Journal of Scientific & Industrial Research
Vol. 79, June 2020, pp. 499-502

Boosting a Hybrid Model Recommendation System for Sparse Data using Collaborative Filtering and Deep Learning

P Valarmathi[1], R Dhanalakshmi[2,]* and Narendran Rajagopalan[1]

[1]National Institute of Technology Puducherry
[2]Indian Institute of Information Technology Tiruchirappalli

Received 12 September 2019; revised 23 February 2020; accepted 23 April 2020

The exponential increase in the volume of online data has generated a confront of overburden of data for online users, which slow down the suitable access to products of pursuit on the Web. This contributed to the need for recommendation systems. Recommender system is a special form of intelligent technique that takes advantage of past user transactions on products to give recommendations of products. Collaborative filtering has turn out to be the commonly adopted method of providing users with customized services, except that it endures the problem of sparsely rated inputs. For collaborative filtering, we introduce a deep learning-based architecture which evaluates a discrete factorisation of vectors from sparse inputs. The characteristics of the products are retrieved using a deep learning model, denoising auto encoders. The traditional collaborative filtering algorithm that predicts and uses the past history of consumer interest and product characteristics are updated with the characteristics obtained by deep learning model for sparse rated inputs. The results of sparse data problem tested on MovieLens data set will greatly enhance the user and product transaction.

Keywords: Collaborative filtering, Neural network, Sparse inputs

Introduction

Collaborative filtering (CF)[1] is a way of determining a consumer's interest on a particular product by studying the background of various other consumers' preferences.[2] It is turning out to be increasingly more significant in the achievement of e-trade and is being utilized in different applications in industries like Ebay, Netflix[3] and Google. Recommendation system has a massive effect on the business accomplishment of these organizations as far as income and client satisfactions are concerned. Collaborative filtering algorithms are usually divisible into two groups, memory based approach and approach based on model. The approaches based on memory attempt to forecast a consumer interest by more related apps or products driven on the scores.

The locality-sensitive hashing is a common method in memory-based approach, which applies the nearest neighbourhood technique. Memory-based approach utilizes the product or client information that continues to be accessible during execution in the memory. It gives a proposal with high exactness. Through various algorithms, model-based approaches are built to discover similar transactions based on the training set of data. Standard model-based approaches comprise the Bayesian models[4], clustering models[5] and latent semantic models.[6]

But, it is generally realized that CF approach[7] experiences sparsity issues. Merely few items are given ratings by the users. In many rating datasets, only about 1 percent of matrix components earn a score from users. It is extremely challenging to assess the interaction between users and products to make efficient recommendations by using a sparse rating matrix.

Deep learning methods have drawn tremendous importance in the domain of recommender systems due to extraordinary achievements in diverse areas like natural language processing[8], image processing.[9] In addition, deep learning techniques are also utilized in collaborative filtering. The sparse rating matrix is illustrated in Table 1. Sparse inputs also require

Table 1 — Illustration of sparse rating matrix

Consumer/Product	P_1	P_2	P_3	P_4	P_5	P_6	P_7
C_1	5	3	4	—	3	5	—
C_2	4	—	—	—	—	—	—
C_3	—	—	—	5	—	—	—
C_4	3	3	—	5	—	—	2
C_5	—	—	—	—	—	—	2
C_6	1	—	—	—	—	—	1

*Author for Correspondence
E-mail: r_dhanalakshmi@yahoo.com

scores to be considered for enhanced recommendation and they must be exactly suggested to the customers for effective recommendation. Else, the sparse rated items possibly will move into an unnecessary period of not receiving ratings. For collaborative filtering, we introduce a deep learning-based architecture which evaluates a discrete factorisation of vectors from sparse inputs.

Materials and Methods

Problem Formulation

A typical architecture to combine the graph based collaborative filtering method with the deep learning model[10] is presented which enhances the performance of recommendation system for sparse inputs. The important factor is on the retrieval of properties of products by the deep learning model[11] and updating these properties to the graph based collaborative filtering algorithm.

Since the ratings for sparse inputs cannot be estimated by conventional CF algorithms[12], further specifications for the products are acquired by deep learning model for the better recommendation. Product features are derived from definitions of the items and used for the ranking calculation of sparse input with the graph based CF algorithm.

A matrix M of products and consumers is obtained after the pre-processing of data, where $M\epsilon\ M_{UxI,\ I}$ refer the number of products and $_U$ refers the number of consumers. The consumer-product matrix M also contains the sparse inputs where the majority of the elements are left without ratings.[13] The other non-zero ratings are partitioned into training and testing data. Every single element in the matrix, mu, i is labelled with a ranking of integers 1 to 5, reflecting the preference of viewer to the movie. The training matrix S is stated as $= \begin{cases} m_{u,i}, (u,i)\epsilon trainset \\ 0, nullrating \end{cases}$. The metric used to evaluate the performance of the hybrid model is RMSE[14], which is defined as

$$\text{RMSE}=\sqrt{\frac{\Sigma_{j=1}^{N}(P_j-O_j)^2}{N}} \qquad \dots (1)$$

Where P_j is the predicted rating, O_j refers actual rating and N refers the number of observations.

The proposed recommendation model

For collaborative filtering, we introduce a deep learning-based architecture, HRCD for sparse inputs. The recommender system is supposed to have users, U and items or products, I which can be represented as a matrix. The elements of the matrix, $m_{u,i}$ are the scores assigned for the products I by the consumers U. The challenge of recommendation is to predict the missing ratings in the matrix based on previous history of user's transactions.

The actual dataset is divided into training and testing set. The work flow of the proposed model is illustrated in Fig. 1. Based on the timestamp of the ratings obtained from the users, the movies are organised in an order. The model is trained by incorporating the required parameters and data into the training set. The scores of the testing data set are predicted using the trained model. The accuracy of the prediction then is calculated and compared.

The normalized data which are classified as user vector and item vector are passed as input to the auto encoder. The respective rating given by the users is utilized as training label. Batch normalisation technique is applied in all layers. The conditional probability of the rating is computed using Softmax function which is defined as

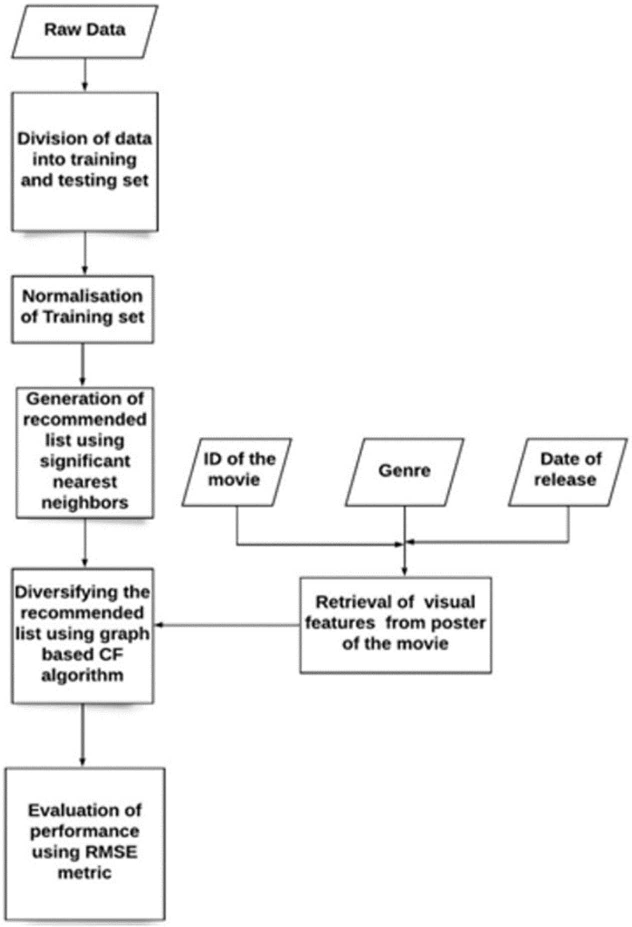
Raw Data
Division of data into training and testing set
Normalisation of Training set
Generation of recommended list using significant nearest neighbors
ID of the movie
Genre
Date of release
Retrieval of visual features from poster of the movie
Diversifying the recommended list using graph based CF algorithm
Evaluation of performance using RMSE metric

Fig. 1 — Work flow of the proposed model

$$\text{Softmax (y)} = \frac{\exp(y)}{\Sigma\exp(y)} \quad \text{... (2)}$$

The missing ratings cannot be considered as zero, which subsequently allows the model to view the rating as zero, resulting in deprived performance. The probability of the non-rated items in the matrix is given by

$$P(n) = \frac{Total\ number\ \ of\ missing\ \ scores}{Total\ number\ \ of\ items} \quad \text{... (3)}$$

The normalized consumer vector and the product vector are fed to the neural net as input data. Predicted ranking is derived on the basis of the conditional probability measured using the softmax function as the output from neural network. Root Mean Square Error (RMSE) is the efficiency metric used for evaluations.

Denoising autoencoder is a form of artificial neural network which is used in an unsupervised way to learn effective data coding. Autoencoder allows discovering the encoding for a data set. The network is required to use hidden layers, causing an implicit decrease in data dimensionality. The technique of batch normalization is applied in all layers including the output layer to avoid the problem of over-fitting.

We have also computed the correlation between the visual features and the average ratings of movies. We have additionally processed the connection between the visual characteristics and the ratings of movies. The importance of a picture has a significant effect on the product's click rate. The item's image plays a key role in recommender system. From the product's image, we can infer the role of the product approximately without knowing the details about the item attribute. In this recommendation model, one of the image attribute, poster of the film is used. Features of the sparse rated movie posters are retrieved and similarity between sparse rated movie posters and previous images is calculated using cosine similarity. A group of significant neighbours with a high degree of similarity is considered and ratings for all unrated elements in the matrix are calculated. The predicted ratings are sorted from the highest to the lowest, and the top N items with the highest rating are given as the recommended list.

Results and Discussion

Dataset

The proposed model is assessed using a huge dataset, MovieLens, formed by the GroupLens research. This hybrid model is tested along with two separate MovieLens datasets, MovieLens100k and MovieLens1M. The ratings against movies given by users for these datasets are indicated in Table 2. Every rating is on the scale of integer 1 to 5, with worst movie assigned a score of 1 and best movie with 5. The data set is divided into training data and test data on the footing of a share of 70 and 30. The proposed model is evaluated by using root mean square error.

Performance Evaluation

As the training advances, models are found to achieve lesser value of RMSE. For the MovieLens, 100k dataset 2,625 (943 + 1682) rating vector is given as input to the neural network. The problem of over fitting is expected to rise with growing amount of epochs. As the input to the neural net 9,992 rating vector is given for the MovieLens1M dataset the learning factors like weight W and biases γ, β are initialized accordingly. The values of RMSE for various methods are shown in Table 3 for two datasets.

The results of the recommended model are finer than traditional methods such as SVD, Autoencoder method. In addition, the outcomes of this hybrid method are improved than the model based on the restricted Boltzmann machine. The performance criteria of MovieLens1M dataset are higher than MovieLens 100k because of huge volume of training dataset. RMSE value of the proposed model is lesser than other models.

The proposed model is assessed as far as performance metric, RMSE and stood out from different models. The correlation of RMSE estimations of different models with the new HRCD model for MovieLens1M dataset is provided in Fig. 2.

Table 2 — Movielens datasets

Dataset	Ratings	Users	Items
MovieLens 100k	100,000	943	1682
MovieLens 1M	1,000,000	6040	3952

Table 3 — RMSE values of various models

Method	MovieLens 100k	MovieLens 1M
SVD	0.942	0.889
Autoencoder	0.937	0.881
Proposed model (HRCD)	0.895	0.798

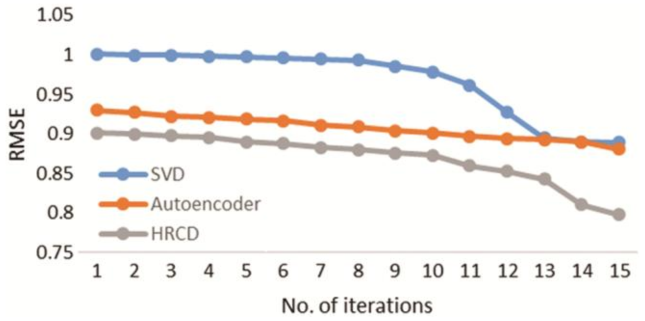

Fig. 2 — RMSE estimations of different models for MovieLens 1M dataset

Conclusions

We have proposed a hybrid model with collaborative filtering and neural network for sparse inputs. Recommendation of sparse inputs is demanding and even an exploration subject. To effectively deal with the non-rated elements, the neural network architecture is provided with the normalized consumer vector and the normalized product vector as input. Results of MovieLens data set tests show that the new model outdoes the traditional CF algorithms like SVD and other methods. In addition, our method can be broadened to other e-commerce applications based on collaborative filtering. So we can predict the preference of a recently arrived user on a product with no further learning. The recently arrived ratings can be used for training the model. The proposed model functions perfectly in those situations where the users and products characteristics changes over time.

References

1 Wang H, Wang N & Yeung D Y, Collaborative deep learning for recommender systems, *Proc Int Conf know disc data min* 2015, 1235–1244.

2 Suvash S, Aditya K M, Scott S & Lexing X, Auto Rec: Autoencoders Meet Collaborative Filtering, *WWW '15 Compan: Proc 24th Int Conf World Wide Web*, 2015, 111–112, https://doi.org/10.1145/2740908.2742726.

3 Bennett J, Lanning S, The Netflix prize, *Proc KDD Cup and Workshop*, 2007.

4 Campos P G, Dez F & Cantador I, Time-aware recommender systems: A comprehensive survey and analysis of existing evaluation protocols, *User Model User-Adap Inter*, **24(1–2)** (2014) 67–119.

5 Ocepeka U, Rugeljb J & Bosnica Z, Improving matrix factorization recommendations for examples in cold start, *Exp Sys Applns*, **42(19)** (2015) 6784–6794.

6 Hofmann T, Latent semantic models for collaborative filtering, *ACM Trans Inf Syst*, **22(1)** (2004) 89–115.

7 Krizhevsky A, Sutskever I & Hinton G, Imagenet classification with deep convolutional neural networks, *Adv Neu Inf Process Syst*, 2012.

8 Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, Huang Z, Karpathy A, Khosla A, Bernstein M, Berg A C. & Fei-Fei L, ImageNet large scale visual recognition challenge, *Int J Comput Vis*, **115(3)** (2015) 211–252. https://doi.org/10.1007/s11263-015-0816-y

9 Mikolov T, Sutskever I, Chen K, Corrado G & Dean J, Distributed representations of words and phrases and their compositionality, *Adv Neu Inf Process Syst*, 2013.

10 Salakhutdinov R, Mnih A & Hinton G, Restricted Boltzmann machines for collaborative filtering, *Proc Int Conf Mach Learn*,2007, 791–798.

11 Georgiev K & Nakov P, A non-IID framework for collaborative filtering with restricted Boltzmann machines, *Proc Int Conf Mach Learn*, 2013, 1148–1156.

12 Oord A, Dieleman S & Schrauwen B, Deep content-based music recommendation, *Adv Neu Inf Process Syst* 2013, 2643–2651.

13 Yuanxin O, Wenqi L, Wewnge R & Zhang X, Autoencoder-based collaborative filtering, *Proc Int Conf Neu Inf Process*, 2014, 284–291.

14 Ioffe S & Szegedy C, Batch normalization: Accelerating deep network training by reducing internal covariate shift, *Proc Int ConfMach Learn* 2015.

15 Srivastava N, Hinton G, Krizhevsky A, Sutskever I & Salakhutdinov R, Dropout: A simple way to prevent neural networks from overfitting, *J Mach Learn Res* (2014), 1929–1958.