

# Graph neural network based entity augmented representation for recommendation system

Huinan Zhao, Lei Mu\*, Xiuzhuo Wei, Suhua Wang\*

<sup>a</sup>Changchun Humanities and Sciences College, Changchun, 130117, China.

## ABSTRACT

A lot of rich auxiliary information can be integrated into the scoring and show good performance, which plays a good role in many recommended methods. In this paper, we use the rich semantic of knowledge graph to explore user preferences and improve the algorithm performance by using item attributes. A large number of experiments have been carried out on two data sets to prove the validity of the graph convolution recommendation model.

**Keywords:** Recommendation system, constitutional model, recommendation strategy, entity enhancement

## 1. INTRODUCTION

The recommendation system is a model for solving problems by using machine learning to build various algorithms, which provides channels of interest to the user to create commercial value. A good algorithm can improve the accuracy of the recommendation system, improve the coverage rate, and perceive changes in users' interests in time.

In recent years, generalized Matrix<sup>1</sup> decomposition has become a popular method in recommendation systems. In mathematics, matrix is a collection of complex or real numbers arranged according to a rectangular array, which originally comes from the square matrix composed of coefficients and constants of equations. Matrix decomposition (decomposition, factorization) is the matrix to disassemble the product of several matrices, it can be classified into Triangular Factorization, full rank Decomposition, QR Factorization, Jordan decomposition, and SVD (Singular Value) decomposition. However, the disadvantage of matrix decomposition is that the scalar product of user semantic vector and item semantic represented by matrix decomposition cannot accurately express more complex interactions.

In order to solve the problem, we propose a depth collaborative filtering method based on the perception of film and television characters, and at the same time use the side information of the item to improve the performance of the algorithm to recommend movies. Since the data of collaborative filtering is scarce, the knowledge graph is considered to add auxiliary information to establish the corresponding semantic. The more semantic there are, the more accurate the predictive test will be.

In many machine learning and deep learning algorithms, machines cannot acquire accurate analysis and reasoning abilities like humans, but knowledge graphs can be trained to make accurate predictions from the perspective of probability and correlation.

The classical algorithms of deep learning include gradient descent (BGD)<sup>2</sup> and Newton's method. Gradient descent method is one of the most commonly used methods for solving unconstrained optimization problems using iteration. Newton method is to use the second derivative of a large number of calculations in the process of iteration to make the convergence speed is fast. Table 1 below compares classical algorithms of deep learning.

\*Lei Mu: voodoo777@163.com; \*Suhua Wang: wangsuhua@ccrw.edu.cn  
Huinan Zhao:zhaohn555@163.com; Xiuzhuo Wei:xiaoyumao1010@163.com

Table 1. Comparison of Advantages and Disadvantages of Classical Algorithms for Deep Learning

Classic algorithms for deep learning	Advantages	Disadvantages
Gradient Descent	The full data set of BGD can well represent samples to achieve global optimization. The one-time iteration implements parallel computation for all samples.	The locality of hesitancy convergence leads to the decrease of accuracy and the sample cannot be implemented in parallel.
Newton's Method	The matrix decreases continuously in the iterative process, which makes the convergence speed read faster	The inverse computation of matrix is complicated and costly.

The development of deep learning to its present glory is inseparable from the hard creation of researchers in many fields. Before deep learning got so much attention, there were two iconic stages of development that laid the groundwork for it. The first is the DBN, or Deep belief network, proposed by Hinton and his team. This network forms non-directional memory through the first two layers of the network, so that the deep and directional network can be learned. This method greatly reflects the research value of the depth of the generalization algorithm. Another landmark development was also suggested by Hinton and his team. The idea is to reduce the dimension of the data in the original network. It is an effective initialization value to make the automatic encoder learn, so as to reduce the dimension of the data. These two theories have greatly promoted the explosive development of modern deep learning.

Other scholars have also made indelible contributions to deep learning. In 2007, Salakhutdinov R and his team tried to combine collaborative filtering with other networks to solve the problem that networks cannot handle and recommend large amounts of data. They also applied the method to movie recommendations, and the experiment also proved that the method is excellent.

The AutoRec algorithm was developed by Australian scientists. The beauty of this algorithm is that it connects the encoder with collaborative filtering, creating a three-layer structure similar to the multi-layer perceptron.

Considering the limitations of these algorithms, we consider using the side information of the item to improve the performance of the algorithm. Using two common data sets for training, after training to achieve a very accurate state.

The results obtained by our algorithm are summarized as following:

- (1) We use the rich semantic of knowledge graphs to explore user preferences
- (2) Improving algorithm performance by using item edge information.
- (3) A large number of experiments have been carried out on two data sets to prove the validity of the knowledge convolution recommendation model.

## 2. RELATED WORK

### 2.1 Deep learning

Deep learning<sup>3</sup>, a branch of machine learning, is an algorithm that attempts to abstract data at a high level by using multiple processing layers to consist of complex structures or multiple nonlinear transformations. Deep learning is an algorithm based on representation learning of data in machine learning. So far, there has been several deep learning frameworks, such as convolution neural networks, deep belief network and recursive neural networks, which have been applied in computer vision, speech recognition, natural language processing, audio recognition and bio-informatics and obtained excellent results.

Deep learning based on artificial neural network can effectively train the algorithm in neural network, so that the algorithm can achieve good results through repeated tuning.

## 2.2 Knowledge graph

On May 17, 2012, Google formally proposed the concept of Knowledge Graph<sup>4</sup>, whose original intention is to optimize the results returned by search engines and enhance the search quality and experience of users.

With the development of intelligent information, the knowledge graph is widely used in the field of intelligent search. At the earliest, the knowledge graph is constructed by top-down method, but this method needs to define ontology and data well. In recent years, the knowledge graph has been constructed from the bottom up, and we can extract entities from some open data.

The recommendation method based on the concept of knowledge graph can be divided into two embedding methods, namely the embedding based on graph and the searching based on path. In the former, the key difficulty is how to integrate the knowledge graph with one-hot, while the latter is how to construct a reasonable path. In 2016, not so long ago, an algorithm combining knowledge graph and collaborative filtering was proposed. This approach uses knowledge maps to mine structured data, including structures, views, and text. Information about an item is found from the knowledge graph and, combined with vectors, represented as its eigenvalues. This aspect of technology is actually a heterogeneous network with physical attributes, so it contains too much semantic information. It plots all the side information about the user and the item in a single graph, with each piece of information as a node. These nodes can be interrelated and interdependent, forming very rich data information.

## 2.3 Auxiliary information

Auxiliary information is often mined as supplementary data in the recommendation system. Since the explicit data of score miss seriously for the high frequency, it needs to be extracted from all the data of edges and corners as auxiliary. We use the side information of the item to improve the algorithm performance, such as the age and theme of film and television characters.

The mining of side information greatly supplements the problem of insufficient original data and is also a powerful means to solve the two fatal weaknesses of cold start and insufficient data. The data are also inextricably linked. By mining these data, we can get unexpected results and make up for the deficiency of the algorithm in data quantity.

# 3. SYMBOLIC REPRESENTATION

In order to better describe the experimental model later on, here we first list some notations and representations of the model description. Some further formulas and representations will also be described in detail in Section 4. Here, the inputs and outputs of which are represented as follows.

$$a_1 = \sigma (b + w * a_0) \quad (1)$$

“ $a_1$ ” stands for the output of the hidden layer;  $a_0$  stands for the input of the hidden layer; and  $*$  stands for convolution operation<sup>5</sup>, which is also a convolution neural network.

# 4. MODEL DESCRIPTION

Our model is based on the recommended method of movie attribute graph network. A user - user adjacency matrix is generated by using the user - item interaction bipartite graph. Multiply by the user score matrix and then multiply by weight matrix to get an aggregated user feature matrix.

At the object side, the object - object adjacency matrix is generated by using the object attribute knowledge graph. Multiply by the item attribute matrix and then multiply by weight matrix to get the feature matrix of the item side. For a particular user-item pair, the user id is used to index the user's representation into the user characteristics. Then use the item ID to index representation of the item into the item feature matrix. The two representations are computed interactively and fed into a mulct-layer perceptron, which outputs a predictive score. See Figure 1.

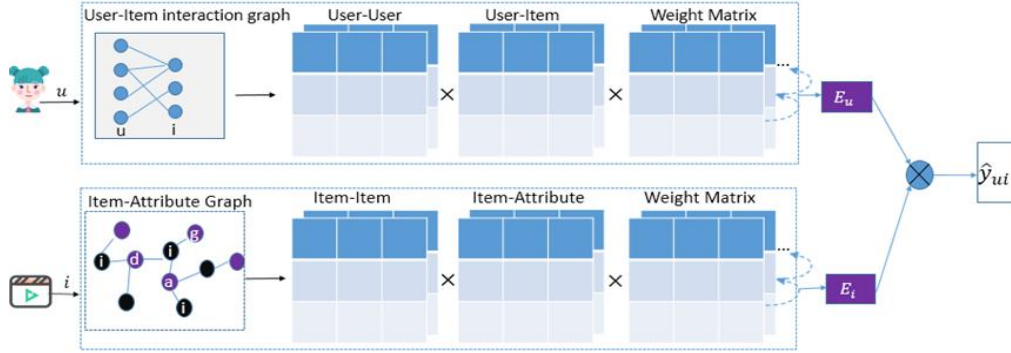


Figure 1. Model Diagram

#### 4.1 Input layer

In Figure 1, the user id and movie id belongs to the input layer. In this layer, suppose the quantities of the two are “m” and “n”, respectively. The user will participate in the calculation as one-hot, and the item itself will be calculated as one-hot data. Both aspects need to use their numbers or other vectors.

#### 4.2 Polymerization layer

All vectors need to be aggregated. On the user side<sup>6</sup>, the user/item interaction diagram in Figure 1 illustrates the relationship between the two. User-user is the adjacency matrix between users, and user-item represents the User's rating of all items. Multiply the two together to get an enhanced feature vector for each user. Because the multiplication makes the features of the adjacent user aggregated to the user, the feature information of the user can be strengthened and the information content can be improved. Then multiply the value with the weight matrix, so that the strong information can be promoted more, and the information value of the data can be further improved.

In terms of items, Item-Item is the matrix that reflects the relationship between items, while Item-Attribute matrix reflects the relationship between items and their attributes. When these two matrices are multiplied together, you get the enhanced item vector. Since each item aggregates attribute from its neighbors, this enhances the vector content of the item. Similarly, multiplying by weight matrix further elevates the feature.

#### 4.3 Interaction layer

Generally, the way to obtain the score value in the recommendation algorithm is to calculate the interactive operation between the feature vector of the user and the item. The interactive algorithm between them is calculated using element-product/wise<sup>6</sup>. When the eigenvalues of both the user and the item aggregate enough neighbor information, they are then fed into the interactive calculation to get an ideal value, which is also larger. Conversely, if the two data points do not sufficiently aggregate other information, the value will be small after the interactive calculation. The interaction value directly affects the accuracy of the final score prediction value, so these values are very important.<sup>7</sup>

After the interactive calculation of the vector of the user and the item, it is further sent into the neural network for calculation. After multiple layers propagation and function activation in the neural network, we get the desired score prediction value. The details are as follows.

$$h_0 = e_u^{new} \otimes ei$$

$$h1 = Relu(h0 * W1 + b1) \tag{2}$$

$$\hat{Y}_{ui} = hL - 1 * WL + bL \quad \text{or} \quad \hat{Y}_{ui} \sigma(hL - 1 * WL + bL)$$

In the above formula, the symbol of  $\hat{Y}_{ui}$  represents the predicted score value, while  $W_k$  and  $b_k$  represent weight and bias respectively, which are passed through the layers of  $h_i$ . “h1” was activated to obtain the final prediction result.

#### 4.4 Computational optimization

In the calculation model, the results of each step need to be optimized to achieve the highest accuracy<sup>8</sup>. In the calculation process of this paper, we need to use the corresponding optimization formula to transfer the calculation. Since the final result we want to calculate is the predicted score value, we need to use the regression error calculation MSE loss calculation formula:

$$loss = \frac{1}{|R|} \sum_{(u,r) \in R} (Y_{ui} - \hat{Y}_{ui})^2 + \lambda \|w\|^2 \quad (3)$$

In (1),  $R$ ,  $Y_{ui}$ ,  $\hat{Y}_{ui}$ ,  $\lambda \|w\|^2$  respectively rating matrix, u-i rating, u-i predict rating and additional term.

In (1),  $R$ ,  $Y_{ui}$ ,  $\hat{Y}_{ui}$ ,  $\lambda \|w\|^2$  respectively represent the rating matrix, u-i rating, u-i predict rating and additional term.

## 5. EXPERIMENT CONTENT

### 5.1 Data set introduction

We used publicly available data set<sup>9</sup>: hetrec2011-delicious-2k, hetrec2011-lastfm-2k, and hetrec2011-movielens-2k. These three data sets were used to evaluate the performance of the GEAR

Next, we explained the experimental process in detail in the following order and analyzed the experimental results. Table 2 below shows the analysis results and statistics of the data set.

Table 2. Statistics of data sets

Data-set	Hetrec2011-movielens-2k	Hetrec2011-lastfm-2k	Hetrec2011-delicious-2k
#items	movies:10,279	artists:17,236	bookmark:104,325
#users	2,303	1,853	1,767
#interactions	855,893 (ratings)[1-5]	92,542 (user-listened artist relations)	437,421 (tag assignments)
#attribute1	20,901 (genre assignments)	165,519 (tag assignments)	67,626 (bookmark titles)
#attribute2	4,051 (directors)	17236 (artist names and Website)	68,266 (bookmark Website)
#attribute3	94,314(actor assignments)	12,656 (user friend relations)	7641 (user relations)

The sparsity of these three data sets is 3.62%, 0.232%, and 0.245%, respectively, with the latter two datasets being about 15 times more sparse than the first.

All the data sets were randomly assigned to 80% and 20%, respectively. The former is for training, the latter for testing. Moreover, in order to avoid over-assembly of common diseases, 3% was randomly verified. Experimental results show that this decision is very suitable.

## 5.2 Baseline method and experimental results

The baseline methods used are: DKN<sup>10</sup>, Ripple Net and K-NOR. Both of them and GEAR were experimentally verified on the above three data sets.

In order to prove the effectiveness of GEAR<sup>11</sup>, RMSE indexes of all models are compared, and the results are shown in Table3.

Table 3. Comparison of RMSE

Method	Hetrec2011- delicious -2k		Hetrec2011-lastfm-2k		Hetrec2011- movie lens -2k	
	RMSE	GEAR improve	RMSE	GEAR improve	RMSE	GEAR improve
DKN	0.7787	2.95	0.7835	4.80	0.7701	4.38
Ripple Net	0.7782	2.80	0.7626	2.19	0.7542	2.36
K-NOR	0.7776	2.73	0.7629	2.23	0.7539	2.32
GEAR	0.7564	-	0.7459	-	0.7364	-
Improve Average	2.83%		3.07%		3.02%	

In order to further prove the effectiveness of GEAR, we carried out a hyper parametric experiment. The number of interaction layers is controlled and experiments are carried out respectively. We set the number of interaction layers as 1 ~ 6 layers to obtain the corresponding experimental results, which is compared and displayed in the graph. See Figure 2(a)-(c):

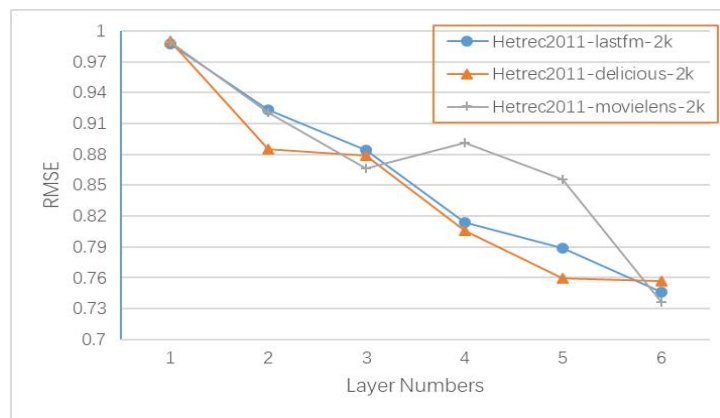


Figure 2(a). Relationship between the number of layers and the RMSE

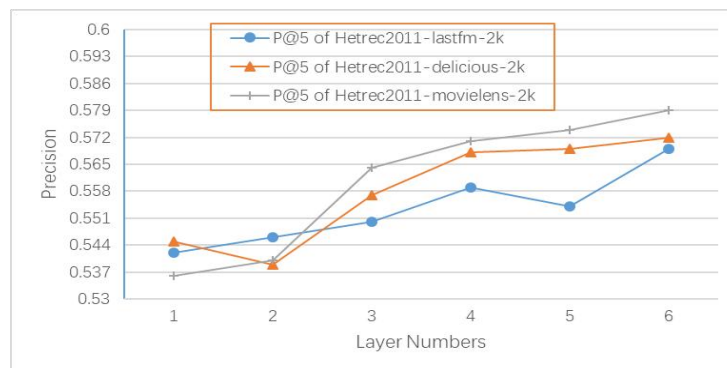


Figure 2(b). Relationship between the number of layers and the precision

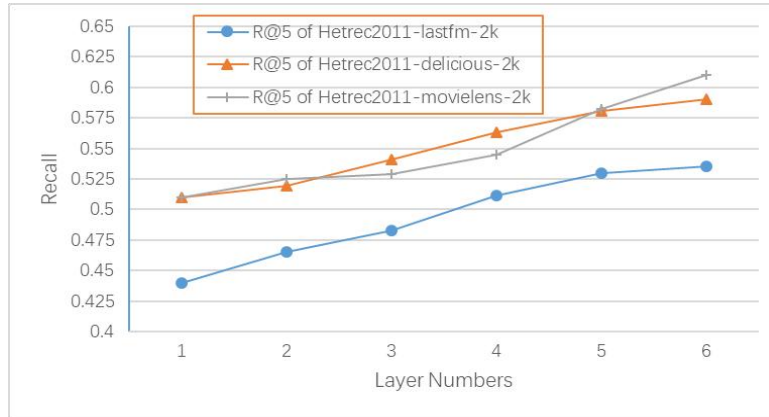


Figure 2 (c). Relationship between the number of layers and the Recall Figure 2 Impact of layers

## 6. CONCLUSION

In traditional recommender systems, the representation of users is often based on one-hot vectors only, and all researchers focus on how to improve the representation of items. However, in fact, users and items are the two indispensable elements of a recommendation system. A user representation based on ID alone is certainly not sufficient and is very lacking in semantic information. The aim of this paper is to enhance the user representation. This is done by treating all users who have seen the same movie as neighbors and generating a user-user adjacency matrix about all users. Each user's ratings of all movies are used as features of that user to generate a feature matrix of all users. The user-user adjacency matrix and the user feature matrix are aggregated together by a graph convolutional network. to make each user fuse the information of all his neighbors in the representation. In the experiments, the second-order third-order and even n-order neighbor information can be aggregated by multi-layer graph convolution operations, in addition to aggregating the information of direct neighbors. The experiments demonstrate that our proposed method effectively improves the user's representation and obtains a significant improvement in the accuracy of the recommendations compared to the traditional methods.

In order to more effectively mine the complex property information in the data set and effectively remove the influence of some confounding factors, we will introduce the causal inference mechanism into the existing methods in the future. The goal is to eliminate some false correlation information in human behavior and find out the real causal information.

In general, causal inference is an idea, and there is no standard algorithm. This is different from machine learning. Therefore, how to effectively combine causal inference and machine learning is also our future direction of research.

## REFERENCES

- [1] Zhou, H., et al., Photonic matrix multiplication lights up photonic accelerator and beyond. *Light: Science and applications*, 11(2), 21 (2022).
- [2] Wang, K., et al., Use of bag-filter gas dust in anaerobic digestion of cattle manure for boosting the methane yield and digestate utilization. *Bioresource Technology*, 348,126729 (2022).
- [3] Lee, D., et al. Deep learning methods for 3D structural proteome and interactome modeling. *Current Opinion in Structural Biology*, 73,102329 (2022).
- [4] Luijt, B. V., Verhagen, M., Bringing Semantic Knowledge Graph Technology to Your Data. *IEEE Software*, 37(2):89-94 (2020).
- [5] Sang, L., et al., Knowledge graph enhanced neural collaborative recommendation. *Expert Systems with Applications*, 164, 113992 (2021).
- [6] Ma, D., et al., SGNR: A Social Graph Neural Network Based Interactive Recommendation Scheme for E-Commerce. *Tsinghua Science and Technology*, 28(4):786-798 (2023).

- [7] Li, Z., et al., Hydrogel-elastomer-based stretchable strain sensor fabricated by a simple projection lithography method. *International Journal of Smart and nanomaterials*, 12(3),13 (2021).
- [8] Legger, G. E., et al., EULAR/PRES recommendations for vaccination of paediatric patients with autoimmune inflammatory rheumatic diseases: update 2021. *Annals of the Rheumatic Diseases*, 46(N):414-22 (2022).
- [9] Zhao, Z., et al., HetNERec: Heterogeneous network embedding based recommendation. *Knowledge-Based Systems*, 204(8),106218 (2020).
- [10] Abolghasemi, R., et al., A personality-aware group recommendation system based on pairwise preferences. *Information Sciences*, 595, 1-17 (2022).
- [11] Pan, Y., et al., Exploiting relational tag expansion for dynamic user profile in a tag-aware ranking recommender system. *Information Sciences*, 545(6):448-464 (2021).