Mining Product Relationships for Recommendation Based on Cloud Service Data

Yuanchun Jiang(✉), Cuicui Ji, Yang Qian, and Yezheng Liu

School of Management, Hefei University of Technology,
Hefei 230009, Anhui, People's Republic of China
ycjiang@hfut.edu.cn

Abstract. With the rapid growth of cloud services, it is more and more difficult for users to select appropriate service. Hence, an effective service recommendation method is need to offer suggestions and selections. In this paper, we propose a two- phase approach to discover related cloud services for recommendation by jointly leveraging services' descriptive texts and their associated tags. In Phase 1, we use a non-parametric Bayesian method, DPMM to classify a large number of cloud services into an optimal number of clusters. In Phase 2, we recommend a personalized PageRank algorithm to obtain more related services for recommendation among the massive cloud service products in the same cluster. Empirical experiments on a real data set show that the proposed two-phase approach is more successful than other candidate methods for service clustering and recommendation.

Keywords: Cloud service · Cluster · DPMM · Personalized PageRank

1 Introduction

The emerging cloud computing technology offers a new computing environment which enables us to access computing resources, storage and network infrastructure through the Internet without up-front infrastructure costs [1, 2]. With the rapid development of cloud computing technology, many information resources are wrapped and released as cloud services on public servers [3] and companies such as Google, IBM, Microsoft and Amazon opt to provide cloud service products through the public servers [4]. Because a public server usually has massive cloud service products, cloud service recommendation is necessary to provide right services to right users.

Many methods have been proposed to construct selection and ranking models for service products. Among them, QoS (quality of services)-based service selection model [7–9], AHP-based cloud service ranking model [10], trust-aware service selection model [11] and selection method based on collaborative filtering mechanism [12] are popular models. In these models, quantitative criteria are employed to evaluate service quality and the textual information (e.g. service descriptions) is rarely considered.

F. Daniel et al. (Eds.): BPM 2018 Workshops, LNBIP 342, pp. 374–386, 2019.
https://doi.org/10.1007/978-3-030-11641-5_30

This paper proposes an approach to recommend cloud services with the textual description information and tags. We first propose a non-parametric Bayesian model to cluster cloud services. The model is constructed based on Dirichlet process mixture model (DPMM), which can infer the number of clusters automatically without specifying the number of clusters in advance and work well with large-scale datasets [6]. Then, we proposed a personalized PageRank algorithm to generate cloud service rankings based on service tags and clusters we obtained.

The major contributions of this paper are summarized as follows:

(1) This paper employs textual information to recommend cloud services. Compared with service title and click records, the textual information implies rich service features which can help us understand the service functions and make accurate recommendations. To the best of our knowledge, this is the first research to recommend cloud services based on textual description information.
(2) We propose a nonparametric DPMM to classify cloud services into an optimal number of clusters while the number of clusters is identified endogenously. To cluster cloud services, managers usually do not have knowledge on how many clusters exist and which cloud services belong to which cluster. The nonparametric model is particularly suitable for cloud service clustering because it requires no predefined number of clusters, instead it optimizes the number automatically based on data.
(3) We propose a personalized PageRank algorithm to rank the cloud services in each cluster obtained by the proposed DPMM method. The personalized PageRank algorithm can rank cloud services by tags and textual descriptions, and recommend services to meet users' personalized requirements.
(4) We conduct a set of experiments based on a real-world dataset from Programmable Web. Our experiment shows, compared with the baseline methods, the proposed model achieves a significant improvement.

The remainder of this paper is organized as follows: Sect. 2 reviews the related works in literature. Section 3 introduces the proposed approach. Then, in Sect. 4, carries out experiments on some real-world data sets to validate the performance of our approach. Finally, we conclude our work by presenting summary and future directions in Sect. 5.

2 Related Work

2.1 Cloud Service Recommendation

Since Weiss [13] first proposed the concept of cloud computing, research on cloud computing is becoming more and more popular. Formerly, most of the researches on service selection and recommendation were based on the QoS values. However, sometimes it is difficult for us to get the exact QoS values, so scholars began to focus on evaluating and predicting the missing QoS values [14]. In [7], they presented an evaluation approach of QoCS (Quality of Cloud Service) in service-oriented cloud computing which combines the cloud users' preferences evaluation of cloud service

providers employing fuzzy synthetic decision with uncertainty calculation of cloud services based on monitored QOCS data for cloud users. Han [8] proposed a recommendation system which creates ranks of different cloud services based on the network QoS and Virtual Machine (VM) platform factors of different cloud providers. Considering that collaborative filtering technology (CF) is the most mature and widely used technology in the recommend system, CF is also widely used in service recommendation based on QoS [12, 15]. In reality, collaborative filtering is vulnerable to the sparse data and is extremely time-consuming with the enlargement of data.

In [16], the author introduced the cloud broker who is responsible for the service selection and developed impactful service selection algorithms to rank potential service providers and aggregate them. Yu [17] put forward a new train of thought that integrates Matrix Factorization (MF) with decision tree learning to bootstrap service recommendation systems. Ding [18] proposed a ranking-oriented prediction method and the method consists of two parts: ranking similarity estimation and cloud service ranking prediction that takes the customer's attitude and expectations for service quality into account.

2.2 Text Clustering Based on Topic Model

Clustering is a widely researched data mining problem in text domain and the popular method in probabilistic description clustering is topic modeling [19]. Topic model is a probabilistic generation model for finding abstract topics in a series of descriptions and it has been widely applied in information retrieval, natural language processing and machine learning.

Topic models, such as Probabilistic Latent Semantic Analysis (PLSA), has been applied to service discovery [20]. Zhang [22] applied the LDA model to cluster the services and extracted service goals from the textual descriptions of services so that they can help users improve their initial queries by recommending similar service goals. The above service clustering models need to specify the number of clusters in advance. Given the limitations of managers' expertise, time and energy, they may not be flexible enough.

Existing cloud service selection approaches rarely consider some important data sources, such as tags, which have been proved to be very powerful in many domains and have been widely used in search engines, social medias, such as Facebook [23].

For cloud service recommendation, we develop a novel model consisting of two phases: cloud services clustering based on Dirichlet Process Multinomial Mixture model (DPMM) and cloud service ranking based on service tags and clusters we obtained. Details of our model are discussed next.

3 The Proposed Model

Our cloud service recommendation system recommends a set of related cloud service products for users by jointly leveraging the textual description information and tag data. Our approach consists of two main phases. In Phase 1, we propose a non-parametric DPMM model to cluster cloud services based on the textual information.

In Phase 2, we propose the Personalized PageRank algorithm to rank the cloud services in each cluster obtained by the proposed DPMM method. The approach framework is illustrated in Fig. 1.

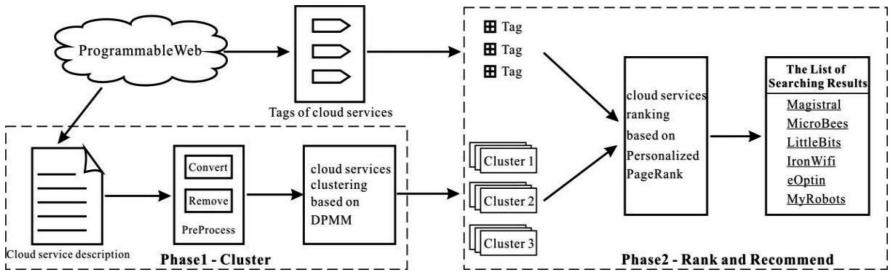

Fig. 1. The framework of the cloud services recommendation.

3.1 Phase1-The Topic Modeling of Web Cloud Service Using DPMM

The DPMM Model. The DPMM is a powerful non-parametric Bayesian method [24] which means that the method can cluster according to the actual situation without specifying the number of clusters in advance. The probabilistic graph of DPMM is shown in Fig. 2 Here, d represents each cloud service description. z represents the cluster label of cloud service description. Multinomial Φ is distributed according to Dirichlet prior β. Multinomial Θ is distributed according to stick-breaking prior α (Table 1).

Table 1. Notations

D	Number of the whole cloud service descriptions set
V	Size of the vocabulary
$\boldsymbol{d}$	Descriptions in the cloud service descriptions set
$\boldsymbol{z}$	Cluster labels of each description
m_z	Number of descriptions in cluster z
N_d	Number of words in description d
N_d^{ω}	Number of occurrences of word ω in description d
n_z	Number of words in cluster z
n_z^{ω}	Number of occurrences of word ω in cluster z

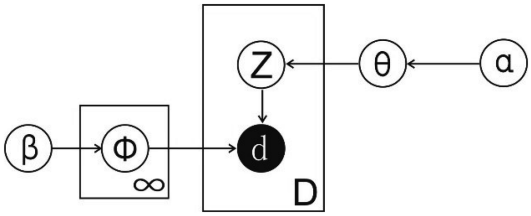

Fig. 2. The probabilistic graph of DPMM.

The generative process of our DPMM is described as follows:

(1) When generating description, the DPMM first selects the cluster $z_d|\Theta \sim Multinomial(\Theta)$ for description d and z_d is distributed according to multinomial Θ.
(2) Then, generating the description $d|z_d, \{\Phi_k\}_{k=1}^{\infty} \sim Multinomial(\Phi_z)$ by the selected the cluster z_d from multinomial Φ_{z_d}.
(3) Generating the weight vector of clusters, $\Theta|\alpha \sim GEM(1, \alpha)$ by a stick-breaking construction with the hyper-parameter α.
(4) Generating the cluster parameters $\Phi_z|\beta \sim Dirichlet(\beta)$ by a Dirichlet distribution with a hyper-parameter β.

Choosing an Existing Cluster. To classify description d to an existing cluster z, the conditional probability can be calculated as follows:

$$\begin{aligned} & p(z_d = z|\boldsymbol{z}_{\neg d}, \boldsymbol{d}, \alpha, \beta) \\ \propto\ & p(z_d = z|\boldsymbol{z}_{\neg d}, \boldsymbol{d}_{\neg d}, \alpha, \beta) p(d|z_d = z, \boldsymbol{z}_{\neg d}, \boldsymbol{d}_{\neg d}, \alpha, \beta) \\ \propto\ & p(z_d = z|\boldsymbol{z}_{\neg d}, \alpha) p\left(d|z_d = z, \boldsymbol{d}_{z,\neg d}, \beta\right) \end{aligned} \tag{1}$$

Here, we apply the Bayes Rule in Eq. (1) and use the properties of D-Separation [24] in Eq. (1) where $\neg d$ means the description d does not include and $d_{z,\neg d}$ represents other descriptions allocated to cluster z.

The first expression in Eq. (1) means the probability of description d choosing cluster z given the cluster assignments of other descriptions. It can be derived as follows:

$$\begin{aligned} & p(z_d = z|\boldsymbol{z}_{\neg d}, \alpha) \\ =\ & \int p(\Theta|\boldsymbol{z}_{\neg d}, \alpha) p(z_d = z|\Theta) d\Theta \\ =\ & \int Dir(\Theta|\boldsymbol{m}_{\neg d}) Mult(z_d = z|\Theta) d\Theta \\ =\ & \frac{m_{z,\neg d}}{D - 1 + \alpha} \end{aligned} \tag{2}$$

The second expression in Eq. (1) indicates a predictive probability of description d given $\boldsymbol{d}_{z,\neg d}$. We can derive the second expression as follows:

$$\begin{aligned} & p\left(d|z_d = z, \boldsymbol{d}_{z,\neg d}, \beta\right) \\ =\ & \int p\left(\Phi_z|\boldsymbol{d}_{z,\neg d}, \beta\right) p(d|\Phi_z, z_d = z) d\Phi_z \\ =\ & \int Dir\left(\Phi_z|\boldsymbol{n}_{z,\neg d} + \beta\right) \prod\nolimits_{\omega \in d} Mult(\omega|\Phi_z) d\Phi_z \end{aligned}$$

$$= \frac{\prod_{\omega \in d} \prod_{j=1}^{N_d^{\omega}} \left(n_{z,\neg d}^{\omega} + \beta + j - 1 \right)}{\prod_{i=1}^{N_d} \left(n_{z,\neg d} + V\beta + i - 1 \right)} \tag{3}$$

Now we can get the probability of description d choosing an existing cluster z when we know the information of other descriptions and their cluster assignments as follows:

$$p(z_d = z | \boldsymbol{z}_{\neg d}, \boldsymbol{d}, \alpha, \beta) \propto \frac{m_{z,\neg d}}{D - 1 + \alpha} * \frac{\prod_{\omega \in d} \prod_{j=1}^{N_d^{\omega}} \left(n_{z,\neg d}^{\omega} + \beta + j - 1 \right)}{\prod_{i=1}^{N_d} \left(n_{z,\neg d} + V\beta + i - 1 \right)} \tag{4}$$

Choosing a New Cluster. We denote a new cluster as $K + 1$, the conditional probability description d belonging to a new cluster z can be calculated as follows:

$$\begin{aligned} & p(z_d = K + 1 | \boldsymbol{z}_{\neg d}, \boldsymbol{d}, \alpha, \beta) \\ & \propto p(z_d = K + 1 | \boldsymbol{z}_{\neg d}, \boldsymbol{d}_{\neg d}, \alpha, \beta) p(d | z_d = K + 1, \boldsymbol{z}_{\neg d}, \boldsymbol{d}_{\neg d}, \alpha, \beta) \\ & \propto p(z_d = K + 1 | \boldsymbol{z}_{\neg d}, \alpha) p(d | z_d = K + 1, \boldsymbol{d}_{z,\neg d}, \beta) \end{aligned} \tag{5}$$

We can derive the first expression in Eq. (5) as follows:

$$p(z_d = K + 1 | \boldsymbol{z}_{\neg d}, \alpha) = 1 - \sum_{k=1}^{K} p(z_d = k | \boldsymbol{z}_{\neg d}, \alpha) = \frac{\alpha}{D - 1 + \alpha} \tag{6}$$

Then, the second expression in Eq. (5) can be derived as follows:

$$\begin{aligned} & p\left(d | z_d = K + 1, \boldsymbol{d}_{z,\neg d}, \beta\right) \\ & = \int Dir(\Phi_{K+1} | \beta) \prod\nolimits_{\omega \in d} Mult(\omega | \Phi_{K+1}) d\Phi_{K+1} \end{aligned}$$

$$= \frac{\prod_{\omega \in d} \prod_{j=1}^{N_d^{\omega}} (\beta + j - 1)}{\prod_{i=1}^{N_d} (V\beta + i - 1)} \tag{7}$$

Finally, we can get the probability of description d choosing a new cluster:

$$p(z_d = K + 1 | \boldsymbol{z}_{\neg d}, \boldsymbol{d}, \alpha, \beta) \propto \frac{\alpha}{D - 1 + \alpha} * \frac{\prod_{\omega \in d} \prod_{j=1}^{N_d^{\omega}} (\beta + j - 1)}{\prod_{i=1}^{N_d} (V\beta + i - 1)} \tag{8}$$

After Gibbs Sampling, we can get the representation of clusters by Φ. For each cluster z, we can derive the posterior of Φ_z as follows:

$$p(\Phi_z|\boldsymbol{d}, \boldsymbol{z}, \alpha, \beta) = \frac{1}{\Delta(\boldsymbol{n}_z + \beta)} \prod_{\omega=1}^{V} \Phi_{z,\omega}^{n_z^w + \beta - 1} = Dir(\Phi_z|\boldsymbol{n}_z + \beta) \tag{9}$$

where $\boldsymbol{n}_z = \left\{n_z^{\omega}\right\}_{\omega=1}^{V}$.

Using the expectation of the Dirichlet distribution, we can infer $\Phi_{z,\omega}$ as follows:

$$\Phi_{z,\omega} = \frac{n_z^{\omega} + \beta}{n_z + V\beta} \tag{10}$$

3.2 Phase2-Cloud Service Ranking Using Personalized PageRank Algorithm

In Phase1, cloud service products are classified into different clusters based on the proposed DPMM algorithm. However, it is still difficult to recommend the appropriate services to users among the massive cloud service products in same cluster. Here we propose the Personalized PageRank algorithm [25] to rank the cloud service products in same cluster.

The proposed Personalized PageRank algorithm employs random walk to rank nodes of a graph consisting of cloud services and tags as nodes and it is a variation of PageRank [26]. PageRank model random-walk process on the web graph composed of numerous pages as nodes and during the process a random surfer will stay the current page i as the next step with probability 1-ε and access to other pages with probability ε. Once the surfer decides to access to other pages, he will uniformly choose a hyperlink contained in the current page. Thus, the random access probability of each page can be calculated as:

$$PR(i) = \frac{(1-\varepsilon)}{N} + \varepsilon \sum_{j \in in(i)} \frac{PR(j)}{|out(j)|} \tag{11}$$

where $PR(i)$ represents the probability of a node to be selected. N is the number of all nodes. $in(i)$ represents the node set pointing to node i and $out(j)$ represents the node set pointed by node j. The first part of Eq. (11) means the probability of the surfer staying on the current page i when it is the starting pointing and the second part means the probability of the surfer jumping back to the current page i by clicking on other pages.

For calculating the access probability of a cloud service node in Personalized PageRank, we substitute $\frac{(1-\varepsilon)}{N}$ to $(1-\varepsilon)\gamma_i$ where γ_i is 1 if the node is our target service and others γ_i is 0. In this way, we can get the relevance of all services relative to the target cloud service.

The Personalized PageRank algorithm will quickly converge to a stable state by recursively calculating and updating the probability of each node. As a result, we can

use the value $PR(i)$ of each node as the rank score and recommend Top-k cloud services by selecting cloud service nodes in the node set for the target cloud service.

4 Experiments and Results

4.1 Data Sets and Preprocessing

Experimental data is obtained from Programmable Web, which provides detailed profile information of massive cloud services. The information of cloud services contains services' name, descriptive text and tags. Our data set consists of 799 cloud services and 790 distinct tags. Many tags exist in multiple services, totally 2,745 tags are included in these services. In addition, the average length (i.e., number of words) of each text description is 71.

Because the raw data of the descriptive texts are very noisy, we conduct the following preprocessing: (1) Convert letters into lowercase; (2) Remove meaningless words such as stop words, low frequency words, high frequency words and characters not in Latin.

4.2 Baseline Methods

In the experimental study, we compare DPMM with two typical service clustering methods for service texts nowadays. The details of them are shown below.

K-Means: K-means [27] is probably the most widely used method for clustering. Before being able to utilizing k-means on a set of text descriptions, the texts must be represented as mutually comparable vectors. To achieve this task, each text description can be represented using the TF-IDF score [28].

LDA: We consider the topics found by LDA [29] as clusters and assign each cloud service to the cluster with the highest value in its topic proportion vector.

Some automatic evaluation metrics are proposed in the past few years to measure the quality of the clusters discovered. The typical metric is the coherence score [30], which indicates that a cluster (or topic) is more coherent if the most probable words in it co-occurring more frequently in the corpus. We can calculate the coherence value of a cluster k as follows:

$$C_k = \sum_{m=2}^{M} \sum_{l=1}^{m-1} log \frac{D\left(v_m^{(k)}, v_l^{(k)}\right)}{D\left(v_l^{(k)}\right)} \tag{12}$$

where $v_m^{(k)}$ is one of the most M probable words in cluster k; $D\left(v_l^{(k)}\right)$ represent the description frequency of word l; and $D\left(v_m^{(k)}, v_l^{(k)}\right)$ is the co-description frequency of words.

4.3 Parameter Setting

For DPMM, we set $K = 1$, $\beta = 0.01$. We also assume $Gamma(1, 1)$ priors over the parameters α_0 that can be optimized in Gibbs sampling procedure [31]. In LDA model, we place $\alpha = 50/k$ and $\beta = 0.1$ where K is the number of topics assumed by LDA.

4.4 Results of Service Clustering

Before presenting the final comparisons of baseline methods, we first show the results of cloud services clustering discovered by DPMM. We run Gibbs samplers for 3000 iterations and finally obtain 26 clusters. Figure 3 shows our cluster results with word cloud. Our methods exhibit effectiveness in grouping related cloud services and semantically coherent words together. For instance, Cluster 1 includes cloud-based services designed to handle description, optical character recognition (OCR), and email formats. Cluster 2 offers cloud-based software-as-a-service platforms for enterprise or business. Cluster 3 presents dedicated servers and cloud hosting services for computing. Cluster 4 is about Internet of Thing (IoT) platforms for connections between the clouds and different kinds of devices or appliances. Cluster 5 is about communication technologies that can integrate voice, messaging and email into application.

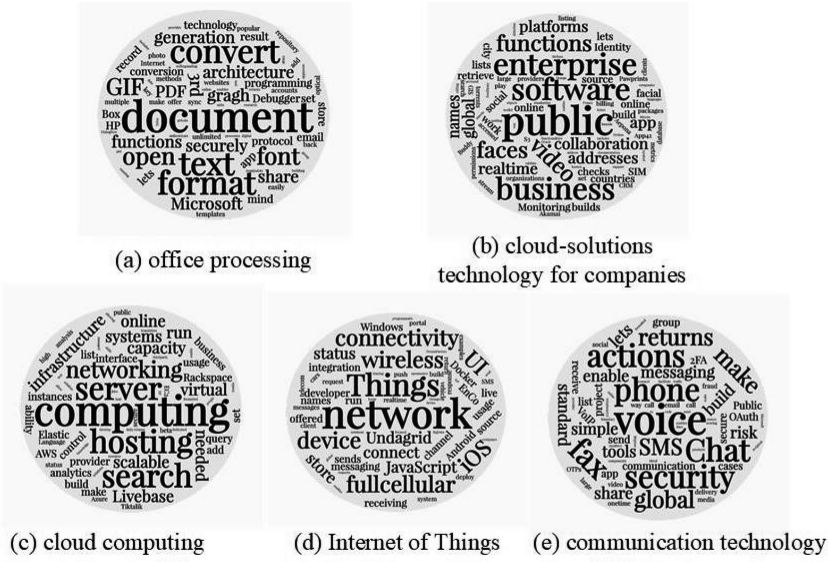

Fig. 3. Word clouds of the cluster results.

To evaluate the overall quality of a cluster set, we analyze the average coherence score, namely $\frac{1}{K}\sum_{k=1}^{K} C_k$, for each method. The result is listed in Table 2, where the number of top words ranges from 5 to 25. As shown in Table 2, we find that DPMM obtains the highest coherence score in all the settings. It demonstrates that the DPMM is able to achieve better performance for cluster quality compared with K-means and LDA.

Table 2. Comparison of coherence scores among different methods. A larger score indicates better performance for cluster quality.

Method	Kmeans (K = 10)	Kmeans (K = 20)	Kmeans (K = 30)	LDA (K = 10)	LDA (K = 20)	LDA (K = 30)	DPMM
Top5	−9.16	−7.03	−8.21	−8.04	−8.06	−10.19	**−5.18**
Top10	−64.73	−53.79	−54.66	−55.48	−60.26	−65.30	**−43.39**
Top15	−179.18	−152.96	−155.78	−145.70	−176.94	−178.71	**−142.01**
Top20	−338.39	−314.32	−323.83	−299.303	−356.74	−360.40	**−308.05**
Top25	−562.25	−530.30	−540.17	−521.32	−599.56	−602.32	**−522.03**

4.5 Results of Recommendation

In this section, we show the results of cloud services recommendation. Using personalized PageRank algorithm for each cluster discovered by DPMM, we obtain a ranking list for each cloud service based on the relevance score. For assessing the performance of our results, we adopt Jaccard coefficient, which is an alternative approach to measuring the correlation between products [32, 33]. The Jaccard coefficient is defined as:

$$Jaccard(A, B) = \frac{|d_A \bigcap d_B|}{|d_A \bigcup d_B|} \tag{13}$$

Where A is the given product and B the recommended product; d_A and d_B are the textual descriptions of product A and B respectively. $d_A \bigcap d_B$ is the intersection between two sets d_A and d_B. Thus $d_A \bigcap d_B$ reveals all words which are in both sets. $d_A \bigcup d_B$ is the union between two sets d_A and d_B, which represents all words in two sets.

In our tasks, we calculate the averaged Jaccard coefficient of different recommendation lists which are obtained by three methods (Cosine similarity with TF-IDF on textual descriptions, Personalized PageRank on tags, our two-phase approach by jointly leveraging textual descriptions and tags). Each recommendation list contains L highest recommended cloud service resulting. For a given L, the result with a higher averaged Jaccard coefficient is better, and vice versa. The averaged Jaccard coefficient for some typical lengths of recommendation list are shown in Fig. 4, as shown in the Figure, our recommendation results achieve better performance than other two methods, which strongly guarantee the validity of our two-phase approach.

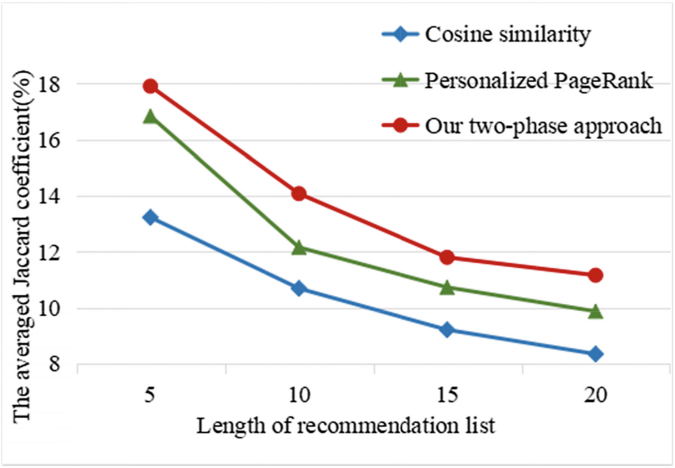

Fig. 4. Comparison of the averaged Jaccard coefficient of different methods.

5 Conclusion

In this paper, we have presented a novel two-phase method by utilizing service text descriptions and tags, to extract latent relations among different cloud services, to generate relevant cloud service recommendation results for aiding users in discovering the available combination of cloud services. Our method is designed to successfully address the cloud service clustering and recommendation. With experiments on a real-world dataset consisting of 799 cloud services and 790 distinct tags obtained from Programmable Web, we demonstrate the effectiveness of this method.

References

1. Michael, A., Fox, A., et al.: Above the clouds: a Berkeley view of cloud computing. Electr. Eng. Comput. Sci. **53**(4), 50–58 (2009). EECS Department University of California Berkeley
2. Katzan, H.: Cloud software service: concepts, technology, economics. Serv. Sci. **1**(4), 256–269 (2013)
3. Chan, H., Chieu, T.: Ranking and mapping of applications to cloud computing services by SVD. In: Network Operations and Management Symposium Workshops, pp. 362–369. IEEE (2010)
4. Marston, S., Li, Z., Bandyopadhyay, S., et al.: Cloud computing - the business perspective. In: Hawaii International Conference on System Sciences, pp. 1–11. IEEE Computer Society (2011)
5. https://en.wikipedia.org/wiki/Mashup_(web_application_hybrid)
6. Yin, J., Wang, J.: A model-based approach for text clustering with outlier detection. In: IEEE, International Conference on Data Engineering, pp. 625–636. IEEE (2016)
7. Wang, S., Liu, Z., Sun, Q., et al.: Towards an accurate evaluation of quality of cloud service in service-oriented cloud computing. J. Intell. Manuf. **25**(2), 283–291 (2014)

8. Han, H., Mehedi, M., et al.: Efficient service recommendation system for cloud computing market. Commun. Comput. Inf. Sci. **63**, 839–845 (2009)
9. Newton, P.C., Arockiam, L.: A Novel Prediction Technique to Improve Quality of Service (QoS) for Heterogeneous Data Traffic. Springer New York, Inc., New York (2011)
10. Garg, S.K., Versteeg, S., Buyya, R.: A framework for ranking of cloud computing services. Future Gener. Comput. Syst. **29**(4), 1012–1023 (2013)
11. Kong, D., Zhai, Y.: Trust based recommendation system in service-oriented cloud computing. In: International Conference on Cloud and Service Computing, pp. 176–179. IEEE Computer Society (2012)
12. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. In: Multimedia Services in Intelligent Environments, pp. 734–749. Springer International Publishing (2013)
13. Weiss, A.: Computing in the clouds. Networker **11**(4), 16–25 (2007)
14. Ding, S., Xia, C., Wang, C., et al.: Multi-objective optimization based ranking prediction for cloud service recommendation. Decis. Support Syst. **101**, 106–114 (2017)
15. Li, J., Zeng, X., Xia, J., et al.: Recent advances in approaches of Web service selection based on QoS. Appl. Res. Comput. (2015)
16. Sundareswaran, S., Squicciarini, A., Lin, D.: A brokerage-based approach for cloud service selection. In: IEEE International Conference on Cloud Computing, pp. 558–565. IEEE (2012)
17. Yu, Q.: Decision tree learning from incomplete QoS to bootstrap service recommendation. In: IEEE International Conference on Web Services, pp. 194–201. IEEE (2012)
18. Ding, S., Wang, Z., Wu, D., et al.: Utilizing customer satisfaction in ranking prediction for personalized cloud service selection. Elsevier Science Publishers B. V. (2017)
19. Aggarwal, C.C., Zhai, C.X.: A Survey of Text Clustering Algorithms. Mining Text Data, pp. 77–128. Springer, US (2012)
20. Ma, J., He, J., He, J.: Efficiently finding web services using a clustering semantic approach. In: International Workshop on Context Enabled Source and Service Selection, Integration and Adaptation: Organized with the, International World Wide Web Conference, p. 5. ACM (2008)
21. Chen, L., Wang, Y., Yu, Q., Zheng, Z., Wu, J.: WT-LDA: user tagging augmented LDA for web service clustering. In: Basu, S., Pautasso, C., Zhang, L., Fu, X. (eds.) ICSOC 2013. LNCS, vol. 8274, pp. 162–176. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-45005-1_12
22. Zhang, N., Wang, J., He, K., et al.: An approach of service discovery based on service goal clustering. In: IEEE International Conference on Services Computing, pp. 114–121. IEEE (2016)
23. Lin, M., Cheung, D.W.: An automatic approach for tagging Web services using machine learning techniques (2016)
24. Bishop, C.M.: Pattern Recognition and Machine Learning. Information Science and Statistics. Springer New York, Inc., New York (2006)
25. Kleinberg, J.M.: Authoritative sources in a hyperlinked environment. J. ACM **46**(5), 604–632 (1999)
26. Kamvar, S.D., Haveliwala, T.H., Manning, C.D., et al.: Extrapolation methods for accelerating PageRank computations. In: International Conference on World Wide Web. pp. 261–270. ACM (2003)
27. Hartigan, J.A., Wong, M.A.: Algorithm AS 136: a k-means clustering algorithm. J. Roy. Stat. Soc.. Ser. C (Appl. Stat.) **28**(1), 100–108 (1979)
28. Larson, R.R.: Introduction to information retrieval. J. Am. Soc. Inform. Sci. Technol. **61**(4), 852–853 (2010)

29. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. J. Mach. Learn. Res. **3**, 993–1022 (2003)
30. Mimno, D., Wallach, H.M., Talley, E., et al.: Optimizing semantic coherence in topic models. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 262–272. Association for Computational Linguistics (2011)
31. Escobar, M.D., West, M.: Bayesian density estimation and inference using mixtures. J. Am. Stat. Assoc. **90**(430), 577–588 (1995)
32. Netzer, O., Feldman, R., Goldenberg, J., et al.: Mine your own business: Market-structure surveillance through text mining. Mark. Sci. **31**(3), 521–543 (2012)
33. Humphreys, A., Jen-Hui Wang, R.: Automated text analysis for consumer research. J. Consum. Res. (2017)