An Algorithm for Personalized Tourism Recommendation Based on Time Series

Yeping Peng

Software College of Jishou University, Zhangjiajie, Hunan Province, China

Keywords: Algorithm; Recommendation; Time Series; Spot

Abstract: For travelling, a tourist's choice of one destination spot is inevitably influenced by both their previous favourite and successive plan. In view of this, this paper proposes a recommendation algorithm based on time series incorporating three clustering algorithms, i.e., K-Means, MCA and Build Classification. The idea of this algorithm is to find the authoritative users of a spot, cluster their evaluated resources, find the previous and successive relevance of this spot according to the evaluation time, and then recommend the successive spot of the current focus to the user in sequence.

1. Introduction

In general, traveling is indiscrete continuum in that travellers' choice of one destination spot is inevitably influenced by both their previous favourite and successive plan. As a tourist, or being an Internet user at the same time, may search for needed resources online, the rule of choice of tourist destinations step-by-step according to the browsing records of authoritative users, and then recommend the spots in a personalized way to relevant users, who will feel helpful when deciding where to go for travel. Based on three typical clustering algorithms, i.e., K-Means, MCA and Build Classification, this paper proposes an information recommendation algorithm based on time series, which enable recommendations of successive travel candidates to users in the most accurate time to better meet their needs.

2. Literature Review

2.1 K-Means Clustering Algorithm

K-means clustering algorithm is the most commonly used partition-based method. It takes K as a parameter and divides n objects into K clusters, remaining highthe similarity within the clusters and that between the clusters lowest. The calculation of the similarity is based on the average value of all objects in a cluster, which is regarded as the centre of gravity of the cluster. The algorithmic process is presented as follows:

(1) K objects are randomly selected from n objects, each of which initially represents a cluster centre.

(2) For each remaining object, it is assigned to the nearest cluster according to its distance from each cluster centre. Given C={X_1, X_2…Xp} is the centre of the cluster concerned and O={O_1, O_2...Op} is an object to be allocated, then the most commonly used measure for the distance between C and O is Euclidean metric, and other distance measures include Manhattan distance, Minkowski distance, etc. The Euclidean distance formula is:

$$d(i,j)=\sqrt{\left|x_{i1}-x_{j1}\right|^2+\left|x_{i2}-x_{j2}\right|^2+\cdots+\left|x_{i1}-x_{j1}\right|^2}.$$

(3) Recalculate the average value for each cluster after change.

(4) It willnot go to the second step until the convergence of the criterion function is over. Among them, the more representative is the average error criterion, which is defined as follows: $E=\sum_{p=1}^{k}\sum_{j\in o_p}|j-Avg_p|^2$, wherein E is the sum of the square errors of all objects in the database, j is the point in space, representing the given data object, Avg_p is the average value of the cluster O_p. If the value of E is less than a threshold, the clustering process will terminate. [1]

DOI: 10.25236/isrme.2019.066

2.2 Matrix Clustering Algorithm

Matrix clustering algorithm is the most commonly used density-based method, which extracts dense regions from sparse matrix for clustering and transforms the relationship between users and resources into 1 or 0.The result of clustering represent that a user group is interested in a certain kind of resource set. With MAC algorithm, data object users or resource items may be distributed in multiple clusters. Assuming that all attribute variables of data objects are binary variables with values of 0 or 1 and that the area of a matrix is defined as the number of rows multiplied by the number of columns, the density of a matrix is defined as the number of matrices divided by the area of the matrix.

The values of m data objects on n attribute variables form a matrix, which has m rows and n columns. It is assumed that the value of all attribute variables of data objects in the matrix cells of binary variables $c_{ij} = (i = 1, 2 \dots m_i,\ j = 1, 2 \dots n)$is either 1 or 0. The area of the matrix is defined as below.

The area of Matrix D S_D equals the product of the number of its rows and that of its columns, i.e., S_D=m*n.

The density of Matrix D d_D equals the result of the number of the cells with $c_{ij} = 1$divided by its area, i.e., $d_d = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n} W_{ij}}{S_D}$.In addition, d_D is also the result of the number of the cells with $c_{ij} = 1$divided by the total number of the cells in the matrix.

For the given threshold of density ω, submatrices with density more than ω are extracted from the whole matrix. [2]

2.3 Build Classification Algorithm

Build classification algorithm can form a class hierarchy according to the similarity between classes, which can effectively reflect the similarity between classes. Whereas there are not many classes in the class hierarchy, the algorithm adopts a bottom-up hierarchical clustering method. it avoids the hierarchical clustering method that some systems must be clustered according to certain topological structures. In addition, when the similarity is less than a specified threshold, clustering analysis will not be carried out, instead of merging directly into only one class as some hierarchical clustering methods do. Thus, it speeds up the clustering speed. [3]

2.4 User Authority

Different users have different reliability in evaluating resources. Experienced users or authoritative users can evaluate resources more accurately and objectively. It can be seen that the reliability of user evaluation of resources, i.e. the authority of users, reflects the stability of user evaluation. Therefore, the evaluation of resources by authoritative users is more worthy of reference by other users. Authority is determined by the factors as follows.

(1) The amount of resources that users have evaluated. If a user has evaluated a lot of resources on one side, then he has some experience and shows his or his taste in the evaluation. Therefore, the first characteristic of user authority can be reflected by the number of users' evaluation of resources. The formula is presented as:$W_1 = \begin{cases} 1, n_u \geq A \\ \frac{n_u}{A}, n_u < A \end{cases}$,wherein, $n_u$represents the number of the resources that user U has evaluated. A is a constant, also called the number of penalties, with value set as 50. When $n_u < A$,the authority of the user will be weakened; when $n_u \geq A$, the weight is 1.

(2) Users' travelling experience. If a user has not evaluated as many resources as required for reflecting his or her authority, then his or her travelling experience will be taken into account. For example, users with much travelling experience will give more objective evaluations of spots than those with less travelling experience. Therefore, the second characteristic of user authority can be reflected by travelling experience, i.e., W_2equals 1, for users with most experience, $K_1$for users with much experience, $K_2$for users with less experience and $K_3$for users with less or none experience.

The above two aspects are measured from the user's own point of view, hence:

$\alpha_1 W_1 + \alpha_2 W_2$, wherein $\alpha_1 + \alpha_2 = 1$.

(3) Resources' evaluations received from users. When a resource is evaluated sufficiently, the average value of the resource evaluation can be used to measure the real quality or quality of the resource, and an authoritative user can generally correctly evaluate the essential characteristics of things. Therefore, the evaluation value of a user is close to the quality of the resource, i.e., the degree close to the average value of the resource evaluation can be used to describe the third characteristic of user authority. The formula is defined as follows:

$AU_2(u) = \frac{\sum_{i \in v_u}\left(1 - \frac{|v_{u,i} - \overline{v_i}|}{Max - Min}\right)}{n_u}$, wherein $v_u$represents the collection of the resources that User U has evaluated, $v_{u,i}$represents User U's evaluation value of Resource i, $\overline{v_i}$represents the average of the evaluation value of Resource i, and Max and Min stand for the maximum and minimum evaluation value respectively. [4]

3. Travel Destination Recommendation Algorithm Based on Time Series

In view of the indiscrete nature of travelling, this study proposes a recommendation algorithm based on time series. In this algorithm, three clustering algorithms are used to recommend the successive tourism resources to users in the most appropriate time, who can get the most needed information in time. The idea of this algorithm is to find the authoritative users of a spot, cluster their evaluated resources, find the previous and successive relevance of this spot according to the evaluation time, and then recommend the successive spot of the current focus to the user in sequence.

(1) Use MCA algorithm to calculate the types of resources. Generally, the resources that a user studies in a short period of time belong to the same field or reflect the same theme. A matrix for MCA algorithm is used to represent all the resources that a user has visited at different times. Rows denote time, and lists denote keywords of resources that users have visited. This is listed as a keyword rather than a resource because resources are isolated. Most resources are accessed only at one time, but keywords are different. Different keywords can be accessed at the same time, and the same keyword can be accessed at different times. [5] Therefore, the clustered sub-matrix represents similar keywords that have been visited at the same time. With the aid of MCA algorithm, a set of keywords that the users are interested in a certain period of time can be collated.

Firstly, regions with dense more than $\omega$are extracted from the sparse matrix for clustering according to S_D and d_D calculated by the formulas and given threshold value of ω.Then, in the specific implementation, additional constraints, such as the smallest number of rows, the smallest number of columns and other parameters, should be attached. Finally, the output forms all the sub-matrices of clustering. Each submatrix is a class, and the resources in each class are obtained in time order.

(2) Use K-Means algorithm to find the resources included in each resource type. According to the total number of sets in the first step, the number of resource partitions that users have visited can be obtained, i.e., the number of categories in K-Means algorithm, by which the resources contained in each class can be calculated.

(3) Use Build Classification algorithm to find the classes formed by resource categories. Users' interests are diverse, and there are many branches in each resource category, so it is necessary to classify each small category that has been found to its higher one. For example, in the class of Resource A, small categories of resources can be organized in chronological order: a_1, $a_2 \ldots a_n$. According to the characteristics of cognition, a_1 is previous experience to $a_2$whilea$_3$ is successive experience to a_2.

(4) Recommend according to continuity of experience. Users similar to certain user in the large class of resources can be found. By comparing the small category b_n of the resources that the current user has accessed to with similar users' resource sequence Table such as a_i. If a_i, b_n are most similar experience, then a_{i+1} will be the most needed resources for the user.

(5) Recommend Resources. Firstly, the authoritative users are found among similar users

according to the authority calculation method. Secondly, User A {a_1, a_2...a_m} recommends resources to User B {b_1, b_2...b_n}. After User B's last keyword class b_n is got in order, i.e., the resource class that User Bis currently interested in, Resource Class a_i, which is most similar to resource class b_n, can be found among the authoritative users. Thirdly, the most similar category between b_n and a_i is calculated by cosine similarity, represented by Max (Sim (b_n, a_i)), wherein $\mathrm{Sim}(b_n, a_i) = \frac{b_n \cdot a_i}{||b_n||||a_i||} = \frac{\sum_{i=1}^{k} a_{ji} \cdot b_{nj}}{\sqrt{\sum_{i=1}^{k} a_{ji}} \cdot \sqrt{\sum_{i=1}^{k} b_{ni}}}$.Fourthly, the more similar to the classes of authoritative users, the greater the likelihood of subsequent experience will be recommended. Therefore, a component of subsequent experience is added to the recommendation list. If the resource j belongs to the keyword set k, then$P_{a,j} = W_1 * P_{a,j}$, wherein $W_1 = \frac{|k-i|}{m-i}$. If the recommendation value ranks in the top N, then it will enter the final recommendation list and finally recommended to the current user.

4. Summary

This paper has proposed a recommendation algorithm based on time series. In this algorithm, three clustering algorithms are synthetically utilized, and the rule of choosing spot candidates is found according to the browsing records of authoritative users, and the successive candidates are recommended to users in the most appropriate time. The research shows that the recommendation algorithm based on time series improves the recommendation quality obviously, and has certain validity and feasibility.

Acknowledgement

This research was financially supported by the General Project of Hunan Provincial Department of Education (No. 16C1312).

References

[1] T. Kanungo, D. M. Mount, N. S. Netanyahu, et al. Anefficient K-means clustering algorithm: analysis and implementation. IEEE Transactions on Pattern Analysis & Machine Intelligence, 24, 7 (2002)881-892.

[2] A. Cichocki, R. Unbehauen, F. L. Luo. A Minor component analysis algorithm, Neural Networks, 10, 2 (1997)291-297.

[3] N. Abdelhamid, A. Ayesh, F. Thabtah, et al.. MAC: a multiclass associative classification algorithm, Journal of Information & Knowledge Management, 11, 2 (2012)1250011-1-1250011-10.

[4] Q. Zhang, Y. W. Leung. A class of learning algorithms for principal component analysis and minor component analysis, IEEE Transactions on Neural Networks,11, 1 (2000)200.

[5] A. D. Pila, R. Giusti, R. C. Prati, et al.. A multi-objective evolutionary algorithm to build knowledge classification rules with specific properties, International Conference on Hybrid Intelligent Systems, 1 (2006).