# Recommendation System using Geometric Matrix Completion with Recurrent Multi-Graph Neural Networks

REPORT

*RESEARCH PRACTICE*

*BITS G540*

*By*

SHIVAM KUMAR

ID No. 2020H1030060G

*Under the supervision of:*

Dr. BHARAT M. DESHPANDE

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE PILANI, GOA CAMPUS

November 2023

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE PILANI, GOA CAMPUS

# *Abstract*

Masters of Engineering (C.S.I.S)

**Recommendation System using Geometric Matrix Completion with Recurrent Multi-Graph Neural Networks**

by SHIVAM KUMAR

Despite being invented in 1992, **Recommendation Systems** have become one of the most important components of today's information society for dealing with information overload, notably in e-commerce. Recent research has shown that incorporating pairwise relationships between users/items in the form of graphs improves the performance of these strategies.I have introduced **geometric structure for matrix completion problems** e.g. through column and row graphs, showing user and item similarity correspondingly.I have applied **geometrical deep learning approach** to apply classic CNN to graphs and manifolds.To learn significant statistical graph-structured patterns and the non-linear diffusion process that yields the known rating, I used matrix completion architecture that blends graph convolution neural networks with recurrent neural networks. A consistent number of parameters is needed for this neural network system regardless of the matrix size. Finally, the matrix serves as a recommendation mechanism, directing users to the item with **highest ratings**.

# Contents

## 0.1 Introduction

### 0.1.1 Recommendation Systems

From **Tinder to Netflix**, recommendation systems are ubiquitous.
**So, what exactly are these recommendation engines? And why are they so mighty?**
The excess of information in our digital culture has become one of the main problems nowadays. Modern communications technology and the Internet in particular have made access to information simpler than ever. But it is frequently difficult for users to choose from this stream the piece of information which is most beneficial in a specific scenario due to the continual stream of information. Recommendation systems assist to manage the overflow of information. A **Recommendation System** is a type of content filtering system that attempts to forecast how a user would rate or favour an item.The recommendation system employs data analytics to locate things that fit the tastes and interests of the user.

Collaborative and content filtering strategies are two key approaches to recommender systems. **Collaborative Filtering Systems** take consumer evaluations of items and detect similar rating patterns to generate new suggestions.
To promote new products, **Content Filtering Systems** analyse commonalities between items and customers.
Collaborative and content strategies are combined in **Hybrid Systems**.

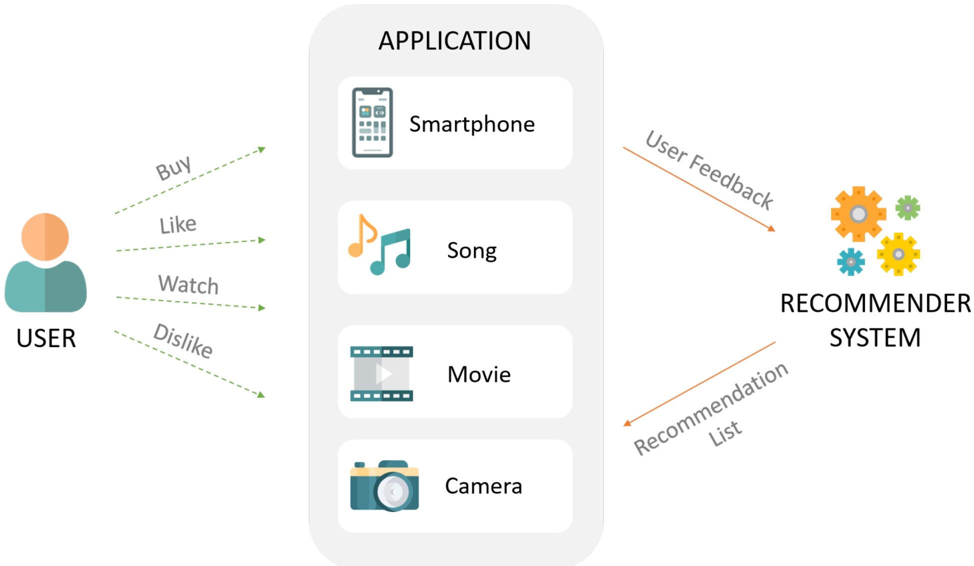


FIGURE 1: Recommendation System

### 0.1.2 Geometric Matrix Completion

A recommendation technique may be modelled mathematically as a **Matrix Completion Problem**, with columns and rows representing users and items, respectively, and matrix values representing a rating deciding whether a user would like an item or not. The objective is to fill in the gaps in the matrix given a limited selection of known elements.

Current findings have demonstrated the performance of such strategies by adding relationships in the form of graphs among users and items. The missing values of a certain matrix can be recovered if a tiny proportion of its entries are supposed to be found within a smaller subspace, that is, if we consider a **matrix of a low rank**.


FIGURE 2: Rating Matrix

### 0.1.3 Deep Learning Approach

#### 0.1.3.1 Convolutional Neural Network (CNN)

Deep Learning has proven to be a particularly useful technique in recent times due to its capacity to manage massive volumes of data. **Convolutional Neural Networks(CNN)** are amongst the most often used deep neural networks. Therefore, we tend to apply CNN in our recommendation systems. However,we can't immediately implement the traditional CNN herein, because it isn't Eucledian structured rather lies on domains of graphs and manifolds.Thereby tried another approach through geometric deep learning that can certainly handle graph-structured data which arises most frequently.

#### 0.1.3.2 MultiGraph CNN(MG-CNN) and RNN

Multiple-Graphs CNN not only represents high-order information using a bipartite user-item interaction network, but also it incorporates proximal data by constructing and analysing user-user and item-item graphs.

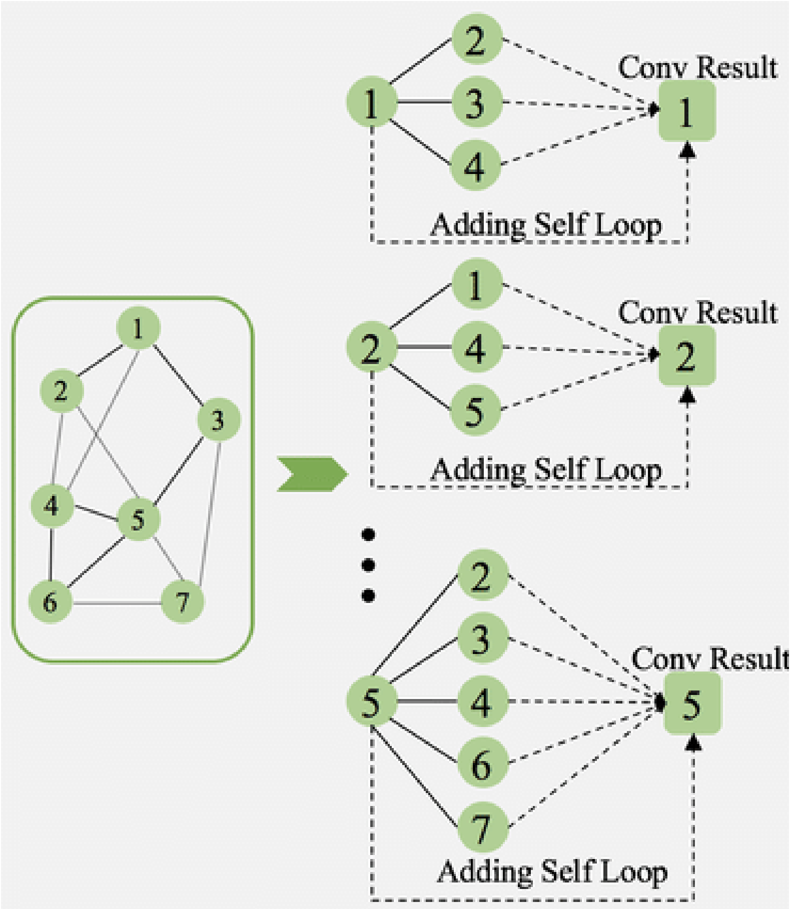


FIGURE 3: Multi-Graph Convolution

The features are extracted from the MGCNN and are fed into the RNN for implementation of score diffusion process.We utilised the standard LSTM RNN architecture for this purpose for its outstanding performance. Herein, exposing model to Multigraph-CNN results in a **m x n x q** output where m is the number of users n is the number of items and q depicts the q-dimensional feature vector. As stated above that the outputs of MGCNN acts as input to RNN which results in high efficacy of dynamical property of data sequences.**LSTM(RNN)** is able to maintain a long term internal state which allows to predict accurate small changes.

## 0.2 Objective

- To build a **Recommendation System using Geometric Matrix Completion with Recurrent Multi-Graph Neural Networks**[10].
- Provide recommendations based on user's needs and preferences information.
- Recovering the missing values of a matrix given a small fraction of its entries.
- To **reduce** computational complexity.
- To deal with high-dimensional data.
- Present and apply the unique **Neural Network Architecture** to draw local stationary patterns from the user's and item's high-dimensional spaces to deduce the nonlinear temporal diffusion process of the ratings.
- Extension of normal CNN to graphs through **Geometric Deep Learning**.

## 0.3 Literature Survey

Building recommender systems may be divided into two categories: **collaborative filtering**[16][5][8] and **content-based filtering**[1][12]. Collaborative filtering suggests products based on the preferences of users with comparable ratings. Content-based filtering, on the other hand, recommends products that are comparable to other items in which the consumer has demonstrated interest.

The current advancement in deep learning approaches[7] has enhanced the latest technology in a number of machine learning applications. Classical techniques to deep learning with a Euclidean or grid-like structure with an integrated notion of invariance are most successful. However, real-world data often features a **non-Euclidean** structure, including graphs networks.

For instance, the rating matrix in the recommendation system may be seen as data connected with a network of similarity between an underlying user-user or item-item. It is not easy to generally use standard deep learning approaches to deal with such data, especially when network and graph convergences are lacking in well-defined processes.

General matrix completion methods cause high computational complexity and ambiguous result . These type of models recommend highly inaccurate result to the user . **General matrix completion models**[3] fail when there is a large data to deal with it , Due to its time complexity of O(mn) it's very hard to implement this into the real world. so, we have to come up with such a method whose computational complexity should be low as which give more accurate recommendation for the user .

**Geometric deep learning**[2] is a branch of developing deep learning techniques that leverage new concepts and ideas created by **graph signal processing**[14], a fast-growing discipline, to generalise conventional deep learning approaches to networks and graphs-related data .

Early efforts to apply neural networks to graphs are related to the formulation of[9] CNN-like deep neural architectures on spectral graphs, using the analogy of traditional Fourier transformations and projections on the graphic operator Laplacian[15].

In a follow-up study, [4] proposed an effective filtering system employing recurrent **Chebyshev polynomials** presented that decreases graphic CNNs' complexity to the same complexity as regular CNNs .This approach was further developed to cover dynamic data[13]. [6] suggested to simplify the Chebyshev networks by simply filtering the 1-hop graph neighbours. [11] has developed a CNN generalisation of the spatial domain to local patch operators as Gaussian mixed models, which has a substantial benefit in generalising these models via various graphs.

## 0.4 Approach and Working

Missing matrix values are handled with the hypothesis of a smaller subspace variable, i.e. a low-grade matrix. Introducing geometric structure i.e the application of laplacian and fourier transformations upgrades the normal matrix completion to geometric completion problems.

The **geometric completion matrix** approach boils down the complexity to **O(m x n )** comprising of n users and m items matrix.

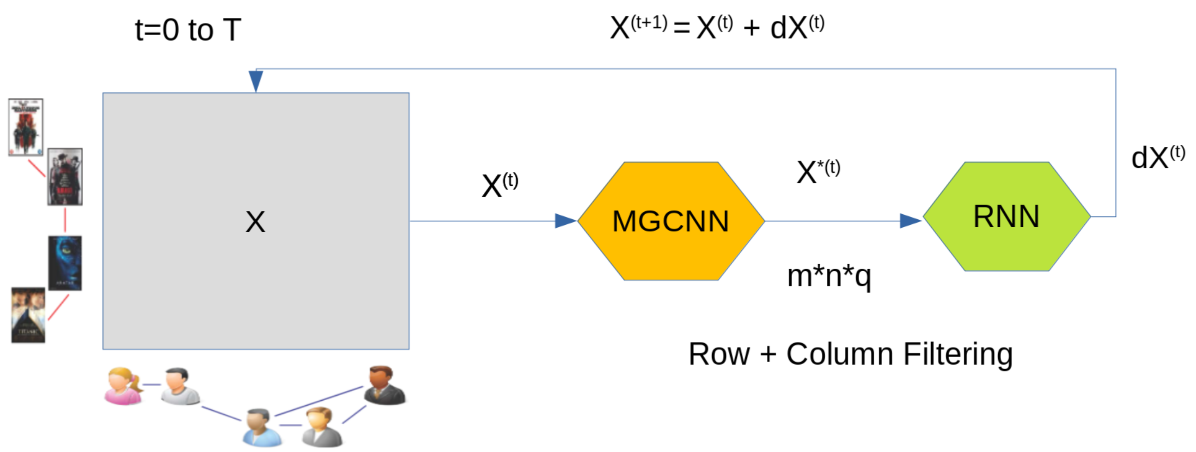


FIGURE 4: Full matrix completion model using RGCNN

These Geometric Matrix completion problems often deal with solutions that are distinctive and sturdy. However,challenges with these approaches are in formulating variables as they comprise a full matrix, making it difficult to measure these procedures up to big matrices .In order to tackle such challenges, **factorised representation** is utilised.

$$X = W * H^t$$

where W,H are m $\times$ r and n $\times$ r matrices, respectively, with r<<min(m,n). Using W,H the number of degree of freedom is decreased from **O(mn) to O(m+n)**.

Later did we introduce the concept of deep learning on graphs i.e extending classic CNN to graphs.

The general method here is based on the idea of polynomial Chebyshev. **Chebyshev polynomials** is an effective filter system that decreases the complication of CNNs to the same complexity as regular CNNs (on grids).

Chebyshev polynomials form an orthogonal basis for the space of smooth functions on $[-1,1]$ and are thus convenient to compactly represent spectral filters.

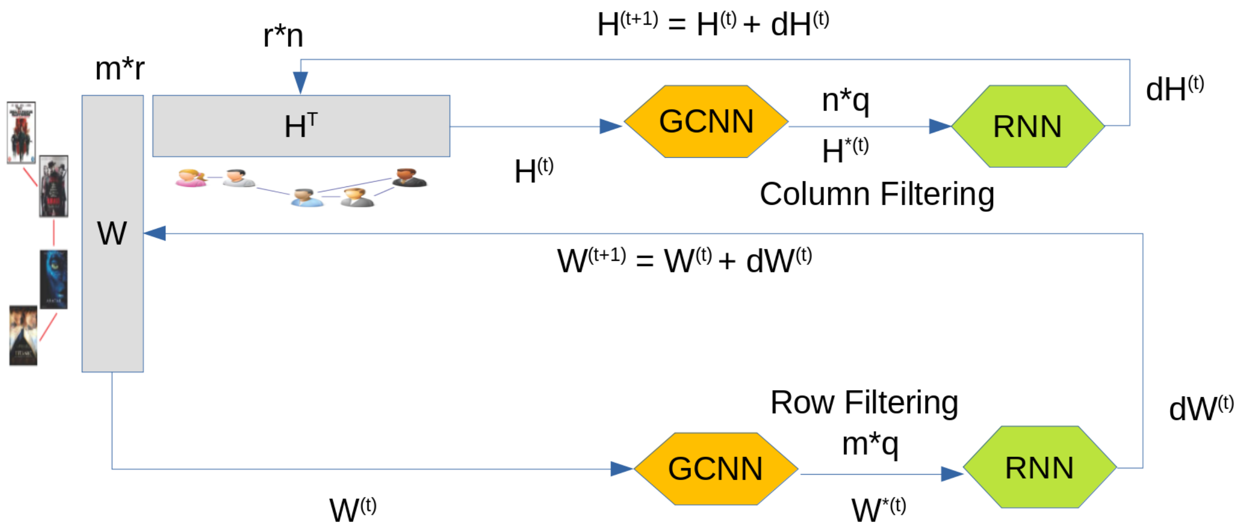


FIGURE 5: Factorized matrix completion model using Separable RGCNN

As an appreciable diffusion process applied to the score values, we suggest the formulation of matrix completion. For this reason, the deep learning architecture includes a **spatial component** extracting spatial characteristics from the matrix (we investigate two distinct methods to the complete and factored matrix models), and a **temporal component** utilising a recurrent LSTM network.The architecture includes the full matrix completion model is used in the **Recurrent GCNN (RGCNN)** architecture, which operates on the matrix's rows and columns concurrently. For each member of the input matrix, the **Multi-Graph CNN (MGCNN)** module produces a q-dimensional feature vector.

The number of learning parameters is **O(1)** and the learning complexity is **O(mn)**.

Chebyshev polynomials with an order of p=5 are utilised here that when applied with MG-CNN results in output of 32 dimensional features.This results in the computational complexity of O(mn).

In view of the factorised form of matrix $X = W * H^t$, simplification of the Multigraph convolution is achieved and single dimensional convolution is applied in each factor in the relevant graph.This factorized version depicts the **Separable Recurrent GCNN(sRGCNN)** architecture that works independently on the parameters W,$H^t$ rows and columns. For each input line/column the output of the GCNN Module is a q-dimensional feature-vector. The number of learning parameters is **O(1)** and the learning complexity is **O(m+n)**.

**Matrix Diffusion With RNN** is the major step involved in this. The extracted features from MG-CNN are passed to Recurrent Neural Network(RNN) that implements the score diffusion process.Here,we used traditional **LSTM RNN** architecture as it is efficient in learning dynamical property of data sequences.

So with this we reached to final two versions of the matrix completion model.First approach using **Algorithm 1** was the RGCNN approach with a computational complexity of **O(mn)** and for larger matrices we can opt **Algorithm 2** , the standard GCNNs that processes the rows and columns separately and scales linearly down to **O(m+n)**.

**Algorithm 1: Full matrix completion model using RGCNN**

input m × n matrix $X^{(0)}$ containing initial values

1. **for** t= 0 :T **do**
2. Apply the Multi-Graph CNN on $X^{(t)}$ producing an m×n×q output $X^{(t)}$ containing a q-dimensional feature vector for each matrix element.
3. **for** all elements (i,j) **do**
4. Apply RNN to feature vector producing the predicted incremental value $dx_{ij}^{(t)}$
5. **end for**
6. Update $X^{(t+1)}$=$X^{(t)}$+$dX^{(t)}$
7. **end for**

**Algorithm 2: Factorized matrix completion model using sRGCNN**

input m × r factor $H^{(0)}$ and n×r factor $W^{(0)}$ representing the matrix $X^{(0)}$

1. **for** t= 0 :T **do**
2. Apply the Graph CNN on $H^{(t)}$ producing an n × q output $H^{(t)}$.
3. **for** j= 1 : n **do**
4. Apply RNN to feature vector producing the predicted incremental value $dh_{ij}^{(t)}$
5. **end for**
6. Update $H^{(t+1)}$=$H^{(t)}$+$dH^{(t)}$
7. Apply the Graph CNN on $W^{(t)}$ producing an m × q output $W^{(t)}$.
8. **for** i= 1 : m **do**
9. Apply RNN to feature vector $W_i^{(t)}$=( $w_{i1}^{(t)}$,..., $w_{iq}^{(t)}$) producing the predicted incremental value $dw_i^{(t)}$

10. **end for**

11. Update $W^{(t+1)}=W^{(t)}+dW^{(t)}$

12. **end for**

## 0.5 Results

RGCNN Settings:order p=5 for **Chebyshev polynomial** and outputs **32 dimensional features**. LSTM cell with 32 features and T=10 stages of diffusion. Rank r=10 was employed for factorised models for **Movielens100k dataset**. This technique has been cross-validated for hyperparameters. Adam stochastic optimization algorithm is used for training factorized model with learning rate of $10^{-3}$ . Diagram given below shows the loss curve of both Training and Testing after being evaluated under **4000 iterations** resulting with a **Root Mean Square Error(RMSE) of 0.969728**.

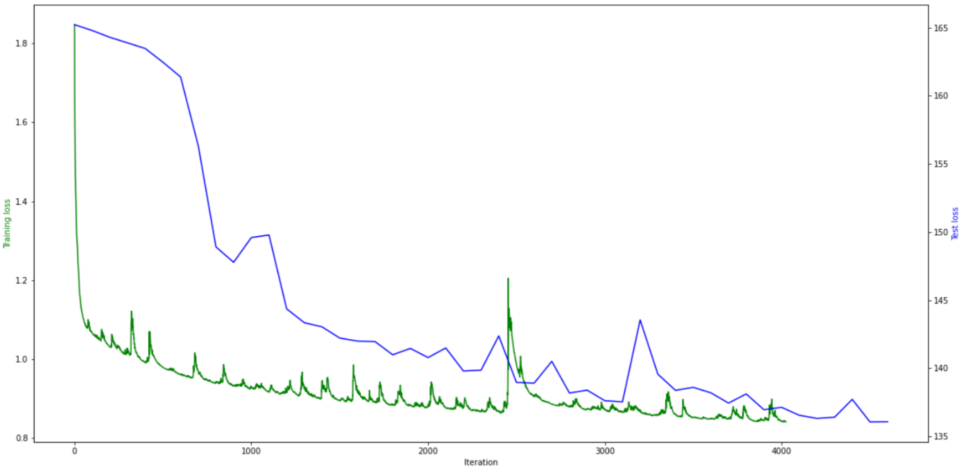


FIGURE 6: Loss in Training and Testing

The graphic below shows the **first matrix** with a few entries. The values for that matrix are within the range of **[0,5]**. The matrix is subsequently handled by the model of factorization and also being applied with **singular value decomposition**.

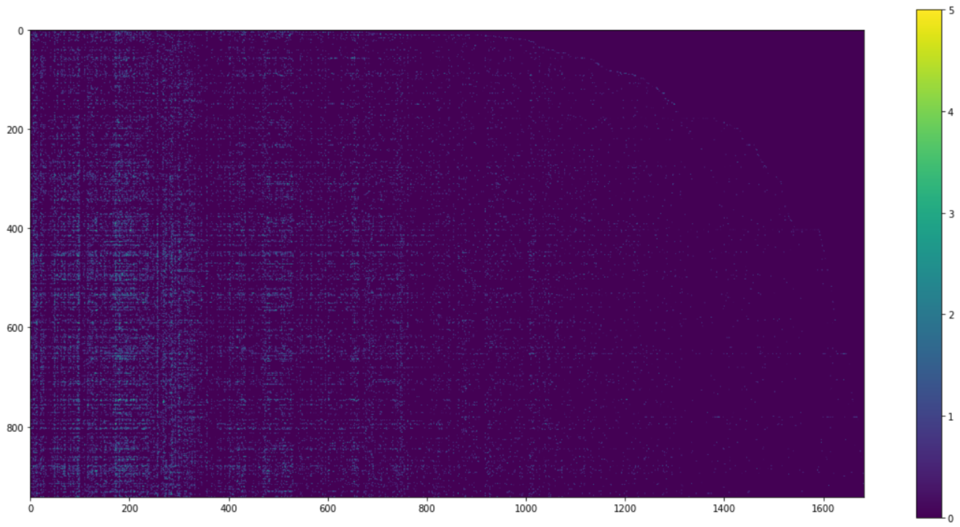

FIGURE 7: Initial Matrix

We acquired the following matrix after the factorization technique employed to the above matrix. In order to deal with huge matrices and to decrease the extent of degree of freedom, the **geometrical model of factorisation** is applied.Current model is then exposed to the deep learning approaches **GCNN and RNN architecture.**

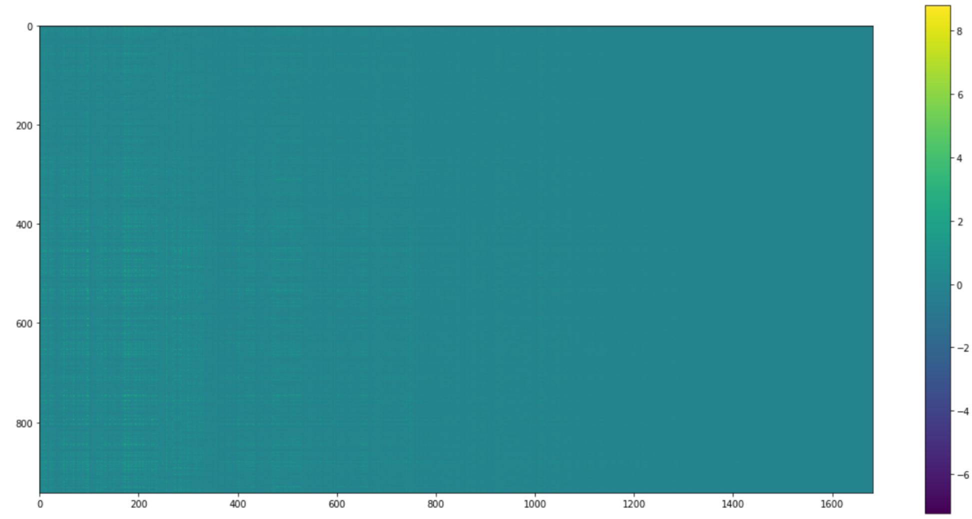

FIGURE 8: Factorized Matrix

Below is the final matrix resulting after the aforesaid model is exposed to the architecture **GCNN and RNN**,a deep learning approach for matrix completion. After it is further exposed to **laplace and fourier transformations** with **Chebyshev polynomial**, all entries are filled with this matrix. The matrix now shows the lowest to highest rated items for user recommendations.

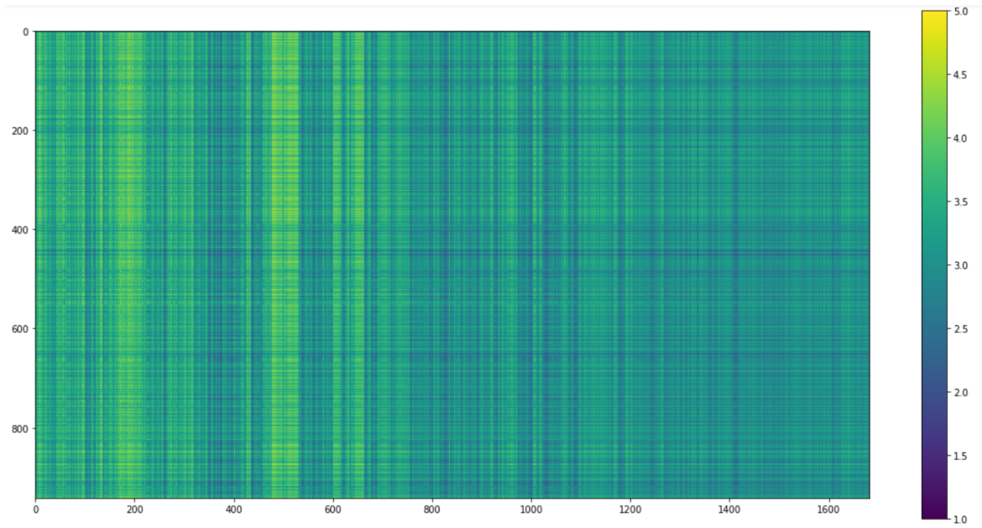

FIGURE 9: Final Matrix for Recommendation System

## 0.6 Conclusion

I have introduced a new profound **Deep Learning Approach** based on a specially constructed multi-graph convolutionary network architecture for matrix completion designed for recommendation systems.Shown that using deep learning for matrix completion hikes it's performance.The original models when exposed to deep learning techniques of **MGCNN and LSTM (RNN)** resulted in a reduced **q-dimensional feature vector** for each input row/column.It shows the high efficacy of dynamic data sequence learning. The algorithm used has a **low computing complexity** and a consistent amount of degree of freedom regardless of the size of the matrix. It demonstrates the possibilities in **non-Euclidean areas** of the emerging subject of geometric deep learning.The findings indicate that the missing values of a given matrix are entirely covered and thus **recommend users** for items that meet their needs and preferences shown by different ratings.

# Bibliography

[1] Asim Ansari, Yang Li, and Jonathan Z Zhang. "Probabilistic topic model for hybrid recommender systems: A stochastic variational Bayesian approach". In: *Marketing Science* 37.6 (2018), pp. 987–1008.

[2] Michael M Bronstein et al. "Geometric deep learning: going beyond euclidean data". In: *IEEE Signal Processing Magazine* 34.4 (2017), pp. 18–42.

[3] Emmanuel J Candès and Benjamin Recht. "Exact matrix completion via convex optimization". In: *Foundations of Computational mathematics* 9.6 (2009), pp. 717–772.

[4] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. "Convolutional neural networks on graphs with fast localized spectral filtering". In: *arXiv preprint arXiv:1606.09375* (2016).

[5] Chein-Shung Hwang and Yu-Pin Chen. "Using trust in collaborative filtering recommendation". In: *International conference on industrial, engineering and other applications of applied intelligent systems*. Springer. 2007, pp. 1052–1060.

[6] Thomas N Kipf and Max Welling. "Semi-supervised classification with graph convolutional networks". In: *arXiv preprint arXiv:1609.02907* (2016).

[7] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. "Deep learning". In: *nature* 521.7553 (2015), pp. 436–444.

[8] Dokyun Lee and Kartik Hosanagar. "How do product attributes and reviews moderate the impact of recommender systems through purchase stages?" In: *Management Science* 67.1 (2021), pp. 524–546.

[9] Yujia Li et al. "Gated graph sequence neural networks". In: *arXiv preprint arXiv:1511.05493* (2015).

[10] Federico Monti, Michael M Bronstein, and Xavier Bresson. "Geometric matrix completion with recurrent multi-graph neural networks". In: *arXiv preprint arXiv:1704.06803* (2017).

[11] Federico Monti et al. "Geometric deep learning on graphs and manifolds using mixture model cnns". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 5115–5124.

[12] Umberto Panniello, Michele Gorgoglione, and Alexander Tuzhilin. "Research note—In CARSs we trust: How context-aware recommendations affect customers' trust and other business performance measures of recommender systems". In: *Information Systems Research* 27.1 (2016), pp. 182–196.

[13] Youngjoo Seo et al. "Structured sequence modeling with graph convolutional recurrent networks". In: *International Conference on Neural Information Processing*. Springer. 2018, pp. 362–373.

[14] David I Shuman et al. "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains". In: *IEEE signal processing magazine* 30.3 (2013), pp. 83–98.

[15] David I Shuman et al. "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains". In: *IEEE signal processing magazine* 30.3 (2013), pp. 83–98.

[16] Benjamin Van Roy and Xiang Yan. "Manipulation robustness of collaborative filtering". In: *Management Science* 56.11 (2010), pp. 1911–1929.