Meta-Path Based Service Recommendation in Heterogeneous Information Networks

Tingting Liang[1(✉)], Liang Chen[2], Jian Wu[1], Hai Dong[2], and Athman Bouguettaya[2]

[1] College of Computer Science and Technology, Zhejiang University, Hangzhou, China
{liangtt,wujian2000}@zju.edu.cn

[2] School of Computer Science and Information Technology, RMIT, Melbourne, Australia
{liang.chen,Hai.dong,athman.bouguettaya}@rmit.edu.au

Abstract. In the scenario of service recommendation, there are multiple object types (e.g. services, mashups, categories, contents and providers) and rich relationships among these objects, which naturally constitute a heterogeneous information network (HIN). In this paper, we propose to recommend services for mashup creation by exploiting different types of relationships in service related HIN. Specifically, we first introduce meta-path based measure for similarity estimation between mashups along different types of paths in HIN. We then design a recommendation model based on collaborative filtering and meta-path based similarities, and employ Bayesian ranking based optimization algorithm for model learning. Comprehensive experiments based on real data demonstrate the effectiveness of the HIN based service recommendation approach.

1 Introduction

The increasing adoption of Service-Oriented Architecture (SOA) leads to a surge of services in forms of Web services, cloud services, APIs, mashups, etc. As a consequence, several service repositories, such as ProgrammableWeb (PW)[1] and Mashape[2], have emerged to incessantly accumulate services and their compositions in recent years. As a web application generated through service composition, mashup has become a popular technique for integrating applications and data over the Web. However, creating a mashup may be difficult and time consuming for inexperienced developer due to the increasing presence and adoption of services on the Internet. Therefore, how to effectively recommend and select services for mashup creation is becoming an urgent problem.

Previous studies for service recommendation and selection can be mainly classified into several categories which are respectively based on semantic similarity [5,7], collaborative filtering (CF) technique [4,14], and quality of services

[1] ProgrammableWeb: http://www.programmableweb.com.
[2] Mashape: https://www.mashape.com.

Q.Z. Sheng et al. (Eds.): ICSOC 2016, LNCS 9936, pp. 371–386, 2016.
DOI: 10.1007/978-3-319-46295-0_23

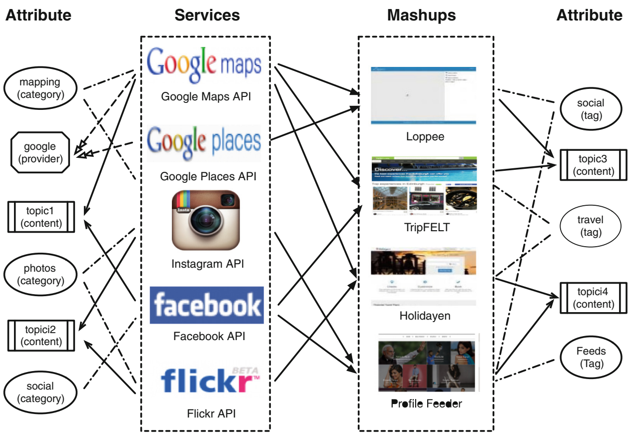

Fig. 1. A heterogeneous information network organized by objects and relationships in the scenario of service recommendation for mashup creation

(QoS) [6,19]. The limitation of semantics-based and CF-based methods is mainly caused by the singleness of employed data. The drawback of QoS-based recommendation approach is that the QoS information is not always available, which limits its generalization ability [17]. Recently, some studies try to recommend services by exploiting information network analysis [3,9]. However, most of information network-based methods only utilize a single type of relationship, e.g., composition relationship between services and mashups.

In reality, service recommender system generally includes multiple object types (e.g. services, mashups, and related attributes) and rich relationships among those objects, which naturally constitute a HIN. An illustration of a HIN in the scenario of service recommendation can be seen in Fig. 1. In this example, except for services, mashups, and the service-mashup interaction relationships, there exist many other object types (e.g. categories, contents, providers, and tags) and kinds of relationships among objects. For instance, *Instagram* and *flickr* are connected as they belong to the same category (*photos*); both of *TripFELT* and *Holidayen* are tagged by *travel*. Moreover, the paths composed by multiple relationships in the HIN have different semantics, based on which similarities with different meanings between objects can be evaluated. For example, *Google maps* and *Google places* can be considered similar as they belong to the same category (*mapping*), from another perspective, they are similar because of the same service provider. Refer to the traditional CF idea, service recommendation based on the similarity between mashups is workable because the recommended service meets the need of target mashup which has the similar demand with mashups having interactions with the service. The more detailed discovery of different semantics of paths can be found in Sect. 3.1. The combination of objects similarities with different semantics can improve the recommendation quality since allround demands of mashups for services are considered.

In this paper, we propose a meta-path based service recommendation approach called PaSRec for mashup creation. PaSRec effectively combines the heterogeneous relationship information to generate a service recommendation model and provides high-quality recommendation results with model learning algorithm. Specifically, we first introduce the meta-path based similarity measurement, based on which the similarities between objects along different paths in HIN are calculated. Considering the fact that the services employed by a mashup should be complementary but not similar with each other, we compute the similarities between mashups. By setting multiple meta-paths, a recommendation model is proposed to combine heterogeneous information referring to the idea of collaborative filtering. We adopt a Bayesian ranking based optimization algorithm [12] to learn the model with implicit feedback data. Note that the feedback in service recommendation is implicit since mashups employ services without expressing their tastes explicitly (e.g. ratings). Experiments on a real world dataset crawled from PW are implemented for the demonstration of the effectiveness of PaSRec.

In particular, the main contributions of our paper are summarized as follows:

1. We propose a service recommendation approach called PaSRec for mashup creation by exploiting service related heterogeneous information network, which is first proposed in the field of service recommendation.
2. We introduce a meta-path based similarity measurement to build a service recommendation model, and adopt a Bayesian ranking based optimization algorithm to learn it with implicit feedback data.
3. Empirical studies based on a real world dataset crawled from PW demonstrate the power of the proposed PaSRec.

The remainder of paper is structured as follows. Section 2 introduces the background and preliminaries of this work. The the meta-path based recommendation model and model learning algorithm are described in Sect. 3. Section 4 illustrates the concrete evaluation process and performance analysis. Finally, we discuss related work in Sect. 5 and make a conclusion in Sect. 6.

2 Background and Preliminaries

In this section, we present the background and some preliminary knowledge of this study.

2.1 Heterogeneous Information Network

An information network represents an abstraction of the real world, focusing on the objects and the interactions among these objects. Refer to [15], an information network is defined as follows.

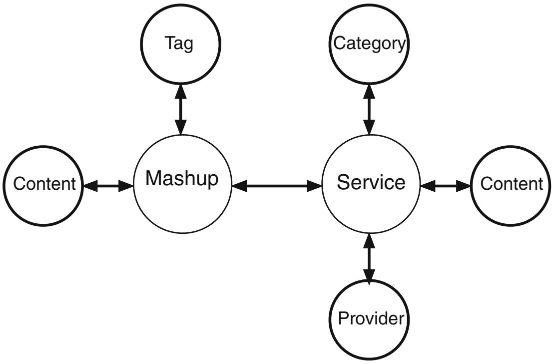

Fig. 2. Network schema of service network

Definition 1 ***(Information Network).*** *An information network is defined as a directed graph $G = (\mathcal{V}, \mathcal{E})$ with an object type mapping function $\varphi : \mathcal{V} \rightarrow \mathcal{A}$ and a relationship type mapping function $\psi : \mathcal{E} \rightarrow \mathcal{R}$, where each object $v \in \mathcal{V}$ belongs to one particular object type $\varphi(v) \in \mathcal{A}$, each link $e \in \mathcal{E}$ belongs to a particular relationship type $\psi(e) \in \mathcal{R}$. If two links belong to the same relationship type, the two links share the same starting object type as well as the ending object type.*

Different from the traditional network definition, we explicitly distinguish object types and relationship types in the network. The information network is called **heterogeneous information network** if the types of objects $|\mathcal{A}| > 1$ or the types of relationships $|\mathcal{R}| > 1$; otherwise, it is a **homogeneous information network**. As mentioned in Sect. 1, Fig. 1 shows an example of heterogeneous information network in service recommender system.

For better understanding the object types and relationship types in a complex heterogeneous information network, it is essential to provide the meta-level (schema-level) description of the network. Therefore, the concept of network schema is proposed as follows to describe the meta structure of a network.

Definition 2 ***(Network Schema).*** *The network schema, denoted as $T_G = (\mathcal{A}, \mathcal{R})$, is a meta template for a heterogeneous network $G = (\mathcal{V}, \mathcal{E})$ with the object type mapping $\varphi : \mathcal{V} \rightarrow \mathcal{A}$ and the link type mapping $\psi : \mathcal{E} \rightarrow \mathcal{R}$, which is a directed graph defined over object types $\mathcal{A}$, with edges as relationships from $\mathcal{R}$.*

Network schema identifies how many types of objects there are in the heterogeneous information network and where the links exist.

For the network in service recommender system described in Sect. 1, the network schema is shown as Fig. 2. Links exist between services and mashups denoting the composing or composed-by relationships, between contents and mashups denoting describing or described-by relationships, between providers and services denoting providing or provided-by relationships, etc. In addition, two objects in a network can be connected via different paths and these paths have different meanings. For instance, mashups can be connected via "Mashup-Service-Mashup" (M-S-M) path, "Mashup-Tag-Mashup" (M-T-M) path, "Mashup-Service-Category-Service-Mahsup" (M-S-Ca-S-M) path and so on. These paths are called meta-paths that are the combinations of a sequence of relations between object types.

2.2 Meta-Path Based Similarity

Refer to [16], the meta-path based similarity is defined as follows.

Definition 3 ***(Meta-Path based Similarity).*** *Given a symmetric meta path $\mathcal{P}$, the meta-path based similarity between two objects of the same type x and y is:*

$$s(x,y) = \frac{2 \times |\{p_{x \to y} : p_{x \to y} \in \mathcal{P}\}|}{|\{p_{x \to x} : p_{x \to x} \in \mathcal{P}\}| + |\{p_{y \to y} : p_{y \to y} \in \mathcal{P}\}|} \tag{1}$$

where $p_{x \to y}$ is a path instance between object x and y.

The similarity $s(x, y)$ is defined in terms of two parts: (1) their connectivity is defined by the number of paths between them following $\mathcal{P}$; (2) the balance of their visibility which is defined as the number of paths between themselves.

3 Services Recommendation Method PaSRec

In this section, we illustrate the proposed service recommendation method for mashup creation called PaSRec. PaSRec includes a meta-path based recommendation model and a model learning algorithm with implicit feedback data, which are described in Sects. 3.1 and 3.2, respectively.

3.1 Meta-Path Based Recommendation Model

This subsection proposes a meta-path based services recommendation model for mashup creation. Considering the fact that the services employed for a mashup should be complementary but not similar with each other, we compute the similarity between mashups for our recommendation model. Specifically, the model first evaluates the meta-path based similarity of mashups, and infers the predicted scores of candidate services according to the similar mashups of the target mashup. It should be noted that the meta-paths between mashups have different semantics. For example, "Mashup-Service-Mashup" (M-S-M) means mashups that are composed by the same services with the target mashup. Following that path, it will recommend services used for mashups having the similar composition with the target mashup. Another example is that "Mashup-Service-Category-Service-Mashup" (M-S-Ca-S-M) means mashups that are composed by the services belonging to the same category with that of the target mashup. According to M-S-Ca-S-M, it will recommend services that are classified to the same categories with components of the target mashup. Table 1 demonstrates the other representative paths and their semantic meanings. Note that for M-Co-M and

Table 1. The meanings of representative meta paths

No.	Meta-Path	Semantic meaning
1	M-S-M	mashups that are composed by the same services with the target mashup
2	M-T-M	mashups that are labeled by the same tags with the target mashup
3	M-Co-M	mashups that have the same topics extracted from contents with the target mashup
4	M-S-Ca-S-M	mashups that are composed by the services belonging to the same categories with that of the target mashup
5	M-S-Co-S-M	mashups that are composed by the services having the same topics extracted from contents with that of the target mashup
6	M-S-P-S-M	mashups that are composed by the services offered by the same providers with that of the target mashup

M-S-Co-S-M, "Co" represents the content (description) of services (mashups). As content of each service (mashup) is unique, we try to extract topics from content by Latent Dirichlet Allocation (LDA) model [1] and connect two services (mashups) through those topics. Based on different meta-paths, the mashups can receive quite different recommended service lists. How to effectively integrate these recommendation results generated by different meta-paths is a critical problem.

By measuring the meta-path based similarity of mashups, we can find the similar mashups to a target mashup with a given path. Refer to the idea of collaborative filtering, the rating of the target mashup on a service can be inferred depending on the ratings of its similar mashups on the service. Under a meta-path $\mathcal{P}_l$, the predicted rating of a mashup m on a service i, denoted as $r^{(l)}(m,i)$ can be calculated as follows,

$$r^{(l)}(m,i) = \sum_{n \in D(m,K) \cap N(i)} s^{(l)}(m,n) r(n,i), \tag{2}$$

where $D(m,K)$ is a set of K mashups that are most similar with mashup m, $N(i)$ denotes a set of mashups having been composed by service i, $s^{(l)}(m,n)$ is the similarity of mashup m and n under meta-path $\mathcal{P}_l$, and $r(n,i)$ denotes the rating score of mashup n on service i. Note that feedback in the scenario of service recommendation is not explicit but implicit and $r(n,i) = 1$.

By repeating the process formulated as Eq. (2) under all L meta-paths, we can obtain L predicted ratings of a mashup m on a service i. The different predicted ratings under diverse meta-paths may have different importance. For example, services may be more likely to be used for mashup creation because of their topics rather than their providers. With this intuition, the final predicted rating under all meta-paths, denoted as $r(m, i)$, can be generated by the weighted sum of predicted ratings under L meta-paths,

$$r(m, i) = \sum_{l=1}^{L} \theta^{(l)} \cdot r^{(l)}(m, i), \tag{3}$$

where $\theta^{(l)}$ denotes the weight for the predicted rating of a mashup on a service under meta-path $\mathcal{P}_l$.

With the recommendation model as shown in Eq. (3), we can predict recommendation scores of all services and accordingly rank these services for a given mashup. The estimation of the parameters in the recommendation model would be presented in the next subsection.

3.2 Learning Model with Implicit Feedback

In this subsection, we introduce a learning algorithm for the proposed meta-path based recommendation model. The recommendation model takes advantage of the heterogeneous relationships in information network. Specifically, we intend to integrate meta-path based similarities with parameters indicating the importance of the corresponding paths in a HIN.

Since the feedback generated by mashups in the scenario of service recommendation is not explicit, we use these implicit feedbacks as training data. Inspired by [18], we introduce BPR method [12] which is widely accepted in implicit feedback recommendation to learn the proposed model. We set value 1 for positive (observed) feedback (mashups have been composed by services) in implicit feedback data while set the value 0 for a mixture of negative feedback (services are not appropriate for the composition of mashups) and unobserved potential interactions (mashups are not aware of such services). Then an objective function is defined to rank mashups with 1 values higher than those with 0 values for each service. The assumption is that services are more likely to be used for the mashups with value 1 in training data than the rest of the mashups.

Bayesian Ranking Based Optimization. Let M be the set of all mashups and S be the set of all services. We use $>_i \subset M^2$ to denote a total ranking of all mashups for service i. $m >_i n$ certainly represents service i is more appropriate for the creation of mashup m than that of n. The Bayesian formulation of finding the optimal ranking for all mashups is to maximize the posterior probability as follows,

$$p(\theta| >_i) \propto p(>_i |\theta)p(\theta), \tag{4}$$

where $\theta = \{\theta_1, ..., \theta_L\}$ denotes the model parameters, and $p(>_i |\theta)$ represents the probability that all mashup pairs can be ranked correctly.

We assume the services are chosen independently by mashups and the ordering of each pair of mashups (m, n) for a specific service is independent. Thus, the likelihood function $p(>_i|\theta)$ can be represented as a product of single densities and then be combined for all service $i \in S$ as follows,

$$\prod_{i \in S} p(>_i|\theta) = \prod_{(i,m,n) \in Z} p(m >_i n|\theta), \tag{5}$$

where $Z \subset S \times M \times M$ denotes the set of triples (i, m, n) where feedback of mashup m on service i is positive and value of mashup n on i is 0.

We formulate $p(m >_i n|\theta)$ as:

$$p(m >_i n|\theta) = \sigma(r(m, i) - r(n, i)), \tag{6}$$

where σ is the logistic sigmoid function $\sigma(x) = \frac{1}{1+e^{-x}}$.

We introduce normal distribution with zero mean and variance-covariance matrix $\Sigma_\theta = \lambda I$ as the prior density $p(\theta)$. Based on the likelihood and probability discussed above, the objective function can be derived as follows,

$$\begin{aligned} OPT &= -\ln p(\theta| >_i) = -\ln p(>_i|\theta)p(\theta) \\ &= -\ln \prod_{(i,m,n) \in Z} p(m >_i n|\theta)p(\theta) \\ &= -\sum_{(i,m,n) \in Z} \ln p(m >_i n|\theta) + \lambda \|\theta\|_2^2 \\ &= -\sum_{(i,m,n) \in Z} \ln \sigma(r(m, i) - r(n, i)) + \lambda \|\theta\|_2^2, \end{aligned} \tag{7}$$

where λ is a model specific regularization parameter.

With the minimization of objective function OPT, the recommendation parameter θ can be learned from the implicit feedback data.

Learning Algorithm. As Eq. (7) is differentiable, gradient descent based algorithms are obvious choices to estimate parameter θ. In this paper, we employ the stochastic gradient descent (SGD) method [2] to learn the parameters for evaluation. The gradient of Eq. (7) with respect to θ is:

$$\begin{aligned} \frac{\partial OPT}{\partial \theta} &= -\sum_{(i,m,n) \in Z} \frac{\partial}{\partial \theta} \ln \sigma(r_{mn,i}) + \frac{\lambda}{2} \frac{\partial}{\partial \theta} \|\theta\|_2^2 \\ &= -\sum_{(i,m,n) \in Z} \frac{e^{-r_{mn,i}}}{1 + e^{-r_{mn,i}}} \frac{\partial}{\partial \theta} r_{mn,i} + \lambda \theta, \end{aligned} \tag{8}$$

where $r_{mn,i} = r(m, i) - r(n, i)$.

The whole process of service recommendation PaSRec can be found in Algorithm 1. The time complexity of PaSRec is analyzed as follows. PaSRec contains two main parts as shown in Algorithm 1. The first part described in lines 1–5 is the computation of similarities and predicted ratings along with different meta-paths. The main time-consuming component is the calculation

Algorithm 1. Heterogeneous Information based Service Recommendation

Input:
  Information network and implicit feedback data
Output:
  Recommended list $RecList$
1: Decide L meta-paths
2: **for** $\mathcal{P}_l \in \mathcal{P}$ **do**
3:   Evaluate mashup similarity s^l with Eq. (1)
4:   Calculate predicted rating r^l with Eq. (2)
5: **end for**
6: Initialize θ
7: **repeat**
8:   Draw (m, n, i) from Z
9:   Calculate $\frac{\partial OPT}{\partial \theta}$ with Eq. (8)
10:   $\theta \leftarrow \theta - \alpha \frac{\partial OPT}{\partial \theta}$
11: **until** convergence
12: Rank predicted ratings calculated with Eq. (3)
13: **return** $RecList$

of similarity which can be completed offline. Another part in lines 6–11 is the model learning process, and its time complexity is $O(|M|^2|S||\mathcal{P}|)$ where $|M|$ is the number of mashups, $|S|$ is the number of services, and $|\mathcal{P}|$ denotes the number of decided meta-paths.

4 Empirical Study

In this section, we present the empirical studies of the proposed service recommendation approach. We implement PaSRec along with several general or state-of-the-art recommendation methods to demonstrate the effectiveness of the proposed approach.

4.1 Data

We evaluate the proposed service recommendation approach PaSRec on the dataset crawled from PW. The dataset includes 6,340 mashups and 1,399 API services with 13,685 composition relationships, and the sparsity of the interaction matrix is about 99.8 %. The dataset includes the attribute information of APIs and mashups. The detailed description of this dataset can be found in Table 2, and their network schema is shown as Fig. 2. The average degree of a object related a relationship is calculated by dividing the number of relationship instances by the number of object instances.

For overall comparison, we use 60 % of the composition records as training set to predict the remaining 40 %. Besides, we assign different training data settings (10 %–90 %) to show the comparison results in different data sparseness.

4.2 Evaluation Metrics

We employ three widely used metrics, precision, recall and mean reciprocal rank (MRR), to measure the performance of PaSRec.

$$\text{Precision} = \frac{1}{|M|}\sum_{m\in M}\frac{|rec(m)\cap test(m)|}{|rec(m)|}, \tag{9}$$

$$\text{Recall} = \frac{1}{|M|}\sum_{m\in M}\frac{|rec(m)\cap test(m)|}{|test(m)|}, \tag{10}$$

where $rec(m)$ is a recommended list for mashup m, and $test(m)$ is a set of services that have interactions with m in test set.

$$\text{MRR} = \frac{1}{|M|}\sum_{m\in M}(\sum_{i\in test(m)}\frac{1}{rank(m,i)}), \tag{11}$$

where $rank(m,i)$ represents the position of service i in the recommended list for mashup m.

Table 2. Statistics of ProgrammableWeb dataset

Relations (X-Y)	Number of X	Number of Y	Number of (X-Y)	Ave. degrees of A/B
Mashup-API	6340	1399	13685	2.16/9.78
Mashup-Tag	6300	393	19013	3.02/48.38
Mashup-Content	6340	50	28267	4.46/565.34
API-Category	1367	323	3540	2.59/10.96
API-Content	1399	50	6652	4.75/133.04
API-Provider	1100	952	1100	1.00/1.16

4.3 Methods for Comparison

To demonstrate the effectiveness of the proposed service recommendation approach, we compare PaSRec with other four recommendation methods described as follows.

- **SVD**: This method is a traditional matrix factorization technique in recommender systems [11].
- **CF**: Collaborative filtering is a classical and widely used technique in recommender systems [13].
- **BPR-SVD**: This method learns SVD model using implicit feedback data with Bayesian Personalized Ranking (BPR) method proposed in [12].
- **BPR-kNN**: This method learns k-Nearest Neighbor (kNN) model using implicit feedback data with BPR method.

In the implement of our PaSRec, we select 6 meaningful meta-paths as shown in Table 1 for experiments. The parameter α in PaSRec is 0.001 and λ is set as 0.001 for the best performance. The parameters in those comparison methods are set with the best performances.

4.4 Learning Rate Decision

The learning rate constant α in the model learning algorithm is a key parameter, which can be determined experimentally. Given the training set, we fix the regularization parameter $\lambda = 0.001$ and set the number of iterations as 15. Then we run the learning algorithm with different values of $\alpha \in \{0.1, 0.01, 0.001, 0.0001\}$. Figure 4 records the change of loss values with the increasing number of iterations. It can be observed that the middle values of α (i.e., 0.01 and 0.001) are the relatively satisfactory selections, which is judged according to (i) the slow convergence of $\alpha = 0.0001$; and (ii) the big fluctuation of curve caused by $\alpha = 0.1$. Between 0.01 and 0.001, we choose $\alpha = 0.001$ in this paper, since its loss curve shown in Fig. 4 is more smooth, which can enhance the robustness of the learning algorithm.

Table 3. Performance comparison

Method	Precision@k			Recall@k			MRR@k		
	$k=3$	$k=5$	$k=10$	$k=3$	$k=5$	$k=10$	$k=3$	$k=5$	$k=10$
SVD	0.0747	0.0832	0.0585	0.1404	0.2659	0.3802	0.1217	0.1655	0.1880
BPR-SVD	0.1082	0.0944	0.0622	0.2148	0.2983	0.3988	0.2311	0.2637	0.2839
BPR-kNN	0.0737	0.0646	0.0497	0.1472	0.2053	0.3026	0.1245	0.1474	0.1707
CF	0.1648	0.1338	0.0915	0.2953	0.3975	0.5276	0.3609	0.4012	0.4334
PaSRec	**0.3421**	**0.2371**	**0.1338**	**0.6946**	**0.7704**	**0.8281**	**0.7051**	**0.7417**	**0.7625**
Improvement	51.83 %	43.57 %	31.61 %	57.49 %	48.40 %	36.29 %	48.82 %	45.91 %	43.16 %

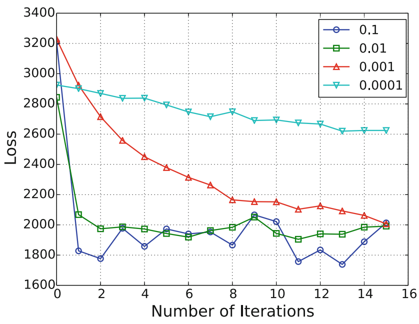

Fig. 3. Loss of PaSRec as a function of the number of iterations

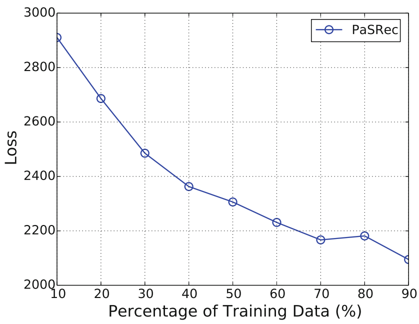

Fig. 4. Loss of PaSRec with the change of percentage of training data

4.5 Performance Comparison

We evaluate and compare all the methods listed in Sect. 4.3. Table 3 shows the performances of all 5 methods in the dataset crawled from PW, from which we can draw some observations (Fig. 3).

SVD, BPR-SVD and BPR-kNN achieve worse performance among all the baselines. SVD model is more appropriate for recommendation problem with explicit rating data since it is a matrix factorization based method. BPR-SVD introduces the idea of pairwise based ranking and combines the basic SVD model with BPR learning algorithm for implicit feedback. The improvements made by BPR-SVD compared to SVD respectively reach up to 30.96 %, 34.64 % and 47.34 % in terms of three metrics (precision@3 = 0.1082, recall@3 = 0.2148, MRR@3 = 0.2311). BPR-kNN achieves the worst performance due to the lack of extra information such as content and network information.

CF method outperforms other baseline methods overall. Essentially, this method leverages the relationships between mashups and services to measure the similarity. CF achieves precision@5 = 0.1388 (compared to 0.0832 with SVD, 0.0944 with BPR-SVD, and 0.0646 with BPR-kNN), and the similar results are shown in terms of other two metrics. The performance of CF method proves the usefulness of the information network constructed with mashups and services.

Our proposed service recommendation model PaSRec, which sufficiently leverages heterogeneous information network and the implicit feedback data, overwhelmingly beats all baseline methods. From Table 3, we can find that PaSRec improves precision@5, recall@5 and MRR@5 by 43.57 %, 48.40 % and 45.91 % compared to CF method. The obvious improvement demonstrates the effectiveness of the related heterogeneous information network and our proposed PaSRec approach can significantly improve the recommendation quality.

Moreover, we compare the proposed service recommendation approach PaSRec with all the baseline methods with respect to different amounts of training data. Specifically, we select first 10 %–90 % of the dataset as training data and assign the last 10 % for testing at each run. The results are presented in Figs. 4 and 5. Figure 4 shows the loss of PaSRec along with the change of percentage of training data. We can find that the loss obtained when the algorithm converges reduces with the increase of training data, which is quite reasonable. Figure 5 illustrates the performances of all methods based on three metrics. It is obvious that our PaSRec approach is superior to the other four methods with any percentage of training data. In addition, PaSRec makes significant improvements when the training data increases, which can be explained by following: more triple samples can be generated with the raise of training data proportion, which can improve the quality of parameters learning. In addition, it can be easily observed that SVD and BPR-SVD make nearly no improvement when the percentage of training data changes. We infer it is caused by the sparsity of the training data, which reaches 99.8 % in the whole dataset as mentioned above. It indicates that our proposed PaSRec can alleviate the data sparsity issue to a certain extent.

Overall, the proposed recommendation approach outperforms all comparison methods in the dataset crawled from PW. The empirical studies verify that exploiting heterogeneous information network can bring positive influences for service recommendation.

5 Related Work

In recent years, there is a surge of research works of service discovery and recommendation. Previous work can be mainly classified into several types, semantics-based, CF-based, QoS-based, and information network-based. The semantics-based approaches generally consider the semantic similarity of services which have been frequently introduced, and here we discuss the rest three types of service recommendation methods.

Inspired by the techniques used in recommender systems, many works started to introduce the widely employed collaborative filtering into service recommendation. Cao et al. [4] handled service recommendation problem with a cube model, based on which a standard deviation based hybrid collaborative Filtering approach is proposed. Sun et al. [14] designed a new similarity measure to compute similarity between services and proposed a normal recovery collaborative filtering method to recommend services for consumers. In [20], Zhong et al. provided a

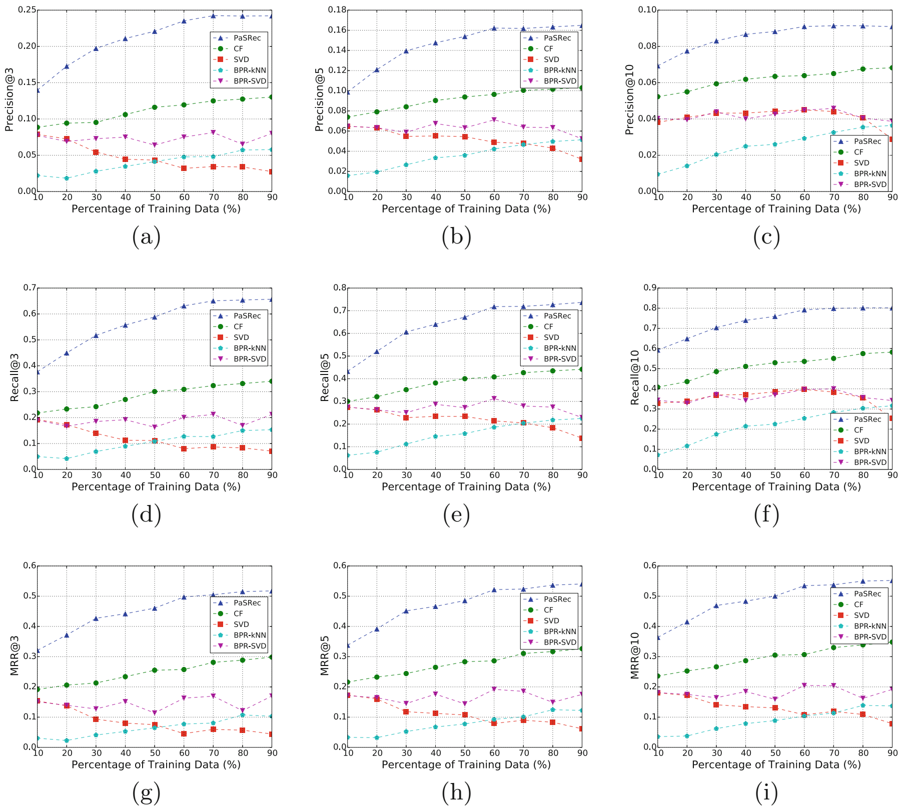

Fig. 5. Performance comparison in terms of precision, recall and MRR based on different percentages of training data.

time-aware service recommendation approach for mashup creation with the combination of service evolution, collaborative filtering and contend matching.

Furthermore, a number of works focus on QoS-based service recommendation. Zheng et al. [19] proposed a QoS prediction approach by combining traditional user-based and item-based collaborative filtering methods, and employ the predicted QoS for service recommendation. Chen et al. [6] designed a large-scale web service recommendation approach by employing the characteristic of QoS and achieved considerable improvement on the recommendation performance. However, QoS information is not always available.

Another direction in service recommendation is introducing information network analysis. In [3], Cao et al. proposed an framework to recommend mashup services, which considered users' interests mined from their usage history and the information network based on relations among mashup services, APIs and tags. Huang et al. [9] built a service network prediction method based on rank aggregation, and presented how to recommend potential compositions, top services and service chains by using this network prediction model. Maaradji et al. [10] introduced a framework named SoCo to provide dynamic recommendations for services discovery by using users' interactions and a social network built from interactions between users and services. Gao et al. [8] recommended services using manifold ranking algorithm which incorporate relationships between services and mashups.

However, most information network-based approaches only consider homogeneous relations between objects in the constructed service network. In this paper, we propose a service recommendation approach for mashup creation by sufficiently utilize heterogeneous information in the network including services, mashups, and their attributes.

6 Conclusion

In this paper, we propose a service recommendation approach for mashup creation called PaSRec, which exploits heterogeneous relationships in service related HIN. PaSRec takes advantages of different types of relationships between objects for the similarity computation under different semantic meanings. We design a recommendation model based on collaborative filtering and meta-path based similarities. A Bayesian ranking based optimization algorithm is applied to learn the model parameters with implicit feedback data. In the part of evaluation, we compare the proposed approach with several widely employed recommendation techniques, and the results show the superiority of our approach PaSRec. We also study the impact of learning rate in the process of model learning.

In our future work, we plan to use an online version of PaSRec to improve the efficiency of service recommendation, which makes it available for the approach to scale to a massive number of services, mashups, and the related attributes.

Acknowledgment. This research was partially supported by the Natural Science Foundation of China under grant of No. 61379119, National Science and Technology Supporting Program of China under grant of No. 2015BAH18F02, the Fundamental Research Funds for the Central Universities under grant of No. ZH2016007.

References

1. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. J. Mach. Learn. Res. **3**, 993–1022 (2003)
2. Bottou, L.: Large-scale machine learning with stochastic gradient descent. In: Lechevallier, Y., Saporta, G. (eds.) Proceedings of COMPSTAT 2010, pp. 177–186. Springer, Heidelberg (2010)
3. Cao, B., Liu, J., Tang, M., Zheng, Z., Wang, G.: Mashup service recommendation based on user interest and social network. In: 2013 IEEE 20th International Conference on Web Services (ICWS), pp. 99–106. IEEE (2013)
4. Cao, J., Wu, Z., Wang, Y., Zhuang, Y.: Hybrid collaborative filtering algorithm for bidirectional web service recommendation. Knowl. Inf. Syst. **36**(3), 607–627 (2013)
5. Chen, L., Wu, J., Zheng, Z., Lyu, M.R., Wu, Z.: Modeling and exploiting tag relevance for web service mining. Knowl. Inf. Syst. **39**(1), 153–173 (2014)
6. Chen, X., Zheng, Z., Liu, X., Huang, Z., Sun, H.: Personalized qos-aware web service recommendation and visualization. IEEE Trans. Serv. Comput. **6**(1), 35–47 (2013)
7. Dojchinovski, M., Kuchar, J., Vitvar, T., Zaremba, M.: Personalised graph-based selection of web APIs. In: Cudré-Mauroux, P., et al. (eds.) ISWC 2012. LNCS, vol. 7649, pp. 34–48. Springer, Heidelberg (2012). doi:10.1007/978-3-642-35176-1_3
8. Gao, W., Chen, L., Wu, J., Gao, H.: Manifold-learning based api recommendation for mashup creation. In: Proceedings of the 22nd IEEE International Conference on Web Services, pp. 432–439. IEEE (2015)
9. Huang, K., Fan, Y., Tan, W.: Recommendation in an evolving service ecosystem based on network prediction. IEEE Trans. Autom. Sci. Eng. **11**(3), 906–920 (2014)
10. Maaradji, A., Hacid, H., Daigremont, J., Crespi, N.: Towards a social network based approach for services composition. In: 2010 IEEE International Conference on Communications (ICC), pp. 1–5. IEEE (2010)
11. Paterek, A.: Improving regularized singular value decomposition for collaborative filtering. In: Proceedings of the KDD Cup Workshop, vol. 2007, pp. 5–8 (2007)
12. Rendle, S., Freudenthaler, C., Gantner, Z., Schmidt-Thieme, L.: Bpr: bayesian personalized ranking from implicit feedback. In: Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence, pp. 452–461. AUAI Press (2009)
13. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Item-based collaborative filtering recommendation algorithms. In: Proceedings of the 10th International Conference on World Wide Web, pp. 285–295. ACM (2001)
14. Sun, H., Zheng, Z., Chen, J., Lyu, M.R.: Personalized web service recommendation via normal recovery collaborative filtering. IEEE Trans. Serv. Comput. **6**(4), 573–579 (2013)
15. Sun, Y., Han, J.: Mining heterogeneous information networks: a structural analysis approach. ACM SIGKDD Explor. Newsl. **14**(2), 20–28 (2013)
16. Sun, Y., Han, J., Yan, X., Yu, P.S., Wu, T.: Pathsim: meta path-based top-k similarity search in heterogeneous information networks. Proc. VLDB Endowment **4**(11), 992–1003 (2011)

17. Wu, J., Chen, L., Feng, Y., Zheng, Z., Zhou, M.C., Wu, Z.: Predicting quality of service for selection by neighborhood-based collaborative filtering. IEEE Trans. Syst. Man Cybern. Syst. **43**(2), 428–439 (2013)
18. Yu, X., Ren, X., Sun, Y., Sturt, B., Khandelwal, U., Gu, Q., Norick, B., Han, J.: Recommendation in heterogeneous information networks with implicit user feedback. In: Proceedings of the 7th ACM Conference on Recommender Systems, pp. 347–350. ACM (2013)
19. Zheng, Z., Ma, H., Lyu, M.R., King, I.: Qos-aware web service recommendation by collaborative filtering. IEEE Trans. Serv. Comput. **4**(2), 140–152 (2011)
20. Zhong, Y., Fan, Y., Huang, K., Tan, W., Zhang, J.: Time-aware service recommendation for mashup creation in an evolving service ecosystem. In: Proceedings of the 21st IEEE International Conference on Web Services (ICWS), pp. 25–32. IEEE (2014)