Scraping functionalities in StackOverflow

Camilo Velázquez-Rodríguez
Vrije Universiteit Brussel
Belgium
cavelazq@vub.ac.be

***Index Terms*—software ecosystems, StackOverflow, natural language processing**

I. Extended Abstract

Software ecosystems such as Maven[1] or npm[2] provide several libraries within their ecosystem. In software ecosystems like Maven, developers can find common categories (e.g., Mocking, Dependency or HTTP Clients) containing several libraries under the same domain. However, for most of the libraries, their functionalities are hidden to the final user and currently, there does not exist a ranking of the libraries based on these features.

To mitigate the absence of this ranking, developers often pose questions in Q&A websites like StackOverflow [3] to gain information about which library is the best ranked given a specific functionality. For example, this post [1] asks about possible ways to delete a directory in a recursive manner using Java.

Several works have been published in topics related to the detection of functionalities/features, API recommendation, exploration of similar libraries and analysis of code snippets from Q&A forums. In this regard, Chen et al. [2] also mine StackOverflow and recommend similar libraries which target the same domain in different programming languages. Our work looks for API level recommendation in libraries within the same domain in the Maven software ecosystem.

Al-Subaihin et al. [3] cluster several mobile applications based on the features present in their descriptions, also known as featurelets. Gu et al. [4] based their work in a Recurrent Neural Network Encoder-Decoder with the purpose of generating API usage sequences from natural language queries. Our approach consists of analysing posts from StackOverflow where you can find pairs of natural language and code corpora.

To compute such ranking this work is focused on the extraction of functionalities in libraries within Maven categories through the use of StackOverflow posts as a source of information. We believe this is one of the first objectives that must be achieved to obtain the mentioned ranking.

On a high level, our approach is based on the analysis of the information available in Maven and Libraries.IO [4] on the one hand and the StackOverflow posts available through the public SOTorrent dataset on the other hand.

From Maven, we extract for a select group of Java libraries all the identifiers of public classes and methods and created vectors of words for further processing. Similarly, from StackOverflow, we made a similar process for the snippets of code within the posts. Is worth to mention that a single post in StackOverflow can contain several answers. We then selected the identifiers in the code fragment of the answers of every post and group them in a vector.

The vectors of words for both the category and the post, allowed us to transform them into numeric vectors for comparison purposes. To achieve this, we employed a word2vec model that was training from GitHub repositories that make use of the libraries considered in the analysis. Through the use of this trained model, we compare the categories and posts to obtain the similarities between them, given that for every word in either vector we extracted its numeric representation.

Finally, from the group of selected posts, we manually selected the most representative tags to remove false positives in the data. Some of the characteristics of the tags in StackOverflow are their repetitiveness, which allowed us to establish a relationship between frequency and importance; and their accuracy towards the topic in discussion reported in previous work [2].

Through the application of a topic modelling technique in the title of the selected group of posts, we could infer some common functionalities within a Maven category. Future work includes 1) The automatic identification of the libraries within the posts and 2) The evaluation of several functionality-based comparison techniques.

References

[1] Delete directories recursively in Java, "https://stackoverflow.com/questions/779519/delete-directories-recursively-in-java,". Accessed 10-November-2019.

[2] Chunyang Chen, Zhenchang Xing, and Yang Liu. 2018. What's Spain's Paris? Mining analogical libraries from Q&A discussions. Empirical Software Engineering 333 24, 3 (2018), 1155–1194.

[3] A.A.Al-Subaihin, F.Sarro, S.Black, L.Capra, M.Harman,Y.Jia, and Y.Zhang. 2016. Clustering Mobile Apps Based on Mined Textual Features. International 323 Symposium on Empirical Software Engineering and Measurement

[4] Xiaodong Gu, Hongyu Zhang, Dongmei Zhang, and Sunghun Kim. 2016. Deep 346 API Learning. Proceedings of the ACM SIGSOFT Symposium on the Foundations of Software Engineering 13-18-Nove (2016), 631–642. https://doi.org/10.1145/2950290.2950334

[1] https://mvnrepository.com

[2] https://www.npmjs.com

[3] https://stackoverflow.com

[4] https://libraries.io