Model-Based Search and Ranking of Web APIs across Multiple Repositories

Devis Bianchini, Valeria De Antonellis, and Michele Melchiori
Dept. of Information Engineering University of Brescia
via Branze, 38, 25123 Brescia, Italy
{devis.bianchini,valeria.deantonellis,michele.melchiori}@unibs.it

Abstract. Web API search and reuse for agile Web application development may benefit from selection criteria that combine several perspectives: they can be performed based on features used to describe APIs, or according to the co-occurrence of Web APIs in the same applications, or they can be driven through ratings assigned by designers who used the Web APIs for their own mashups. Nevertheless, different Web API repositories usually focus on a subset of these perspectives, thus providing complementary Web API descriptions. In this paper, we propose a unified model for Web API characterization. The model enables a cross-repository search of Web APIs and mashups, based on different kinds of similarity between them, identified regardless the complementarity of their descriptions. This unified representation improves retrieval results if compared with a Web API search performed over multiple repositories considered separately.

1 Introduction

Web API selection and aggregation, performed for mashup and short-living application development, may benefit from the adoption of criteria that combine different perspectives [1]: a component perspective (based on features used to describe Web APIs); an application perspective (i.e., information about mashups composed of the Web APIs); an experience perspective (including ratings assigned by web designers, who used Web APIs to develop their own mashups). The advantages coming from a multi-perspective Web API search have been confirmed by several approaches, that combined categories, tags and technical features like the adopted protocols and data formats in Web API descriptions with the co-occurrence of APIs in the same applications [2], with a quality-based model for Web APIs [3], with the network traffic around APIs and mashups, as an indicator of their success, and ratings assigned by designers [4]. Existing approaches rely on a single Web API repository. The ProgrammableWeb repository [1] is the most common one for sharing Web APIs and mashups. It contains over 11,500 Web APIs, where about 1,200 of them have been registered in the last year. APIs have been used in more than 7,400 mashups, while over 2,800 mashup

1 http://www.programmableweb.com/
owners are registered in the repository. Nevertheless, different repositories emphasize complementary aspects to be considered for Web API search. Although ProgrammableWeb constitutes a well-known meeting point for the community of mashup developers, it does not provide a comprehensive Web API model that includes all the perspectives: it is mainly focused on a feature-based description of Web APIs (through categories, tags and technical features) and on the list of mashups that have been developed using the Web APIs. Another repository, Mashape\textsuperscript{2}, a cloud API hub leveraging a twitter-like organization, associated each Web API with the list of developers who adopted or declared their interest for it, denoted as consumers and followers, respectively. Other public repositories, such as apigee or Anypoint API Portal\textsuperscript{3}, focus on different and only partially overlapping aspects as well. This scenario brings to situations where: (i) the same Web APIs or mashups are registered multiple times within different repositories; (ii) Web APIs (resp., mashups) are searched and ranked according to distinct criteria in separate repositories, to meet different Web API (resp., mashup) descriptions (for instance, in ProgrammableWeb Web APIs are ranked with respect to the number of mashups they have been used in, Web API ranking performed on Mashape repository depends on the number of API followers). As proved in \cite{1}, performing Web API search and ranking on a comprehensive API descriptor, that includes different and complementary descriptive aspects, would improve retrieval results. This implies that it is not enough to search for APIs within distinct repositories considered separately and simply merge search results, but a real unified view over the repositories before starting the search is required. In this sense, similarity between Web APIs and mashups across different repositories should be exploited to enrich search results. Current Web API search scenarios lack of a model that provides a unified representation of Web APIs and mashups, to ease the identification of similar resources regardless the complementarity of their descriptions across different repositories \cite{5}. Behind the advantage of avoiding multiple copies of the same API among the search results, although described with different properties depending on the repository from where API has been extracted, such a unified view would improve the retrieval outcomes as expected.

In this paper, we discuss about the definition of this model such that: (i) its unified representation covers the three perspectives mentioned above in Web API description, namely component, application and experience perspectives; (ii) it is part of a framework that enables the identification of different kinds of similarity between Web APIs and mashups, to provide a cross-repository search of these resources; (iii) it is integrated with a Web API and mashup search engine, that exploits similarity measures to properly access complementary information across repositories. A preliminary experimental evaluation confirms the improved search results, obtained by applying our approach, compared with Web API search performed on multiple repositories considered separately.

\footnotetext[2]{https://www.mashape.com/}
\footnotetext[3]{https://api-portal.anypoint.mulesoft.com}