CS 643, F ‘19: Cloud Computing

Analyzing 3 Million GitHub Repositories to identify trends of GitHub Repositories over the 10 years

Kejul Vyas (kv283)
Namita Agarwal (na479)
Jainisha Mithani (jm956)
Charanpreet Kaur Dhir (ckd22)

Outline

- Motivation
- Design
- Implementation
- Evaluation
- Conclusions

Motivation(1)

- GitHub is one of the largest social coding communities around right now.
- GitHub can integrate with common platforms such as Amazon, Microsoft and Google Cloud, services such as Code Climate to track your feedback, and can highlight syntax in over 200 different programming languages.

Motivation(2)

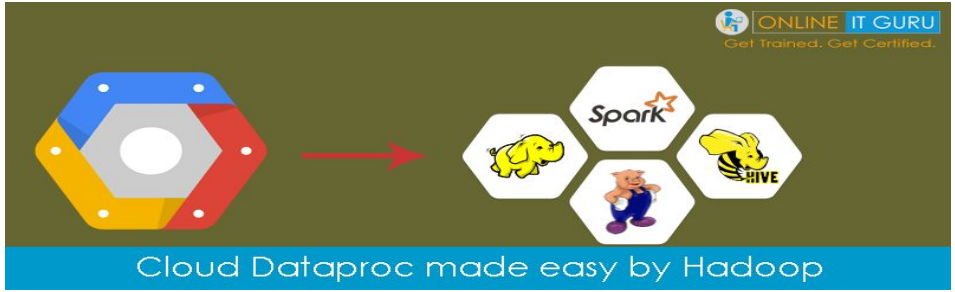
- Software developers of various levels keep entering Github, and use Github to save their public and private software projects as well as use the repositories and datasets available in GitHub to do their personal or academic projects.
- The large amounts of software developers and software repositories on Github are posing new challenges to the world of software engineering.
- So analyzing the importance and influence of Github repositories is important.
- By analysing Github Repositories and Github data, we investigate the collaboration of Github users based on their activities on repositories.
- We can also analyse the study language importance, or predict the trends of popular programming languages.

Design

- We are using Github_repos public dataset for our project.
- This 3TB+ dataset comprises the largest released source of GitHub activity to date.
- It contains a full snapshot of the content of more than 2.8 million open source GitHub repositories.
- We have analysed 50GB of the data, as there is a restriction of 1TB per month for free google account.
- This dataset is available via Google Cloud Platform's Marketplace, GitHub Activity Data, as part of GCP Public Datasets.
- You can use the BigQuery Python client library to query tables in this dataset in Kernels.
- For more information: Google Cloud Marketplace - Github data

Dataproc

- We are using DataProc provided by Google Cloud Platform for implementing our project in Apache Spark.
- Dataproc is fast, easy-to-use, fully managed cloud service for running Apache Hadoop and Apache spark clusters in a simpler, more cost-effective way.
- It provides built-in integration with BigQuery, so it much easier for us to use and implement the dataset hosted in BigQuery for our project.
- We are using Spark for implementing our project because of its higher computational speed.
- For more information:
  Dataproc quickstart experience



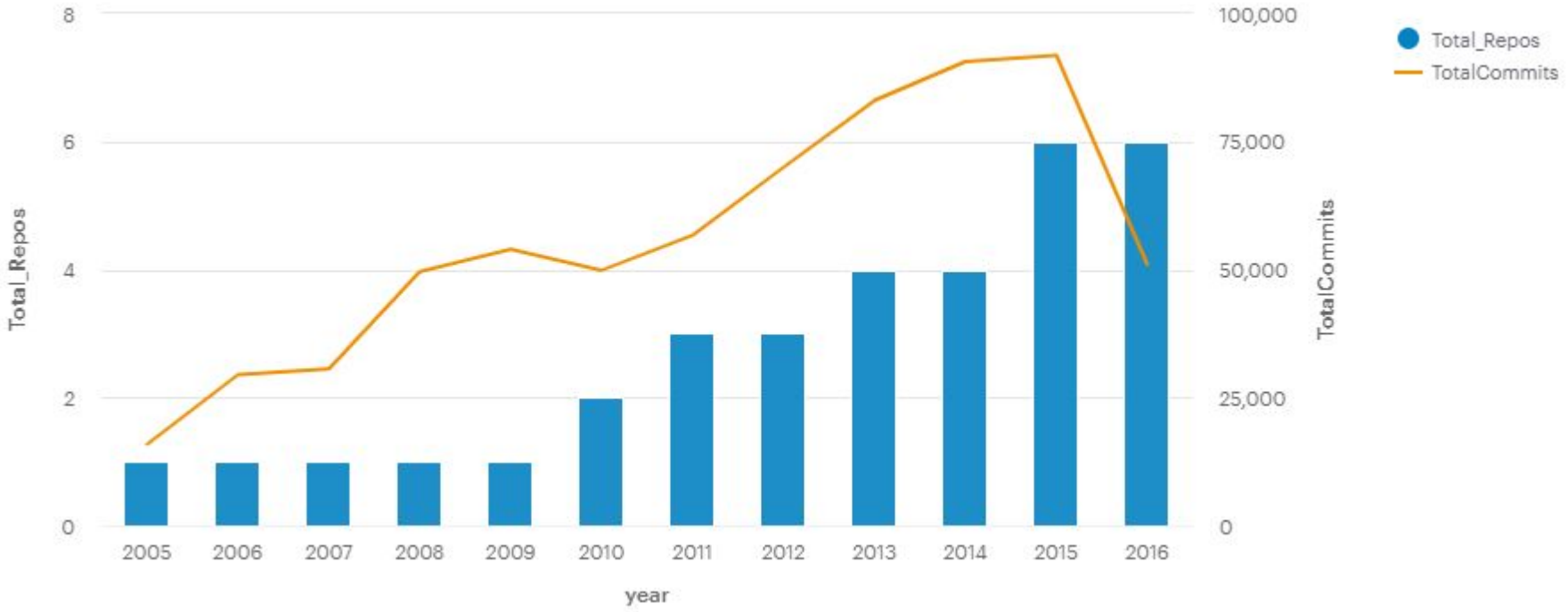
PySpark

- We are using pyspark (Python for Apache Spark) as it is easier to learn and use.
- It is comparatively easier to learn for object-oriented language programmer because of its syntax and standard libraries.
- The interface is simple and comprehensive.
- Talking about the readability of code, maintenance and familiarity with Python API for Apache Spark is far better than Scala.

- One of the major reasons why Python is preferred over Scala for Apache Spark is that the later lacks proper data science libraries and tools unlike Python.
- For more information: BigQuery connector with Spark

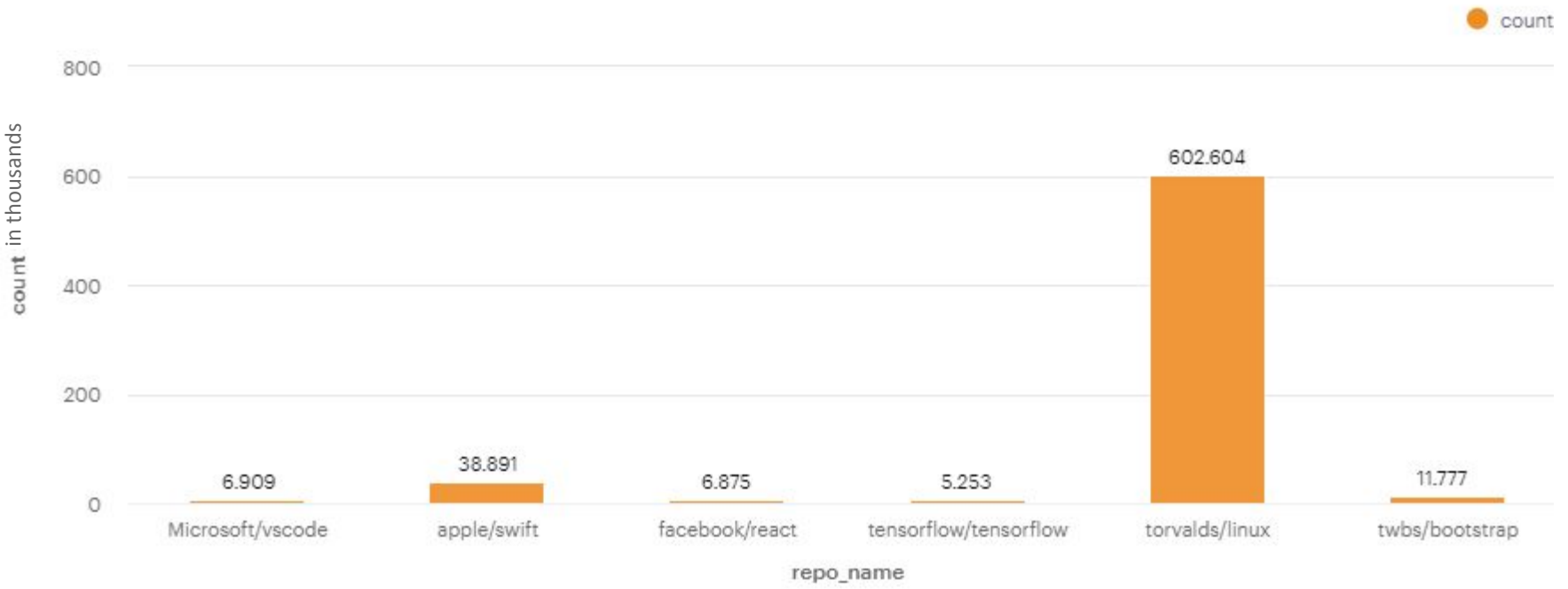
Implementation (1)

Trend of Github Repositories and commit over the years



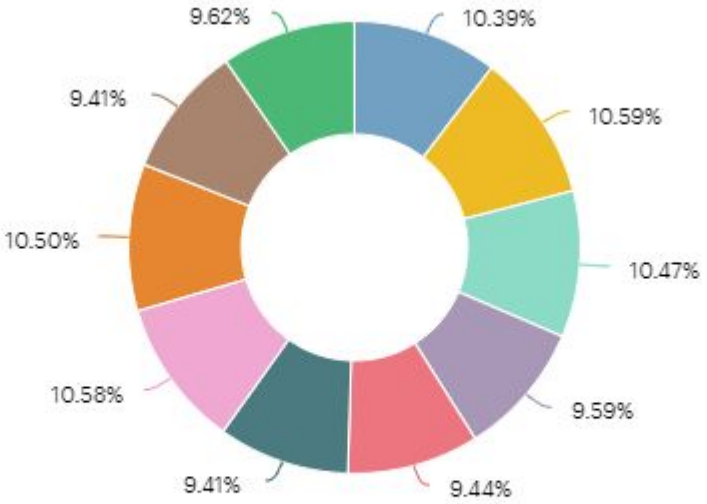
Implementation(2)

Popular repositories based on commit count



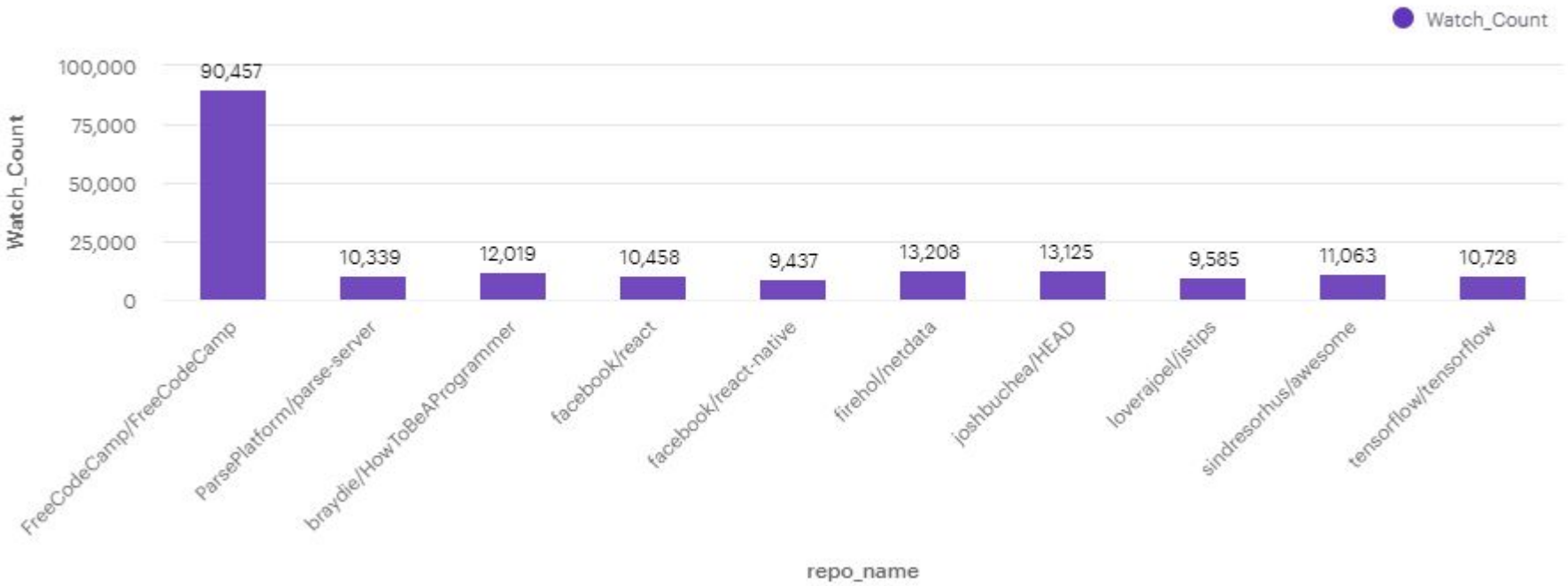
Implementation(3)

Popular repositories based on byte count



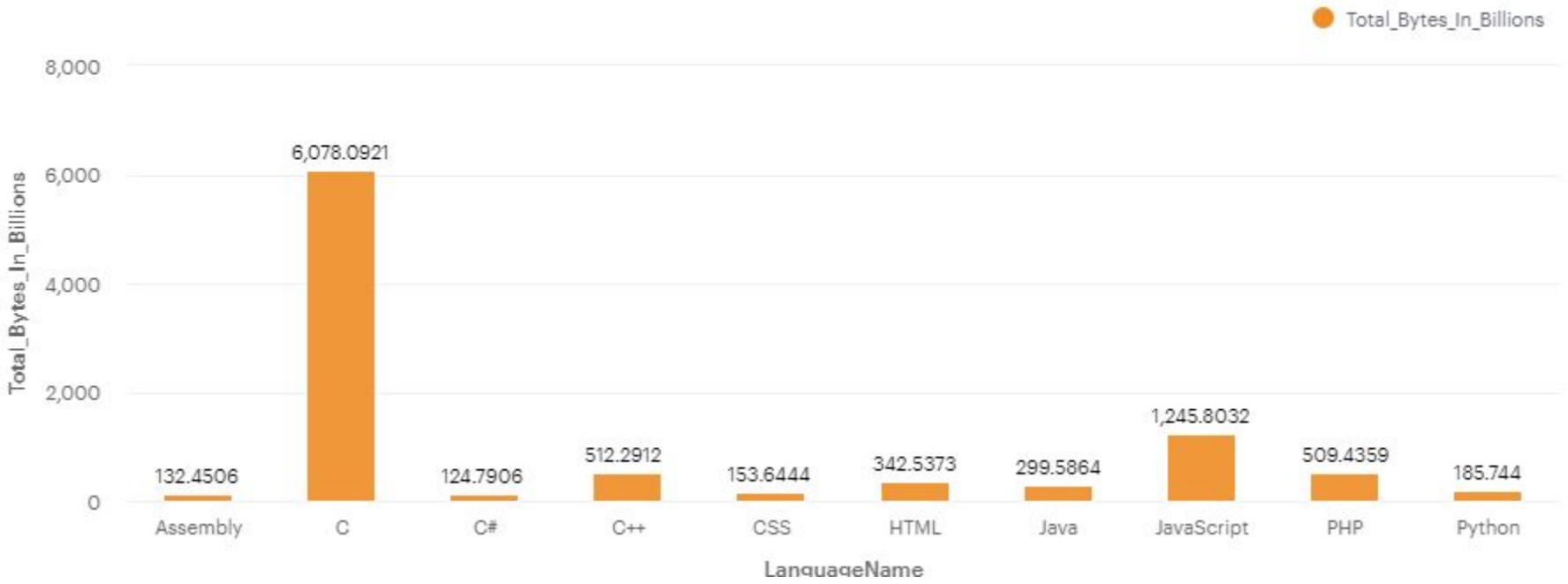
Implementation(4)

Popular repositories based on watch count



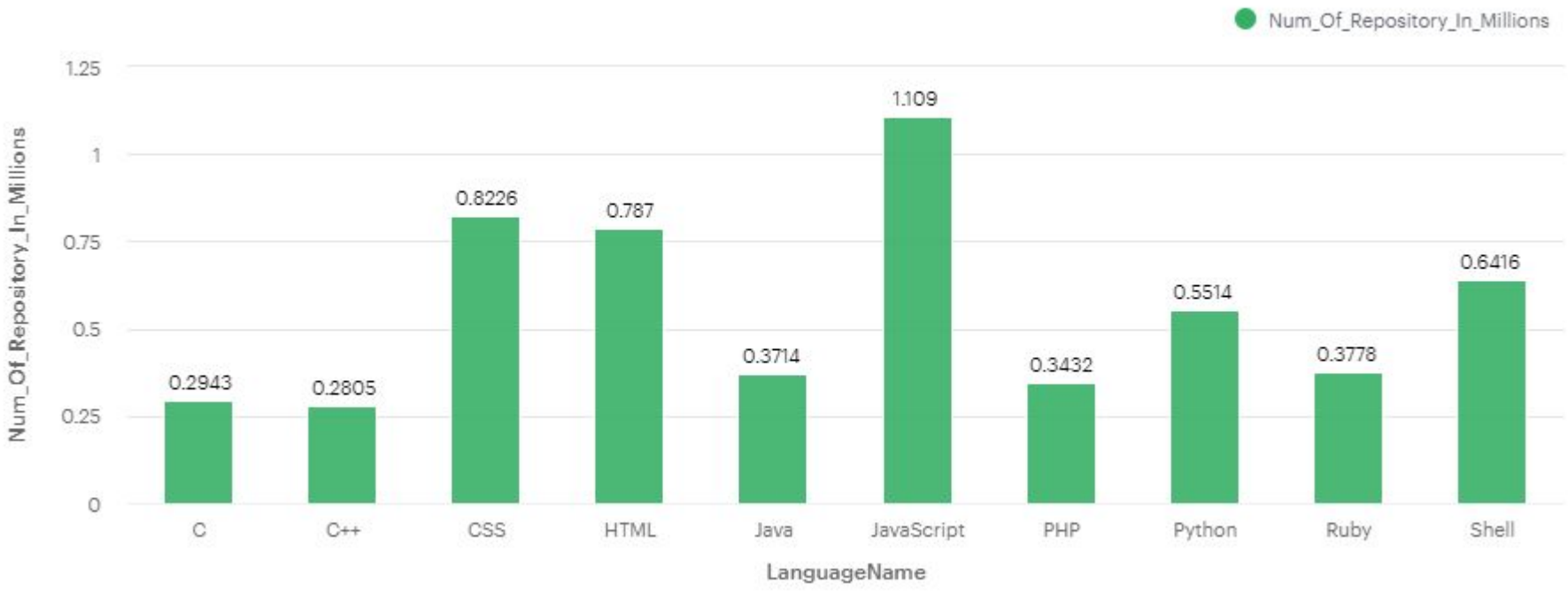
Implementation(5)

Popular languages based on byte count



Implementation(6)

Popular languages based on how many repositories used each language



Evaluation (Scalability)

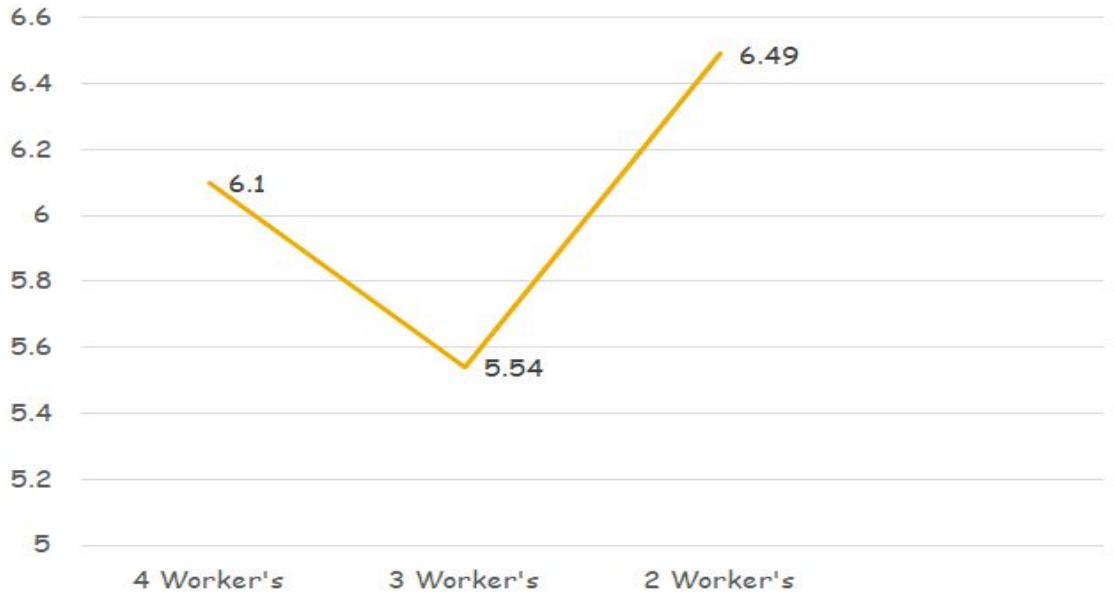
- Initially we started with 1 master node and 4 slave nodes.
- And then we added more Workers to our cluster.

Total No of Workers	Time Taken (seconds)
4	6 min 1 sec
6	6 min 35 sec
8	6 min 48 sec
10	6 min 50 sec

Evaluation (Fault-Tolerance)

- Initially we started with 1 master node and 4 slave nodes.
- And then we killed two workers one by one in our cluster.

Total No of Workers	Time Taken (seconds)
4	6 min 1 sec
3	5 min 54 sec
2	6 min 49 sec



Conclusion

- Thus we can conclude that there is increase in the usage of GitHub over time by technical professional for managing their projects.
- Due to increasing popularity of GitHub, it also provides easy integration with popular platforms like Amazon, Microsoft and Google.
- We can also conclude that based on different user criteria (like based on byte count,based on no. of repositories use language) we can get different ranking for the popular languages.
- We can conclude from output 6 that even though C language was used in less no. of repositories, it has max. no of lines of Code. So we can not conclude from output 5 that the C language is the most popular language.
- What do you think is the best way to gauge the popularity of a programming language?