Learning mobility patterns with latent Dirichlet allocation –

A Readme file for the code (v1)

Author: Eyal Ben Zion

Reference:

> Ben Zion, E**.,** and Lerner, B. Learning human behaviors and lifestyle by capturing temporal relations in mobility patterns, In *European Symposium on Artificial Neural Networks* (*ESANN*), 2017.

This package finds the lifestyle of a user according to his points of interest (POIs) using Google Location History. Then it builds trajectory and week clusters from the POIs, and on these clusters, runs the 16 LDA models described in the paper. The first stage begins with preprocessing to extract the (Matlab) mat file from a JSON file of Google Location History by a required package, and creates a proper raw data file [prepardata]. The package also creates KML files, which upload the POIs to Google My Maps as a custom layer [createKML]. In order to reduce the experiments' runtime, this package creates the datasets (e.g., trajectory clusters and week clusters) for the LDA in different functions [createtrajectories], and creates all the input files for the LDA experiments. [runexp] runs all 16 LDA models on the datasets, and the output is the -LL on the test sets for every model.

File main.m in the package contains an example of the algorithm and its outputs. The chosen input in the main.m file is optimal. You will need to optimize the input for each user separately.

1. prepardata.m

   function [] = prepardata(username)

   Input: User name as a string.

   Output: This function saves these files:

   1. User name + 'AllData': mat file of the latitude, longitude, and date vector of the timestamp of the JSON file as a double matrix.
   2. User name + 'Data': mat file without blanks of the latitude, longitude, and date vector of the timestamp of the JSON file as a double matrix.

   This function reads the User1.mat file from the 'Location' folder in the package. The User1.mat file in the Location folder is the output of the matlab-json package. The function saves User1AllData.mat and User1Data.mat files in the 'Users' folder in the package.

2. createKML.m

function [] = createKML(username, rCell, iThresh)

Input:

1. User name as a string.
2. rCell: Double number which represents the cell size in the grid map.
3. iThresh: Double number 0 to 1. The threshold represents the number of weeks considered as rare.

Output: This function saves this file: KML file of all the user's POIs.

This function reads the User1Data.mat file from the 'Users' folder in the package. The function saves User1StopArea.kml in the 'Users' folder in the package.

3. createtrajectories.m

function [] = createtrajectories(username, weekday, rCell, iThresh, trajectoryKcluster)

Input:

1. User name as a string.
2. weekday: 1–7 day week (include weekend), ‘otherwise’–5 day week (not including weekends).
3. rCell: Double number which represents the cell size in the grid map.
4. iThresh: Double number 0 to 1. The threshold represents how many weeks to consider when choosing the POIs.
5. trajectoryKcluster: Integer indicates on the trajectory cluster number.

Output: This function saves this file:

'LDAinput' + user name: mat file which includes:

a. 'daysList': Cell vector with all the trajectory strings, size Nx1 (N is the number of unique trajectories).
b. 'daysAllClusters': Double matrix with UxN (U is the number of candidate clusters). Each column in the matrix is an Nx1 size vector, which represents the cluster number for each trajectory. For each column, the overall cluster number is different.
c. 'weekList': Double vector of all the week's numbers, size 1xM (M is the number of weeks)

This function reads the User1Data.mat file from the 'Users' folder in the package. The function saves User1LDAinput.mat in the 'Users' folder in the package.

4. runexp.m

function [] = runexp(username, i1, i2, i3, i4)

Input:

1. User name as a string.
2. i1: Weekdays: 1–7 day week (including weekend), otherwise–5 day week (not including weekend).
3. i2: Integer number of weeks in the test set.
4. i3: Integer number of lifestyles.
5. i4: Integer number of column to choose in 'daysAllClusters'.

Output: 'LDAExp' + user name + all the input parameters: mat file of the LL test set results as a double matrix, size (test set number) x16 (in which 16 is the number of models).

This function reads the User1LDAinput.mat file from the 'Users' folder in the package. The function saves LDAExpUser1_1_1_3_5.mat in the 'Results' folder in the package.

Required package: https://github.com/bayesnet/bnt