A report on

Recruitment Engine

Submitted as part of partial fulfillment for
CS F415 : Data Mining

G06

Priyank Lodha	2014A7PS0021G
Abhimanyu Singh Shekhawat	2014A7PS0030G
Puneeth Saladi	2014A7PS0075G

Objective

Hiring software development talent – real talent – is a multi-faceted skill that lies at the crossroads of technical acumen, process management, social networking, and intuition. Anyone who has ever had hiring responsibility understands all-too-well the extent and depth of the hiring challenge. The main objective of our project is to assist recruiters in this complex task. We have focused our analysis on GitHub, which is home to the world's largest community of developers and their projects.

Approach

For our analysis, we have considered the top 256 contributors on GitHub for the time period of June 2015 to June 2016. We built a web scraper which extracted useful information from contributors' GitHub profile which included number of repositories, forked repositories, starred repositories, followers, active years, repositories in each language, their general bio data, etc. Now the main challenge was to decide which features among these were distinctive and the amount of correlation among them. We came up with the following research questions -

1. What is the correlation among the major languages used by developers?
2. What is the most widely used language?
3. What is the variance/jerkiness in the contributions made?
4. During what time of the week do most developers work?
5. How are the best contributors spread geographically?
6. What is the correlation among various other features?

Answering these questions provided us with enough insights which aided us in building our final recruitment engine. Depending on the requirements given and the insights that the above mentioned questions provided, our engine provides a comparison of the top candidates. The final decision however lies with the recruiter.

The underlying algorithm of our recruitment engine in the most basic terms is scoring each candidate according to the given requirements. The major requirements that can be given are - languages,years active, activities(average of),followers and variance. The recruiter would provide a significance(between 0 and 1) of each of these requirements. Let us say, the recruiter gives a significance of 0.4 to languages, 0.1 to years active, 0.1 to activities, 0.2 to follower and 0.2 to variance. Now he provides 3

languages - python, java and html and provides α, language importance score for each. As languages are highly correlated in similar domains, we calculate α for all other languages from the provided α's and take an average of it in order to finally end up with α of all major 50 languages. The formula to calculate α(c++) of one language w.r.t. to let us say α(python) and α(java) is given by -

$$\alpha(c++) = (\alpha(python)\beta(python, c++) + \alpha(java)\beta(java, c++))/2$$

Where, β is the correlation between 2 languages.

For each candidate then, his/her number of repositories in a particular language is multiplied with its α and then multiplied with significance of language as a whole (which was provided earlier). For other requirements i.e years active, activity, followers and variance, their respective values are multiplied with their significance. Now for each of the candidate, these 54 values(50 languages + 4 other requirements) are added to get the final score for each user. The users are sorted according to decreasing order of their score and an analysis is shown among top n candidates. Here the n can be specified by the recruiter.

Learning Outcomes

This project provided us with ample opportunities to learn about various techniques and technologies related to raw data scraping, processing, extraction of insights and visualization. Some of them are -

1. The quantity, quality and format of data stored by GitHub on user profiles, repositories and user connectivity.
2. The working of version control system, as we ourselves used it to collaborate on the project.
3. Various python libraries,
    a. BeautifulSoup used for scraping websites
    b. pandas and numpy used for data manipulation
    c. Matplotlib, seaborn, plotly and gmplot used for data visualization
    d. geocoder used for getting geographical data
4. The practical aspects, complexities and considerations associated with data mining.

Inferences

The major inferences extracted from the graphs which we created in order to answer our research questions are -

1. As seen from the heat map of languages, languages in the same domain have more correlation with each other when compared to languages in other domain. For example, if we look at Javascript, contributors with more number of repositories in javascript have equally more repositories in HTML, CSS, Python and Ruby in decreasing order.
2. The winner of most widely used language is Javascript with a total of over 16000 repositories among the top 256 contributors. Although no other language comes near to this number, the next most popular languages are Ruby and PHP. This would imply that the major contributors on GitHub focus on web development than any other domain.
3. The mean standard deviation of contributions over the last 3 years has seen a shift from 11.45 in 2015 to 10.56 in 2016 to 8.39 in 2017. This means that the major contributors are tending to work regularly more and more each year.
4. When we plotted the frequency histogram of the ratio of commits on weekend to all days, the mean turned out to be 0.20. If developers actually worked equally during all days, the ratio should have been 0.28. This implies that developers prefer working on open source projects during weekdays.
5. Major contributors seem to be concentrated in developed areas of Europe and US. Rather being spread out over the entire globe, they occur in concentrated patches. This hugely affects the recruitment process as these developers will prefer to remain in these areas itself.
6. We found small correlation between the following -
    a. Repositories and language count
    b. Starred repositories and following
    c. Years active and repositories
    d. Language count and forked repositories

Results

The results from our recruitment engine looks promising. It is able to give us an apt comparison of candidates. This can be seen in the graphs attached with this report.

Conclusion

Our recruitment engine performs the intended job well enough. It takes into account not the strict requirements of the recruiter but also the general trends and modifies the search criterion appropriately.

The future work for our project would be to build a GUI that would pipeline the entire process, so that when the recruiter provides their organization's requirements, everything is automated.