LEARNING THE TRIP SUGGESTION FROM LANDMARK PHOTOS ON THE WEB

Rongrong Ji† Ling-Yu Duan* Jie Chen* Shuang Yang* Hongxun Yao† Tiejun Huang* Wen Gao†

*Institute of Digital Media, Peking University, Beijing, 100871, China
†Visual Intelligence Laboratory, Harbin Institute of Technology, Heilongjiang, 150001, China
{lingyu,cjie,sysyang,tjhuang,wgao}@pku.edu.cn  {rrji,yhx}@vilab.hit.edu.cn

ABSTRACT

In this paper, we introduce a novel touristic trip suggestion system to facilitate the traveling of mobile users in a given city. Given the current user location and his touristic destination, our system can suggest a shortest trip path that visits as many popular landmarks as possible. To this end, we collect geographical tagged photos from Flickr [1] and Panoramio [2], then we build a geographical graph by modeling photos as vertices and connecting them based on their geographical and visual closenesses, where visual closeness is measured by GIST [8] distances. Then, we quantize this graph by grouping nearby vertices, then trim unpopular subgraphs to output a dominant subgraph within each city, which largely improves the online recommendation efficiency while avoiding the visit of duplicated landmarks. In online suggestion, both the user location and his target location are mapped to the nearest vertices in this dominant graph respectively. Our system then suggests a trip in this dominant graph, which ensures traveling path as short as possible, while enabling the mobile user to view as many popular landmark as possible during his traveling.

Related Works: Towards trip suggestion, there are no directly related works, while most previous works aim at touristic recommendations, such as mining landmarks and their photographing views. For instance, Kori et al. [5] analyzed typical travel patterns from local blogs, where blog texts and extracted location names are phased based on [6]. Then, they ordered the appearance of the location names in the blog to indicate the landmark traveling order sequence. Typical travel patterns are subsequently found by association rules based pattern mining over the extracted location names, time periods, and types of experiences. Hao et al. [3] proposed a location overview generation approach, which first mines location-representative terms from travelogues and then uses such terms to retrieve Web images. The learnt terms and retrieved images are presented to provide an informative overview for a given location. Ji et al. [4] also proposed to mine city landmarks from blogs, and subsequently suggested personalized tourism suggestion, in which context, content, and community information are fused in both PageRank and HITS style to simulate both static and dynamic ranking of landmark photos. When a blogger uploads a tourism article, work in [4] automatically suggests the cities, landmarks, and views based on his/her tourism logs. As a successive work, Arase et al. proposed to mine people’s trips from large scale geo-tagged photos [7], which mainly relied on trip segmentation and frequent trip patterns mining.

Figure 1 outlines our proposed framework. Section 2 introduces our dominant graph mining from touristic Web photo

1. INTRODUCTION

With the ever growing popularization of mobile phone cameras, there is a great potential to utilize mobile phones for visual search and location related applications, such as landmark identification, location recognition, tourism recommendation, as well as photographing suggestion.

In this paper, we propose a novel application scenario named Touristic Trip Suggestion: Given the current user location and his target location or trip destination (both e.g. GPS coordinates), we aim to recommend a shortest trip route online, which meanwhile covers as many popular landmarks as possible. We achieve this by exploring the geographical tagged photos on the Web community, based on which we mine the best trip from the consensus of community users.

To this end, we propose a graph based trip suggestion framework. In offline learning, we first collect trip photos from Flickr [1] and Panoramio [2], then we build a geographical graph by modeling photos as vertices and connecting them based on their geographical and visual closenesses, where visual closeness is measured by GIST [8] distances. Then, we quantize this graph by grouping nearby vertices, then trim unpopular subgraphs to output a dominant subgraph within each city, which largely improves the online recommendation efficiency while avoiding the visit of duplicated landmarks. In online suggestion, both the user location and his target location are mapped to the nearest vertices in this dominant graph respectively. Our system then suggests a trip in this dominant graph, which ensures traveling path as short as possible, while enabling the mobile user to view as many popular landmark as possible during his traveling.
collections. Then, Section 3 presents the online trip suggestion strategy. Finally, Section 4 shows quantitative results in a 0.5 million geographical tagged photo collection, with comparisons to several alternative approaches.

2. MINING DOMINANT GRAPH FROM COMMUNITY TOURISTIC PHOTOS

We aim to mine a dominant trip graph from the geographical tagged, user traveling photos collected from both Flickr and Panoramio. Mapping such photos into the geographical map forms our basis to show the user traveling activities e.g. popular landmark locations. We then mine a dominant trend from this geographical distribution map, which is utilized to give online trip suggestion for mobile users.

2.1. Geographical Graph Construction

Given the geographical photo collection as \( \{P_i\}_{i=1}^{N} \), first a fully connected graph \( G \) is built to model \( \{P_i\}_{i=1}^{N} \) as vertices \( \{g_i\}_{i=1}^{N} \) and the connection strength of vertices \( g_i \) and \( g_j \) as:

\[
D(g_i, g_j) = D_{Geo}(g_i, g_j) \times D_{Visual}(g_i, g_j)
\]

(1)

Here, \( D_{Geo}(g_i, g_j) \) denotes their geographical distance on the map measured by the L2 distance of their (latitude, longitude) vectors; \( D_{Visual}(g_i, g_j) \) is their visual similarity measured by their GIST descriptor distance [8].

Such combination is to incorporate the visual similarity to refine the imprecise geographical tagging, as well as to highlight the most popular photos (making them more dense in the subsequent mining). It is reasonably assumed that the amount of the amount of visually similar (or near-duplicated) photos within a given geographical region may reveal its popularity.

2.2. Dominant Trip Graph Mining

In our subsequent trip suggestion, the best matched touristic trip is achieved by searching the shortest path in this dominant subgraph. We consider two issues in mining a dominant trip graph from this geographical graph:

- **Efficiency**: The online recommendation complexity largely relates to the number of graph vertices. Therefore, this search is inefficient in the original graph which contains, for instance, one million vertices.
- **Robustness**: In addition, we want to filter out unpopular locations before making touristic trip planning to improve the trip suggestion robustness.

**Graph Quantization**: Our first step is to reduce the scale of vertices by quantizing the original graph. Our quantization aims to reduced graph \( G \) by conducting k-means clustering as shown in Algorithm 1. As a result, a quantized graph \( G_{Quantized} = \{g'_i\}_{i=1}^{K} \) (K nodes) is output for the subsequent graph trimming and shortest path mining.

**Popularity based Graph Reduction**: We further reduce the quantized graph \( G_{Quantized} \) via cutting vertices based on their popularity. This popularity comes from:

- The geographical distribution density of photos within this vertex, whose higher density indicates that the quantized vertex comes from an attractive region. Such vertex may be subsequently applied to suggest the traveling trip online.
- Whether those photos falling into this quantized vertex is visually similar, since it is commonsense that popular landmarks tend to produce near-duplicated views of the landmark location in tourism photographing.

To fulfill above two criteria, we propose the following geographical density estimation with visual rectification as:

\[
Popularity(g'_i) = \sum_{Quan(g_j) = g'_j} \sum_{Quan(g_k) = g'_i} \exp(-D(g_j, g_k))
\]

(2)

Therefore, a vertex in \( G_{Quantized} \) that contains more visually similar photos is more popular. Then, we filter out unpopular vertices in \( G_{Quantized} \) by only retaining vertices that:

\[
G_{Dominant} = \{g'_i | Popularity(g'_i) > T \}
\]

(3)

![Fig. 1. The proposed trip suggestion framework for mobile users from Web touristic photo collections.](image)
Algorithm 2: Finding the shortest path from \(g'_{\text{Current}}\) to \(g'_{\text{Target}}\) based on the Dijkstra search [9] on \(G_{\text{Dominant}}\)

1. **Input**: The locations of the user and his or her target as \(L_{\text{Current}}\) and \(L_{\text{Target}}\) respectively;
2. **Output**: The shortest path from \(L_{\text{Current}}\) to \(L_{\text{Target}}\);
3. **Initialization**: match both \(L_{\text{Current}}\) and \(L_{\text{Target}}\) to their nearest vertices in \(G_{\text{Dominant}}\), denoted as \(g'_{\text{Current}}\) and \(g'_{\text{Target}}\);
4. while \(\{\text{Open}\} \) is not empty \{do
5. Traverse the shortest nodes to \(g'_{\text{Current}}\) that is unchecked, put it into \(\text{Open}\);
6. Find the closest node \(g'_j\) in \(\text{Open}\), find its connected nodes \(\{g'_i\}\), put \(g'_j\) into \(\text{Close}\);
7. Traverse each \(g'_j\) in \(\{g'_i\}\), calculate their distances to \(g'_{\text{Current}}\) as \(\{\text{Distance}\}\), push \(g'_j\) into \(\text{Open}\);
8. end

3. **ONLINE TRIP SUGGESTION**

**Trip Suggestion as Shortest Path Learning**: In online scenario, given the user location and his target location, we suggest the shortest path from \(L_{\text{Current}}\) to \(L_{\text{Target}}\) over the dominant trip graph \(G_{\text{Dominant}}\), which covers as many popular landmarks as possible. Our trip recommendation is conducted with two consecutive steps as shown in Algorithm 2.

While searching over the original graph \(G\) the shortest path would yield even shorter path, we still prefer to search in the quantized graph \(G_{\text{Dominant}}\) due to:

- **Popularity**: We aim to suggest the user to travel popular landmarks in his trip. However, this is not guaranteed using the shortest path search in the original graph.
- **De-Duplicated**: We also aim to avoid the trip that suggests the mobile user to have multiple stops in a given landmark. This may be unavoidable in the original graph \(G\), since near-duplicated photos are geographical concurrent, that would be typically included in forming the shortest path.

4. **EXPERIMENTAL RESULTS**

**Data Collections**: We have collected over 0.5 million geotagged touristic photos from both Flickr [1] and Panoramio [2]. This dataset covers typical areas including Beijing and New York City. To validate our approach not only for popular landmarks but also general locations, from the geographical map of each city, we selected the top 10 most popular landmarks and 10 random locations, we then simulate each pairwise locations including an initial user location and a target location, respectively. To obtain the ground truth, we ask a group of 10 volunteers who has rich traveling experiences (or serve as local residents) in this city. Each volunteer is asked to label the best trip route from the starting point (the simulated user location) to the target point (the simulated target location). We then manually pick up the most dominant route from all labeled routes for each <start location, target location> pair as its ground truth label.

**Performance Evaluation**: We measure the similarity between our suggested trip (a set of graph vertices) to the ground truth trip for each <start location, target location> pair based on the Dynamic Time Warping (DTW) [10] distance. DTW finds an optimal matching between two given sequences that are “warped” non-linearly in the time dimension. Similarly, we measure the distance of two trip paths \(g_{\text{suggest}}=<g_0, g_1, g_2, ..., g_n>\) and \(g_{\text{labeled}}=<g'_0, g'_1, g'_2, ..., g'_n>\) as:

\[
\text{DTW}(g_{\text{suggest}}, g_{\text{labeled}}) = \min \left\{ \begin{array}{l} \text{DTW}(g_{\text{suggest}}, \text{Tail}(g_{\text{labeled}})) \\ \text{DTW}(\text{Tail}(g_{\text{suggest}}), g_{\text{labeled}}) \\ \text{DTW}(\text{Tail}(g_{\text{suggest}}), \text{Tail}(g_{\text{labeled}})) \end{array} \right\}
\]

\[
||g_i - g'_j||\text{ can be the L2 distance. Similar to [10], we denote the tail of each sequence as Tail}(g) = <g_1, ..., g_m> \text{ by removing its first element. We have DTW}(< >, < >) = 0 \text{ and DTW}(g_{\text{suggest}}, < >) = \text{DTW}(< >, g_{\text{labeled}}) = \infty. \text{ Optimization is performed using a dynamic programming.}
\]

**Comparison Baselines and Quantitative Results**: To demonstrate our effectiveness, we provide the quantitative comparisons in Figure 3 using three following baselines:

- **Without Visual Diversity Embedding**, which ignores the visual diversity in graph quantization, hence the photo vertices are only quantized based on their geographical distances. Therefore, it is not guaranteed that our suggested trip can visit landmark locations. Instead, many parts of the route without visual diversity embedding are not commonsense comparing with the human labeling (Figure 2). Figure 3 further shows that it quantitatively performs worse than our final approach.

- **Directly Nearest Path Search**, which directly computes the nearest path from the initial geographical tagged graph without quantization. In such case, many noisy graph vertices (unpopular locations) would be also included into the suggested trip, which degenerates the possibility that the suggested trips approximate the manually labeled ones. This is quantitative proven in Figure 3.

- **Weighted Nearest Path Search**, which computes the weighted nearest path from our initial geographical tagged graph, plus the popularity of each vertex measured by their density in Section 2. This alternative approach introduces noises as it is done over the entire graph, rather than the dominant subgraph, which may incur less popular places when the density estimation is not well-tuned. Figure 3 shows that, the quantitative performance is degenerated. Furthermore, since the time complexity using Dijkstra algorithm is \(O(n^2)\), it is unacceptable in the original graph that contains 0.2 to 0.3 million vertices.
Fig. 2. The visualized examples of the recommended trips and the dominant photos on these trips. Blue: Ground truth; Black: Search in original graph without visual; Pink: Search in original graph with visual; Green: Search in dominant subgraph without visual; Yellow: Our approach.

Fig. 3. The quantitative evaluations of our proposed trip suggestion approach with comparisons to baseline approaches.

**Trip Visualization:** Figure 2 further shows a group of exemplar recommendation trips via the Google earth API, containing four typical trips in Beijing and New York. In each subfigure, we show the recommendation trip on the left, and visualize photos within their dominant nodes on the right.

In online suggestion, the only upstream transmission from a mobile user to the remote server is the pair of <start location, target location>, hence it’s almost real time. On the other hand, computing suggested trips on the server is also efficient, in the case of using the dominant graph.

5. CONCLUSION

In this paper, we propose an online trip suggestion system to facilitate the trip planning of mobile users. Our main idea is to learn the shortest trip route from the Web, more especially, from the geo-tagged touristic photos collections crawled from the Web community. To this end, given the user location and his target location, we model the trip suggestion as the problem of selecting the shortest path to minimize the traveling cost, meanwhile enabling the user to see as many popular landmarks as possible. To ensure efficient and robust trip suggestion computing, we further include both visual and geographical diversities to quantize the photo graph. The online suggestion is an efficient shortest path search process, which is implemented using Dijkstra search [9] algorithm. We have conducted extensive experiments over 0.5 million photos collected in Beijing and New York from Flickr [1] and Panomira [2]. Superior performances are reported comparing with several alternative baselines.

6. ACKNOWLEDGEMENTS

This work was supported by the National Basic Research Program of China under contract no. 2009CB320902, in part by grants from the Chinese National Nature Science Foundation under contract no. 60902057 and 61071180, and in part by the CADAL Project Program.

7. REFERENCES

[1] www.flickr.com