This could probably be contained inside a GitHub section. We could explain breifly about github and what it offers for data collection. Such as an api etc. and then below explain what we chose to do for collecting metadata and why. Then other subsections could explain about each of the methods of collecting source that we used, commits and projects.

1 Github Metadata

Perhaps a little introduction to why we are getting metadata and what we need it for.

1.1 Ghtorrent

Ghtorrent is an 'offline mirror of data offered through the GitHub REST API' (from website). It contains large quantities of data on GitHub users, projects, commits from as far back as 2012. This data is being collected (by whom) to support research on software repositories and is available to the public for other research on Github.

We chose to use Ghtorrent because it allowed us to filter and collect spcific sets of commits and projects without having to use the github api. The github api has a rate limit and in order to process the amount of data we were able to with ghtorrent would have taken a very long time. Ghtorrent allowed us to filter out data that was for specific languages as well as make sure we were only looking for projects that had a country attached to them.

1.2 Google BigQuery

The dataset is available for download, but it is very large (100 Gb for the mysql). To make it possible to run queries on it without downloading it the dataset has been made available on Google BigQuery. This make accessing it much easier and faster, though not without it's issues (should we list some of them here or later). We were able to use SQL queries to collect only the information that we needed for our research. We were then able to download these results to use them to facilitate collecting source code for valid projects and commits.

1.3 The Data

Our research centers on how different sociological features of a programmer affect their use of programming languages. So far we have been exploring things like the gender and the region of the programmer. We were looking for commits and projects in langauges like C++, Python, and Java. We needed these to have regional data associated with them and ghtorrent provides country, state, and city for a user. All projects have a programming language attached to them as well so we were able to filter by language easily.

Using the dataset we were able to filter out millions of commits for each language. Using SQL we further took random samples of 500k rows for each

language. We then were able to download them and use the github api to get the content of the commits. We were also able to filter out 150k plus projects for each language that had not been forked. From these we took random samples of 150k projects and used some python libraries to download projects and collect source files from them.

1.4 Issues

Because the Ghtorrent database holds a lot of older data it is not all still available. Some of the projects are marked as deleted so we filtered these out. There are also fake users (which means what?) that we mad sure to filter out. Whe taking projects specifically we filtered out forked repositories to give ourselves a better chance of finding single author code. We also had as mentioned above to filter for users with countries and our desired programming languages. Even with our attempts to filter these things. Some projects have been made private since their data was cached so we could no longer access them. Also even though a project is in a specific language it does not mean that it won't have other types of files in it. So it is necessary when collecting source code to filter for only files that are in the desired language. This can easily be done by checking the files extension. While using Ghtorrent makes it much easier to get metadata it does not offer everything that one might want.

There were a few specific things were were not able to get. While user logins can be used to connect commits and projects to an author, there is no name information available for users on Ghtorrent. Since we are doing analysis on gender it is necessary to get names in order to determine gender. If name information is available on a user, it is obtainable through the github api. There is also no source code or anything of that sort in the ghtorrent dataset. It only contains things like user info and project names and urls, etc. It is a great resource, but for us it is only the first step in getting what we need.