Project Proposal
Big Data (DS-GA 1004)

Diogo Mesquita (dam740) Kenil Tanna (kyt237) Mihir Ujjwal Rana (mur214)

1 Project Title

Recommender System for Amazon.com Products.

2 Problem Description and Goal

Physical delivery systems are characterized by a scarcity of resources, thereby provisioning only for simple recommendations, e.g., displaying only the most popular books in a bookstore. On the other hand, online stores can make anything that exists available to the customer, e.g., Amazon offers millions of books. This distinction between the physical and online worlds is called the Long Tail Phenomenon, which forces online institutions to recommend items to individual users. It is not possible to present all available items to the user, the way physical institutions can.

The goal of this project will be to build a personalized recommender system for a subset of the total products offered on Amazon.com. We believe that this satisfies the project requirements and is an interesting industry-oriented problem to work on, which will be useful to us after our graduate studies.

3 Previous Works and References

The argument regarding the importance of the long tail in online systems and the various Collaborative Filtering methods are discussed in Chapter 9 [1]. The item-item similarity and design of collaborative filtering algorithms for Amazon product recommendations are discussed in [2]. There are three papers describing the three algorithms that, in combination, won the NetFlix challenge, which was an open competition for the best collaborative filtering algorithm to predict user ratings for films, based on previous ratings without any other information about the users or films. They are [3], [4], and [5].

4 Data Set

The data can be found here. The total size is nearly 3 GB and comprises 83.68 million rows (although due to computational constraints, we will only work with a subset of the data), which is apt to be run in a map-reduce framework. User-user (or item-item) collaborative filtering is a parallelizable algorithm which can be implemented on Hadoop/Spark since at its core, it consists of vector inner prodycts, matrix multiplication, factorization, etc. (all of which are parallelizable). Every row in the data is of the form *(user_id, product_id, product_rating)*.

5 Method and Algorithms

The input data will be divided into train and test sets, with a subset of products for each user in both the sets (to avoid cold-start problem). The following algorithms will then be used to fully populate the Utility (user-product) Matrix:

1. User-User Collaborative Filtering (CF)
2. Item-Item CF
3. Model-Based CF [Time Permitting]

Concretely, the rating for each product given by each user will be predicted on a scale of 1-5 or 0/1 (i.e., like/dislike). For measuring similarity, multiple metrics such as Jaccard Distance and Cosine Distance will be explored.

6 Evaluation Criteria

Traditionally, both Mean Absolute Error (MAE) and Root-Mean-Squared Error (RMSE) are used for evaluating the predictions of recommendations [6], and we will explore both metrics for our project.

References

[1] Jure Leskovec, Anand Rajaraman, and Jeffrey David Ullman. *Mining of Massive Datasets*. Cambridge University Press, New York, NY, USA, 2nd edition, 2014.

[2] Greg Linden, Brent Smith, and Jeremy York. Amazon.com recommendations: Item-to-item collaborative filtering. *IEEE Internet Computing*, 7(1):76–80, January 2003.

[3] Yehuda Koren. 1 the bellkor solution to the netflix grand prize, 2009.

[4] Martin Piotte and Martin Chabbert. The pragmatic theory solution to the netflix grand prize, in: Netflix prize documentation, 2009.

[5] Andreas Töscher, Michael Jahrer, and Robert M. Bell. The bigchaos solution to the netflix grand prize, 2009.

[6] Claude Sammut and Geoffrey I. Webb. *Encyclopedia of Machine Learning*. Springer Publishing Company, Incorporated, 1st edition, 2011.