TUTORIAL: EVALUATION OF SIMILARITIES AMONG OPEN SOURCE SOFTWARE

Phuong Nguyen, Juri Di Rocco, Riccardo Rubei, Davide Di Ruscio

Software Engineering and Architecture (SEA) Group

Department of Information Engineering, Computer Science and Mathematics

Università degli Studi dell'Aquila

Via Vetoio 2 -- 67100 L'Aquila, Italy

Email: {phuong.nguyen, juri.dirocco, riccardo.rubei, davide.diruscio}@univaq.it

We would like to thank you for your participation in our evaluation. This user study is conducted to evaluate the performance of 4 different tools for finding similar software projects, namely: **MUDABlue**, **CLAN**, **RepoPal** and **CrossSim** (proposed by the UDA team). In this document, we are going to present you a quick overview on how to properly perform the human evaluation.
A query project is the one that needs recommendation which contains a list of projects being similar to it. Given a query project, by using a similarity tool, we get a ranked list of 5 retrieved projects. We would like to know that if the 5 retrieved projects are really **similar to the query project** according to human perception. To this end, we kindly ask you to give a score to each pair of **<query project, retrieved project>** using the following guidelines.

1. Introduction

We consider two open source software (OSS) projects similar if they implement similar features, or they are described by the same abstraction. Two OSS projects are similar if there is significant overlap in the requirements and functionalities of the software. In orther words, we evaluate the similarity between two projects solely with regards to their functionalities, irregardless of their implementation.

There are the following examples:

- If two OSS projects implement cryptographic services to protect information then they are similar to a certain degree, even though they may have other different functionalities for different domains.
- Two text editors that are implemented by different programmers, but share many features: copy and paste, undo and redo, saving data in files using standard formats, are similar.
- An OSS project p_1 that performs the sending of files across a TCP/IP network is somehow similar to an OSS project p_2 that exchanges text messages between two users, i.e. Score(p_1,p_2)=3. However an OSS project p_3 with the functionalities of a pure text editor is dissimilar to both p_1 and p_2, i.e. Score(p_1,p_2)=Score(p_1,p_3)=1.
- An OSS project p_4 which possesses the functionalities of a normal text editor, e.g. those by p_3, however it embeds the ability to send text files over the network, then it is similar to all the above mentioned project p_1, p_2, and p_3 i.e. Score(p_1,p_4)=3, Score(p_2,p_4)=3, and Score(p_3,p_4)=3.
- The following apps are considered to be highly similar to each other: Viber, WhatsApp, Messenger since they are all applications that exchange text and multimedia messages over the TCP/IP network.

2. How to perform the evaluation

Each of you will be given 5 queries. For each query, you need to evaluate the results of two different similarity tools, among the four tools mentioned above (MUDABlue, CLAN, RepoPal, and CrossSim). To avoid bias against a specific tool, however you will not know exactly from which tool the query-retrieved projects are produced.

For each query, there are 5 retrieved projects and your task is to evaluate to which degree a pair of <query project, retrieved project> are similar. For each project, there will be a hyperlink to the actual GitHub repository where you can read the **Readme.md** file to better understand the functionalities. In case the Readme.md file is too brief and doesn't provide enough information to make a judgement, please try to investigate the project by consulting its source code files to ascertain its functionalities. It is often the case that you will not be able to understand everything about a project just by looking at its **Readme.md**.

3. Similarity Level

Given a pair of <query project, retrieved project>, you will give a score that specifies the level of similarity between them according to the following descriptions:

Similarity Level	**Description**	**Score**
Completely dissimilar	There is absolutely nothing in the retrieved project that is similar to the query project, nothing in it is related to the task and the functionality of the query project.	1
Mostly dissimilar	Only few remotely related requirements are located in the query and retrieved project.	2
Mostly similar	A somewhat large number of implemented requirements are located in the retrieved project that are similar to ones in the query project.	3
Highly similar	The query and the retrieved projects share the same semantic concepts expressed in the task	4

Table 1. Similarity scores

4. Interface

Each of you will be assigned an Excel sheet where there are five query projects, each corresponds to ten retrieved projects which are the outcomes of **two** similarity tools among the ones mentioned above (MUDABlue, CLAN, RepoPal, and CrossSim). In total, you have to give score for 50 pairs of <query, retrieved> projects. Figure 1 shows an example of an Excel sheet for evaluation. To unhide the link to the actual GitHub project, please hover the mouse on its name.

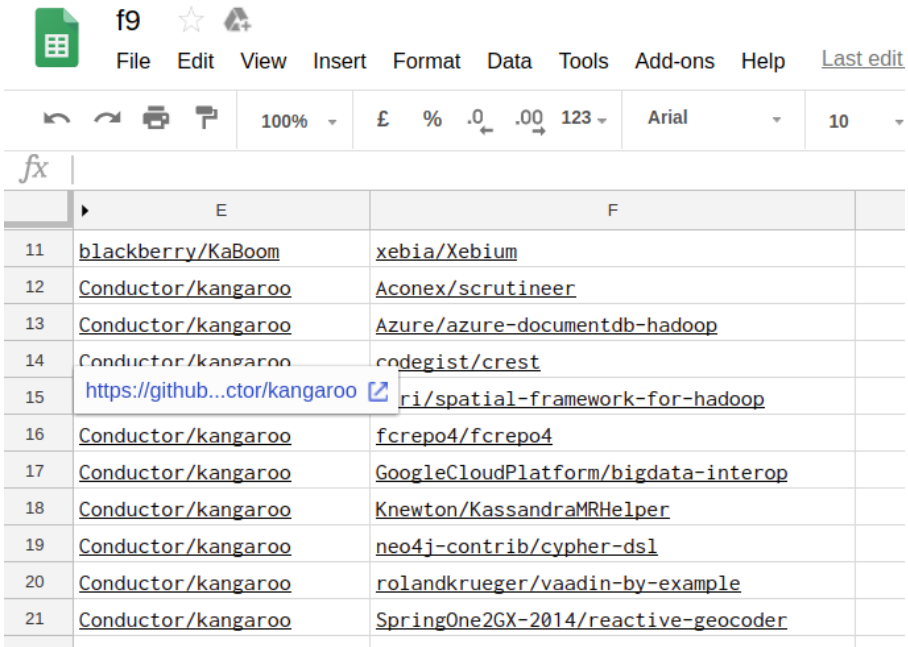


Figure 1. Evaluation interface

Sometimes, you may encounter a query project with two retrieved projects that are identical as shown in Figure 2. This is normal since the two tools consider the same project as similar to the query project. In this case, you need to give the same score for both projects.

22	datasalt/pangool	alexholmes/hadoop-utils
23	datasalt/pangool	alexholmes/hadoop-utils
24	datasalt/pangool	alexholmes/hiped2
25	datasalt/pangool	Azure/azure-documentdb-hadoop
26	datasalt/pangool	castagna/jena-grande
27	datasalt/pangool	chubbyjiang/MapReduce
28	datasalt/pangool	hopshadoop/hops
29	datasalt/pangool	laserson/avro2parquet
30	datasalt/pangool	msathis/SQLToNoSQLImporter
31	datasalt/pangool	socrata/datasync

Figure 2. A query project with an identical retrieved project

The third column of the sheet is where you put your score (Figure 3). Please note that similarity scores range from 1 to 4 as specified in Table 1, values that are out of this range will not be accepted.

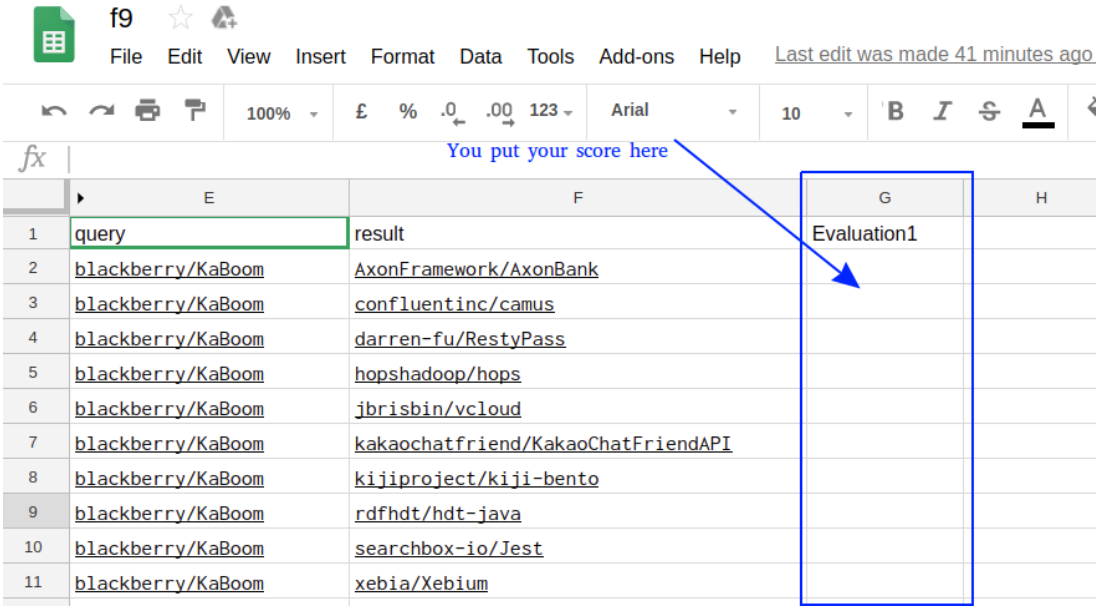


Figure 3. Put your score in the third column

If you encounter any problem during the evaluation, please do not hestitate to contact us using the information on the first page of this document.

Again, thank you so much for your kind support!

5. Examples

The following projects are considered to be similar:

Query project	Retrieved project	Score
https://github.com/AskNowQA/AutoSPARQL **Description from Readme.md**: AutoSPARQL TBSL is a graphical user interface, which allows to answer natural language queries over RDF knowledge bases. It is based on algorithms implemented in the DL-Learner Semantic Web machine learning framework.	https://github.com/neo4j-contrib/sparql-plugin **Description from Readme.md**: Sparqlify is a scalable SPARQL-SQL rewriter whose development began in April 2011 in the course of the LinkedGeoData project. This tool can create RDF dumps from CSV file based on SML view definitions.	3

Both projects work with SPARQL to query and create RDF data. You may know that RDF is a format for storing Linked Data and SPARQL is the language used to query RDF data. Thus, the two projects share common functionalities. As a results, they are considered to be similar and Score(AskNowQA/AutoSPARQL, neo4j-contrib/sparql-plugin)=3.

The following projects are highly similar:

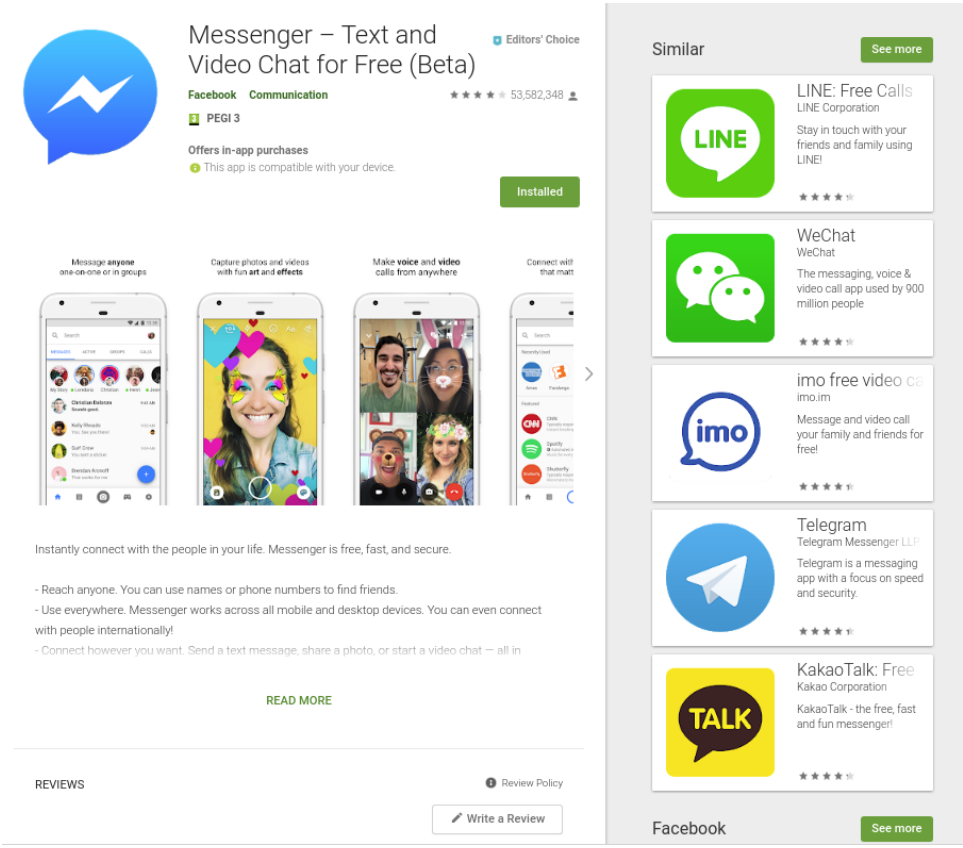
Query project	Retrieved project	Score
https://github.com/psaravan/JamsMusicPlayer **Description from Readme.md**: Jams is a free, powerful and elegant music player for Android. Jams used to be a trial/paid app on the Play Store.	https://github.com/TheAndroidMaster/Pasta-Music **Description from Readme.md**: Pasta Music is a material design music player for android that attempts to create a better user experience than the standard music players which can have too many features or be generally confusing to users. It was created to show an improvement in design and to allow older and slower devices to have quicker access to local music files.	4

Both projects are Android music players so they share common functionalities, such as: open, play music files. Therefore the similarity score is 4 for this pair. We don't care about their internal implementation.

To further elaborate the similarity between software applications concerning their functionalities, we introduce the following examples extracted from Google Play:

a. Messenger

Similar apps: LINE: Free Calls, WeChat, imo free video, KakaoTalk. These are considered to be similar to Messenger because they are all used for exchanging text and mutlimedia messages.



b. Outlook

Similar apps: Email App for Android, Email Exchange, Email-Fast & Secure for Android, Skype for Business, Universal Email. Almost all these apps are used to send Email.