Movie Recommender System

Network Tour of Data Science

Mladen Korunoski, Blagoj Mitrevski, Ljupche Milosheski, Alen Carin

December 2019

Project proposal

Utilizing the notions of graph frequency and graph filters in order to enhance rating prediction is a relatively new idea in recommender systems [1]. Recent works have also shown that we can employ different deep learning techniques for link prediction on graphs to boost performance [2, 3].

In this project we are interested in combining both approaches and develop recommender system enhanced using the recent advances in graph signal processing. We plan to utilize the Movielens 100k dataset and possibly enrich the data with various movie information downloaded from OMDb (The Open Movie Database) and additional sources.

Our main goal is to construct two graphs, one using the movies description data, G_m, and the other one using the users description data, G_u, enriched as described above. We plan to employ different similarity measurements and define custom ones in order to capture the similarities between entities. Possible similarity measurements for the graph encoding movie data are: genre, actors, director, etc. On the other hand, defining similarity measurement for users data will be more difficult. Fortunately, we can use the available user information, such as: age, profession, etc.

After constructing the graphs, we plan to apply matrix factorization on the original data matrix. More formally, we define $\mathbf{X}$ to be a $D \times N$ sparse matrix, where: $D =$ number of movies and $N =$ number of users, and:

$$\mathbf{X}_{i,j} = \begin{cases} rating & \text{if user j gave rating for movie i,} \\ 0 & \text{otherwise.} \end{cases}$$

Then, the matrix factorization problem is defined as follows: $\mathbf{X} \approx \mathbf{W}\mathbf{Z}^\top$, where we aim to find $\mathbf{W}$ and $\mathbf{Z}$ with dimensions $D \times K$ and $N \times K$ respectively, with K being the latent space dimension.

For our baseline model, we plan to use the $\mathbf{W}$ and $\mathbf{Z}$ factors, to predict the rating that user j would give for a movie i as a simple dot product between the row $\mathbf{W}_{i,:}$, representing the features for the i-th movie with the column $\mathbf{Z}^\top_{:,j}$ representing the features for the j-th user, thus the predicted rating will be $\mathbf{W}_{i,:} \cdot \mathbf{Z}^\top_{:,j}$.

To incorporate the information from the previously mentioned graphs, we plan to perform a graph Fourier transform on the matrix $\mathbf{W}$ using G_m, and on $\mathbf{Z}$ using G_u. Doing this transformation we hope to obtain denoised versions of the two matrices and we define them as $\mathbf{W}^*$ and $\mathbf{Z}^*$. Then, we can use the denoised matrices to:

- predict the rating that user j would give for a movie i as a simple dot product between the row ${\mathbf{W}^*}_{i,:}$, representing the features for the i-th movie with the column ${\mathbf{Z}^*}^\top_{:,j}$ representing the features for the j-th user, thus the predicted rating will be ${\mathbf{W}^*}_{i,:} \cdot {\mathbf{Z}^*}^\top_{:,j}$, similar to our baseline model, and
- use the denoised matrices $\mathbf{W}^*$ and $\mathbf{Z}^*$ to train a neural network to predict the rating a user would give for a movie.

With this approach, we can only improve the predictions of our baseline model since we can always use a constant filter with value 1 to keep our matrices $\mathbf{W}$ and $\mathbf{Z}$ unchanged. At the end, we can use these rating predictions to build several recommender systems and compare their performance.

References

[1] Weiyu Huang, Antonio G Marques, and Alejandro Ribeiro. "Matrix completion via graph signal processing". In: *43rd IEEE Int. Conf. Acoust., Speech and Signal Process., Calgary, AB*. 2018, pp. 15–20.

[2] Federico Monti, Michael Bronstein, and Xavier Bresson. "Geometric matrix completion with recurrent multi-graph neural networks". In: *Advances in Neural Information Processing Systems*. 2017, pp. 3697–3707.

[3] Rianne van den Berg, Thomas N Kipf, and Max Welling. "Graph convolutional matrix completion". In: *arXiv preprint arXiv:1706.02263* (2017).