Tourist Attraction Recommendation Based on Knowledge Graph

Phatpicha Yochum[(✉)], Liang Chang, Tianlong Gu, Manli Zhu, and Weitao Zhang

Guangxi Key Laboratory of Trusted Software,
Guilin University of Electronic Technology, Guilin 541004, China
mink.phatpicha@gmail.com

Abstract. This paper focuses on building recommendation model based on knowledge graph in the tourism field. A knowledge graph for tourist attractions in the Bangkok city is constructed, and a tourist attraction recommendation model based on the knowledge graph is presented. Firstly, we collect tourism data in Bangkok and generate a tourist attraction knowledge graph by using the Neo4j tool. Then, by applying the network representation learning method Node2Vec, we generate the feature vectors of both attractions and tourists, and calculate the correlation scores between tourists and attractions according to the cosine similarity. Finally, we normalize the correlation scores to generate the recommended list. This model presented in the paper can overcome the sparsity problem of tourist knowledge graphs and can be used in large scale knowledge graph.

Keywords: Knowledge graph · Recommendation system · Node2vec Network representation learning

1 Introduction

With the rapid development of Internet technology, it is an important factor that the tourism services are changing from the past. The modern traveler has an easy access to large amount of travel information in seconds via the Internet. The travel websites such as TripAdvisor, Expedia, and TourismThailand plan their own trip. However, it is necessary to spend more time for searching about tourism information but finding or identifying the most relevant information about tourist attraction, routing, point of interest and planning a trip for each day of each trip are difficult. Therefore, one solution for this problem is the development of a recommendation system (RS) based on knowledge graph (KG). The high availability of information has huge benefits for the tourism domain.

RS is the information filtering system that deals with the problem of information overload by filtering vital information fragment out of large amount of dynamically generated information according to user's preferences, interest, or observed behavior about item [1]. RS is used in variety of areas including movies, restaurants, social tags and products in general. The tourism field is one of the most potential application areas of RS. The authors [2] collected tourism information from social media, extracted data, and find similarity with the relations among information which method gained high

effectiveness performance. In [3], RS was constructed to solve the problem of generating road-trip itineraries and activity duration with time of the day is more relevant to the user by using data from community opinion and location. In [4], the authors applied machine learning algorithms, such as K-nearest neighbors, decision tree, switching and weighted, to overcome the cold start problem in tourism field.

The concept of KG is highlighted by Google Corporation. The key technologies include the extraction of entities with their attribute information, and the relationships between entities [5]. In [6], KG was presented for recommending travel attraction in a new city that the user is going to by the use of semantic information has been exploited. In recent years, network representation learning has been proposed and aroused considerable research interest, most notably models known as word2vec [7], node2vec [8], and entity2rec [9]. It aims to learn the low-dimensional representations of vertexes in a network, while structure and inherent properties of the graph is preserved. Therefore, we propose a novel approach to organize and share tourism information in a large scale, and use a KG to connect all information regarding to the tourism e.g. tourist attraction, location, category, time and make it easy and universally accessed by everyone. We are to best of our knowledge the first who analyze the evolution of a KG and make use of this novel information to design tourist attraction recommendation system based on KG in Thailand.

A data set of this research is 5-year tourism data of TripAdvisor and TourismThailand of Bangkok in Thailand since 2014. This research has been conducted under three objectives: (1) to analyze tourism data set for exploring the tourism domain, (2) to design and implement a tourist attraction KG that can present the relationship among data, monitor and retrieve required information and (3) to propose a tourist attraction recommendation model based on KG that adopts a flexible random walk procedure based on Node2Vec.

2 Design and Implementation of Tourist Attraction Knowledge Graph

The conceptual framework is shown in Fig. 1. Tourism data in Bangkok is collected by the data aggregator from TripAdvisor and TourismThailand, the end of retrieved data is April 16, 2018. Specifically, we use the Knowledge of Information Retrieval tool (KnIR) which is developed in JAVA language for gathering data aggregator after that putting the data into the centralized database.

From data aggregator, we design the six fact attributes in tourist attraction: location, category, open time, user ID, rating and comment. We purpose to generate graph that represent a relationship between tourist attraction and other related information in triples, such as (Chatuchak Weekend Market, Category of, Street Markets). The Neo4j is used to generate the tourist attraction knowledge graph as shown in Fig. 2.

Our data set contains 1,411 attractions, 66,461 reviews rated by 41,765 users. There are 23 categories and 35 sub categories. The total number of triples in knowledge graph is 87,544.
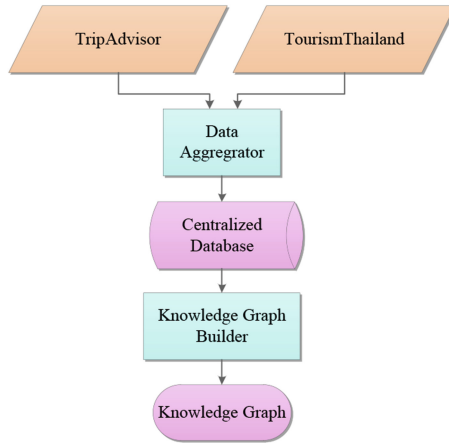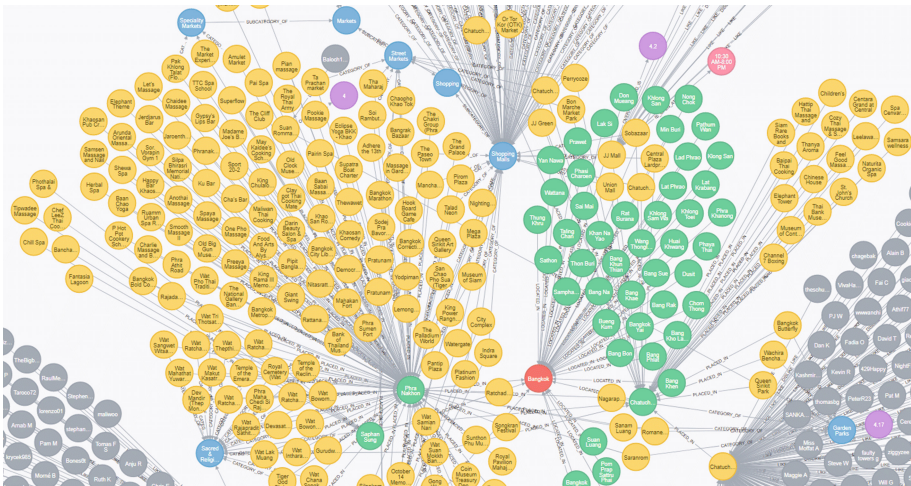
Fig. 1. The conceptual framework

Fig. 2. The schema of the tourist attraction knowledge graph

3   Tourist Attraction Recommendation Model Based on Knowledge Graph

In this section, we design a flexible random walk procedure based on Node2Vec for tourist attraction recommendation. Using the network representation learning method Node2Vec models the tourist attraction. The node sequence model takes attribute sub-graph as input to the generate node sequence, and produces the feature vectors of an attraction or tourists as the output which obtained by learning sequence feature. Next, uses the cosine similarity to calculate the correlation scores between tourists and

attractions. Finally, we normalize the correlation scores to generate the recommended list. This model is able to overcome the sparsity problem of the tourist knowledge graphs. The framework is shown in Fig. 3.
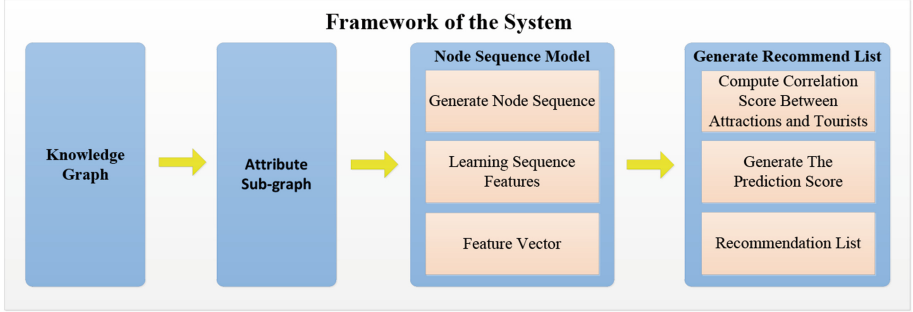

Fig. 3. The framework of the system

3.1 Generate Node Sequences

First, divide KG into several sub-graphs and generate the corresponding sequence $S = [u_1, u_2, \ldots \ldots u_n]$ by using random walk strategy. Formally, u is a given source node, we simulate a random walk of flexible length l which related to the scale of a subgraph. u_j is the i_{th} node in the walk, starting with $u_0 = u$. Nodes u_j are generated by the following distribution:

$$P(u_j = x | u_{j-1} = v) = \begin{cases} \frac{\pi_{vx}}{N} & if \ (v, x) \in E \\ 0 & otherwise \end{cases} \tag{1}$$

where E represents the set in the knowledge graph, π_{vx} is the unnormalized transition probability between v and x, and N represents the normalizing constant. Edge (t, v) is just traversed by a random walk which now resides at node v. The walk needs to compute the transition probabilities π_{vx} on edges (v, x) for selecting the next node. The unnormalized transition probability is as follows

$$\pi_{vx} = \alpha_{pq}(t, x) * w_{vx} \tag{2}$$

where w_{vx} is the weights on edges (v, x), in the case of no weighted graphs the default values are 1, and α_{pq} is as follows

$$\alpha_{pq}(t, x) = \begin{cases} \frac{1}{p} & if \quad d_{tx} = 0 \\ 1 & if \quad d_{tx} = 1 \\ \frac{1}{q} & if \quad d_{tx} = 2 \end{cases} \tag{3}$$

where d_{tx} is the shortest path distance between t and x and the value of d_{tx} is one of the set {0, 1, 2}. P and q are the return parameter and in-out parameter respectively.

3.2   Learning Sequence Features

In Sect. 3.1, we have divided KG into several sub-graphs, and here we use one of these sub-graphs as an example. Suppose $S = [u_1, u_2, \ldots\ldots.u_n]$ is a node sequence of the sub-graph, we build the node sequence representation model by using three lawyer neural network. The objective function which is a log-likelihood function is as follows

$$\sum_{j=1}^{N} \log \Pr\left(u_j | x^{u_j}\right) \tag{4}$$

where N denotes the number of nodes in this sequence, X^{u_j} denotes a feature vector which is composed of the context nodes of u_j.

3.3   Recommend List Generation

In Sects. 3.1 and 3.2, tourists and attractions in the knowledge graph have been represented as vectors based on the attributes of features. Here we use **v(attract)** represents the feature vector of the attraction, and **v(user)** represents the feature vector of the user. After get the feature vector of the attractions and visitors in the same vector space, we use the following formula to measure the correlation between a tourist and attractions and generate the prediction score.

$$\mathrm{Re}l(attract, user) = sim(\mathbf{v}(\mathbf{attract}), \mathbf{v}(\mathbf{user})) \tag{5}$$

where *sim* is the cosine similarity.

Finally, the system returns a recommendation list according to the prediction score. This model is able to overcome the sparsity problem of the tourist knowledge graphs.

4   Conclusion and Future Work

In this paper, we designed and implemented a tourist attraction knowledge graph that can present the relationship among data, monitor and retrieve required information. We also proposed a tourist attraction recommendation model based on knowledge graph that adopts a flexible random walk procedure based on Node2Vec. Firstly, we collect tourism data in Bangkok by using KnIR tool for gathering data aggregator from TripAdvisor and TourismThailand and use the Neo4j tool to generate the tourist attraction knowledge graph. After that the network representation learning method Node2Vec is applied to obtain the feature vectors of an attraction or tourists, and then use the cosine similarity to calculate the correlation scores between tourists and attractions. Finally, we normalize the correlation scores to generate the recommended list. This model is presented for overcoming the sparsity problem of the tourist

knowledge graphs and can be used in large scale knowledge graph. In the future work, we will enrich our knowledge graph in depth with all cities in Thailand and develop recommendation system about Thailand.

References

1. Kzaz, L., Dakhchoune, D., Dahab, D.: Tourism recommender systems: an overview of recommendation approaches. Int. J. Comput. Appl. **180**(20), 9–13 (2018)
2. Shen, J., Deng, C., Gao, X.: Attraction recommendation: Towards personalized tourism via collective intelligence. Neurocomputing **173**, 789–798 (2016)
3. Mittal, R., Sinha, V.: A personalized Time-Bound Activity Recommendation System. In: 7th Annual Computing and Communication Workshop and Conference (CCWC). IEEE, USA (2017)
4. Kbaier, M., Masri, H., Krichen, S.: A personalized hybrid tourism recommender system. In: 14th IEEE/ACS International Conference on Computer Systems and Applications, pp. 244–250. IEEE, Tunisia (2017)
5. Juel, K.: Ethics of Google's Knowledge Graph: some considerations. J. Inf. Commun. Ethics Soc. **11**(4), 245–260 (2013)
6. Lu, C., Laublet, P., Stankovic, M.: Travel attractions recommendation with knowledge graphs. In: Blomqvist, E., Ciancarini, P., Poggi, F., Vitali, F. (eds.) EKAW 2016. LNCS (LNAI), vol. 10024, pp. 416–431. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-49004-5_27
7. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient Estimation of Word Representations in Vector Space. CoRR abs/1301.3781 (2013)
8. Grover, A., Leskovec, J.: node2vec: scalable feature learning for networks. In: 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 855–864. ACM, USA (2016)
9. Palumbo, E., Rizzo, G., Troncy, R.: entity2rec: learning User-Item Relatedness from Knowledge Graphs for Top-N Item Recommendation. In: Eleventh ACM Conference on Recommender Systems, pp. 32–36. ACM, Italy (2017)