ITCS-6190/8190: Cloud Computing for Data Analysis – Project Report

RECOMMENDATION ENGINE

Lakshmanan Ramu Meenal, Prasanna Kumar Rajendran, Senthil Kumar Karthikeyan

Project Overview:

In present generation of Computing revolution, recommendation systems are integral part of any intelligent information systems. e.g. Search engines (Google, Bing, Yahoo), Netflix, Amazon, YouTube and so on, recommends the article or entities which might interest users. For a system to be intelligent, it needs have informative data about user and about the entities he/she was interested in. In this project, we have developed a Book recommendation engine (stand-alone) which is used to recommend books using the User profile and User rating details. The rating system is designed with two recommendation algorithms, 1. Collaborative based and 2. User demographic profile (User location and age).

Approach and Implementation:

Collaborative based Recommendations:

Collaborative based recommendation engine is basically build based on the collaboration of different user's contribution on a book. The parameter which is taken into consideration is the different user's rating.

- We have grouped all the books each user has rated, for all the users, and sorted them in descending order of the ratings. We have ignored the low rating book that the user has.
- Now, we get ordered pairs of interest for books each user has. So, we have dropped the user information from the pairing.
- Then, we calculated the similarity between every book with every other book which are rated by the same user.
- Finally, we combined the similarity score we calculated with for each book with other book we have calculated in the previous step.
- We have implemented the Collaborative based recommendation engine using Hadoop MapReduce and the programming language used was Java.

Recommendations filtering based on Demographic data:

- We grouped all the data in such way that we can perform clustering based on location or country the user belongs to. We used the unsupervised learning algorithm K-means clustering for clustering the data based on the Country. We can even include age or any other user information to cluster data. This shows the user's recommendations which falls close to his/her country cluster.
- We have implemented this using the Hadoop MapReduce, as this has got the iterative computation, in Java.

Motivation:

We tried to check how this recommendation engine works in the real-world scenario by giving one of our teammate's information as a new user profile detail in the dataset and also his ratings to the books which he has already read from the dataset. Interestingly, recommendation engine suggested some books which was aligned to his interest based on the rating details which he provided. This was very useful and interesting about our project.

Dataset:

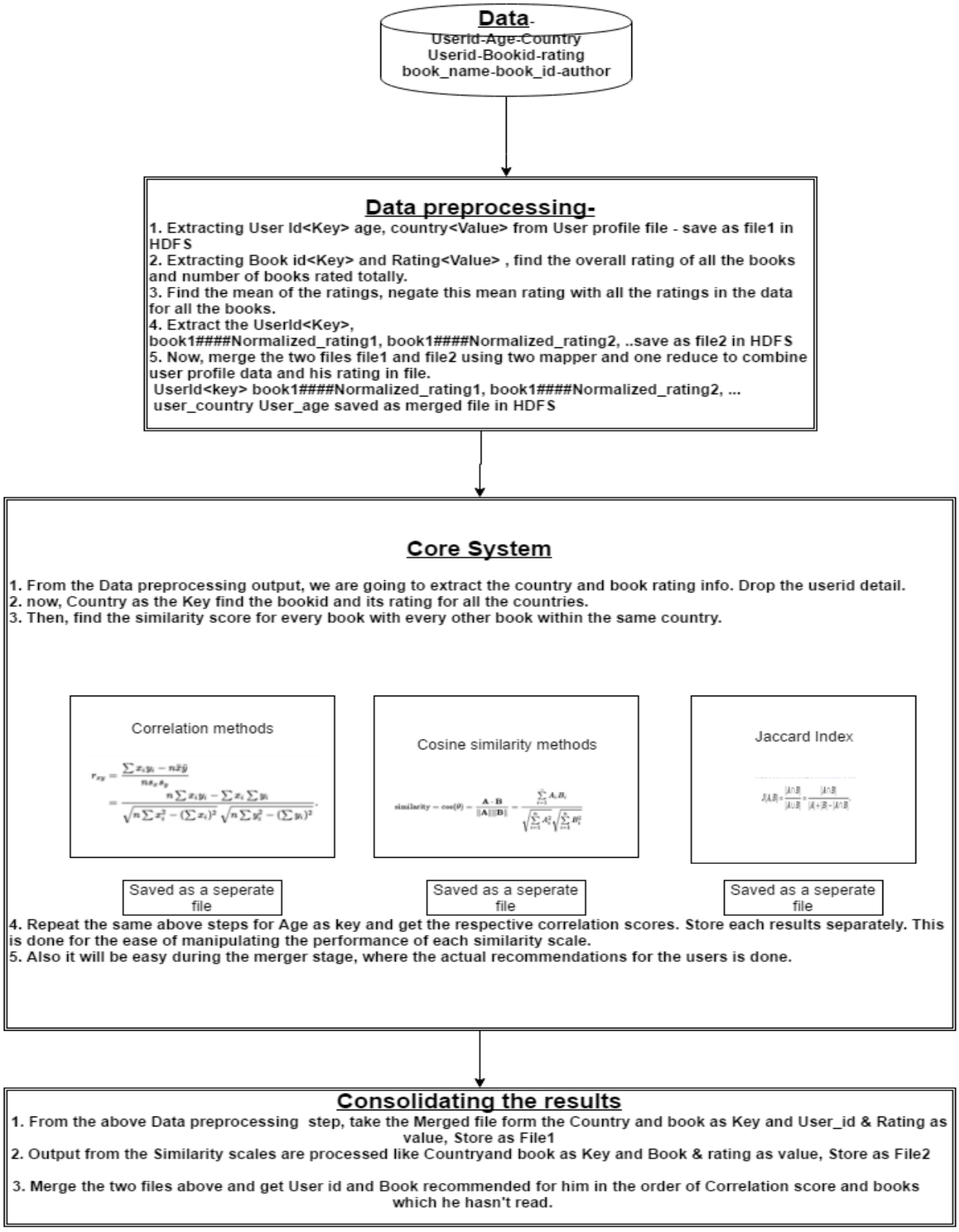
We are using the Book-Crossing Dataset which was mined by Cai-Nicolas Ziegler, DBIS Freiburg, from the Book-Crossing Community. This dataset contains 278,858 users (anonymized but with demographic information) providing 1,149,780 ratings (explicit / implicit) about 271,379 books. This dataset is freely available for research purpose from http://www2.informatik.uni-freiburg.de/~cziegler/BX/.

Data Description:

The Book-Crossing dataset comprises 3 tables in comma-separated values (CSV) files.

- **BX-Users**
  Contains the users. Note that user IDs ('User-ID') have been anonymized and map to integers. Demographic data is provided ('Location', 'Age') if available. Otherwise, these fields contain NULL-values.
- **BX-Books**
  Books are identified by their respective ISBN. Invalid ISBNs have already been removed from the dataset. Moreover, some content-based information is given ('Book-Title', 'Book-Author', 'Year-Of-Publication', 'Publisher'), obtained from Amazon Web Services. Note that in case of several authors, only the first is provided. URLs linking to cover images are also given, appearing in three different flavors ('Image-URL-S', 'Image-URL-M', 'Image-URL-L'), i.e., small, medium, large. These URLs point to the Amazon web site.
- **BX-Book-Ratings**
  Contains the book rating information. Ratings ('Book-Rating') are either explicit, expressed on a scale from 1-10 (higher values denoting higher appreciation), or implicit, expressed by 0.

Architecture:



Framework:

The entire Recommendation Engine in this project was built in Hadoop MapReduce framework using Java.

Environments:

For development, we used Cloudera.

For testing, we used UNCC Hadoop DSBA cluster.

For implementation and demo, we used Amazon AWS Cluster.

Accomplished:

- *Definitely will accomplish:* As we proposed, we have successfully implemented the Book Recommendation Engine using the Collaborative + Demographic recommendation model from the scratch.
- *Likely to accomplish:* As we mentioned, we also tried to implement the recommendation engine using content-based recommendation engine but to time constraints we were not able to design it completely.
- *Would ideally like to accomplish (in future):* A responsive UI for the system and showing the recommendation in a webpage and storing any new user data in the database for adding him to the existing dataset we have and include his data for further analysis and recommendation computation.

Roles and Responsibilities:

This project basically had Environment setup (Hadoop, SPRAK, DATA set up), Design, Coding, Testing, Documentation and setting up meetings on regular basis for project status update.

Specific Task assignments:

1. Lakshmanan Ramu Meenal
    a. Environment Setup
    b. Codingsrt
2. Prasanna Kumar Rajendran
    a. Setting up meetings on regular basis
    b. Design
    c. Coding
3. Senthil Kumar Karthikeyan

a. Documentation

b. Coding

References:

- Improving Recommendation Lists Through Topic Diversification,

  Cai-Nicolas Ziegler, Sean M. McNee, Joseph A. Konstan, Georg Lausen; Proceedings of the 14th International World Wide Web Conference (WWW '05), May 10-14, 2005, Chiba, Japan. http://www2.informatik.uni-freiburg.de/~cziegler/BX/ (*for dataset*).