1 Detailed Problem Description

Goodreads is a website launched in early 2007, which lets "people find and share the books they like and improve the process of reading and learning throughout the world." It is the world's largest site for readers and book recommendations with a user base of about 30 million members along with 34 million reviews from 900 million books as recorded in 2015 [**?**].

Goodreads provides a multitude of features to its users. It goes beyond the traditional rating and reviewing of books by allowing users to make friends and join and form reading groups based on their literary tastes. Users can not only see what their friends have read, but they can also meet new people with similar reading interests. They can make recommendations to friends, follow authors, track the books they are currently reading, have read and want to read. In addition, goodreads provides personalized recommendations to book readers by analyzing the user data.

Basically, we want to detect and study goodreads communities based on books reviewing. Two people who has given similar rating for the same book share a similar interest, and thus are closed. We also want to improve this homophily by giving a negative score to different rating of the same novel. We wonder if communities based on book tastes are analogous to friends communities, or what the average homophily score is between two friends?

2 Related Work

3 Data Acquisition

Goodread provides well documented and publically accessible API to query various features supported by the website. All we need is a developer key, which one gets after signing up on goodreads, and an OAuth access to use numerous APIs. The data of the users on a website, being the concern of privacy, is not readily available by means of goodread APIs. Aside from this a goodread user has liberty to keep his/her profile private making it accessible only to the friends on goodreads. Given the large userbase of goodreads, we hope to get sizeable dataset to run our analysis.

Althought the data scraping is laborious and time consuming, its is vital in our study. For a fair study, the data must be unbiased. The method which we use to gather the data may add some bias to it. So selection of scraping method is critical to data acquisition. There are two ways one can collect the data:

Retrieve User IDs from a group Goodreads hosts various reading groups catered to various genres and reading interests. User can become member of such groups or can initiate a group. In a group reading activities are supported wherein members can schedule the book readings so that they can share their opinions and conduct a healthy conversation. *Goodreads Authors/Readers* is one of the major groups on goodreads. According to goodreads, this group is dedicated to connecting readers with goodreads authors. It is divided by genres, and includes folders for writing resources, book websites, videos/trailers, and blogs.
Goodreads provides an API to get the list of users given the ID of the group. One can get the ID of the group from the URL of the corresponding group on goodreads.

Retrieve User IDs from reviews of books The list of various books genres is available at **??**. One can select top rated books from the highly reviewed genres. So this collection of "best-selling" books forms a base set for the books. So we can get the list of users who have read

these books. But goodreads fails to support an API which retrieves reviews for a particular book given its ID. Despite this, a meticulous analysis of *Javascript* calls, the webpage makes to show the reviews of a book to the user, reveals a bacdoor API *book/reviews/*. We can retrieve user realted information User ID and the rating by parsing the response to the *Javascript* callback using regular expressions.

The list of User IDs thus scraped serve as a base user set. We later collect *1-neighbourhood* of this set of users, namely friends of them and the books read by these friends, to get a substantial dataset.
We have decided to go with the first strategy. By scraping a toy dataset, we observe the data to be unbiased. Plus the data related to the friends assures the sizeable amount for the analysis.

4 Description of the method

5 Discussion of project progress

We have queried the first reading group entirely. The statistics computed based on this first scraping pass are:

- 567461 ratings
- 7815 public profiles
- 2564 private profiles
- 7203 profiles without a review
- 1948 profiles

We did not know we would get so many fake or private profiles. Within a group of 20000 members, we got less than 8000 users.

We decided to scrape all the friends of the users we have found to get more data. Among those 128213 friends, we have removed the profiles already in the group, which are private or do not have any review to store a list of 95503 active profiles who have likely written more than 5 million ratings.
To accelerate the scraping, we used multiprocessing and we set up a Tor connection, even if we did not have used it. While getting the last data, we focused on the algorithm part.