End-to-End Web Service Recommendations by Extending Collaborative Topic Regression

Bing Bai, Department of Automation, Tsinghua University, Beijing, China
Yushun Fan, Department of Automation, Tsinghua University, Beijing, China
Wei Tan, IBM Thomas J. Watson Research Center, Yorktown Heights, NY, USA
Jia Zhang, Department of Electrical and Computer Engineering, Carnegie Mellon University, Moffett Field, CA, USA
Keman Huang, Sloan School of Management, Massachusetts Institute of Technology, Cambridge, MA, USA
Jing Bi, School of Software Engineering, Beijing University of Technology, Beijing, China

ABSTRACT

Mashup has emerged as a lightweight way to compose multiple web services and create value-added compositions. Facing the large amount of services, effective service recommendations are in great need. Service recommendations for mashup queries suffers from a mashup-side cold-start problem, and traditional approaches usually overcome this by first applying topic models to mine topic proportions of services and mashup queries, and then using them for subsequent recommendations. This solution overlooks the fact that usage record can provide a feedback for text extraction. Besides, traditional approaches usually treat all the usage records equally, and overlook the fact that the service usage pattern is evolving. In this article, the authors overcome these issues and propose an end-to-end service recommendation algorithm by extending collaborative topic regression. The result is a generative process to model the whole procedure of service selection; thus, usage can guide the mining of text content, and meanwhile, they give time-aware confidence levels to different historical usages. Experiments on the real-world ProgrammableWeb data set show that the proposed algorithm gains an improvement of 6.3% in terms of mAP@50 and 10.6% in terms of Recall@50 compared with the state-of-the-art methods.

KEYWORDS
Mashup Development, Topic Modeling, Web Service Recommendations, Web Services

1. INTRODUCTION

With the wide adoption of Service-Oriented Architecture (SOA) and Cloud Computing, web services, usually in the form of web APIs, have grown rapidly both in quantity and diversity (Zhang and Zhang, 2013; Wu et al., 2015). To create value-added software efficiently and effectively, reusing existing services to composite mashups has become a promising solution (Cao et al., 2015; Wang et al., 2014). To deal with the large amount of services, performing service recommendations based on the massive historical usage data has been proved to be an expedient method (Zhong et al., 2015).

Apart from some non-personalized approaches (Ni et al., 2016; Huang et al., 2014), existing personalized service recommendation approaches can be roughly divided into two categories,

DOI: 10.4018/IJWSR.2018010105

Copyright © 2018, IGI Global. Copying or distributing in print or electronic forms without written permission of IGI Global is prohibited.
i.e., Quality-of-Service (QoS)-based service recommendations and function-based service recommendations. For QoS-based ones, recommendations are achieved by predicting non-functional characteristics such as reliability, availability and response time (Xiong et al., 2009; Zheng et al., 2009; Wei et al., 2012). For function-based service recommendations, researchers focus on finding services that meet the functional requirements the best (Zhong et al., 2015; Xia et al., 2014). In this paper, we focus on the latter one.

Function-based service recommendations suffers from an issue of, what we call, Mashup-side Cold-start problem. Different from traditional (and the most common) matrix completion recommendation tasks (QoS-based recommendations mainly belongs to this category), for function-based service recommendations, service usage of the mashup queries is what need to be predicted and completely unknown to recommender systems. The only information about the queries are the descriptions in the form of text (Zhong et al., 2015; Xia et al., 2014). Figure 1 illustrates the differences between these two tasks.

In other recommender systems, such a problem is often referred as a user-side cold-start problem. When new users join a system, it is difficult to recommend for them due to lack of information. Common recommendation algorithms, like standard collaborative filtering or matrix factorization, cannot be directly applied (Adomavicius and Tuzhilin, 2005).

In recent years, Latent Dirichlet Allocation (LDA) (Blei, Ng and Jordan, 2003) has been applied to extract service topics from textual information, in order to address the mashup-side cold-start problem (Zhong et al., 2015; Xia et al., 2014; Li et al., 2013). However, our study has revealed that traditional approaches may lead to the following two issues, which prohibit high-quality service recommendations.

1. **Overlooking the fact that usages can provide feedback and guide the content extraction:** Traditional methods apply LDA and then use the topic proportions as service domain proportions to provide features for subsequent processing (Zhong et al., 2015; Xia et al., 2014; Li et al., 2013). As a result, content modeling and usage modeling are totally separated. However, existing work (Wang, Wang and Yeung, 2015) shows that usages can provide a feedback to guide the extraction

![Figure 1. Illustration of matrix completion recommendations and function-based service recommendations for mashup development. Unlike the former one, service usages of the query are totally unknown, thus service recommendations suffers the mashup-side cold-start problem.](image-url)
Dynamic, Flow Control-Based Information Management for Web Services
www.igi-global.com/chapter/dynamic-flow-control-based-information/31216?camid=4v1a

Data Mining in Web Services Discovery and Monitoring
www.igi-global.com/article/data-mining-web-services-discovery/3115?camid=4v1a