spot, i.e., a combination of CIRCLE and NAME.</td>
</tr>
<tr>
<td>Address</td>
<td>ADDRESS</td>
<td>We converted each of the target spots and tweet locations into an address by reverse geocoding using the Yahoo! Geocoder API. This method outputs the tweet set whose address corresponds to that of the target spot.</td>
</tr>
<tr>
<td>All tweet-based One-Class SVM</td>
<td>TWEET + OCSVM</td>
<td>This method considered all now-tweets as the training data, and outputs the positive tweet set as determined by OC-SVM based on the training data.</td>
</tr>
<tr>
<td>Tweet-based One-Class SVM with DBSCAN</td>
<td>TWEET + DBSCAN + OCSVM</td>
<td>This method takes the modified now-tweets given by DBSCAN as training data, and outputs the positive tweet set as determined by OC-SVM based on the training data.</td>
</tr>
<tr>
<td>Tweet-based Graham’s Scan with DBSCAN</td>
<td>TWEET + DBSCAN + GRAHAM</td>
<td>This method uses the modified now-tweets given by DBSCAN, and outputs the tweet set included in the convex hull found by Graham’s scan based on the training data.</td>
</tr>
<tr>
<td>Tweet-based Density Reachability with DBSCAN</td>
<td>TWEET + DBSCAN + DR</td>
<td>This method uses the modified now-tweets given by DBSCAN as training data, and outputs the tweet set that is density reachable from the training data.</td>
</tr>
<tr>
<td>Photo-based One-Class SVM with DBSCAN</td>
<td>PHOTO + DBSCAN + OCSVM</td>
<td>This method considers the modified spot-photos given by DBSCAN as training data, and outputs the positive tweet set as determined by OC-SVM based on the training data.</td>
</tr>
</tbody>
</table>

Fig. 2. Precision, recall, and F1-measure of each method.

Table 2. Number of mapped tweets and granularity of principal tourist spots.

<table>
<thead>
<tr>
<th>Spot name</th>
<th># tweets</th>
<th>Ave. distance (m)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kiyomizu-dera Temple</td>
<td>3546</td>
<td>187.63</td>
</tr>
<tr>
<td>Rokuon-ji (Kinkaku-ji)</td>
<td>1398</td>
<td>127.52</td>
</tr>
<tr>
<td>Nijo Castle (Ninomaru Garden)</td>
<td>1283</td>
<td>229.55</td>
</tr>
<tr>
<td>Fushimi Inari Taisha</td>
<td>743</td>
<td>62.54</td>
</tr>
<tr>
<td>Shrine</td>
<td>432</td>
<td>47.98</td>
</tr>
<tr>
<td>Ginkaku-ji (Jisho-ji) Temple</td>
<td>432</td>
<td>47.98</td>
</tr>
<tr>
<td>Tofukuji Temple</td>
<td>1147</td>
<td>571.42</td>
</tr>
<tr>
<td>To-ji Temple</td>
<td>867</td>
<td>130.78</td>
</tr>
<tr>
<td>Sanjusangendo Temple</td>
<td>412</td>
<td>54.83</td>
</tr>
<tr>
<td>Kodai-ji</td>
<td>462</td>
<td>83.68</td>
</tr>
</tbody>
</table>

5.3. Comparative methods

We used the methods shown in Table 1 for comparison. Here, CIRCLE, NAME, CIRCLE+NAME, and ADDRESS are the baseline methods, and TWEET+OCSVM, TWEET+DBSCAN+OCSVM, TWEET+DBSCAN+GRAHAM, TWEET+DBSCAN+DR, and PHOTO+DBSCAN+OCSVM are our proposed methods. In Section 5.4, we use $\epsilon = 120m$, MinPts = 7 in DBSCAN. These values were found to be optimal in preliminary experiments. We omit the results of the preliminary experiments on account of the space.

5.4. Comparison with baseline methods

Figure 2 compares the results of the baseline methods with those of our proposed approaches. As shown in Figure 2, PHOTO+DBSCAN+OCSVM produced the highest F1-measure. From the viewpoint of this metric, our proposed methods outperformed all baseline methods.

CIRCLE exhibited high recall but low precision. This performance is easy to understand. When we set a bigger circle, more positive tweets are included, but more negative tweets are also included. This indicates that the simple circle cannot appropriately represent the spot region.
In contrast, NAME produced high precision but low recall. If the tweet text includes the target spot name, it is probably related to the spot. However, relatively few tweets explicitly include the spot name. Therefore, we cannot expect a high recall from such tweets. Rather than giving an improvement, the combination of CIRCLE and NAME actually reduced the F1-measure.

ADDRESS was also inferior to our proposed methods. This shows that it is not always that the spot address appropriately represents the spot region, but that we must consider where people actually visit.

The F1-measure of TWEET+DBSCAN+OCSVM was 0.641, whereas that of TWEET+OCSVM was 0.562. Thus, our modification of the training data by DBSCAN appears to be effective. As shown in Figure 1(b), the now-tweets contain some noise. The use of DBSCAN helps to remove this noise and improve the accuracy.

The F1-measures of TWEET+DBSCAN+OCSVM (0.641), TWEET+DBSCAN+GRAHAM (0.567), and TWEET+DBSCAN+DR (0.575) indicate that OC-SVM is superior to Graham’s scan and density reachability in estimating spot regions.

The F1-measure of TWEET+DBSCAN+OCSVM (0.641) is somewhat lower than that of PHOTO+DBSCAN+OCSVM (0.742). This is despite there being fewer photos than tweets, suggesting that the photo-based method is superior to the tweet-based method. This is because geotagged photos have a definite target. People aim at the object when taking a photo, and upload the selected photos from many that they have taken. Therefore, geotagged photos are of high quality, and are thus useful as training data.

5.5. Analysis of granularity

We analyzed the granularity of mapped tweets for each spot. We mapped all 4,559,643 tweets in our database according to the regions estimated by PHOTO+DBSCAN+OCSVM. Table 2 lists the number of mapped tweets at each of the principal tourist spots. The most popular spot appears to be Kiyomizu-dera Temple, with 3,546 mapped tweets.

Table 2 also gives a measure of the granularity as the average distance between the spot location and each tweet. For instance, the granularity for Kinkaku-ji is 127.52 m. The finest granularity is 47.98 m for Ginkaku-ji Temple. Even in the roughest case, the granularity is only 571.42 m for Tofukuji Temple. The average granularity is 144.85 m.

Thus, we have successfully mapped tweets to tourist spots with a fine, spot-level granularity.

6. Conclusions

We have proposed a method for learning spot regions by training data from now-tweets and spot-photos. To improve accuracy with which spot regions can be inferred, we modified the training data using DBSCAN. Experimental results demonstrated that photo-based mapping with sophisticated training data gives the best performance. We also found that the accuracy of the spot region can be improved by adjusting the parameters of DBSCAN and OC-SVM. Applying our method to 4,559,643 geotagged tweets, we were able to map the tweets to tourist spots with an average spot-level granularity of 144.85 m.

In future work, we would like to analyze the differences between the tweet-based and photo-based methods. For instance, which method is better for major (or minor) spots based on the number of now-tweets? We will also examine the combination of these methods, e.g., by considering the intersection of the extracted regions. Although we analyzed average parameters in this paper, we intend to examine how they can be adjusted for specific spots. We focused on the Kyoto area in this paper. If we consider wider areas, the disambiguation problem of toponyms may occur. We must therefore consider how to overcome this problem.

In resolving the above issues, we will try to characterize tourist spots based on mapped tweets to develop a tourist spot recommender system.
Acknowledgments

This work was supported by JSPS KAKENHI Grant Number 15K12151.

References