Recommendation System based on Disambiguating Social Network Entities with Freebase

by Agustin Baretto

The purpose of this project will be to create a books and movies recommendation system for any of the Facebook friends of a user account based on the Facebook Pages each of them liked. To do this, we will use the Collaborative Filtering technique in order to make predictions about interests of each friend, based on those items and tastes information from the rest of users within the same graph. We will also make use of Freebase knowledge base (in particular its reconciliation feature) to perform disambiguation of Facebook Pages in an iterative manner. This will allow us to minimize redundancy between different Pages referring to the same real world entities and therefore have a more accurate collection of items, and also be able to even extend our recommendations to items that are not within our collection but which we can obtain by making use of Freebase data (like other movies from the same genres or directors for example).

Keywords: **Freebase, Facebook Pages, Name Entity Disambiguation, Naturally Disambiguated, Type Taxonomy**

1. Introduction

With the growing availability and popularity of Social Networking and blogging sites and the proliferation of mobile devices, new opportunities and challenges appear as people can now actively generate contents that provide a unique compilation of information that is more updated and inclusive that traditional media. These contents also allow us to shape those users behind the posts and get some more information about them other than just their nicknames. Representing the semantics of individual users activities and modeling the interests of users can help to improve contents personalization, establish user profiling strategies and countervail the information overload.

2. Related Work

There are several ways to analyze data generated from Social Networks and one of the most popular is **NLP analysis.** This technique is applied on entities (e.g., products, organizations, people, etc.) in posts and becomes a rapid and effective way of gauging public opinion for business marketing or social studies. However, their unique characteristics (like tweets limited length and their noisy and informal nature) give rise to new problems for current semantic analysis methods, which originally focused on large opinionated corpora such as movie reviews. It is hard to perform complex analysis on such short pieces of texts and algorithms need to rely only on simple methods which do not provide significant results when tested against real data.

Existing **disambiguation methods** fall into three categories: unsupervised approach, semi-supervised approach and supervised approach:

- The *unsupervised methods*[1] are based on unlabeled corpora, and do not exploit any manually tagged corpus to provide a choice for a phrase in context. They mainly rely on some heuristic rules, or the predefined similarity metrics. These methods are very simple and easy to implement. But the overall performance can't achieve better result than supervised methods.

- The *supervised approaches*[2] employ the annotated data set. They first extract various appropriate features and then design supervised classifiers to select the best target entity for the phrase. Compared to unsupervised methods, the supervised methods can achieve better results, but it is more expensive and time consuming to acquire large numbers of data labeled by humans.

- *Semi-supervised approaches*[3] rely on generative models that try to describe the generation process of a corpus to employ small numbers of labeled data and large numbers of unlabeled data.

Many of these problems have been already addressed and researchers seemed to have found good workarounds for quickly training data (such as training classifiers based on sets of data containing emoticons[4] or using dictionaries of specifics words) and analyze short pieces of text using unsupervised methods[5], but none of them seems to have found an accurate method to disambiguate data and make a proper use of such information.

3. Facebook Pages and Ambiguity

Our approach will not be to improve the NLP methods currently being used to analyse text content from users but rather focus on their actions, which happen to be quite powerful when trying to understand their interaction with real world entities (something specially important for disciplines such as Marketing, Sociology, Economy and Politics). Social Networks such as **Facebook**, give users far more abilities than just microblogging. Users can seek out information and contents related to the things they like or feel sympathy for (a rock band for example) and this way have the chance to meet other people with similar tastes. This usually takes the shape of **Pages** and Groups in the platform, which are communities created by users which allow other users to join and interact. In order to have access to those groups users need to subscribe to them by "liking" them. This action can

[1] C. Wang, K. Chakrabarti, T. Cheng, and S. Chaudhuri, "Targeted disambiguation of ad-hoc, homogeneous sets of named entities,"
[2] X. Han and J. Zhao, "Named entity disambiguation by leveraging wikipedia semantic knowledge,"
[3] Sergei Brin. 1998. "Extracting patterns and relations from the World Wide Web."
[4] Kun-Lin Liu, Wu-Jun Li, Minyi Guo, "Emoticon Smoothed Language Models for Twitter Sentiment Analysis"
[5] Lei Zhang, Riddhiman Ghosh, Mohamed Dekhil, Meichun Hsu, Bing Liu "Combining Lexicon-based and Learning-based Methods for Twitter Sentiment Analysis"

also be shared with the public in order to let the rest of the users know about such interaction between user and page.

As said before, Facebook Pages (and their respective contents) can be created by any user without any supervision from the community other than suggestions sent to the creators or abuse reported to Facebook. Due to their nature of virtual communities, Pages do not aim to have the objectivity and accuracy visitors would expect from a collaborative site (such as *Wikipedia*[6]) where a community of users edits and reviews articles created by other members. This becomes an impediment in our research as again, there is ambiguity and data redundancy: different pages might refer to the same real world entity (*There are more than 100 pages of "The Lord of the Rings"*), categories assigned to a page might be incorrect (*the Page dedicated to a book writer -JR Tolkien, for example- might appear under "Book" category instead of "Author"*), titles of pages might be misspelled or include words that do not correspond to the actual titles (*"Lord of the Rings, official page"*), titles of pages might be in different pages in different languages (*"Lord of the Rings", "El senor de los Anillos"*) to name a few of the integration challenges we face.

Facebook has been already trying to address this problem in many ways. On the one side: it now allows users to add suggestions to the page creators. So a user could potentially suggest the creator of a Page to change its title or moreover tell him that the Page refers to an already existing entity. However, the decision is still on hands of the Page Admin, who is probably aware of such information and will hardly modify the page or even remove it.

On the other hand, Facebook launched some years ago its Social Graph. People, places and some other entities are stored as nodes and the relationships between them as edges that can be queried and modified by apps and sites integrated to Facebook. This is not completely open to the public (only to developers agreeing with Facebook Terms and Conditions) and it is limited to Facebook's data only. This movement was seen by some[7] as a way to compete with Google which on its side acquired Metaweb[8], creators of Freebase an "open, shared database of the world's knowledge". Freebase[9] is a massive, collaboratively edited database of cross-linked data which unlike Facebook Graph, offers links to external sources all over the web.

4. Freebase

Our approach will be to use **Freebase** in order to reduce duplication, correct deviations from correct terminology and converge different references to the same entities. This will allow us to perform semi-automated mapping between Facebook Pages and Freebase Entities that can scale to large numbers of nodes. Freebase (Bollacker et al. 2008) is a large collaborative

[6] http://en.wikipedia.org/wiki/Wikipedia
[7] "A new emphasis has been put on finding ways to harness the site's hundreds of millions of members as a kind of human indexing squad, a flesh and blood version of Google's Web crawler." MIT Technology Review (http://www.technologyreview.com/news/511591/facebook-nudges-users-to-catalog-the-real-world/)
[8] http://techcrunch.com/2010/07/16/google-acquires-metaweb-to-make-search-smarter/
[9] http://www.freebase.com

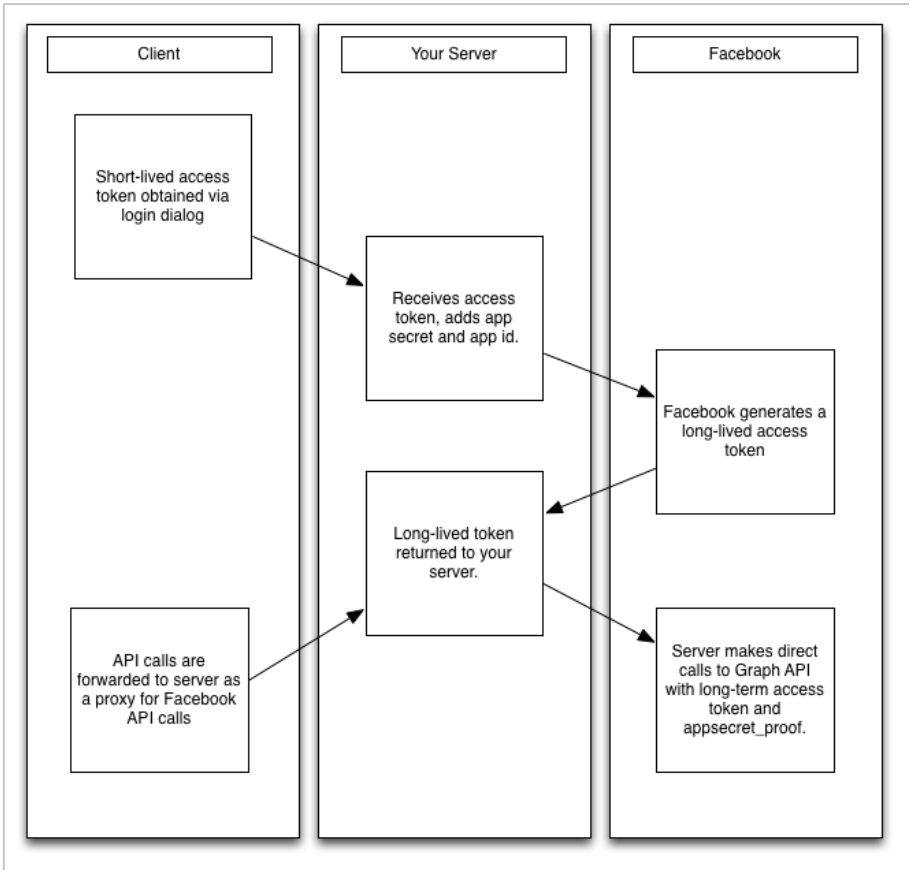
database of general human knowledge. It is publicly readable and writable using an HTTP-based query language[10]. There is some research done on entity disambiguation using Freebase[11] [12] but none focused on its implementation on Facebook's user generated Sites.

Freebase has a rich taxonomy and well defined schemas for the entities, and is therefore considered more as a structured database than other alternatives such as Wikipedia. The types of an entity determine the schema of attributes for that entity. Freebase represents data using objects and properties. Every object has properties */type/object/name* and */type/object/type*, so for any given string we can easily query Freebase for all objects with that name, and generate a list of possible types. Since an object's type determines what other properties it possesses, we can eventually use these candidate types to get candidate properties to map generate recommendations based on such properties (for example, by suggesting books that belong to the same genre than those preferred by the user getting recommendations).

5. Implementation Data Gathering

To better illustrate the different steps taken and the results from every action, we will use the same example accross this document by referring to the same Facebook account graph. The first step consisted on connecting to the Facebook Open Graph in order to get each of the user friends, and for each of those contacts, the list of Pages under "Books" and "Movies" categories. This is done by (1) authenticating the app with a set of credentials provided by Facebook Developers, (2) signing to Facebook using the selected account, (3) giving permissions to the app to access the user account's information (4) receiving an OAuth token and (5) using that token in order to generate HTTP queries to the graph.

[10] Alan Ritter , Evan Herbst, "Interactively Querying and Updating Freebase with Web Tables"
[11] Zhicheng Zheng, Xiance Si, Fangtao Li, Edward Y. Chang, Xiaoyan Zhu, Google Inc "Entity Disambiguation with Freebase"
[12] Qingqing Cai, Alexander Yates, "Semantic Parsing Freebase: Towards Open-domain Semantic Parsing"



The result of the query consists of a list with each of the friends and a list of books and a list of movies (if any) for each of those friends in a format as the following (real ids have been obfuscated for the sake of privacy):

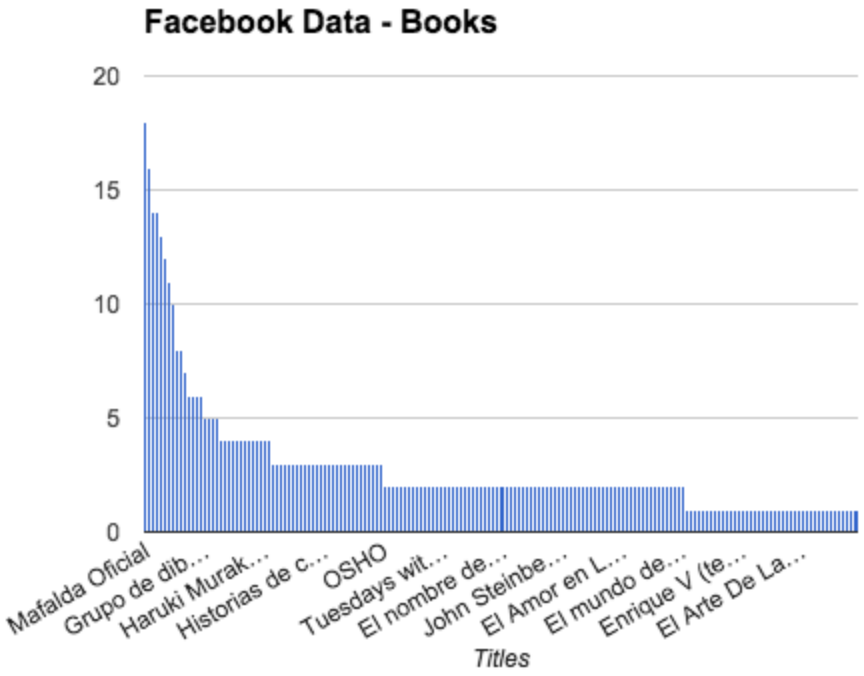

```
"id": "XXXXXXXXX",
        "books": {
          "data": [
            {
              "category": "Book",
              "name": "On the Road",
              "created_time": "2014-05-01T23:56:02+0000",
              "id": "613324775416871"
            },
            {
              "category": "Book",
              "name": "Steppenwolf (novel)",
              "created_time": "2010-11-01T02:41:57+0000",
              "id": "108099249219091"
            }]
        },
        "movies": {
          "data": [
```

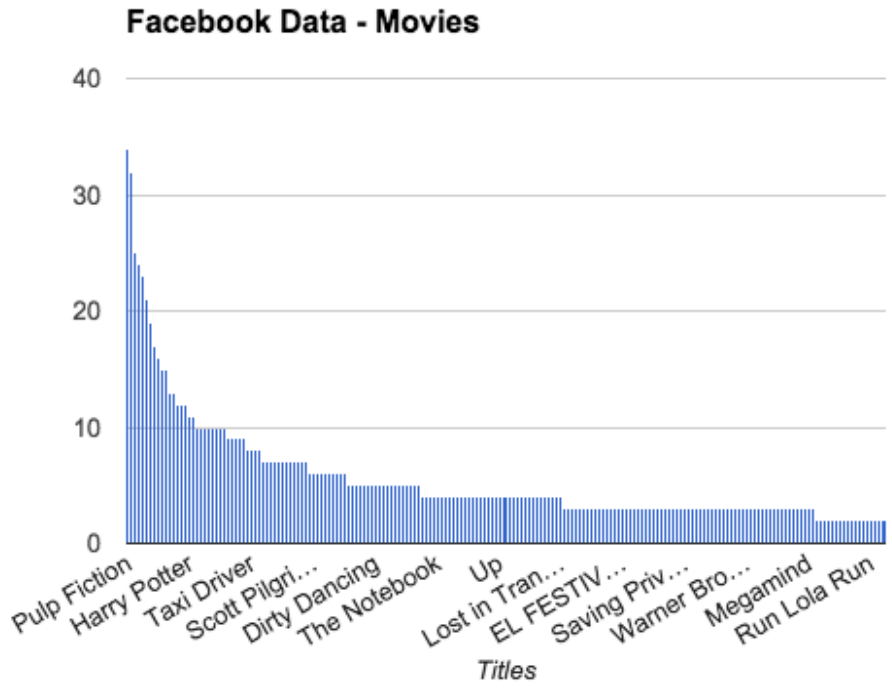
```
        {
          "category": "Movie",
          "name": "Rushmore Movie",
          "created_time": "2014-10-08T07:58:19+0000",
          "id": "1507702766138376"
        },
        {
          "category": "Movie",
          "name": "Commando",
          "created_time": "2013-11-23T07:50:29+0000",
          "id": "1428297240727377"
        }]
    }
```


6. Data Cleaning

We first need to perform some preprocessing on data, so we proceed to reduce dimensions by discarding those fields that are not useful at this step (such as category of the movie, id and created time). Expected common behaviour would be to keep id and discard name because of its uniqueness and ease of use, but the purpose of this stage will be to match different pages and see if they refer to the same real world entity and the only way to do so is by analyzing the strings of text that describe the titles of such pages. We also need to take account for missing values, so we remove those tuples or individuals not having any book or movie associated.

After this, the data is converted into a vertical data format as we have many more items than users and this ordering eases iterating over each single title during the following steps. List is rearranged into 2 separate lists, one for books and another for movies, each of them containing the existing titles and the ids of users interested. At this stage of the process, we have a list of 793 books *(with stdev=1.4933, avg=1.4124)* and 1411 movies *(with stdev=2.4112, avg=1.7647)*.





Facebook Pages can have different titles to be returned for different languages. Many times, pages in languages other than English will also have the English version of the title. However, pages in English will not have titles in each of the other languages in general. The results

shown above were obtained by querying *Facebook Graph* and specifying the locale parameter as ES_LA (*spanish latin america*). However, after benchmarking with other options we discover that the films results vary if we set locale to EN (*english*). New list contains 1386 different results (which is better, as it means 5 Page redundancies were corrected). As for books, there is no difference in results.

7. Facebook Pages Disambiguation with Freebase

Following step is to reduce numerosity of the itemset by disambiguating titles of Facebook Pages against Freebase database. We intend to check if the title of the page we have matches with or is similar to any existing real-world entity uniquely identified in Freebase and if so we ask for such entity's correct name and Freebase id (MID). To do this, we also need to authenticate our app using our Google API credentials and then proceed to query the database. The specific feature or tool we use is **Reconciliation**. The Freebase Reconciliation API is used to uniquely match an entity in Freebase with some structured data about that entity. The Reconciliation API responds with a list of possibly matching MIDs (entity ids) and confidence scores. The confidence score is the probability that this entity is the unique matching entity given the specified property-values.

After running our first reconciliation round, we have a new improved list of books and movies: 788 books (*454 found in Freebase, 334 not found*) and 1310 movies (*872 found in Freebase, 438 not found*). As for the items not found during the first round, we run them against the reconciliation tool, this time specifying spanish as the language to query by. New items that had not been found before appear in Freebase, bringing a set of 787 books (*470 matched, 317 not matched*) and 1300 movies (*911 matched, 389 not matched*). Finally, after analyzing the results not found after this second round, we discover that many of them contain extra words added by the Pages Administrators such as "The movie" or "Official Page" that Freebase is trying to query together with the tiles of the items and is not able to find. We proceed then to create a dictionary of common words we find and automatically remove it for the last set of words and run the new words through the reconciliation tool twice (one time in english and another in spanish in case it did not appear in the first round). End results are encouraging: Books have been reduced from 793 titles to 770 (2.99%) and 468 (60.78%) of

them are associated to a real-world entity which can be further queried in order to get data other than the one proportioned by Facebook. Movies have been reduced from 1411 titles to 1275 (10.67%) and 945 (73.57%) of them have been found in Freebase

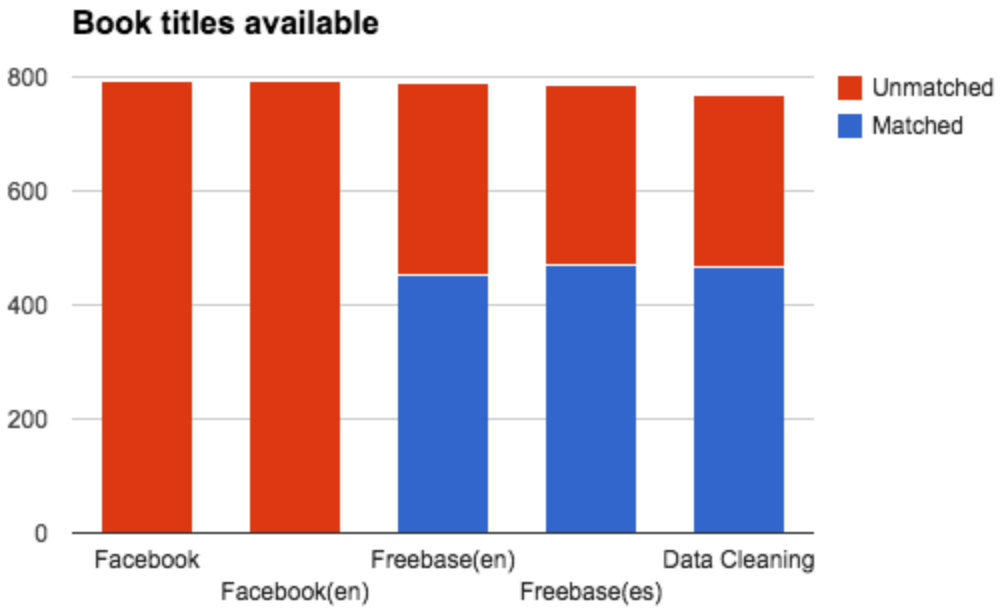


fig.1: book titles disambiguation evolution

	Facebook	Facebook(en)	Freebase(en)	Freebase(es)	Data Cleaning
Matched	0	0	454	470	468
Unmatched	793	793	334	317	302
Total Books	793	793	788	787	770

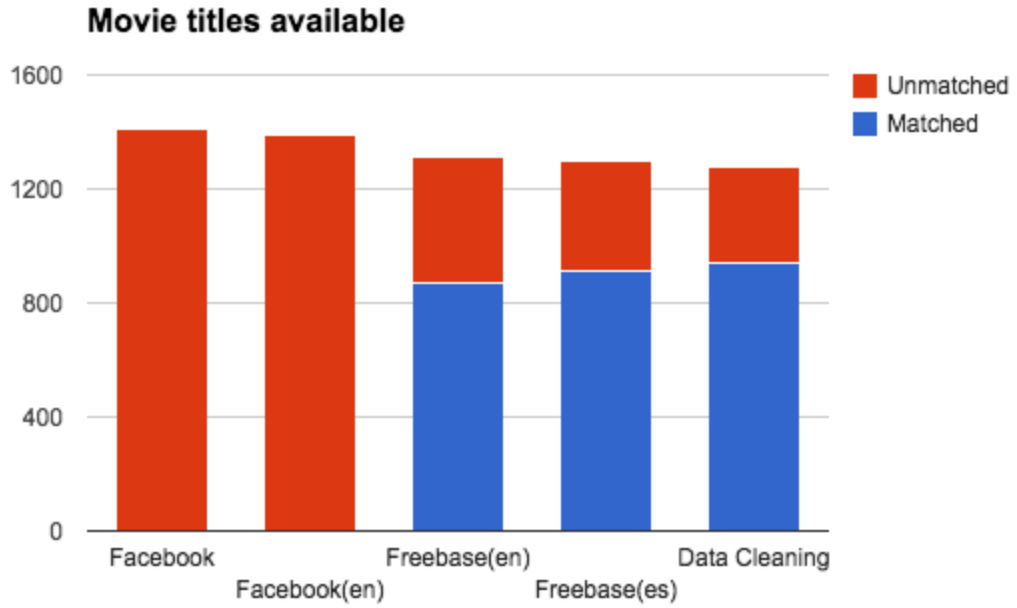


fig.2: movie titles disambiguation evolution

	Facebook	Facebook(en)	Freebase(en)	Freebase(es)	Data Cleaning
Matched	0	0	872	911	938
Unmatched	1411	1386	438	389	337
Total Movies	1411	1386	1310	1300	1275

It is important to consider that items not found on Freebase do not necessarily mean an inconsistency or error margin. After analyzing data individually we found that many of these were recent titles that still did not appear on Wikipedia, or titles of independent movies which are not mainstream and do not have an article as well. Also, we need to take into consideration that the Facebook account being used to illustrate our method was from a native Argentinean, so many of the titles (specially books) are not available at a worldwide level. On the other side, we also do not want this kind of titles which happen to be very specific to a location to be suggested to people from other places which might not have access to them, so it is somehow rather desired to discriminate them in order to take a corresponding action.

8. Collaborative Filtering

After the preprocessing step is done we proceed to focus on the recommendations engine. The aim of this research is not improve or measure the efficiency of recommendation algorithms so we will only implement existing Collaborative Filtering method to find and rank those items that might be of interest for the user being evaluated. The motivation for collaborative filtering comes from the idea that people often get the best recommendations from someone with similar tastes to themselves. Collaborative filtering explores techniques for matching people with similar interests and making recommendations on this basis.[13] Our approach first measures the distance or interests similarity between each member of the graph and then suggest an ordered list of items based on that distance.

9. The application

We developed a PHP and Javascript web application in order to show how Facebook Pages disambiguation with Freebase could be used in the real world and give value to end users. An interesting fact about the recommendations engine is that it can predict not only items that the owner of the account might be interested in reading, but it also allows to get suggestions for any of the members within the graph, which make it rather different from state of the art recommendation systems.

User needs to first login with his/her Facebook account and give permissions to the app in order to access the Recommendation System. Once this is done, user can either pick the Books or Movies option and select one of his/her friends. This person selected will be the subject of the recommendations and a list of potential items of interest will be displayed on the screen. The user can also click on any of the titles in order to do a reverse search. This feature, allows to search instead for individuals who might be interested on a specific item.

Finally, the app also displays a graph containing all of the network contacts of the user and the books and items they like as nodes and the interest relationships as edges. User also has the possibility to click on any member node within that graph and get redirected to the friend suggestions page for that person.

[13] http://en.wikipedia.org/wiki/Collaborative_filtering

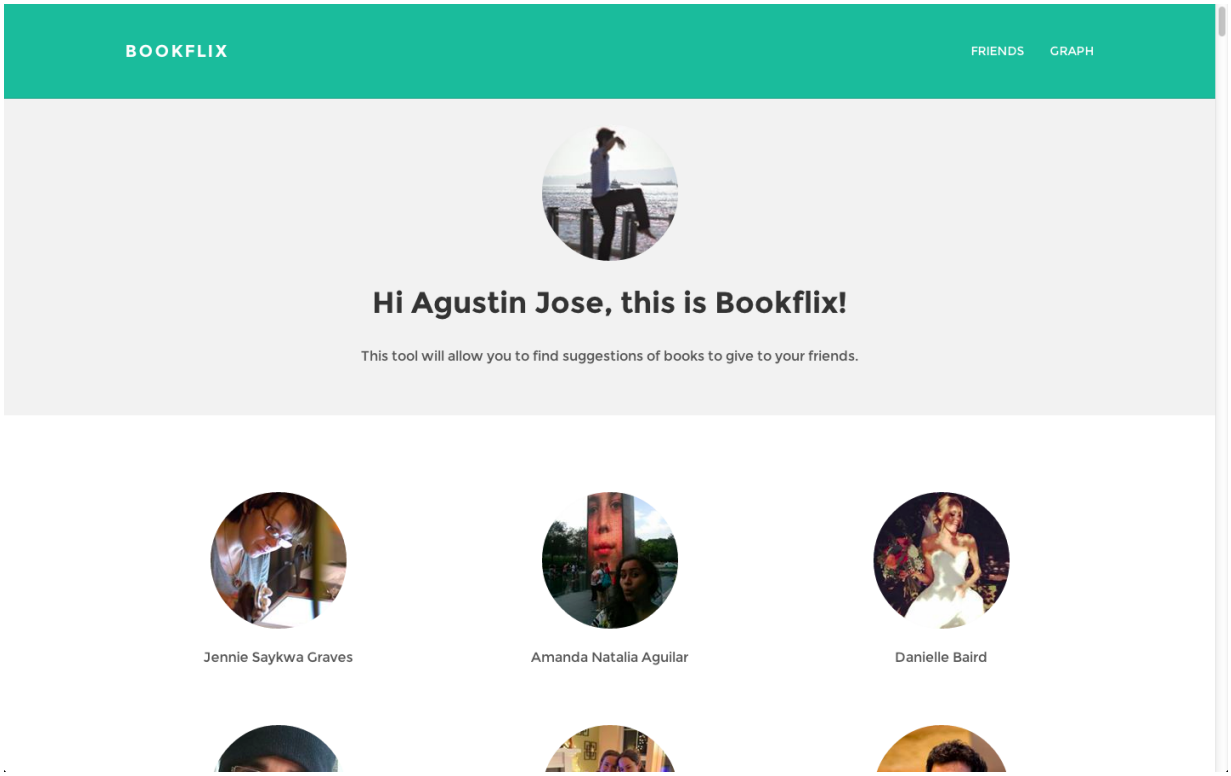


fig.3: Application Main screen: user can select the friend to get recommendations for.

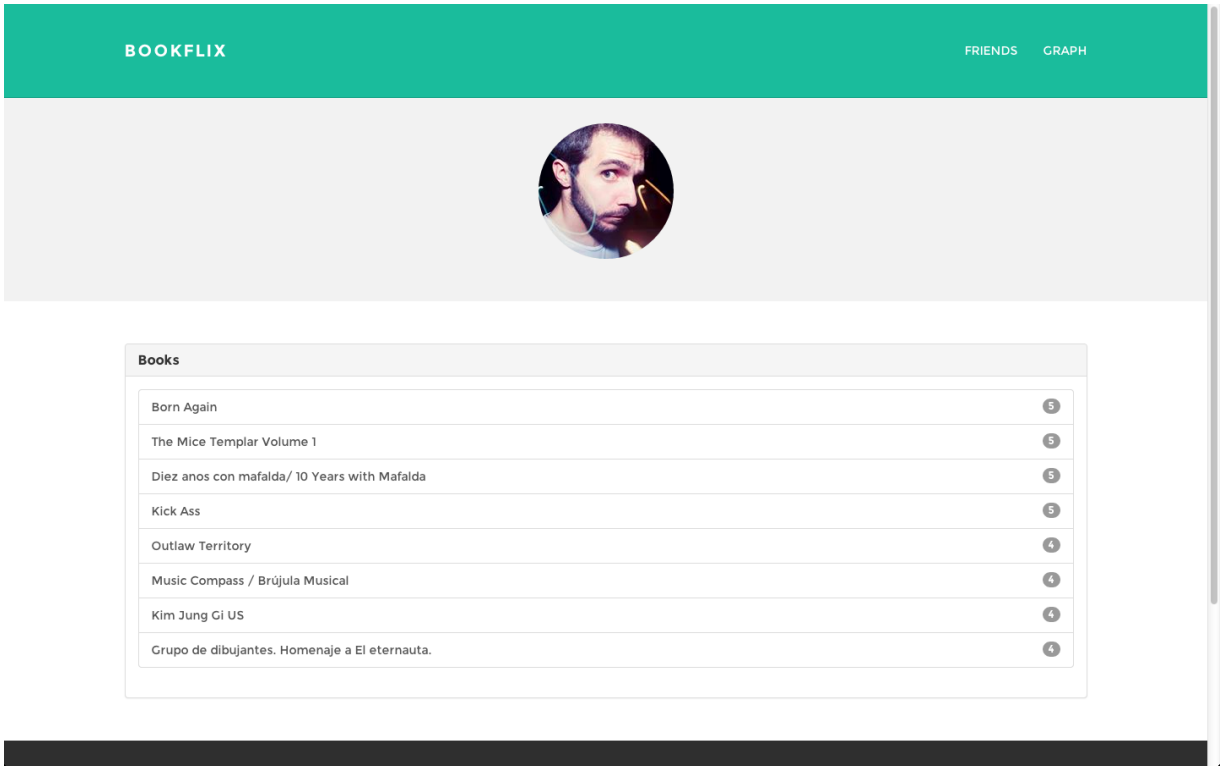
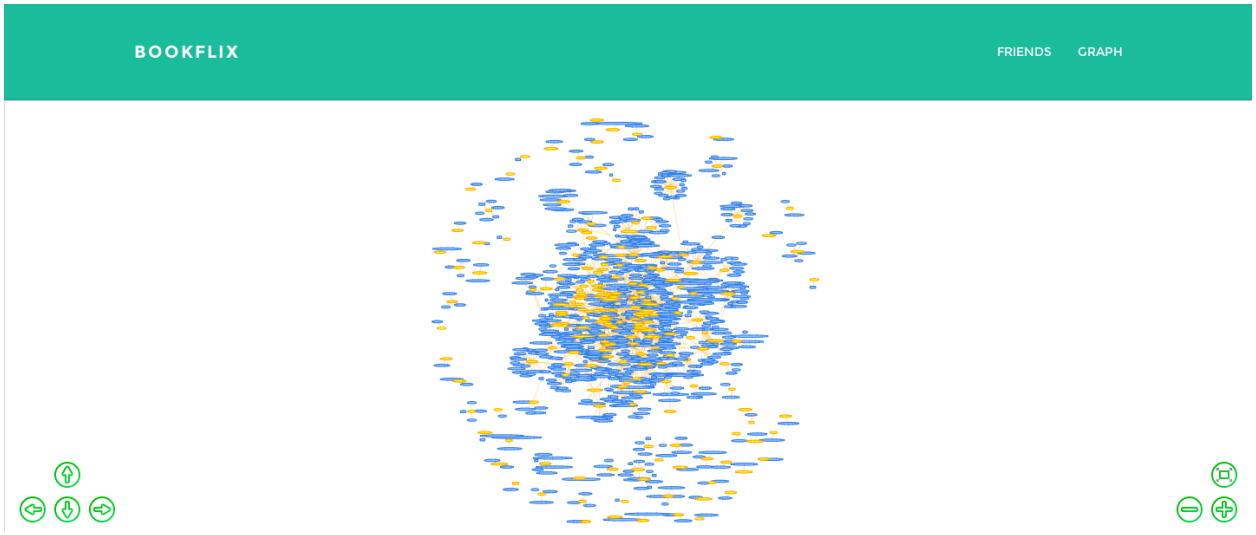


fig.4: Friend Recommendations screen: list of books suggested and ranked.



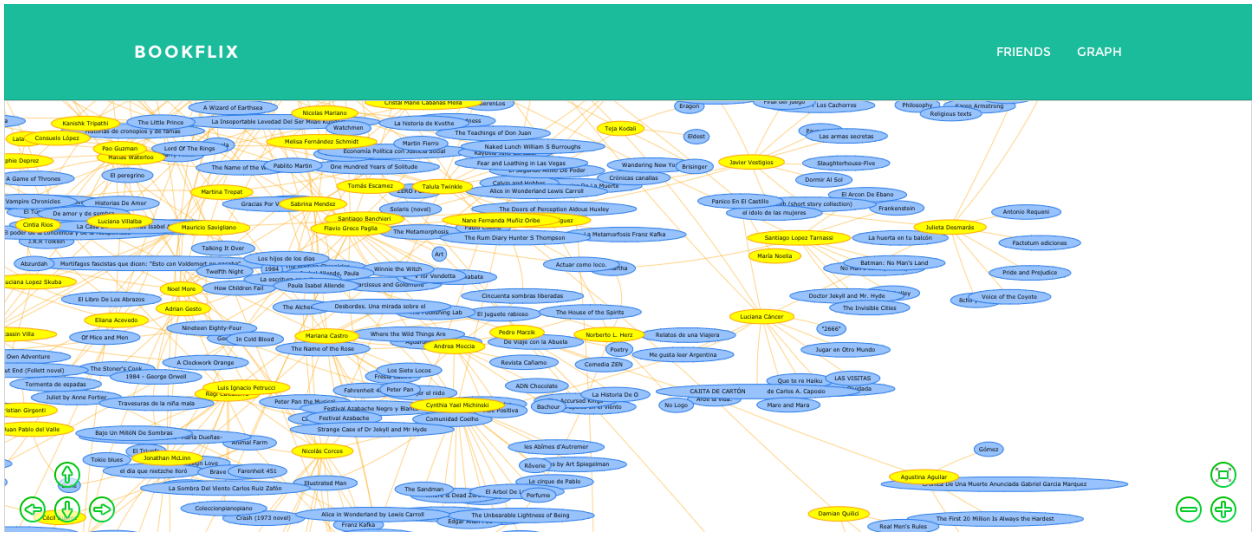


fig.5 and 6: Graph Visualization screen: user can click on any friend node in order to get suggestion for that person.

10. Next steps

Further development could be made in order to allow the user of the app to give more information or items that he/she knows his/her friends like, in order to enrich the data input for the collaborative filter and get more accurate suggestions.

There are some limitations from using Facebook Likes as the main input for the algorithm: the rating given to each item is binary (the user either liked a Page or not) and the absence of a like will not necessarily mean a negative rating, as it could also mean the user is not familiar with the item or maybe is, and is actually happy with it, but just did not take the time

to find the Facebook Page and like it. A future version of the app could also allow users to provide 1 to 5 stars to books and then enlarge the data source.

Also, the users could help as a human input for improving and correcting book and movie titles and also for updating Freebase database. There are many Pages that do not actually correspond to any book or movie which should be filtered out from the list and discarded. Problem is the application has no way to tell if unmatched titles correspond to this type or if they are rather valuable data that could be used to improve Freebase knowledge base. Casual users could become a sort of human filter by voluntarily marking in the application the invalid titles. As for the ones that do pass the filter, these could be harvested to automatically extend Freebase.