CS 435/CS 535
Spring 2018
Semester Long Project
Assigned on 17 January 2018
Coding Component Due on 3 May 2018
Final Due on 10 May 2018

Total Points: 100

This is a semester-long project, encouraging students to dive into the research in data mining. Specifically, this project concerns with the research on developing a recommender system. The project has three phases. Graduate students must complete all the three phases while undergraduate students are required to complete the first and the third phases.

The first phase begins at the beginning of the semester when you have received the assignment. You are asked to conduct the research on recommender systems independently by yourself.

Recommender systems have been studied extensively in the literature and have also been found popularly useful in many real-world applications. In general, a recommender system is considered in the scenario where we have m different users and n different items in a specific application (e.g., in an E-commerce application we have n different customers potentially intending to buy m different commercial items) where a user i may give an item j a rating value k based on this user's preference ($i \in [1,\dots, m]$; $j \in [1,\dots,n]$; $k \in [1,\dots,K]$) where a higher rating value indicates that this user likes this item better. Essentially, the whole user preference rating data can be represented as a matrix of n by m where each element of this matrix is the value k if the user in question gave a rating for the item in question, and 0 if no such rating was given yet. Initially, all the elements of this matrix are 0. After we have collected a certain amount of such user preference rating data, this matrix becomes partially filled out with different non-zero entry values. The goal of a recommender system is that, after we have collected a certain amount of the user preference rating data, the system is able to predict the rating value k of user i for item j if user i has not yet given such a rating. In other words, given such a partially filled out matrix, a recommender system shall be able to fill out all the predicted rating values for those elements with the current value of 0. That is, a recommender system mathematically is a solution to a matrix value completion problem.

In this phase, you study the problem of recommender systems in the literature and either develop a new recommender system by yourself or identify an existing, working recommender system. Then you implement and evaluate this implemented

system till you are happy with the system. You may use whatever language you are comfortable with to implement the recommender system. Now you are ready to test and report how good your system is. You are given a partially filled out rating data matrix with this assignment. The matrix is given in an ASCII text file where each row is a non-zero element of this matrix with three entities: the user ID number, the item ID number, and the non-zero rating value. The rating value is in the range of [1, 5] in integer where 1 means that the user likes the item the least and 5 means that the user likes the item the best. The file name is train_all_txt.txt with 943 users (m = 943) and 1682 items (n = 1682). You may use part of this given training dataset to evaluate your recommender system. When you report your testing result, you must make sure that your code generates an ASCII text file for the whole rating matrix with each element filled out (either as given or as the predicted rating value by your code). The output text file must follow the format of one element of the matrix as one row beginning with the element for the first user for the first item, followed by the element for first user for the second item, …, the element for the first user for the last item, the element for the second user for the first item, …, the element for the second user for the last item, …, the element for the last user for the first item, …, and the element for the last user for the last item; each row of the file has three numbers: user ID number, item ID number, and the rating value, separated by a space. While the given rating values are all in integers, your predicted rating values can be either in real numbers or rounded into integers, up to you to decide. Your final grade for the coding component in this phase is determined by how much total error for the whole matrix between what your system has predicted and the ground truth rating matrix. **Note that we use a script to grade your turned-in file and any violation of this format requirement may result a gross error that leads to a very low grade for you. By noon May 3, you must submit in myCourses a zipped package containing the source code of your implementation of the recommender system with appropriate comments and documentations in the code, a README file to explain how to compile and run your code under what specific environment, and most importantly a text file containing the output matrix following exactly the format requirement stated above.**

The second phase consists of preparing for and giving a presentation about your research on recommender systems. In this phase, you prepare for a 10-minute presentation on your understanding of the literature on recommender systems and the specific system you have implemented. We will announce the presentation schedule later in the semester. Note that while only graduate students are required to complete this phase of the work, all the students including undergraduate students are required to attend the classes while the graduate students are presenting their research.

Specifically, in the presentation, you must answer the following questions based on your research:

- Other than recommender systems, give and explain briefly two more exampes of the real-world application problems with a solution of matrix completion.
- Explain in detail the specific system you have implemented (i.e., the algorithm)

- Report your evaluation results of your system with the data your are given in train_all_txt.txt (i.e., how much of the given data you have used for training and how much for testing as well as the final testing results using what metrics)

In the third phase, you submit an up to one page mini-paper describing the algorithm you have developed/implemented for the recommender system you built for all the students and your presentation material (e.g., your powerpoint file) for graduate students only into myCourses by noon of May 10.

A final note: you must complete this project independently and no collaboration is allowed.