Hindawi
Mathematical Problems in Engineering
Volume 2020, Article ID 5670215, 8 pages
https://doi.org/10.1155/2020/5670215

Research Article

Leverage Label and Word Embedding for Semantic Sparse Web Service Discovery

Chengai Sun[1,2], **Liangyu Lv**[1,2], **Gang Tian**[1,2], **Qibo Wang**[1,2], **Xiaoning Zhang**[1,2] and **Lantian Guo**[3]

[1]*College of Computer Science and Engineering, Shandong University of Science and Technology, Qingdao, China*
[2]*Key Laboratory for Wisdom Mine Information Technology of Shandong Province, Shandong University of Science and Technology, Qingdao, China*
[3]*School of Automation and Electronic Engineering, Qingdao University of Science and Technology, Qingdao, China*

Correspondence should be addressed to Gang Tian; tiangang@whu.edu.cn

Received 15 November 2019; Revised 30 January 2020; Accepted 28 February 2020; Published 24 March 2020

Guest Editor: Weifeng Pan

Information retrieval-based Web service discovery approach suffers from the semantic sparsity problem caused by lacking of statistical information when the Web services are described in short texts. To handle this problem, external information is often utilized to improve the discovery performance. Inspired by this, we propose a novel Web service discovery approach based on a neural topic model and leveraging Web service labels. More specifically, words in Web services are mapped into continuous embeddings, and labels are integrated by a neural topic model simultaneously for embodying external semantics of the Web service description. Based on the topic model, the services are interpreted into hierarchical models for building a service querying and ranking model. Extensive experiments on several datasets demonstrated that the proposed approach achieves improved performance in terms of F-measure. The results also suggest that leveraging external information is useful for semantic sparse Web service discovery.

1. Introduction

In the era of Big Data, a growing number of business enterprises worldwide are driven to deploy their business applications into Web services in both intranet and internet [1, 2]. A number of registry centers, such as, Programmableweb (http://www.programmableweb.com) and Mashape (https://www.mashape.com/) and business enterprises, have built their own service discovery mechanism to provide a convenient way to access these Web services. The search engine-based approaches are widely adopted among these registries. However, the discovery method-based searching engine technology which mainly focuses on keyword-based matching may result in the poor recall problem due to lacking of keywords in Web service descriptions, using of synonyms, or variations of keywords [3].

Two kinds of methods are often adopted to alleviate the poor recall problem in discovering nonsemantic Web services [4, 5]. The first one is to perform a broad searching and get a potentially large number of Web services which may not really be interest to users. The second one is to cluster services into similar functional clusters using the descriptions of Web services to enhance the capability of the search engine. Since this kind of method can effectively reduce the search space, it has attracted higher attentions from researchers [3–7].

There are some new issues emerged when using the aforementioned Web service discovery approaches in recent years. One is the semantic sparsity problem resulting from short text descriptions of Web services that there is no sufficient information to express the full semantics of the Web service. The current Web service marketplaces often briefly describe the main functions, the providers, and the types of a Web service using short sentences which do not contain enough statistic information so as to hinder effective similarity computing and pose challenges to traditional

service retrieval approaches [8, 9]. Faced with this issue, transfer of external knowledge to enrich the semantic representation of short text documents has been proposed such as Tian et al. [5] transfer external knowledge by using Gaussian LDA and the word embedding model from auxiliary information to enhance the semantics of the Web services.

Inspirited by these excellent findings, we propose to introduce the word embeddings which have been shown to capture lexico-semantic regularities in the language. In the embedding space, words with similar syntactic and semantic properties are found to be close to each other [10]. Thus, this feature is particularly suitable to solve the problems of using synonyms/variations of keywords in the query. Furthermore, the context information such as the co-occurrence information in the word embeddings can be effectively used to enrich the semantics of a document. Inspired by this, we propose to introduce word embedding to handle the semantic sparsity problem in the discovery of Web service.

To enhance the clustering performance, extensive research has been carried out on category information [11]. Inspired by this, some topic models can directly integrate these information into the generative process of a topic model to improve topic quality and cluster accuracy. Some excellent work had been done to leverage external meta-information to enhance the topic model [12].

According to the above description, we propose a label-aided neural topic model (LNTM) derived from Gaussian LDA [13] which leverages word embeddings and external label information to improve Web service discovery.

Our main contributions are as follows:

(1) We presented an approach that leverages pretrained word embeddings to enrich the semantics of Web service descriptions

(2) We proposed a label-augmented neural topic model to retrieve the Web services based on word embeddings and categories of the Web services

(3) We experimentally illustrate that the proposed approach outperforms several other approaches with higher evaluation metrics

2. Related Work

Web service discovery provides a mechanism to discover relevant services from different service registries. Base on the description method of services, the Web service discovery can be generally divided into two categories: semantic-based and nonsemantic-based. For instance, the Ontology Web Language for Services- (OWL-S-) based service is a typically semantic description language. In contrast, WSDL, Web Application Description Language (WADL), and natural language are typical nonsemantic description languages. Semantic-based approaches mainly focus on high-level match-making [14, 15], whereas nonsemantic-based discovery methods utilize information retrieval techniques [3–7]. In the proposed approach, we concentrate on nonsemantic Web services.

The nonsemantic-based discovery approaches are fairly different due to different description languages. For example, Elgazzar et al. [3] preprocessed the WSDL document to extract content, types, messages, ports, and service name as main features for the discovery method and utilized information retrieve approach to enhance Web service discovery. The WSDL documents need be preprocessed to construct the features for representing the Web service. If the Web services use different description languages, the WSDL-based methods must be adjusted for the discovery process. In this paper, we focus on the discovery of Web services which have shorter description and may contain less features compared with the services with sufficient information files. Therefore, the above methods may fail to work since they lack ways to handle the semantic sparsity problem.

Several studies have found that it is helpful to leverage external information to handle the semantic sparse problems of information retrieval approach [4, 8, 16]. Chen et al. proposed an augment LDA model to utilize both WSDL and tags for Web service discovery so as to provide effective Web service clustering [16]. There are also different methods to handle with the semantic sparsity problem. For example, Hu et al. proposed to enhance the short text cluster by leveraging world knowledge [17]. Jin et al. utilized a transfer learning model to cluster short texts to embody auxiliary long texts [8]. These approaches can partially handle the semantic sparsity problem; however, they also have some limitations. For instance, Hu et al.'s work in [17] makes the implicit assumption that the auxiliary data are semantically related to the short texts, which may not be true in the real world. Similarly, work [8] makes the assumption that the topical structure of the two domains which is completely identical would not be wholly correct.

Some studies utilized the complex network to handle Web service clustering problems to introduce the capability of network-based software. Many approaches have been performed from a complex network perspective by representing software systems (or service-oriented software systems) as software networks (or software service networks). Ma et al. [18] and Pan et al. [19] analyzed the topological structure of software networks, revealing many shared properties such as small-world and scale-free. Şora and Chirila [20] and Pan et al. [21, 22] proposed approaches to identify key classes in Java systems. Ma et al. [18] and Pan et al. [23–25] proposed software metrics by using parameters in complex networks. Zhou and Wang [26] and Pan et al. [27, 28] proposed an approach to cluster services by using community detection approaches in complex networks. There works are helpful to utilize the capability of the complex network; however, it still has the problem of semantic sparsity.

Faced with above problems, we propose to introduce another solution that introduces external information by word embeddings which have been shown to be beneficial for the semantic sparsity [29].

Latent Dirichlet Allocation (LDA) and extensions have been proved as efficient methods for boosting the discovery performance of Web services [16, 30]. However, due to the

base assumption that the words are discrete multinomial distribution, these probabilistic models cannot benefit from the word embeddings which are continuous vectors. Faced with this, we propose to use the neural topic model to leverage the advantages of both word embeddings and probabilistic models. Category labels can play an important role in the clustering procedures. Inspired by this, leveraging both label information and embeddings to enhance the discovery performance has attracted our attention. As a result, a label-aided neural topic model derived from Gaussian LDA which integrates both word embeddings and external label information is proposed.

3. The Discovery Process of the Proposed Approach

As is shown in Figure 1, the service discovery process of the proposed model consists of four major parts: service preprocessing, service modelling, query modelling, and service ranking. As shown in Figure 1, the Web service is firstly crawled and preprocessed. The description and the label of a Web service are extracted from the collected materials. Then, the service descriptions are taken as the input of the word2vec model to create the word embeddings. After getting the word embeddings, we map the words in the service description label into word embedding to produce one input and take the Web service label as the other input for the proposed model LNTM. The LNTM will convert each Web service into representations of latent factors. To model users' queries, the words in a query are looked up from embeddings and mapped into embeddings. In the service ranking phase, based on LNTM and users' queries, a probabilistic service ranking model is proposed to retrieve relevant services for the users.

In the proposed approach, training word embedding and modelling services are conducted offline, and the efficiency of the proposed discovery model can be guaranteed. Hence, the focus of the approach will be placed on the accuracy of discovery.

3.1. LNTM. For capturing semantic regularities in the language and handling the semantic sparse of Web services, an augmented topic model with word embeddings for Web service discovery is proposed in [31]. In the meanwhile, labels of documents can be used to guide topic learning so as to find more meaningful topics [12]. Therefore, in this paper, a label-augmented neural topic model is proposed to leverage label information and capture semantic regularities for enhancing discovery performance of the Web service in this paper.

In the proposed model LNTM, the word embedding v for each term in a document d at position i is written as $v_{di} \in R^W$, and W is the length of the word embedding. As a result, the words in a document are mapped into continuous vectors in the W-dimensional space. Therefore, each topic k is characterized as a multivariate Gaussian distribution with mean μ_k and covariance Σ_k. The Gaussian parameterization is determined by both analytic convenience and the semantic similarity of embeddings. To govern the mean and variance of each Gaussian, the Gaussian distribution centered at zero and an inverse Wishart distribution for the covariance are placed as the conjugate priors.

Similar to Gaussian LDA, Web service modeled by LNTM is represented as the mixtures over latent topics with proportions drawn from a Dirichlet prior.

To integrate labels, words are indicators for the presence of labels, and then l_d would include 1 in the positions for each label listed on document d and 0, otherwise. The graphical representation of LNTM is shown in Figure 2. Based on above notions, the generative process of LNTM for a document can be summarized as follows:

(1) For topic $k = 1, \ldots, K$,

  (a) For each label $l = 1, \ldots, L$, choose $\lambda_{l,k} \sim \mathrm{Ga}(s, s)$
  (b) Draw topic covariance $\Sigma_k \sim W^{-1}(\Psi, \nu)$
  (c) Draw topic mean $\mu_k \sim \mathrm{Normal}(\mu, (1/k)\Sigma_k)$

(2) For each document d in corpus D,

  (a) For each topic k, compute $\alpha_{d,k} = \prod_l^{L_{\mathrm{doc}}} \lambda_{l,k}^{f_{d,l}}$
  (b) Draw topic distribution $\theta_d \sim \mathrm{Dir}(\alpha)$
  (c) For each word index $i = 1, \ldots, N_d$,

    (i) Draw a topic $z_n \sim \mathrm{Cat}(\theta_d)$
    (ii) Condition on z_{di}, draw a word v_{di} with a probability
    $v_{di}/z_{di}, u_{1,\ldots,K}, \Sigma_{1,\ldots,K} \sim \mathrm{Normal}(\mu_{z_{di}}, \Sigma_{z_{di}})$

3.2. Web Service Modelling Using LNTM. The LNTM is a generative model in which each embedding v in a service description is associated with the latent variable topic z, and each topic z is associated with the service description d. With these two distributions, a Web service can be expressed as two layers: the service topics and the topic embeddings.

After using the LNTM, the service-topic distribution is achieved by the parameter θ ($\theta \in |\mathrm{services}| \times |\mathrm{topics}|$), and topic embedding is achieved by the multivariate Gaussians.

To infer the topic assignments of individual embeddings and the posterior distribution of services over the topics, a collapsed Gibbs sampling method is adopted to derive the topic assignments to each embedding by using the update rule shown in the following equation:

$$p(z_{di} = k, \lambda_{di} = l \mid z_{-di}, l_{-di}, V_d, \zeta, \alpha) \propto \left(n_{l_{di} z_{di}} + \alpha_{z_{di}}\right) \times t_{\nu_k - M+1}\left(v_{di} \mid \mu_k, \frac{\kappa_k + 1}{\kappa_k}\Sigma_k\right), \quad (1)$$

where z_{-di} represents the topic assignments of all word embeddings, excluding the one at the i-th position of service$d$. λ_{-di} represents the label assignments. V_d is the sequence of vectors for service description d; M is the length of the word embedding; a tuple $\zeta = (\mu, \kappa, \Sigma, \nu)$ is the parameters of the prior distribution; and $t_{\nu_l}(x \mid \mu', \Sigma')$ is the multivariate t-distribution with freedom degree $\nu\imath$ and parameters μ' and Σ'.

Note that the first part of equation (1) which expresses the probability of topic k in service description d is derived as that of Gaussian LDA.

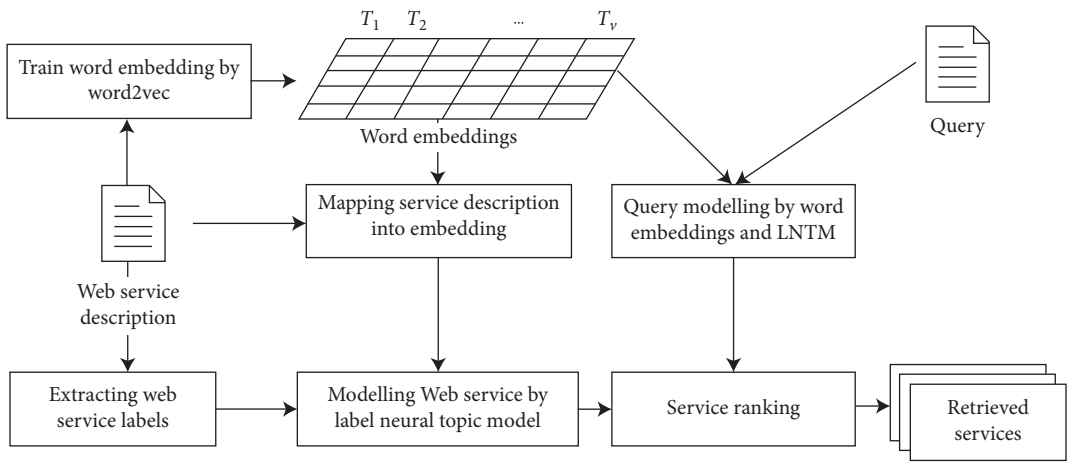

Figure 1: The discovery process.

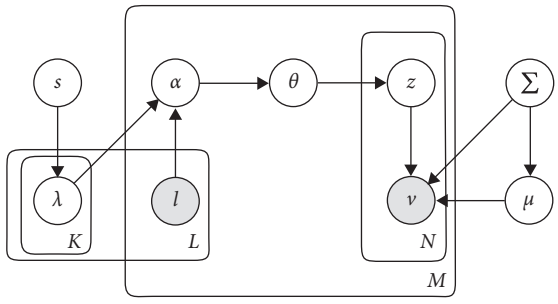

Figure 2: Graphical model of LNTM.

The second part of equation (1) expresses the probability of assignment of topic k to the vector v_{di} given the current topic assignments which represented by a multivariate t-distribution with parameters $(\mu_k, \kappa_k, \Sigma_k, \nu_k)$. These parameters for the posterior distribution are calculated by equation (2) as the Gaussian LDA:

$$\begin{aligned}
\kappa_k &= \kappa + N_k, \\
\mu_k &= \frac{\kappa\mu + N_k\overline{v}_k}{\kappa_k}, \\
\nu_k &= \nu + N_k, \\
\sigma_k &= \frac{\Psi_k}{\nu_k - M + 1}, \\
\Psi_k &= \Psi + C_k + \frac{\kappa N_k}{\kappa_k}\left(\overline{v}_k - \mu\right)\left(\overline{v}_k - \mu\right)^{\top}.
\end{aligned} \tag{2}$$

Here, the parameters $\overline{v}_k$ and C_k are calculated as

$$\begin{aligned}
\overline{v}_k &= \frac{\sum_d \sum_{i:z_{di}} v_{di}}{N_k}, \\
C_k &= \sum_d \sum_{i:z_{di}=k} \left(v_{di} - \overline{v}_k\right)\left(v_{di} - \overline{v}_k\right)^{\top},
\end{aligned} \tag{3}$$

where N_k is the total counts of the words of the topic assignment of k across all descriptions. $\overline{v}_k$ and C_k are the sample mean and the scaled form of sample covariance assigned topic k, respectively. Intuitively, the parameters μ_k and Σ_k are the posterior mean and covariance. The parameters κ_k and ν_k denote the strength of the priors for mean and covariance, respectively. After getting these parameters, we can simply achieve the topic-embedding distribution as discussed above.

3.3. Query Modelling and Ranking. To retrieve relevant services by the proposed model, we firstly translate the user query into embeddings. The words in a query are extracted and mapped into the embeddings by looking up the embedding features.

To rank the retrieved Web service, we use the generated probabilities to calculate the similarity between the user queries and the Web services as the work in [31]. The similarities are represented by $P(Q \mid s_i)$, where Q is the query and s_i is the i-th Web service. Thus, using the assumptions of the LNTM described above, it can be calculated by the following equation:

$$P\left(Q \mid s_i\right) = \prod_{e \in Q} P\left(e \mid s_i\right) = \prod_{e \in Q} \sum_{z=1}^{K} P\left(e \mid z\right) P\left(z \mid s_i\right). \tag{4}$$

Here, $P(e \mid z)$ and $P(z \mid s)$ are the posterior probabilities computed according to above equation (2) and the matrix θ, respectively. Finally, we can obtain a list of retrieved services towards a query according the value of $P(Q|s_i)$.

4. Experiment Setting

To evaluate the proposed approach, we conducted several experiments on the standard Web service test dataset SAWSDL-TC3 (TC3) (http://www.semwebcentral.org/projects/sawsdl-tc) as Tian et al. [31] did. To use TC3, we first parse the WSDL files into a plain text and then removed stop words and lemmatized the remaining words.

As is known to all, the Web services of TC3 do not have explicit category labels. However, there are some implicit categories in the WSDL files. As shown in Figure 3, the node "<xsd:annotation>" of service "FoodMaxpricequantity.wsdl"

Figure 3: The category labels for a service.

has values of "#Food," "#MaxPrice," and "Quantity." As a result, we extracted these values to generate the category labels for each Web service in our experiments. Since the Web services in the real-world service registry all belong to their certain categories, it is easily to collect the category label information for using the proposed approach.

In our experiments, we used precision p, recall r, and F-measure f as the evaluation criterion which is defined in equation (5) for the proposed approach. The larger the F-measure is, the better the performance of the discovery is.

$$p = \frac{|\text{relevant} \cap \text{predicted}|}{|\text{predicted}|},$$

$$r = \frac{|\text{relevant} \cap \text{predicted}|}{|\text{relevant}|}, \tag{5}$$

$$f = 2 \cdot \frac{p \cdot r}{p + r},$$

where relevant is the relevant class labels and predicted is the predicted results of the classification methods.

4.1. Performance of the Proposed Approach. To examine the performance of our approach, we compare the proposed method with three other Web service discovery approaches. These approaches are demonstrated as follows:

(1) LDA: when using LDA, the latent factors learnt from the Web service description are adopted to represent the Web service, and then a discovery approach is used to rank the services [30].

(2) Meta-LDA: in Meta-LDA [12], metainformation such as a category of a Web service is directly incorporated into the generative process. The external metainformation can improve the topic quality and modelling accuracy. We use Meta-LDA to group Web services into different clusters and then employ a probabilistic model to rank the services.

(3) Gaussian – LDA: a Gaussian LDA-based Web service discovery approach which makes use of embeddings for semantic sparsity Web service discovery is conducted based on the work done by Tian at al. [31].

(4) LNTM: for LNTM, we first train the word embeddings by word2vec from the prepared corpus. Then, we train the LNTM by incorporating embedding and service category labels into the generative process and organize the Web service into different clusters. Finally, we represent the query by embedding and utilize the probabilistic discovery model to rank the Web services as illustrated in Section 3.

For LDA and the Meta-LDA model, following the modelling process mentioned above, the topics are generated from the descriptions of the Web services. Then, we tuned the algorithms, respectively, to their best parameter settings by cross validation.

Table 1 shows the experimental data on TC3. According to these experiments, we have several observations: firstly, LNTM outperformed all the competitors in terms of F-measure on nearly all the settings, showing the benefit of using both word embeddings and service category labels which demonstrates the effectiveness of the proposed model.

Secondly, by looking at the approaches using the label information, we can see the significant improvement of these models over LDA, which indicates that document labels can play an important role in guiding topic modelling.

Thirdly, the LNTM and Gaussian – LDA have better performance than the LDA-based method. The results show that the embedding-based approach which takes continuous embeddings as the input may capture more semantically coherent topics compared to the traditional LDA-based method.

Finally, it is interesting to note that the Meta-LDA outperforms LDA and LNTM outperforms Gaussian – LDA, respectively, in this study. These findings are in agreement with the idea that utilizing the category label data of Web services improves the performance of Web service discovery. These results inspire the research work to integrate other external information for effective Web service discovery.

Table 1: Performance of the proposed approach.

Query	LDA			Meta-LDA			Gaussian LDA			LNTM		
	p	r	f	p	r	f	p	r	f	p	r	f
@10	0.64	0.30	0.40	0.78	0.41	0.54	0.76	0.37	0.50	0.91	0.46	0.61
@15	0.57	0.35	0.43	0.80	0.39	0.52	0.69	0.43	0.53	0.82	0.55	0.65
@20	0.50	0.38	0.44	0.74	0.42	0.53	0.61	0.47	0.53	0.75	0.58	0.65
@25	0.45	0.31	0.43	0.69	0.39	0.49	0.58	0.51	0.54	0.71	0.63	0.66
@30	0.41	0.44	0.43	0.63	0.49	0.55	0.55	0.54	0.54	0.69	0.67	0.67
@35	0.38	0.46	0.42	0.67	0.53	0.59	0.51	0.59	0.55	0.65	0.72	0.68
@40	0.36	0.49	0.42	0.61	0.59	0.60	0.49	0.61	0.54	0.63	0.73	0.67

4.2. Validation of Labels. To validate that incorporating category label information can significantly improve the generative topic accuracy, we varied the proportion of services used in training from 20% to 80% and used the remaining for testing. Here, we utilize normalised pointwise mutual information (NPMI) as shown in equation (6) to evaluate the topic quality of LDA, Meta-LDA, and LNTM:

$$\mathrm{NPMI}(k) = \Sigma_{j=2}^{T} \sigma_{i=1}^{j-1} \left(\log \frac{p(w_i, w_j)}{p(w_i)p(w_j)} - \log\left(p(w_i, w_j)\right) \right). \tag{6}$$

The NPMI score of each topic in the experiments is calculated with top 15 words ($T = 15$). As shown in Figure 4, the NPMI scores of both LNTM and Meta-LDA outperform LDA. The result indicates that the label information can enhance the LDA-based model to find more meaningful topics. The details of the two corpus are shown in Table 2.

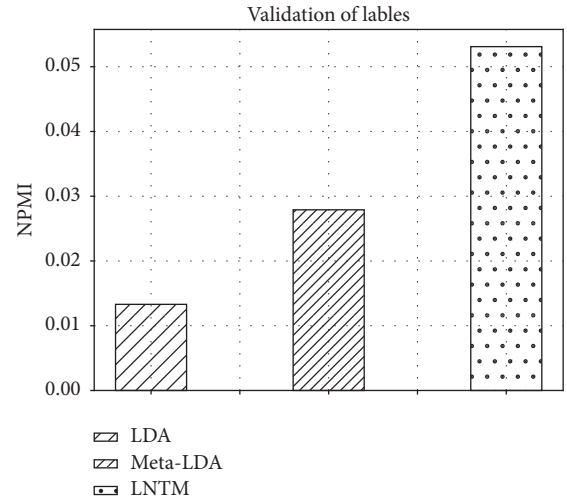

Figure 4: The influence of labels.

4.3. Validation of Embedding. As discussed above, embodying more semantic knowledge by changing the Bag of Words model into the continuous embedding space using LNTM can enhance the performance of the Web service discovery model. Several experiments are conducted so as to validate the result. Figure 5 shows the F-measure performance of the proposed approach with different word embeddings trained by the word2vec model using different corpus TC3 and Wikipedia.

As shown in Figure 5, the proposed approach using TC3 has better F-measure performance than using Wikipedia. The possible explanation for this may be that some words extracted from the WSDL files which do not have enough appearance counts in the Wikipedia corpus are removed when training the embeddings though they are very informative [31].

Table 2: Statistic of word embeddings.

	TC3	Wikipedia
Words	6,895	8,069,236
Documents	1,043	3,758,076
Embeddings	50	50

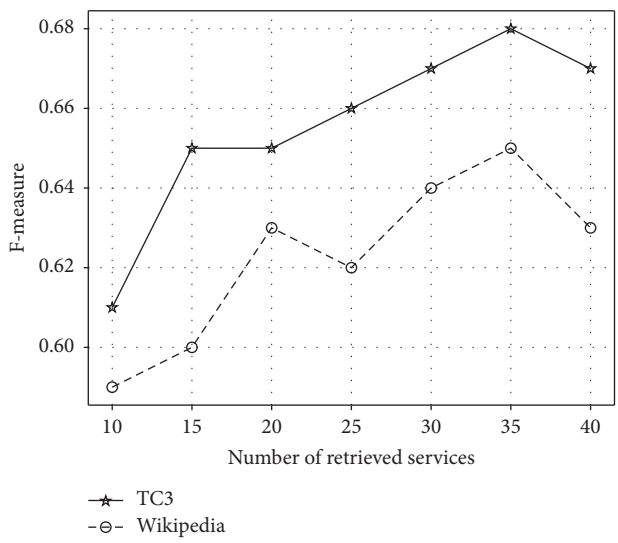

Figure 5: The influence of different embedding sets.

4.4. Influence of Hyperparameters. In LNTM, the parameter α illustrates the weight of language model contribution, μ and Σ control the document contribution, while s contributes to the label information. In our work, hyperparameters are empirically set as $\alpha = 1/K$, $s =$ zero, $\mu =$ zero mean, $\Sigma = 3 * I$, and 1,000 sampling iterations as in work [31]. Here, K is the number of topics, and I is the identity matrix. To check the influence of topic number k, we calculated $P(e \mid k)$ for different k. As shown in Figure 6, the

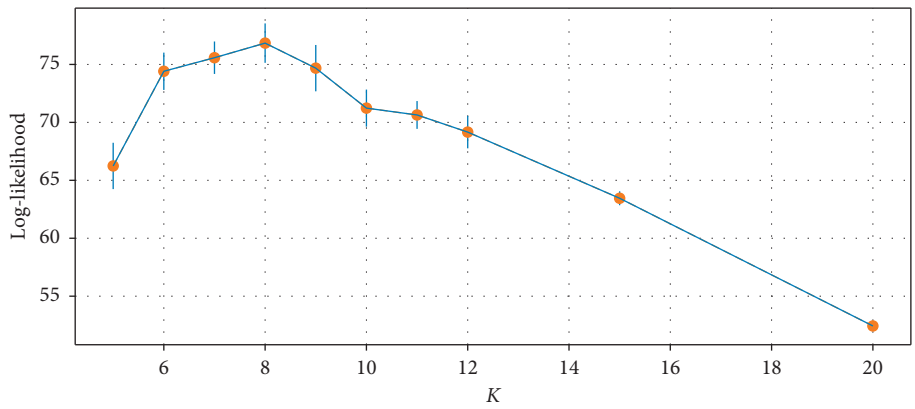

Figure 6: The number k of topics in LNTM.

result suggests that the data are best accounted for the proposed LNTM model incorporating 8 topics.

5. Conclusion

In this paper, we proposed a Web service discovery approach that combines word embeddings and category labels to deal with the poor recall problem in searching semantic sparse Web services. We used word embeddings to map the word into embedding so as to enrich the Web service semantics. We also introduced a label-augmented neural topic model LNTM which organizes the Web services into hierarchies for a probabilistic ranking approach.

Several experiments were conducted on a widely used dataset TC3 to validate the performance of our approach. Experimental results suggested that the proposed approach is feasible, and in particular, the word embeddings and label information both lead to enhanced performance in the Web service discovery process.

Since not all the Web services have their category labels, it is necessary here to clarify exactly how to conduct effective Web service discovery without labels. In the future, there is abundant room to further investigate the usefulness of various metainformation of Web service and propose different forms based on Gaussian – LDA to provide effective service discovery.

Data Availability

The experiments' data are available in http://www.semwebcentral.org/projects/sawsdltc.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under grant nos. 61702305, 11971270, and 61903089, the China Postdoctoral Science Foundation under grant no. 2017M622234 and Science and Technology Support Plan of Youth Innovation Team of Shandong higher School under grant no. 2019KJN2014.

References

[1] M. Xu, W. Tian, and R. Buyya, "A survey on load balancing algorithms for virtual machines placement in cloud computing," *Concurrency & Computation Practice & Experience*, vol. 29, no. 12, p. e4123, 2017.

[2] D. He, X. Yang, Z. Feng, S. Chen, and F. Fogelman-Soulie, "A probabilistic model for service clustering - jointly using service invocation and service characteristics," In Proceedings of the 2018 IEEE International Conference on Web Services (ICWS), San Francisco, CA, USA, July 2018.

[3] K. Elgazzar, E. H. Ahmed, and P. Martin, "Clustering wsdl documents to bootstrap the discovery of web services," in *Proceedings of the 2010 IEEE International Conference Web Services*, pp. 147–154, IEEE, Miami, FL, USA, July 2010.

[4] G. Tian, P. Liu, Y. Peng, and C. Sun, "Tagging augmented neural topic model for semantic sparse web service discovery," *Concurrency and Computation: Practice and Experience*, vol. 30, no. 16, p. e4448, 2018.

[5] G. Tian, S. Zhao, J. Wang, Z. Zhao, J. Liu, and L. Guo, "Semantic sparse service discovery using word embedding and Gaussian lda," *IEEE Access*, vol. 7, pp. 88231–88242, 2019.

[6] W. Chen, I. Paik, and P. C. K. Hung, "Constructing a global social service network for better quality of web service discovery," *IEEE Transactions on Services Computing*, vol. 8, no. 2, pp. 284–298, 2015.

[7] L. De Jesus Silva, D. Barreiro Claro, and D. Cicero Pavão Lopes, "Semantic-based clustering of web services," *Journal of Web Engineering*, vol. 14, no. 3-4, pp. 325–345, 2015.

[8] O. Jin, N. N. Liu, K. Zhao, Y. Yu, and Q. Yang, "Transferring topical knowledge from auxiliary long texts for short text clustering," in *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*, pp. 775–784, ACM, Glasgow, Scotland, October 2011.

[9] S. Seifzadeh, K. F. Ahmed, M. S. Kamel, and F. Karray, "Short-text clustering using statistical semantics," in *Proceedings of the 24th International Conference on World Wide Web Companion*, pp. 805–810, Florence, Italy, May 2015.

[10] T. Mikolov, W.-T. Yih, and G. Zweig, "Linguistic regularities in continuous space word representations," in *Proceedings of the HLT-NAACL*, pp. 746–751, Atlanta, GA, USA, June 2013.

[11] E. Amine, J. Flocon-Cholet, S. Gosselin, and S. Vaton, "Bayesian mixture models for semi-supervised clustering,"

2019, https://hal.archives-ouvertes.fr/hal-02372337/file/Bayesian_Mixture_Models_For_SemiSupervised_Clustering.pdf.

[12] H. Zhao, D. Lan, W. Buntine, and G. Liu, "Metalda: a topic model that efficiently incorporates meta information," 2017, https://arxiv.org/abs/1709.06365.

[13] R. Das, M. Zaheer, and C. Dyer, *Gaussian LDA for Topic Models with Word Embeddings*, Vol. 1, Long Papers, Beijing, China, 2015.

[14] L. D. Ngan and R. Kanagasabai, "Semantic web service discovery: state-of-the-art and research challenges," *Personal and Ubiquitous Computing*, vol. 17, no. 8, pp. 1741–1752, 2013.

[15] P. Rodriguez Mier, C. Pedrinaci, M. Lama, and M. Mucientes, "An integrated semantic web service discovery and composition framework," 2015, https://arxiv.org/pdf/1502.02840.pdf.

[16] L. Chen, Y. Wang, Yu Qi, Z. Zheng, and J. Wu, "Wt-lda: user tagging augmented lda for web service clustering," in *Service-Oriented Computing*, pp. 162–176, Springer, Berlin, Germany, 2013.

[17] X. Hu, N. Sun, C. Zhang, and T.-S. Chua, "Exploiting internal and external semantics for the clustering of short texts using world knowledge," in *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, pp. 919–928, ACM, Hong Kong, China, November 2009.

[18] Y.-T. Ma, K.-Q. He, B. Li, J. Liu, and X.-Y. Zhou, "A hybrid set of complexity metrics for large-scale object-oriented software systems," *Journal of Computer Science and Technology*, vol. 25, no. 6, pp. 1184–1201, 2010.

[19] W. Pan, B. Li, J. Liu, Y. Ma, and B. Hu, "Analyzing the structure of Java software systems by weightedK-core decomposition," *Future Generation Computer Systems*, vol. 83, no. 1, pp. 431–444, 2018.

[20] I. Şora and C.-B. Chirila, "Finding key classes in object-oriented software systems by techniques based on static analysis," *Information and Software Technology*, vol. 116, no. 1, pp. 75–89, 2019.

[21] W. Pan and C. Chai, "Structure-aware mashup service clustering for cloud-based internet of things using genetic algorithm based clustering algorithm," *Future Generation Computer Systems*, vol. 87, pp. 267–277, 2018.

[22] W. Pan and C. Chai, "Measuring software stability based on complex networks in software," *Cluster Computing*, vol. 22, no. 2, pp. 2589–2598, 2019.

[23] W. Pan, H. Jiang, H. Ming, C. Chai, B. Chen, and H. Li, "Characterizing software stability via change propagation simulation," *Complexity*, vol. 2019, Article ID 9414162, 17 pages, 2019.

[24] W. Pan, H. Ming, C. Chang, Z. Yang, and D.-K. Kim, "Elementrank: ranking java software classes and packages using multilayer complex network-based approach," *IEEE Transactions on Software Engineering*, 2019.

[25] Y. Xiang, W. Pan, H. Jiang, Y. Zhu, and H. Li, "Measuring software modularity based on software networks," *Entropy*, vol. 21, no. 4, p. 344, 2019.

[26] S. Zhou and Y. Wang, "Clustering services based on community detection in service networks," *Mathematical Problems in Engineering*, vol. 2019, Article ID 1495676, 11 pages, 2019.

[27] W. Pan, B. Song, K. Li, and K. Zhang, "Identifying key classes in object-oriented software using generalizedk-core decomposition," *Future Generation Computer Systems*, vol. 81, no. 1, pp. 188–202, 2018.

[28] W. Pan, J. Dong, K. Liu, and J. Wang, "Topology and topic-aware service clustering," *International Journal of Web Services Research*, vol. 15, no. 3, pp. 18–37, 2018.

[29] T. Kenter and M. de Rijke, "Short text similarity with word embeddings," in *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pp. 1411–1420, ACM, Melbourne, Australia, October 2015.

[30] C. Gilbert, P. Barnaghi, and K. Moessner, "Probabilistic methods for service clustering," in *Proceeding of the 4th International Workshop on Semantic Web Service Matchmaking and Resource Retrieval, Organised in Conjonction the ISWC*, Citeseer, Guildford, UK, 2010.

[31] G. Tian, J. Wang, Z. Zhao, and J. Liu, "Gaussian LDA and word embedding for semantic sparse web service discovery," *Collaborate Computing: Networking, Applications and Worksharing*, Springer, Berlin, Germany, pp. 48–59, 2017.