School of Information Technology & Engineering
Department of Computer Applications
Fall Semester 2019 – 20

Master Thesis Project 1st Review

An Improved Collaborative Filtering for rating prediction in movie Recommender System

By
Abhishek Kumar
18MCA0156

Guided By
Chemmalar Selvi G
SITE

Abstract

In the age of digitization, where we have immense collection of media which has given an overwhelming number of choices, which is need to be filtered, prioritized and efficiently deliver relevant information in order to alleviate the problem of information overload. This problem can be solved by using recommender system which searches and filters through a large volume of dynamically generated information to deliver a more personalized content and services. Today there are many approaches to build an effective recommendation system one of which is Collaborative Filtering (CF). This conventional method uses ratings given to items by the user to predict the ratings. Since the traditional method initially relies on user data which can be sparse, which can eventually lead to common recommendation problem called Cold Start. Due to all these problems a new approach is taken using the neural network in which LSTM has shown a very competitive performance and it also outperforms in the term of accuracy and efficiency as compared to other methods.

Introduction

With the explosive growth in online entertainment media streaming, there has been an overwhelming growth in content which has created a dilemma in choosing a better option. In order to rescue from this Recommendation System has proven itself a indispensable tool for businesses and their users. Recommendation System are mainly of two types Content Based and Collaborative Filtering, in which Collaborative Filtering has proven its worth by providing a better results and recommendation by analyzing the user similarities with respect to the content they access and helps to delivery a more relevant recommendation.

Collaborative Filtering (CF) recommender approaches are extensively investigated and widely used in industry. As the most popular approach among various collaborative filtering technique is matrix factorization (MF) which learns a latent space to represent a user or an item becomes a standard model for recommendation due to its scalability and flexibility. Another approach include memory based collaborative technique which created a user similarity matrix and item similarity matrix based on cosine similarity to predict the ratings.

Recommendation System are great tool for predicting rating and has proved its worth but they have some issues which eventually leads to less accurate prediction or recommendation and the most common issues are Data Sparsity and Cold Start, traditionally method like matrix factorization usually due to data sparsity so to overcome this problem neural network has been introduced into matrix factorization, Neural Network Matrix Factorization (NNMF) [37] and Neural Collaborative Filtering (NCF) [53] are the two representative works.

In this paper our approach will also be focused on collaborative filtering using matrix factorization and gradient descent to optimize the result which will be used in our deep learning model. Due to the properties like flexibility, Sequence modelling, nonlinear transformation deep learning model are the best approach to eliminate the issues like Data Sparsity and Cold Start which will be our main aim.

Literature Survey

SNo	Paper Detail	Proposed Method	Advantages	Disadvantages
1	Graph Convolutional Matrix Completion by Rianne van den Berg Thomas N. Kipf Max Welling 2017	Graph Auto Encode	Able to resolve cold start by including user item features Model can work one large dataset Standard error less than 0.001	Unable to address the issue of data sparsity Scalability is not easy

2	Deep Neural for Youtube Recommendation by Paul Covington, Jay Adams, Emre Sargin 2016	Deep ranking network with relu activation function	outperforming matrix factorization outperformed previous linear and tree-based methods for watch time prediction Able to handle cold start problem	Can be overfitted Not efficient in predicting click-through rate directly
3	Deep Models of Interaction across sets by Jason Hartford, Devon R Graham 2018	AutoEncoders	Able handle Extrapolation Solves Data Sparsity	Didn't address cold start problem
4	Deep Matrix Factorization Models for Recommender Systems By Hong-Jian Xue, Xin-Yu Dai 2017	Deep Structured Semantic Models (DSSM), uses a deep neural network to rank a set of documents for a given query	Able to solve Data Sparsity	-
5	Deep Learning for Recommender Systems by Alexandros Karatzoglou, Balázs Hidasi 2017	Survey Paper	-	-
6	Collaborative Filtering and Deep Learning Based Recommendation System For Cold Start Items by Jian Wei, Jianhua He 2016	timeSVD++, SADE	Handles cold start problem using two models	approach requires extra storage and computation resources. Does not addresses the Data Sparsity problem

7	Collaborative Deep Learning for Recommendation System by Hao Wang, Naiyan Wang, Dit-Yan Yeung 2015	hierarchical Bayesian model called collaborative deep learning (CDL) and SDAE	Solves Data Sparsity problem by using auxiliary information such as item content information	Does not addresses the Cold Start problem
8	A Recurrent Neural Network Based Recommendation System by David Zhan Liu Gurbir Singh	Multi-stacked bi-directional Gated Recurrent Unit (GRU) and Long Short-Term Memory (LSTM)	Improved recommendation accuracy by using reviews to predict the user interest	Does not helps to solve cold start and data sparsity problem
9	Recurrent Neural Networks for Long and Short-Term Sequential Recommendation by Kiewan Villatel, Elena Smirnova 2018	RNN for Short term and Long Term	Works well even for distanced user item interaction helps reduce data sparsity problem	-
10	Wide & Deep Learning for Recommender Systems by Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen	Wide , Deep and Wide-Deep neural network	Wide & Deep learning jointly trained wide linear models and deep neural networks to combine the benefits of memorization and generalization for recommender systems which helps to eliminate cold start as well as data sparsity problem	-

Module Description

1. Data Preparation

   1.1 Dataset Description
   MovieLens 1M dataset is used, which contains 1 million ratings (between 1-5) , dataset contains column like

   UserID::MovieID::Rating::Timestamp

   - UserIDs range between 1 and 6040
   - MovieIDs range between 1 and 3952
   - Ratings are made on a 5-star scale (whole-star ratings only)
   - Timestamp is represented in seconds since the epoch as returned by time(2)
   - Each user has at least 20 ratings

   1.2 Preprocessing Step
   Since the dataset sparse so we need to make it denser in order to improve the recommendation, so we need to remove the inactive users and items with less rating because this will not help in improving the results, so after removing we need to change the dataset into User-Item matrix which we got of size 943 rows and 1682 columns. This approach has made the dataset denser leading to the reduction of data sparsity problem.

2. Matrix Factorization

   Firstly, we have a set U of users, and a set D of items. Let $\mathbf{R}$ of size $|U| \times |D|$ be the matrix that contains all the ratings that the users have assigned to the items. Also, we assume that we would like to discover K latent features. Our task, then, is to find two matrices $\mathbf{P}$ (a $|U| \times K$ matrix) and $\mathbf{Q}$ (a $|D| \times K$ matrix) such that their product approximates $\mathbf{R}$:

$$\mathbf{R} \approx \mathbf{P} \times \mathbf{Q}^T = \hat{\mathbf{R}}$$

In this way, each row of $\mathbf{P}$ would represent the strength of the associations between a user and the features. Similarly, each row of $\mathbf{Q}$ would represent the strength of the associations between an item and the features. To get the prediction of a rating of an item $d_j$by u_i, we can calculate the dot product of the two vectors corresponding to $u_i$and d_j:

$$\hat{r}_{ij} = p_i^T q_j = \sum_{k=1}^{k} p_{ik} q_{kj}$$

2.1 Decompose

The user-item matrix we got is too large to process so applying matrix factorization will decompose the matrix into constituent parts which will make it simpler to perform operations. The two matrices will be in the m x m and m x n size.

2.2 Gradient Descent Loss function

After getting decomposed matrix it needs to be validated to check for the error rate and should be reduced to get the optimized result, so here gradient descent method will be applied to minimize the error rate will give the most optimized decomposed matrices.

$$e_{ij}^2 = (r_{ij} - \hat{r}_{ij})^2 = (r_{ij} - \sum_{k=1}^{K} p_{ik} q_{kj})^2$$

3. Deep Learning Model

4. Evaluation of Dataset

References

[1] Gintare Karolina Dziugaite and Daniel M Roy. 2015. Neural network matrix factorization. arXiv preprint arXiv:1511.06443 (2015).

[2] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative €ltering. In WWW. 173–182.

[3] Graph Convolutional Matrix Completion by Rianne van den Berg Thomas N. Kipf Max Welling 2017

[4] Deep Neural for Youtube Recommendation by Paul Covington, Jay Adams, Emre Sargin 2016

[5] Deep Models of Interaction across sets by Jason Hartford, Devon R Graham 2018

[6] Deep Matrix Factorization Models for Recommender Systems By Hong-Jian Xue, Xin-Yu Dai 2017

[7] Deep Learning for Recommender Systems by Alexandros Karatzoglou, Balázs Hidasi 2017

[8] Collaborative Filtering and Deep Learning Based Recommendation System For Cold Start Items by Jian Wei, Jianhua He 2016

[9] Collaborative Deep Learning for Recommendation System by Hao Wang, Naiyan Wang, Dit-Yan Yeung 2015

[10] A Recurrent Neural Network Based Recommendation System by David Zhan Liu Gurbir Singh

[11] Recurrent Neural Networks for Long and Short-Term Sequential Recommendation by Kiewan Villatel, Elena Smirnova 2018

[12] Wide & Deep Learning for Recommender Systems by Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen