Analyzing tourist destinations in Belgrade using geotagged photos from Flickr

Vera Milosavljević
vera.milosavljevic@famnit.upr.si
University of Primorska
Faculty of Mathematics,
Natural Sciences and Information Technologies
Department of Mathematics,
SI-6000 Koper, Slovenia

Dejan Paliska
dejan.paliska@fts.upr.si
University of Primorska
Faculty of Tourism
Department for Sustainable
Destination Development
SI-6000 Koper, Slovenia

ABSTRACT

This research aims to analyze tourist destinations in Belgrade by defining trajectories of movement of the users of platform Flickr using geotagged photos on Flickr. We defined tourist movements and used generalization techniques to identify the main tourists locations. We applied several techniques to identify frequently visited locations and predict next possible tourist spots. Our findings provide insights into popular travel patterns and suggest potential areas for tourism development.

KEYWORDS

Tourism data analysis, active user bias, geotagged photo, DBSCAN clustering, trajectory generalization

1 INTRODUCTION

Tourism is a major contributor to the global economy, offering economic benefits and fostering cultural exchange. With the growth of social media and mobile phones usage, tourists now document their journeys through geotagged photos, sharing their experiences with a wide audience. Flickr, a popular photo-sharing platform, contains a wide repository of such geotagged images and publicly aveilable API which makes it a good choice for our analysis. The users of the application are not rarely professional photographers and their focus lies in architecture, nature and country's most beautiful destinations.

Analyzing these photos provides valuable insights into tourists' behavior at destinations, and particularly into their space-time patterns. Traditional tourism data often relies on surveys and official records, which can be bias and limited in scope. In contrast, social media data offers real-time, user-generated information that reflects actual tourist activities and interests. Flickr has many active users who contribute to the platform daily.

In this paper, we focused on Belgrade as a popular tourist destination, but the insights this paper provides can be applicable to any location that has a rich dataset of photos on the platform Flickr.

2 OBJECTIVES

This research has the following objectives:

(1) Collecting geotagged photos from Flickr by using the publicly available Flick API.
(2) Data preprocessing. Defining the set of variables that are important for further analysis.
(3) Generating trajectories of movement by using geo-coordinates for each user.
(4) Trajectory analysis and visualisation for better understanding of the movements.
(5) Clustering, aggregation and generalisation of trajectories.
(6) Prediction of the next tourist movement based on several different modeling tecniques.

3 SIGNIFICANCE

Understanding the location preferences of visitors is crucial for local tourism organizations, travel agencies, and other stakeholders involved in destination development. Information about local attractions, visitor mobility, and intradestiation movement patterns can help with strategic planning, improve destination marketing, and enhance connectivity between at- tractions. By using modern technology and user-generated content, such as geotagged photos from social media, re- searchers can better understand visitor patterns and behaviors. In Belgrade, the relationships between tourist destinations and visitor mobility are under-researched, thus this study aims to address these gaps by analyzing data from Flickr platform. This will allow us to identify tourists POIs (clusters), and their mobility patterns at destinations. Additionally, while Flickr data could also be utilized to analyze tourists' emotions and destination image, these aspects are beyond the scope of this study. This research contributes to the academic field by demonstrating the application of data mining techniques in tourism analytics.

4 STRUCTURE

The rest of this paper is structured as follows: Section V details the methodology, including data collection, preprocessing, clustering, aggregation and generalisation, and the prediction techniques used. Section VI presents the results of our analysis, highlighting key findings and patterns. Section VII concludes the paper and suggests directions for future research.

5 METHODOLOGY

5.1 Tools and Technologies

All data processing and analysis were done using Python programming language, using some of the many data science libraries. The following tools were used:

- Jupyter Notebook: Computing environment that was used for the development and documentation of the code along with miniconda installer and command line.

- Pandas: Used for data manipulation and analysis.
- Numpy: For numerical computations.
- Scikit-learn: Applied for clustering and other machine learning tasks.
- Geopandas: Used for geographic data processing and visualization.
- MovingPandas: Used for spatial analysis and manipulation of geospatial data.
- Mlxtend: Employed for the implementation of the Apriori algorithm and association rule mining.
- SeqMining: Applied for sequence mining to find frequent sequences.
- Matplotlib: Used for data visualization.

5.2 Data Collection

We collected 31019 geotagged photos from 1233 different users from the Flickr application. We used the Flickr public API which was granted by a unique key after becoming a user of the Flickr app. The dataset was focused on the territory of Belgrade, by using the tag "belgrade". Each photo's metadata, including the geocoordinates and timestamps, was extracted and used for further analysis.

5.3 Data Preprocessing

The preprocessing step involved cleaning and structuring the collected data to make it suitable for analysis. We defined a variable ownerID, which represents one user in one day. We defined a variable locID which represents different locations for each user. Algorithm for defining locID is shown as Algorithm 1.

The motivation for defining this variable is because we wanted to tackle the problem of an active user: a situation where one user generates many photos of the same location. We wanted to treat these photos as a single group, with the same value of locID variable. This would ensure that our analysis would be less bias.

The reduced dataset had a structure as shown in the Fig. 1 . We dropped rows which were out of boundaries of Belgrade. The rows included must have longitude variable value between 20.35 and 20.65 and latitude variable value between 44.7 and 45.1 to be considered as a photo captured on the territory of Belgrade. The final reduced dataset had 20723 rows from 550 users.

	id	datetaken	latitude	longitude	owner	tags	location	realname	year	hour	minute	second	dateonly	owner_ID	tdiff	distance	loc_ID
0	6108776710	2011-06-24 12:26:32	44.819289	20.463452	10018259@N04	statue bronze serbia torso d200 belgrade beogr...	Sydney, Australia	Dan Wiklund	2011	12	26	32	2011-06-24	3	0.000000	0.000000	1
1	6220553571	2011-06-24 15:00:30	44.818254	20.455727	10018259@N04	man stairs serbia christian priest d200 belgra...	Sydney, Australia	Dan Wiklund	2011	15	0	30	2011-06-24	3	153.966667	0.620778	2
2	6244571806	2011-06-25 14:28:26	44.797878	20.468398	10018259@N04	building church cathedral serbia religion chri...	Sydney, Australia	Dan Wiklund	2011	14	28	26	2011-06-25	4	0.000000	0.000000	1
3	6059780964	2011-06-25 20:29:37	44.822516	20.449547	10018259@N04	street sunset people orange love loving cozy k...	Sydney, Australia	Dan Wiklund	2011	20	29	37	2011-06-25	4	361.183333	3.120701	2
4	7164418755	2011-06-25 21:05:18	44.818528	20.455749	10018259@N04	street longexposure flowers night booth citysc...	Sydney, Australia	Dan Wiklund	2011	21	5	18	2011-06-25	4	35.683333	0.660990	3
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
20718	7923053300	2012-08-12 18:48:32	44.830613	20.453023	9950018@N07	serbia victor winner belgrade sieger beograd b...	NaN	NaN	2012	18	48	32	2012-08-12	4919	9.016667	0.000000	1
20719	7923053630	2012-08-12 18:53:35	44.830613	20.453023	9950018@N07	serbia victor winner belgrade sieger beograd b...	NaN	NaN	2012	18	53	35	2012-08-12	4919	5.050000	0.000000	1
20720	7923056438	2012-08-12 19:00:11	44.830613	20.453023	9950018@N07	serbia victor winner belgrade sieger beograd b...	NaN	NaN	2012	19	0	11	2012-08-12	4919	6.600000	0.000000	1
20721	41718649222	2018-04-21 17:43:44	44.806640	20.442370	9991563@N06	blackwhite canon 450d eos monument socialist m...	NaN	NaN	2018	17	43	44	2018-04-21	4925	0.000000	0.000000	1
20722	27890740008	2018-04-21 17:44:30	44.806678	20.442348	9991563@N06	monument architecture socialist	NaN	NaN	2018	17	44	30	2018-04-21	4925	0.766667	0.004573	2

Figure 1: Reduced dataset

Data: database with columns 'owner_ID' and 'distance'
Result: database with 'loc_ID' column assigned to every row defining the location group

```
/* Define thresholds                                          */
dist_threshold = 0.0005 cumulative_dist_threshold = 0.001
/* Set initial loc_ID for each row                            */
final_database["loc_ID"] = 1
/* Group data by owner_ID and list distances                  */
owner_ID_map = group database by owner_ID and
  aggregate distances;
/* Initialize row index to zero                               */
i = 0;
foreach key in owner_ID_map do
    prev = 1 /* Initial loc_ID for current owner_ID
       */
    cumulative_dist = 0 /* Cumulative distance for
       current loc_ID                                         */
    foreach list_value in owner_ID_map[key] do
        cumulative_dist = cumulative_dist + list_value ;
         /* Accumulate distance */
        if list_value > dist_threshold or cumulative_dist >
         cumulative_dist_threshold then
            prev = prev + 1
            /* Increment loc_ID                               */
            cumulative_dist = 0
            /* Reset cumulative distance                      */
        end
        database.loc[i, "loc_ID"] = prev
        /* Update loc_ID for current row                      */
        i = i + 1
        /* Move to next row                                   */
    end
end
```

Algorithm 1: loc_ID Algorithm

5.4 Clustering

Various clustering techniques were explored to identify clusters of frequently visited locations. We experimented with DBSCAN, HDBSCAN, and OPTICS algorithms. The centroids of the clusters were identified using mean and median methods. We created visualizations to show the clustering results, trajectories, and connections between clusters.

5.5 Generalization and aggregation

Generalization techniques were applied using the Geopandas library and Douglas-Peucker Generalizer [5]. Different threshold parameters for tolerance were tested to observe the effects on generalization. A comparison was made between Douglas-Peucker Generalizer and Top-Down Time Ratio Generalizer. We defined trajectory collection using the MovingPandas [3] library which is a collection of all trajectories for each OwnerID. We used this collection as an imput parameter for DouglasPeuckerGeneralizer algorithm also defined in MovingPandas library. Fig. 2 shows the

trajectories for each OwnerID. We used TrajectoryCollectionAggregator [4] from MovingPandas library to get significant points and flows between them. The results will be show in the section Results.

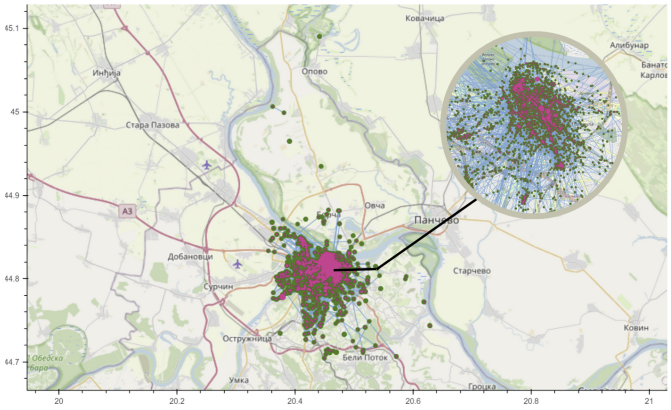

Figure 2: Trajectories for each OwnerID

5.6 Predictive Modeling

In the modeling prediction, our approach was to predict the next cluster in our spatial data analysis based on the set of visited clusters. The clustering technique that we choose was HDBSCAN because we wanted to focus on smaller clusters of tourist destinations within the city. Initially, we defined individual paths for each owner based on their clusters. We excluded outliers marked as -1 (noise) and grouped the remaining data by owner ID. Additionally, we defined a DataFrame with each owner's ID and their corresponding cluster paths. After computing the unique sequence of clusters for each owner, we generated trigrams [2] to identify recurring patterns within the paths. We defined the transition matrix and we normalized transition counts to derive probabilities of transitioning from one cluster to another. Also, we developed functions for predicting the next cluster using Markov chains, Monte Carlo simulations, and association rules derived from Apriori analysis.

6 RESULTS

6.1 Clustering

Here we will present the visualisation of the results of the clustering using HDBSCAN algorithm and median method for calculating the centroid of each cluster. Each different color in Fig. 3 represents different cluster. We calculated the number of transitions between each cluster (If one person visited cluster 1 and than after that they visited cluster 3, we would count that as 1 transition between clusters 1 and 3). Fig. 4 shows the connections between clusters. Purple points represent the centroids of the clusters, thickness of the line and the number represent how many connections are between two clusters..

6.2 Generalization

The Fig. 5 shows different thresholds for generalization by DouglasPeckuer. By analyzing these input parameters, we opted for

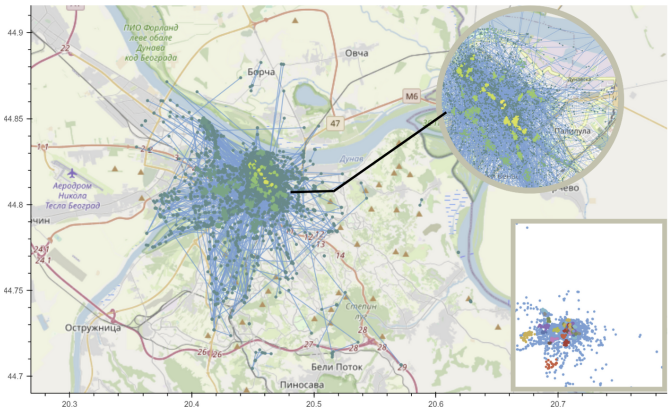

Figure 3: Clustering

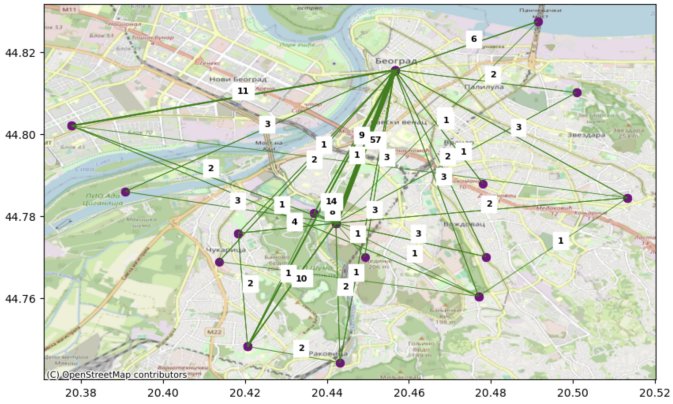

Figure 4: Connections between clusters

threshold of 0.01 as the most suitable for our research. We noticed a small difference between original and generalized trajectories of movements because our original trajectories were already filtered and reduced. We also experimented with TopDownTimeRatio Generalizer. The comparison for a singular trajectory when using DP Generalizer and TDTR Generalizer can be seen in Fig. 6.

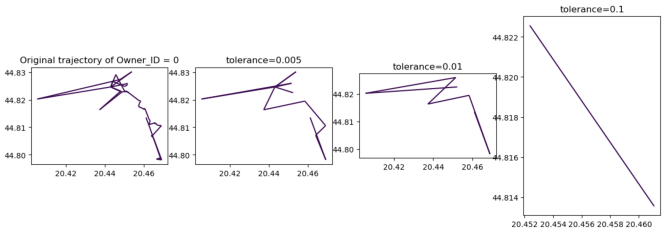

Figure 5: Different thresholds for a singular trajectory

6.3 Aggregation

The input parameter for TrajectoryCollectionAggregator was generalized trajectories with DouglasPeckuer Generalizer as instructed in

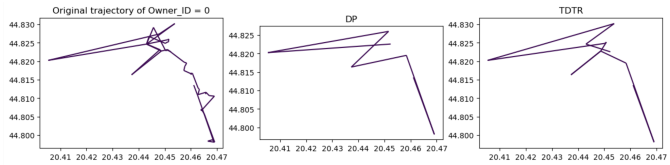

Figure 6: Comparison between TDTR and DP for a singular trajectory

the documentation of the MovingPandas library. Siginficant points and the aggregated flows between them by using this method are shown in the Fig. 7. We can observe that this method of reduction gave us similar results as clustering, identifying the centre of the city with park Kalemegdan as the most popular tourist area.

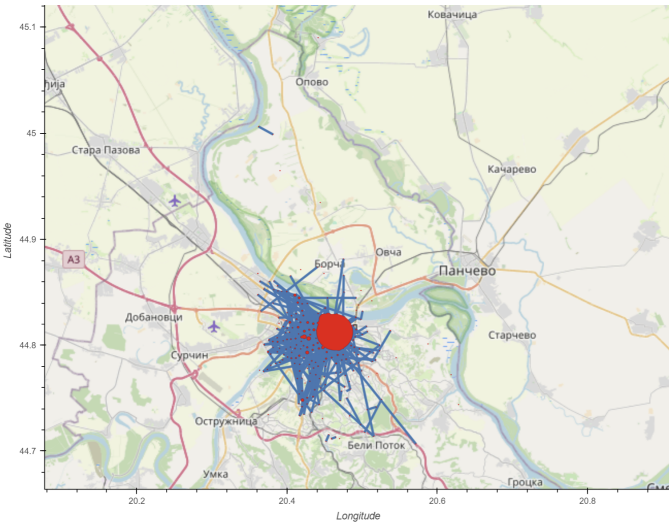

Figure 7: Aggregated flows and more popular areas

6.4 Predictive Modeling

The Markov chain analysis provided insights into the probabilities of transitions between clusters. By normalizing transition counts into probabilities, we gained insights into the likelihood of tourists moving from one cluster to another. Through Markov chain modeling, we determined the most probable next cluster after a given current cluster. For instance, after analyzing a sequence with cluster 5 which corresponds to New Belgrade, the predicted next cluster was 7, which corresponds to Zemun. This aligns geographically since these 2 locations are close. Additionally, employing Monte Carlo simulations, we had more trials to predict future clusters based on starting clusters. Lastly, by using association rules derived from Apriori [1] analysis, we predicted the next cluster by identifying the longest subsequence matching antecedents in the rules. For instance, when the current sequence contained cluster 36 (representing the Victor statue in Kalemegdan), the predicted next cluster was 38 (representing Roman well in Kalemegdan). Example output of the Apriori algorithm can be seen in Fig. 8.

```
Association Rules:
  antecedents consequents  antecedent support  consequent support   support  \
0        (36)        (38)            0.089189            0.162162  0.054054
1        (45)        (46)            0.102703            0.140541  0.051351

   confidence      lift  leverage  conviction  zhangs_metric
0    0.606061  3.737374  0.039591    2.126819       0.804154
1    0.500000  3.557692  0.036917    1.718919       0.801205
```

Figure 8: First two rows of Apriori algorithm output

7 CONCLUSION

This research introduces some improvements in analyzing tourist activities at destinations by using geotagged photos from Flickr, focusing on tourist movement patterns in Belgrade. A key innovation of this study is the development and application of the locID algorithm, which identifies unique tourist POIs and deals with the issue of overrepresentation from highly active users. This approach ensures that the analysis is more representative and less biased compared to other methods. The clustering results provided a view of frequently visited locations, offering valuable insights into the spatial distribution of tourist activity. By employing generalization techniques, the study effectively simplified tourist trajectories, making the data easier to interpret. Predictive modeling techniques, such as Markov chains, Monte Carlo simulations, and the Apriori al- gorithm, were used to predict future tourist movements. These methods demonstrated their potential for practical applications in tourism management, enabling more targeted marketing strategies and improved visitor experience planning.

ACKNOWLEDGMENT

The author would like to thank mentor professor Dejan Paliska for the support, cooperation and availability during the whole process of the research. Also, the University of Primorska and representative for this seminar prosessor Miklos for supporting this project.

REFERENCES

[1] P. Fournier-Viger, "SPMF: An Open-Source Data Mining Library," [Online]. Available: https://www.philippe-fournier-viger.com/spmf/. [Accessed: 03-Jun-2024].
[2] J. Silge and D. Robinson, "Text Mining with R: A Tidy Approach," O'Reilly Media, 2017. [Online]. Available: https://www.tidytextmining.com/ngrams. [Accessed: 03-Jun-2024].
[3] A. Graser, "MovingPandas: A Python Library for Movement Data Analysis," [Online]. Available: https://movingpandas.readthedocs.io/en/main/. [Accessed: 03-Jun-2024].
[4] A. Graser, "Trajectory Aggregator in MovingPandas," [Online]. Available: https://movingpandas.readthedocs.io/en/main/trajectoryaggregator.html. [Accessed: 03-Jun-2024].
[5] A. Graser, "Movement Analysis Tools on GitHub," [Online]. Available: https://github.com/anitagraser/movement-analysis-tools. [Accessed: 03-Jun-2024].