CS5232 Project Proposal

Antoine Francois Pascal Creux (A0123427M)

March 18th, 2014

1 Concept Study

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 worlds 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[3].

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.

To capture the notion of similar interest among the users of goodreads the concept of one-mode analyses of bipartite network data(also known as affiliation network in sociology terminology) can be used [**?**].Such analyses use matrices derived from the affiliation matrix, $\mathcal{A}$, wherein actors are represented in rows and the events are represented in columns.
Co-membership of actors i and j for an event k is identified if the rows a_i and a_j both have 1 in the column ak . Hence, $a_{ik} = 1$ and $a_{jk} = 1$ indicates that both actors participated in the event k. So, the total number of co-memberships for the two actors can be computed from the number of times that $a_{ik} = 1$ and $a_{jk} = 1$ where k takes all possible values. The number of events with which both actors i and j are associated will vary from 0(if both have no affiliated events in common) to h(if both have all affiliated events in common).

2 Problem Description

In this project, we aim to probe the groups of users within goodreads based on their reading interests. As stated earlier, goodreads is more than just a book reviewing website. Users and authors on goodreads can connect amongst themselves forming a rich social network. We do not want to observe the groups/clusters within such a social network, which already exists in goodreads, since this network may not necessarily capture the reading interests of the users in it. We use the information about the books to capture the ties amongst the users.

Let U and B be the set of users and the set of books on goodreads respectively. We will construct an undirected bipartite graph $G(V, E)$ using these two sets where $V = U \cup B$. An edge exists betweem a vertex $u_i \in U$ and a vertex $b_j \in B$ if a user u_i has read book b_j.

We consider the rating given by a user u_i to a book b_j to assign weight to the edge between them. The weight to this edge plays an important role in the semantics of the networks. Consider a scenario wherein a book b_k is read by both the users u_i and u_j. Suppose user u_i has liked the book and hence has positively rated the book whereas user u_j has not liked the book and so has given a low rating to the book. If we do not take this fact into the account then these users might end up being placed in a same reading group. So as to prevent situation, we assign weight to the edges in G. Let r_{ij} be the rating given by user u_i to book b_j. Let $w \to E \times \{1, -1\}$ be the weight runction. For every $(i, j) \in E$,

$$w((i,j)) = \begin{cases} 1 & r_{ij} \geq \alpha \\ -1 & r_{ij} < \alpha \end{cases}$$

where α is some threshold which can be tuned during the experimentation.

Let $\mathcal{M}$ be a $|U| \times |B|$ matrix which represents the underlying graph G. If there exists an edge between user u_i and book b_j then $\mathcal{M}_{ij}^{th}$ entry in the matrix is set to the weight of the corresponding edge otherwise the entry is set to zero. We obtain $\mathcal{U}$, matrix encoding the relationship amongst the users, using $\mathcal{M}$ by calculating $\mathcal{M}\mathcal{M}^T$. The graph encoded by $\mathcal{U}$ is the graph of the interest for finding the user groups. We propose to explore the interesting user groups by using graph partitioning algorithms like Girvan-Newmann algorithm.

3 Underlying Assumptions

4 Data acquisition

Goodreads data is not available online, we have to acquire all data through their APIs. Stanford students have already studied Goodreads platform by building a product recommendation. [2]. However, they have not published the data they acquired and we think that their data acquisition is wrongly implemented based on erroneous assumptions. Indeed, they acquired data through the *book.shelves* api method. Given a *user_id* and a bookshelf name, they were able to query all the books a user may have put in this virtual bookshelf. The major issue is that goodreads do not list every user_id. They then decided to query for user_id randomly to query the books people have put to their public 'reading' shelve. Finally, they only gather 4000 users and their 'reading books'.

We have decided to query the data using other APIs, available at [1]:

- group.list - List groups for a given user
- group.members - Return members of a particular group
- group.show - Get info about a group
- fanship.show - Show author fanship information
- owned_books.list - List books owned by a user
- shelves.list - Get shelves owned by a user

- user.show - Get shelves owned by a user
- user.followers - Get user's followers
- user.following - Get people a user is following

Here, we see we can add many kinds of information related to the network of a user:

- such as in Twitter, a user can follow other users, and can be followed
- a user can take part in reading groups
- a user can be fan of an author
- we have all books stored in all shelves of a use

Goodreads have 30 million users which may be too large for our study, but the goal is to gather an acceptable amount of data. Goodreads provide the top user in certain categories such as the top 50 active users, top 50 readers, top 50 most popular reviewers well as the best reviews. These statistics are computed for the last week, last month, last 12 months and all the time for each country(as well as worlwide). Based on the top users, we have a list of user_id from which we can query the data. On the other hand, for each user, we will store the books read and stored on the shelves, the authors the user is fan of and the groups the user belongs to. Finally, we only can query a user data if the profile is public. Based on our first study, most profile are public and thus should not raise an issue. Even so, we will store the information that a user profile is private.

5 Project Objectives

6 Literature review

We compare n methods for the clustering of our dataset. We give a brief summary of the different clustering methods and state if these methods will be scalable to our own dataset.

Quantitatively, clustering aims at assigning each data point to a cluster.

6.1 Hierarchical Clustering

Hierarchical clustering builds a hierarchy of clusters, Either it is built *bottom up*, as we start with the same numbers of clusters and data points, and we try to group clusters with each other. The *top down* us the opposite: we start with a unique cluster that we are going to split recursevely.

Complexity is very high ($\Theta(n^3)$), and thus is not pertinent to big datasets, even if heuristics lower the complexity to ($\Theta(n^2)$.

6.2 K-Means Clustering

K-means clustering aims at computing k clusters from that will minimize the distance from the data points to the cluster, such as we will split n data points into k groups. It is a centroid-based clustering, and does not apply only to graphs. The most common application of k-means clustering is the iterative algorithm named Lloyd's algorithm. The major issue is that we have to set in advance the number of clusters we want to

1. Initalize (randomly preferably) the center of the k clusters
2. For each point, assign to it the closest cluster
3. Compute the new position of every cluster as the center of all data points assigned to this cluster
4. Repeat 2 and 3 until the clusters do not move (Relative to a ϵ for instance)

7 Proposed Contributions

8 Success Measures

9 Project Plan

10 Deliverables

References

[1] goodreads - APIs documentation. `https://www.goodreads.com/api`.

[2] Book Recommendations on GoodReads.com. `http://cs229.stanford.edu/proj2008/IsaacsonSebastian-GoodReadsRecommendations.pdf`, 2008.

[3] goodreads - About us. `https://www.goodreads.com/about/us`, 2012.