Trust Me, I Know You: Using Clustering to Predict Renter Film Selections in Netflix

Zeina Atrash
Northwestern University
zeina@u.northwestern.edu

Eugenia Gabrielova
Northwestern University
genia@u.northwestern.edu

Lalith Polepeddi
Northwestern University
l-polepeddi@northwestern.edu

ABSTRACT

Prediction systems are becoming increasingly popular as services move online. Netflix launched a competition to help with the accuracy of these suggestions for their online film rentals. In this study, we use spectral clustering in combination with weighted k-nearest neighbor to improve upon suggestion of films. Using data from the Netflix costumer rental history, we make predictions of rental interest by clustering user taste by previous rental ratings.

1. CONCEPT

With prediction systems all around us, we as consumers have become more dependent on these systems to help us learn more about our options. Such systems are valuable in helping consumers take a vast quantity of data and focus on what is most relevant to them. Consumers often log in to virtual systems like Netflix, an online movie rental service, and seen that their personalized recommendations are sometimes enlightening, but more often than not puzzling predictions of our taste.

There are many reasons that rental taste appears dynamic: experimentation, renting for others, providing access for multiple users, accidental renting, and evolution of taste. We propose to investigate an improvement to rental classification that clusters user preference based on similarities among other subscribers. In 2006, Netflix launched a competition to improve its rental suggestion application, Cinematch, using subscriber rental data. The goal of the competition was to produce a ten-percent improvement over the root mean squared error (RMSE) of Cinematch.

2. DATA RESOURCES

2.1 Dataset to Test and Train System

Netflix provided a dataset of customer rental information for its grand prize competitors. The dataset uses over 480 thousand subscribers and 18 thousand movies for a total of over 100 million customer ratings, further details of the dataset composition can be found in the competition description [2]. Given the robust size of the dataset, the entries have been limited further in consideration of processing time and resources for this research, as shown in Figure 1. The dataset used for study, hereafter referred to as the *dense matrix*, has limited users to approximately 23 thousand user-movie rating entries constructed from users with the most number of ratings, the most frequently rated movies, and ratings from the last 5 years of the dataset (2003 - 2007). These ratings are on an integer scale of 1 to 5 stars, where 1 is the lowest rating and 5 is the highest positive rating.

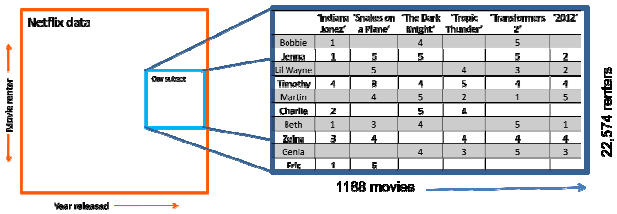

Figure 1. Dense Matrix Dataset | A subset of data was taken from the original Netflix dataset for complexity considerations.

2.2. Existing Software Packages

Python's SciPy and NumPy libraries were used to calculate eigenvalues and corresponding eigenvectors. Microsoft Excel was utilized to implement k-nearest neighbor as well as the resulting RSME computations.

3. APPROACH

3.1 The Learner

In order to create a measurement of distance between users we generate two matrices. First, the *dense matrix* is generated using standard SQL processing of the raw data. This matrix consists of the movies ratings from the most active raters in the system on the most commonly rated movies. This matrix is generated in order to compensate for the sparse original data; with little processing power and time for this pilot study it is important to test the algorithm on a more thorough sample.

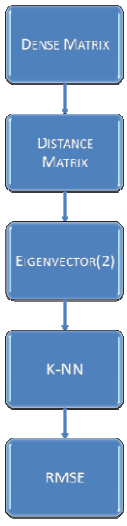

Figure 2. Algorithm Data Flow Chart | The pipeline followed to generate movie recommendation measurements from raw user rating data.

Second, the dense matrix is used to compute distance between all users and generate the user-by-user *distance matrix*. This matrix consists of a measurement of user similarity based on the Euclidean distance of their movie ratings.

Eigenvalues of this matrix are calculated and the eigenvector corresponding to the second largest eigenvalue is retrieved for spectral clustering and k-nearest neighbor processing. The values are partitioned on positive/negative basis for clustering. We find the k-nearest neighbors for a target user in order to suggest a movie using rating data

from other users within that cluster. The weight of each neighbor's ratings is given distance computed on the eigenvalue plot. The weighted ratings for every rental seen by the neighboring subscribers are combined per movie, and the movies with the five highest scores are suggested to the target user.

3.2 Analysis

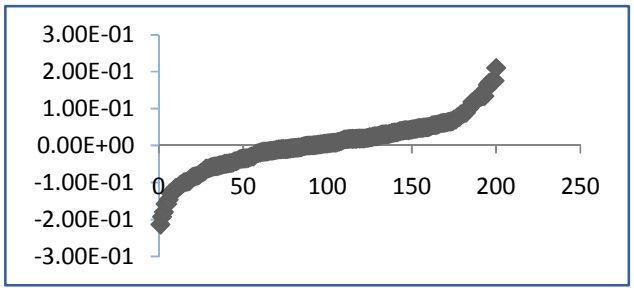

Figure 3. Spectral Clustering of Distance Matrix Data | One of ten generated spectral cluster from the dense data subset.

We ran five experiments at ten trials a piece to generate a final overall RMSE of 1.86. Though this does not achieve the ten percent improvement, many trials should improvement to an RMSE as low as .71. The density of ratings in the matrices played an important role in this value. As more ratings are included, the RMSE lowers; those matrices with fewer ratings result in higher RMSE scores. Therefore, creating denser matrices does help produce more accurate results and suggestions, but also detracts from the reality of the problem at hand. Many users do not rate rentals, or have not seen some selection of movie; these scores, or lack thereof, do influence the suggestion schemes, but must be taken into consideration when developing this type of algorithm.

3.3 Discussion

Netflix uses root mean square error (RMSE) to measure the amount by which a prediction misses the actual score. The prize-winning team was able to lower the RMSE to 0.8567, which means that predictions are off by less than a point from the user's actual ratings. Using RMSE as a basis for our performance evaluation, our system achieves an average RMSE of 1.86. Although these results do not meet the ten percent improvement requested of the competition, we believe that the results do encourage further work to incorporate other ideas.

Though this value is not lower than the winning RMSE, the processing time and low complexity demonstrates considerable advantages towards an overall sturdier system. With proper time and resources, this algorithm can be improved with greater detail, yet maintain its simplicity in processing. We plan to incorporate additional movie description data from The Internet Movie Database, such as plot, characters, or high level view of the movie to generate similarity ratings and classify movie taste more specifically. Including additional data to ratings will help to overcome the effects of data scarcity while maintaining the true nature of unrated data.

4. RELATED PAPERS

[1] Bell, R., Koren, Y., & Volinksy, C. (2008). The BellKor 2008 Solution to the Netflix Prize. http://www2.research.att.com/~volinsky/netflix/Bellkor2008.pdf

[2] Bennett, J., Lanning, S., (2007). The Netflix Prize Netflix Netflix

[3] Netflix Leaderboard. Retrieved December 3, 2009. http://www.netflixprize.com/leaderboard.

[4] Netflix Testing Corpus. Retrieved November 5, 2009. http://archive.ics.uci.edu/ml/datasets/Netflix+Prize

[5] Netflix Training Corpus. Retrieved November 5, 2009. http://archive.ics.uci.edu/ml/datasets/Netflix+Prize

[6] K. Burchett, "Spectral clustering rocks," (2006). Retrieved December 4, 2009. http://www.kimbly.com/blog/000489.html.