Comparison of Four Recommender Systems on Goodreads Dataset

Jay Shi, William Su

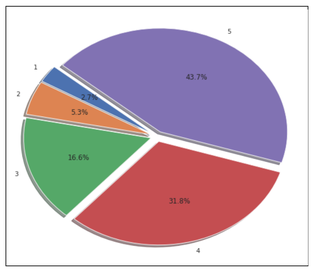
Introduction

- *Goodreads* is an online platform where its users can :
  - 1. “shelf” books that they are currently reading
  - 2. markdown books that they had read
  - 3. rate these books from a scale of 1 to 5.
- *Goodreads* incentives users to have these specific interactions with the books on the platform by recommending the users back books that users may possibly want to read next depending on the books that they have interacted with on the platform.
- In this project, we intend to imitate this platform and build four different recommender systems based on a dataset that provides anonymous users’ specific interactions with any number of books.
- The four models we will be building are:
  - 1. User-User Similarity
  - 2. Weighted User-User Similarity
  - 3. Matrix Factorization With ALS
  - 4. Neural Network Based Matrix Factorization

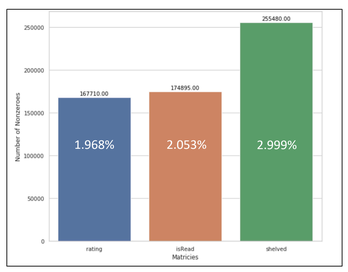
Dataset

- We obtained a user-book interaction dataset from The University of California, San Diego. We extracted a shelf matrix, a isRead matrix and a rating matrix, all of size 20000 by 426, from the dataset.
- We randomly split 10% of nonzero entries in the rating matrix as testing data and the rest 90% is our training data.

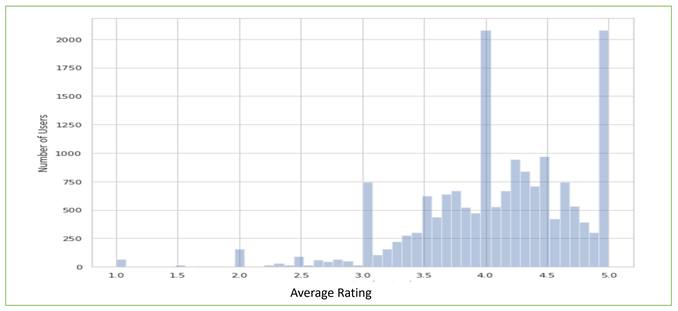
Distribution of Ratings



Number of Nonzero Entries and Sparsity



Distribution of Average Ratings Given by Users



Results

1. User-User Similarity

- Similarity is calculated by Pearson Correlation. Testing MSE comes to 0.9756.

2. Weighted User-User Similarity

- We wanted to utilize the fact that the dataset not only provided rating information but also information about whether a user shelved a book or had read a book. We tried two approaches to give weights to the shelf matrix and isRead matrix. Denote shelf matrix by *S,* isRead matrix by *O*, and rating matrix by *R*. Denote weights assigned to *S, O* by *w1, w2*.

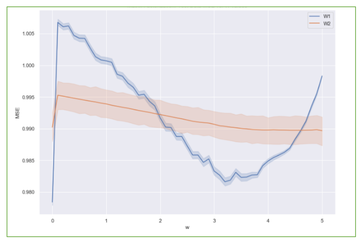
Approach 1: Compute Similarity From Weighted Sum of Three Matrices

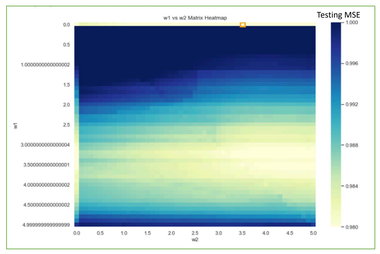
- Let matrix *M = w1 * S + w2 * O + R*.
- Pairwise user-user similarity is computed from *M*. This model turns out to be very inaccurate, with a testing MSE of over 1.5.

Approach 2: Incorporating Shelf and IsRead Information into Nonzero Entries

- We computed *M* by the following.
- For each entry *R[i, j]* of *R*,
  - If *R[i, j] = 0,* *M[i, j] = w1 * S[i, j] + w2 * O[i, j] + R[i, j];*
  - *Else,* *M[I, j] = R[i, j].*
- We swept the parameter space of *w1* from 0 to 5 and *w2* from 0 to 5 with increment of 0.1. The best combination we found was *w1 = 0* and *w2 = 3.6*, which resulted in a testing MSE of 0.9764, still slightly worse than without weights.

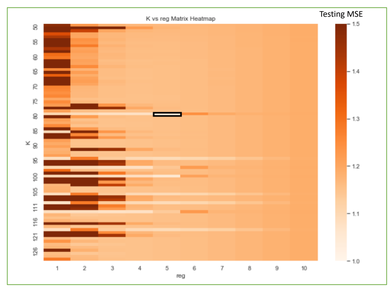
Testing MSE vs. Weights

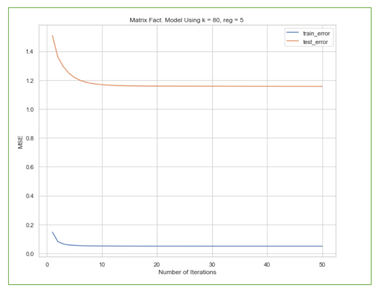




3. Matrix Factorization With Alternating Least Squares

- We swept the parameter space of *k*, the dimension of the latent space, from 50 to 130 and *reg*, the value of the regularizer, from 1 to 10.
- The top five parameter combinations *(k, reg)* are (80, 5), (101, 5), (114, 3), (114, 4), (111, 1), and (101, 3), with testing MSE ranging from 1.032 to 1.078, surprisingly not better than user-user similarity model.

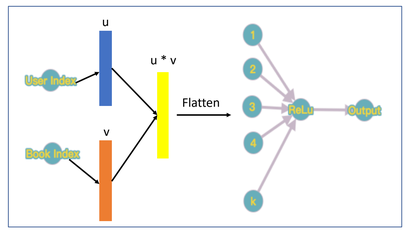


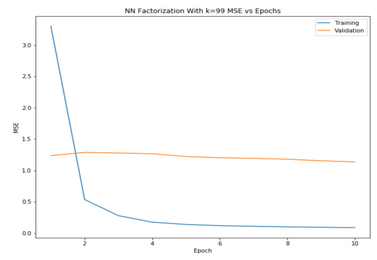


4. Neural Network Based Matrix Factorization

The neural net takes as inputs a user index and a book index. It embeds the user and the book to vectors *u, v in R^k*. Then it multiplies *u, v* element wise and flattens the result. A dense layer of one neuron with activation ReLu comes after that. The ouput is one neuron following the dense layer. With *k=99*, testing MSE comes down to 0.768.

Neural Network Architecture

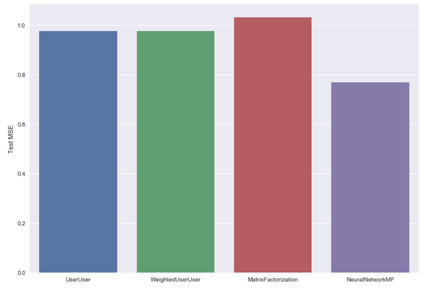




Conclusions

- Out of the four models, the Neural Network Based Matrix Factorization Model has significantly better testing MSE than the others.
- Incorporating shelf matrix and isRead matrix into rating matrix requires more sophisticated approaches.
- It may be more efficient to sweep for optimal *k* using user-user similarity model and use that *k* for training the NN-MF model.

Comparison of Testing MSE of Models



Future Directions

- We would like to investigate more ways of incorporating shelf matrix and isRead matirx into rating matrix. One idea is to represent each entry of the rating matrix as a 3-tuple. Then, user-user similarity can be computed by computing the similarity between two lists of 3-tuples, or two matrices.
- There are more neural network based models for recommender system that we would like to explore.
- We should employ cross-validation when testing the models
- We would like to see how these models perform on other datasets.

References

1. https://sites.google.com/eng.ucsd.edu/ucsdbookgraph/home
2. https://cseweb.ucsd.edu/~m5wan/paper/recsys18_mwan.pdf
3. https://medium.com/recombee-blog/machine-learning-for-recommender-systems-part-1-algorithms-evaluation-and-cold-start-6f696683d0ed
4. Professor Devika Subramanian’s lectures and labs.