CS209A Project Report

11911336 Tan Yajing
11910831 Liu Jidong

May 24, 2022

1 Topics

Our topic is to find out the popular java projects on the github, and analysis the feature of them.

2 Architecture Design

We use *PostgreSQL*, *Vue* and *Springboot* of Java to design the whole project.

2.1 Data Collection, I/O and persistence

2.1.1 Jsoup & RESTful API

We choose *GitHub* as our data source website, because we can track the star, fork, issue and watching on it, which can help us analysis the projects better. During the data collection, we use *Jsoup* and the *RESTful API*. Since each data source has a daily quota for REST requests, we decided to add the *Authorization* and the token to the header.

The contents will be stored as *JSON Object*, sometimes also in *JSON Array*.

2.1.2 JDBC & I/O

We use *PostgreSQL* as the database to store the data and manage the relationship of the data. After collecting, we use *JDBC* to write the data into the database.

Each table in the database has its corresponding class. Some important classes in our projects are as follows:

- Project: It contains some necessary fields such as name, url, star, fork, issues and watch. Besides, each project has a unique ID which can help distinguish it from the other projects.
- Time: It includes the created time and the last modified time of a project. The *time* field is stored as timestamp.
- Topic: It contains the topic of each project, and each project can have many topics.
- TopicCount: It stores the occurrence frequency of each topic for the analysis.

When read from database, the data will be returned as a *List* with different properties.

2.2 Data manipulation and analysis

We have applying different types of algorithms to analysis the data. Firstly, we sort them according to different fields, for example the number of stars, then analysis whether they are popular. *Lambda* and *stream* are used in this part.

One of the most important method is the *findByProjectInfo*, which uses lambda and stream to sort the projects.

We have found the relationship between the stars, forks and watches among different projects, the result is in Figure 1:

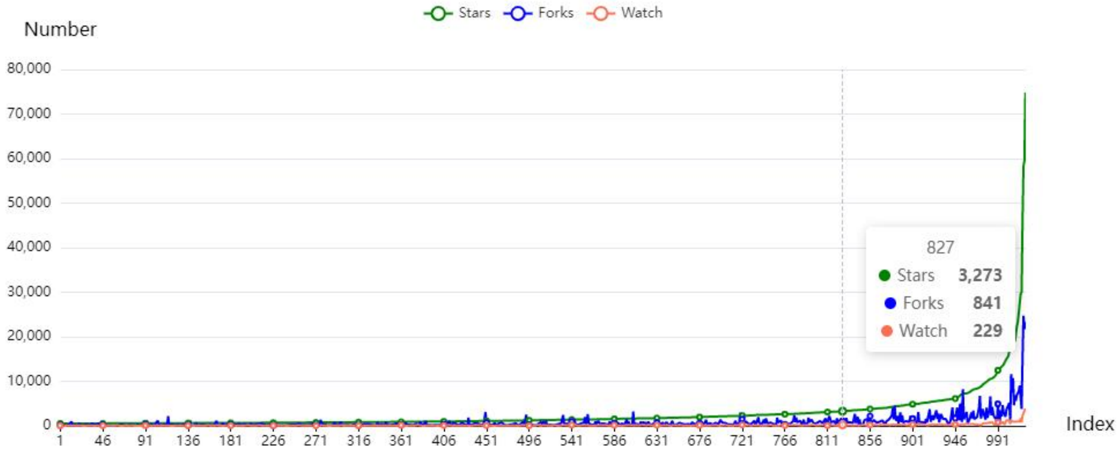
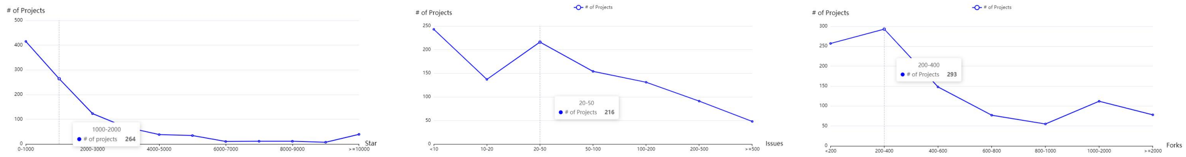


Figure 1: Relationship

We can roughly get the conclusion that the number of stars, forks and watches can reflect the popularity of a project. Although there are some fluctuations, the whole trend of the line chart is that if a project has more stars, it will also have more forks and more watches.

We have also record the statistics of stars, forks and issues. The result is in Figure 2.



(a) Star

(b) Issues

(c) Forks

Figure 2: Line Charts

Most projects that are collected by us have less than 1000 stars, however there are also some projects have more than 10000 stars, which can prove their popularity.

The projects don't have so much issues, since the author may always response or close the issues to solve the problem.

Also, most projects have 200-400 forks, it also proves that they have being noticed and used by many people.

We have also collected the relationship of the project and the time to see the development history of Java projects on the GitHub. The result is in Figure 3.

From Figure 3, we can see that the whole number of the java projects on GitHub have a sustainable growth. This shows that Java is still popular and it is an important programming

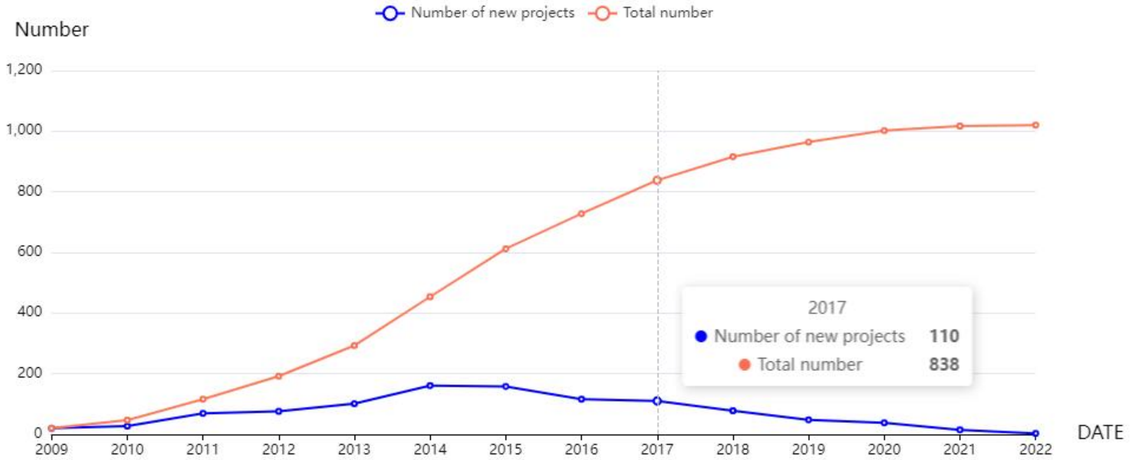


Figure 3: Created Time

language. In 2014 and 2015, the number of newly created projects reaches the maximum value at about 160.

In Figure 4, we have also designed a word cloud. In order to avoid the frequency of *java* being too large and causing visual effect, we removed the *java* topic and use other topics to form the word cloud. From this figure we can see that the most related topic with java is *android*, and some other topics for example *kotlin*, *spring* also appears.

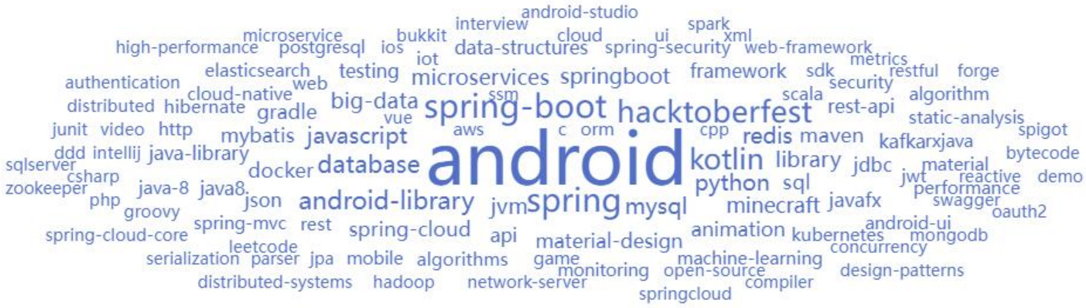


Figure 4: Word Cloud

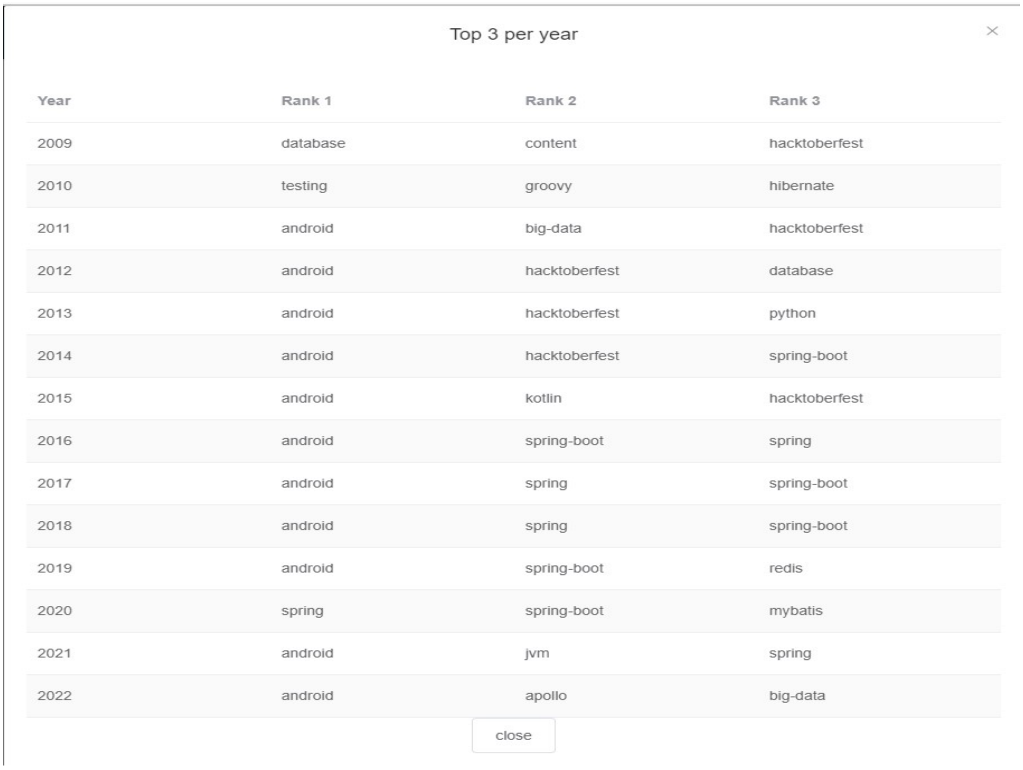
In Figure 5, we design a table to count the three most popular topics for each year from 2009 to 2022. It is obvious to see that *android* has topped the list eleven times, and *spring-boot*, *spring*, etc. are also very popular.

2.3 Data visualization & user experience

2.3.1 Data Visualization

We use *Vue* to do data visualization. The visualization is web-based, which means it can be opened in a browser. The data analysis results are shown as different kinds of charts, including line chart, bar graph and also word cloud.

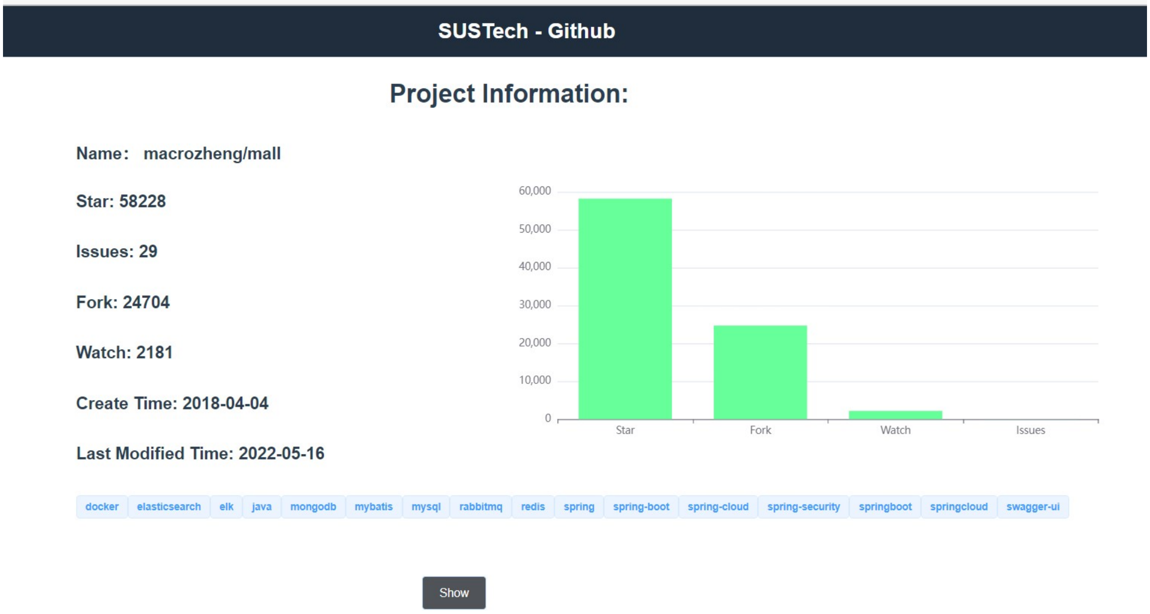
In order to implement word cloud, we use Java to collect the topic and their frequency, sorting them then write them into a Json file. By reading the json file, the front-end can display the word cloud successfully.



Top 3 per year

Year	Rank 1	Rank 2	Rank 3
2009	database	content	hacktoberfest
2010	testing	groovy	hibernate
2011	android	big-data	hacktoberfest
2012	android	hacktoberfest	database
2013	android	hacktoberfest	python
2014	android	hacktoberfest	spring-boot
2015	android	kotlin	hacktoberfest
2016	android	spring-boot	spring
2017	android	spring	spring-boot
2018	android	spring	spring-boot
2019	android	spring-boot	redis
2020	spring	spring-boot	mybatis
2021	android	jvm	spring
2022	android	apollo	big-data

Figure 5: Top 3 Topics



SUSTech - Github

Project Information:

Name: macrozheng/mall

Star: 58228

Issues: 29

Fork: 24704

Watch: 2181

Create Time: 2018-04-04

Last Modified Time: 2022-05-16

Figure 6: Project Information

2.3.2 User Experience

Our project also supports user interaction. In the home page, user can click the tags on the left and the web page will redirect to the list of projects which contains this tag. If a user clicks a project, like Figure 6, the page will show the project's basic information, and also provides a button to show the bar chart of the star, fork, issues value of the project.

Also, users can search keywords, and in the result the keywords will be highlighted. Besides, users can also sort the result by either star, fork, issues and watch. The sort method is also implemented by *lambda* and *stream*.

If a user wants to download the result, he can click the download button and will get the *csv* files of the basic information of projects shown on the current page.

3 Insights

In our analysis result, the conclusion of our topic is that the *java-design-patterns* is the most famous java project among our data sources, since it has the most star, watch and fork value, and also has a lasting time of 8 years. When counting topics, we found that *android* is always the mainstream. Before 2015, *hacktoberfest*, *database*, etc. were more popular, and after that, *spring-boot*, *spring* gradually became the new mainstream.

The features of a popular java project is that it may have many stars, forks and watch, not too many issues because the author will always check and modify the project.

For the whole data sources, we can find out that in 2014 and 2015, there are many new java projects being created on GitHub. The total number of java projects is still growing, which proves the popularity of java as a programming language.