DS Project

Proposal

Team 7

Name	Section	Bench No.
Ahmed Waleed	1	13
Mohamed Saad	2	14
Mohamed Salama	2	15
Essam Wisam	2	2

Supervised by

Dr. Dina Elreedy

Eng. Mohamed Abdullah

March 2023

Table of Contents

Problem Description (Idea)

Github is a well known platform where developers can collaborate together. In practice, it takes collaboration to build good software which makes Github a promising resource of most software existing in today’s world.

In this project, we aim to understand the characteristics and trends of popular and successful projects on Github and how they relate to the programming languages in use. The key metric we will use for project success is the number of stars. Other factors such as the number of watches or pull requests convey the degree of interest and collaboration a project is subject to.

We will be working with the Github Repository Metadata dataset found on Kaggle. The dataset has a record for each of over 3 million repositories, each with at least 5 stars and include info about the population of repositories found on Github between the time 01/01/2009 and 02/01/2023. The is an example record (one repository):

```
{
  "owner": "pelmers",
  "name": "text-rewriter",
  "stars": 11,  "forks": 4,
  "watchers": 3,
  "isFork": false,
  "isArchived": false,
  "languages": [ { "name": "JavaScript", "size": 21769 }, ...],
  "diskUsageKb": 75,
  "pullRequests": 4,
  "description": "Webextension to rewrite phrases in pages",
  "primaryLanguage": "JavaScript",
  "createdAt": "2015-03-14T22:35:11Z",
  "pushedAt": "2022-02-11T14:26:00Z",
  "defaultBranchCommitCount": 54,
  "license": null,
  "assignableUserCount": 1,
  "codeOfConduct": null,
  "forkingAllowed": true,
  "nameWithOwner": "pelmers/text-rewriter",
   "parent": null
 }
```

Note that because the dataset is too large (over 3 million records). We plan to take a large random sample of 200-400K records (stratified over years) to perform our analysis on. We may take additional small samples to test for generalizability while answering some of the questions.

* The questions follow the following format: *[x] Question*, where x indicates the type of the question.

Stated Questions

1. [D] What is the distribution of the average number of stars over different programming languages? What is the distribution of the average number of stars over time for the top 7 languages in stars?

2. [D] What is the distribution of the most common primary language over time? And what is the distribution of the underlying margin compared to the next most frequent programming language?

3. [D] What is the fraction of repositories without a license? What fraction of those with licenses also have a code of conduct?

4. [E] What are the top three sets of programming languages used together over time? What are some strong association rules that can be drawn from these sets?

5. [D] What is the average size of repositories over time for the top 3 programming languages in terms of pull requests?

6. [E] What is the correlation between the number of watchers and number of pull requests for each programming language?

7. [E] What databases tend to occur more often with different backend frameworks? What front-end frameworks tend to occur most often with the three most frequent backend frameworks?

8. [I] Does the popularity of Javascript generalize to prove that dynamically typed languages are more popular than statically typed ones?

9. [I] If <X> is the language with the most archived repositories that don’t allow forking for the period before 2009 and after 2015, then does this generalize to the whole period (2009 to 2023) regardless whether forking is allowed or not?

10. [P] What is the expected number of pull requests over all Python repositories for the year 2023?

11. [P] What programming language is expected to have the most repos archived in 2023?

12. [E] Is there any association between the number of main branch commits, stars and pull requests, and the primary programming language used in a project?

13. [E] What licenses are associated with what primary programming languages and project sizes?

Work Plan For Each Question

1. [D] What is the distribution of the average number of stars over different programming languages? What is the distribution of the average number of stars over time for the language with the most and least stars?

Answering this question is as easy as querying the mean number of stars for each programming language to result in the distribution, followed by finding the max over it. We would then subset the data into different years to find out how the average change for the most and least common language.

2. [D] What is the distribution of the most common primary language over time? And what is the distribution of the underlying margin compared to the next most frequent programming language?

Answering this question is as easy as subsetting the data

> overtime and then finding the language with the max number of repos in each subset . The second part of the question repeats this but where we take the 2nd most frequent and then subtract the number of repos of both distributions.

3. [D] What is the fraction of repositories without a license? What fraction of those with licenses also have a code of conduct?

> Answering this question is as easy as querying the number of repos without a license and seeing their ratio over all repos. Then queries the number of those with license and code of conduct and taking the ratio over all those with license.

4. [E] What are the top three sets of programming languages used together over time? What are some strong association rules that can be drawn from these sets?

> An example for how this question can be answered (exhaustively) is to find the maximum frequency over the present sets of languages found in the data. We can then use the Apriori algorithm over such sets to draw strong association rules that govern the chance language(s) x is used if language(s) y is used.

5. [D] What is the average size of repositories over time for the top three programming languages in terms of pull requests?

After subsetting the dataset relative to time, the argmax language over numbers of pull requests will be found. Likewise will be done to find the 2nd and the 3rd position.

6. [E] What is the correlation between the number of watchers and number of pull requests for each programming language?

In this, after subsetting the data relative programming languages we will study one or more correlation coefficients given the number of watchers and the number of pull requests variables.

7. [E] What databases tend to occur more often with different backend frameworks? What front-end frameworks tend to occur most often with the three most frequent backend frameworks?

A basic approach for this is to exhaustively enumerate the set of (backend framework, database) then find the argmax (database) by maxing over the number of repos. A similar approach applies given a set of (front-end framework, back-end framework). We plan to consider more sophisticated approaches that aren't as exhaustive if they exist.

8. [I*] Does the maximal popularity of Javascript generalize to prove that dynamically typed languages are more popular than statically typed ones?

In this, we shall first check if indeed Javascript is the most popular language over repos using descriptive analytics and then see if this holds for all dynamically typed languages in the same data as well by performing the same analytics and perceiving them as one language and comparing with all statically typed ones as one language.

9. [I] If <X> is the language with the most archived repositories that don't allow forking for the period before 2009 and after 2015, then does this generalize to the whole period (2009 to 2023) regardless whether forking is allowed or not?

In this, we shall first check if the language X satisfies the given original hypothesis which applies on the period between 2009 and 2015 and is over repos with no forking then check if that also applies to the whole dataset. We may use hypothesis testing in the process.

10. [P] What is the expected number of pull requests over all Python repositories for every month in the year 2023?

In this, we shall use time series forecasting (regression) where the target is the expected number of pull requests and the time axis is split by month.

11. [P] What programming language is expected to have the most repos archived in every month 2023?

In this, we shall use time series classification)where the target is the programming language and the time axis is split by month.

12. [P] Is there any association between the number of main branch commits, stars and pull requests, and the primary programming language used in a project?

In this, we shall use an association measure to relate the number of branch commits, stars and pull requests to the primary programming language.

13. [P] What licenses are associated with what primary programming languages? Is there an association with project sizes?

In this, we shall use an association measure that relates different licenses to different programming languages (categorical association) then we shall do the same with project sizes.