Application of SVD Technology in Video Recommendation System

Menghan Yan, Wenqian Shang, Zhenzhong Li
Communication University of China
School of Computer Science
Beijing, China
e-mail:524942428@163.com

***Abstract*: The most direct access to evaluate what kinds of topics are valuable for video producers, and bring them inspiration is to seek subjects which specific groups concern currently. We can obtain massive user information from social networking platforms, large video sites and search engines, and then exploit the data to produce more practical works with the combination of business requirements. In views of the existing disadvantages of inferior scalability, sparsity problem and huge volume test data, the application of Singular Value Decomposition Method(SVD) actualize the unknown prediction score function of set of tests. The simulation results show that scalability, sparsity and omputational efficiency improved effectively.**

Keywords: SVD algorithm; collaborative filtering; video recommendation; matrix decomposition

I. INTRODUCTION

There are massive successful cases of big data technology applied in video industries [1]. In pre period of production cycle, there will be different processes of data mining [2] and data analysis. First, using online access to achieve vast amount of video information; second, getting specific subjects users are interested in through analyzing substantial records already got, such as themes, actors and songs of these resources. Finally, creators produce videos only depend on the analysis of the obtained information to achieve huge video views. One characteristic of current stage of China's search engine is personalized, which is based on user based recommendation, project recommendation, content based recommendation [3]. Collaborative filtering algorithm mentioned above have the problems of poor scalability and user similarity is hard to distinguish. With the development of science and technology, increasing experts and scholars begin to focus on Recommender System [4][5], and propose targeted enhancement being aimed at diversified problems traditional algorithms have. For instance, the expansion of difference and sparse data problems can be improved through the matrix decomposition method [6], such as Principal Component Analysis (PCA) , non- Negative Matrix Factorization(NMF) and Singular Value Decomposition(SVD) algorithm [7][8] can be used to reduce data sparsity and dimensions.

II. THE TRADITIONAL COLLABORATIVE FILTERING ALGORITHM

To solve the relationship between users and resources is the main content in recommender systems. m*n is defined as the relationship between users and resources, in which m set to be user, n set to be program. $R_{i,j}$ set to be the results after user i grade program j.

$$R = \begin{bmatrix} R_{1,1}...R_{1,j}...R_{1,n} \\ ... \\ R_{i,1}...R_{i,j}...R_{i,n} \\ ... \\ R_{m,1}...R_{m,j}...R_{m,n} \end{bmatrix} \quad (1)$$

By seeking the recent leadership for the targeted users, the traditional collaborative filtering algorithm obtains each targeted item score, and then actualize auto recommendation. This kind of recommendation is based on users` current interests, and can effectively explore network resources people be fond of. But due to the limitation of the algorithm, there are also problems such as bad scalability and sparsity exist. The sparsity mainly refers to that the system is hard to find suitable users in the case of deficiency of evaluation. The scalability refers to that with resources and users increasing, both system performance and quality will be more poor.

The score can be expressed as:

$$P_{\mathrm{u,j}} = \frac{\sum_{m=1}^{\mathrm{n}} sim(u,m) \bullet R_{m,j}}{\sum_{\mathrm{m=1}}^{\mathrm{n}} \mathrm{sim(u,m)}} \quad (2)$$

u presents user;

j presents program;

978-1-5090-0806-3/16/$31.00 copyright 2016 IEEE
ICIS 2016, June 26-29, 2016, Okayama, Japan

$R_{m,j}$ presents the results after m grade j;

sim(u,m) presents the Similarity between u and m.

III. Research on SVD Algorithm

SVD (Singular Value Decomposition)is one that should be a part of the standard mathematics and besides being rather intuitive, these decompositions are incredibly useful, which undergraduate curriculum but all too often slips between the cracks. It provides a convenient way for breaking a matrix, which perhaps contains some data some video companies are interested in, into pieces meaningful and simple .

A. The Principle of SVD Algorithm

- Score prediction can be simply seemed as filling blank elements of sparse matrices. However, filling method requires to bringing minimum disruption for rating matrices, namely after filling matrix the characteristic values have little difference with original figures. SVD algorithm can coincidentally meet this demand, using dimension reduction to complete the sparse rating matrices.
- Video recommendation system can handle huge data after users grade diversified videos and movies stored in a matrix form. Score prediction problem can be simply seemed as filling the blank elements in the sparse matrix.
- The average score of each list of item is substitute for the corresponding blank elements of the sparse matrix, in which way the system can get a dense matrix. Supposing that R is a normal m*n sparse rating matrix, after filling the blank elements, R turns to R`. We can get the Singular Value Decomposition of R` as follow:

$$R^{\sim} = U \sum V^{T} \qquad (3)$$

- As shown above, a normal matrix can be decomposed into three different matrix. Why do this? The number of original resources we obtain from social websites is rather formidable in the general case. Apply three matrix to describe one matrix provides a easy way to analyze the data.

B. Specific Steps of SVD

For a normal square matrix, characteristic values are accessable to describe features of the known matrix, in another words, however, characteristic values can not represent non-square matrix. For example, there are N users, each user has respective marks toward M movies, therefore a N*M matrix is used to describe scores which can not be square.

Supposing R is a M*N matrix, using the formula (3) mentioned above to describe R. U is a M*M square matrix, of which the vectors are orthotropic, called Left Singular Vector. Σ is a M*N matrix. The elements in the diagonal line of Σ is called singular values, and all the other elements are 0. V^T is a N*N square matrix of which the vectors are also orthotropic called Right Singular Vector. We can directly perceive the contact between the original matrix and the left three matrix through the senses.

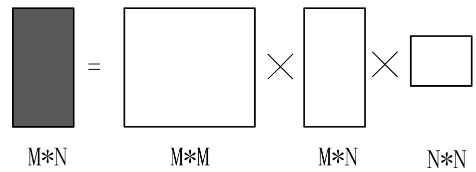

Figure 1

- Supposing R is the original matrix we get, taking the transpose of matrix R which finally turns into R^T. One can make a square matrix by multiplying R and R^T. We can work out the characteristic values of square matrix R* R^T. According to the formula as following

$$(A^T A)\vec{v}_i = \lambda_i \vec{v}_i \qquad (4)$$

can get eigenvalues and eigenvectors of $(A^T A)$.

- The vector $\vec{v}_i$ is the Right Singular Vector mentioned above, besides that, we can also obtain the singular value $\sigma_i = \sqrt{\lambda_i}$, the Left Singular Value $u_i = \frac{1}{\sigma_i} A\vec{v}_i$.
- The effect towards the entire matrix of singular value σ_i is similar with the eigenvalue , both of them provide a direct description of specific matrix. The singular values stored in the matrix Σ fall in a sharp speed, and consequently, in many cases, the first 10%, even 1% singular values occupy almost 99% of the sum of all the values. That is to say that we can use several of the largest values to describe the entire matrix approximately.

$$R_{m\times n} \approx U_{m\times r} \sum{}_{r\times r} V^T{}_{r\times n} \qquad (5)$$

The value of r is far from m and n, so the Multiplication looks like this:

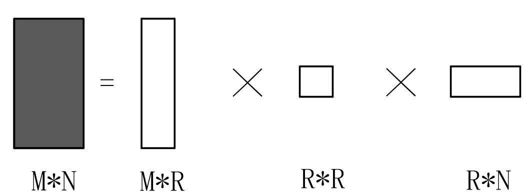

Figure 2

- The multiplication result of three of the right matrix is close enough to the original matrix. The closer r ranges to n, the closer the result ranges to R. The requirement of storage will drop down rapidly with the area of decomposed matrix decreasing. If we want to indicate primary matrix with a compassed storage space , just use only matrix U、Σ、V.

IV. The Application of SVD in Video Recommendation System

The conception of SVD was first applied in message retrieval area, in which the original matrix was complete can be decomposed directly with the method of SVD. In the recommendation system, however, it is hard to decompose the data with the premise of the untreated matrix is incomplete, even using the way of filling the blank elements with average value. Applying SVD in the traditional Collaborative Filtering Algorithm can seemed as an optimisation problem.

A. The Feature Value to Describe Characteristic

Supposing there is no direct connection between movies and users. Defining one dimension called feature to profile some properties of what we want to research, for instance, defining a movie as tragedy or comedy, action movie or love film. Users have a straight-forward relevance with the feature. Some like watching action movies and the others are fond of love films. And there are also relevance between movies and features.

We can decompose a rating matrix,rating=m*n, into a product of two matrix including user_feature and $(item\_feature)^{T.}$

*1) rating=m*n=user_feature*(item_feature)^T*

a) m represents the number of users, n represents the number of movies

*b) user_feature is a m*k matrix, among which k represents the dimension of the feature that can be defined optionally.*

*c) item_feature is a n*k matrix.*

The duty we need to accomplish is to find every element of both matrix, bringing the multiplication of them near to the original matrix as closer as they can.

2) The evaluation formula (6) :

$$E = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{m} I_{ij}(V_{ij} - p(U_i, M_j))^2 + \frac{k_u}{2}\sum_{i=1}^{n}\|Ui\|^2 + \frac{k_m}{2}\sum_{j=1}^{m}\|Mj\|^2$$

a) n represents the number of users. m represents the number of videos. I_{ij} represents weather user i has made the score ,which is defined as V_{ij}. $P(u_i,m_i)$ represents the prediction user i make concerning about video j.

b) We can use the method of Stochastic Gradient Descent to receive a mimimum E:

$$-\frac{\partial E}{\partial U_i} = \sum_{j=1}^{m} I_{ij}((V_{ij} - p(U_i, M_j))M_j) - k_u U_i, i = 1,\ldots,n \quad (7)$$

$$-\frac{\partial E}{\partial M_j} = \sum_{j=1}^{n} I_{ij}((V_{ij} - p(U_i, M_j))U_i) - k_m M_j, j = 1,\ldots,m \quad (8)$$

c) The technological process of the algorethm is as following:

① Compute gradients of matrices U,M

② Set $U \leftarrow U - \mu\nabla U and \nabla M by$

B. Optimize the effect of recommendation algorithm by using SVD

The real data set is more sparse compares with the condition that we expect. Figure 3 is showing a more specific rating matrix.

	movie_1	movie_2	movie_3	movie_4	movie_5	movie_6	movie_7	movie_8
user_1	2	5	4	3	3	2	5	3
user_2	1	4	4	3	3	3	4	4
user_3	4	4	2	1	1	4	4	2
user_4	3	3	3	2	5	5	2	2
user_5	4	2	2	3	5	3	3	1
user_6	4	3	2	3	5	2	2	1
user_7	3	1	3	1	2	5	3	4
user_8	2	5	4	4	1	2	3	2
user_9	4	3	3	4	3	3	4	4
user_10	5	3	3	4	2	2	5	5

Figure 3

We put the entire matrix into the program then calculate the singular value decomposition of the matrix to know how many dimensional characteristics it has.

27.8313	0	0	0	0	0	0	0
0	6.2845	0	0	0	0	0	0
0	0	5.0758	0	0	0	0	0
0	0	0	4.2431	0	0	0	0
0	0	0	0	3.3956	0	0	0
0	0	0	0	0	2.0466	0	0
0	0	0	0	0	0	1.0050	0
0	0	0	0	0	0	0	0.6531
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Figure 4

The next step is to find how many singular values of the diagonal matrix should be retained in R', of which the sum up to more than 90% of total. We call each element in the diagonal line δ, and δ_2 is δ^2. The value of δ_2 is the sum of each δ^2, which in this case, is 325.1461. The 90% of this value is 292.6315 and the energy first 2 elements include is 313.7823.

Finally we can turn a 8-D matrix into a 3-D matrix in this case, because the first three elements include even more than 90% of the total energy. Using SVD to mapping the whole videos in a lower dimensional space, where be able to applying similarity calculation method to the recommendation algorithm value is 292.6315 and the energy first 2 elements include is 313.7823.

user_1	-0.3471	-0.303	0.2755
user_2	-0.3302	-0.2546	0.0595
user_3	-0.286	-0.059	-0.3071
user_4	-0.3153	0.4618	-0.0306
user_5	-0.292	0.4714	0.1889
user_6	-0.2786	0.4178	0.395
user_7	-0.2782	0.1089	-0.6916
user_8	-0.2937	-0.3896	0.3426
user_9	-0.355	-0.0446	-0.0664
user_10	-0.3698	-0.2533	-0.183

Figure 5

movie_1	-0.3625	-0.3768	-0.3432
movie_2	0.3101	-0.3055	-0.2084
movie_3	-0.1859	0.4199	0.0688
movie_4	-0.6296	0.314	0.3286
movie_5	0.4408	0.4912	-0.1936
movie_6	-0.1625	0.0324	-0.2455
movie_7	0.1514	0.4722	-0.2179
movie_8	-0.3128	0.1562	-0.7621

Figure 6

The left singular vector as shown in Figure 5 presents some main characteristics of users, on the contrary, the right singular vector below shows the features of movies. The first column of the left and right singular vector represents relatively every word's frequency.

C. The analysis of the result

In Figure 7 every blue star presents one user whose statistics had been exploited in the SVD process, and on the contrary green starts represents details of all the movies. In this way we can conduct cluster analysis of these key words. For example,user_1、user_2 and user_8 can be seen as similar consumers who can be recommended movies based on two other people for reasons that they always give the same file an analogous score.

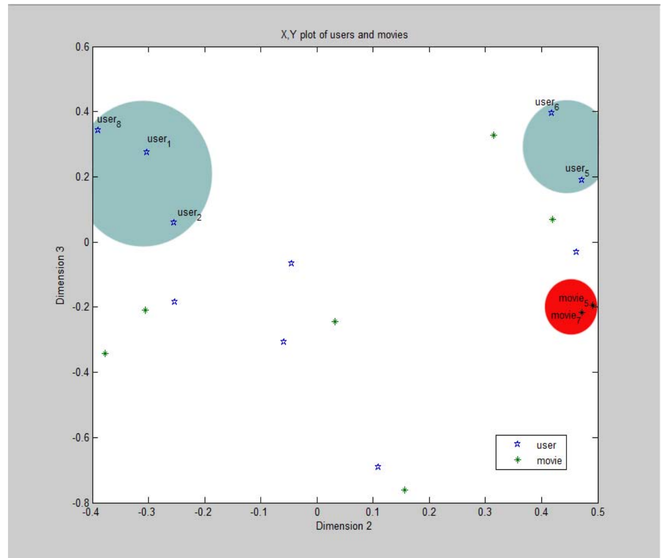

Figure 7

V. Conclusion

Score prediction can be simply seemed as filling blank elements of sparse matrices. SVD provides a brand new train of thought towards how to clean the original figures and in what way finding the most similar items and users in the recommender system. Video recommender system can handle huge data after users grade diversified videos and movies finally stored in a matrix form. By seeking the recent leadership for the targeted users, the traditional collaborative filtering algorithm obtains each targeted item score, and then actualize auto recommendation. But SVD method is working through choosing limited features of the entire data, eliminating the other "noisy" information.

Acknowledgment

This paper is partly supported by "the Excellent Young Teachers Training Project (the second level, Project number: YXJS201508)", "Key Cultivation Engineering Project of Communication University of China (Project number: 3132016XNG1606 and 3132016XNG1608)", "Cultural technological innovation project of Ministry of Culture of P.R.China (Project number: 2014-12)", and partly supported by "The comprehensive reform project of computer science and technology, department of science and Engineering". The research work was also supported by "Chaoyang District Science and Technology Project (CYXC1504)".

References

[1] J. Shlens. A Tutorial on Principal Component Analysis. Center for Neural Science,New York University,Salk Institute for Biological Studies La Jolla, CA 92037 arXiv:1404.1000[cs.LG]

[2] J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

[3] I. S. Jacobs and C. P. Bean, "Fine particles, thin films and exchange anisotropy," in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.

[4] K. Elissa, “Title of paper if known,” unpublished.

[5] R. Nicole, “Title of paper with only first word capitalized,” J. Name Stand. Abbrev., in press.

[6] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, “Electron spectroscopy studies on magneto-optical media and plastic substrate interface,” IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].

[7] M. Young, The Technical Writer’s Handbook. Mill Valley, CA: University Science, 1989.

[8] Thomas H.Cormen, Charles E.Leiserson, Ronald L.Rivest, Clifford Stein. Introduction to Algorithm 3th ed. 2013

[9] Qi Guo,Bo-Wei Chen, Feng Jiang, Xiangyang Ji, Sun-Yuan Kung. Efficient Divide-And-Conquer Classification Based on Feature-Space Decomposition. ArViv:1501.07584[cs.LG] Thu.29 Jan 2015

[10] S.Todd IBM United Kingdom Scientific Centre,Neville Road,Peterlee,
Durham SR8 IBY,England,UK. Algorithm and Hardware for a Merge Sort Using Multiple Processors. IBM Journal of Research and Development (vol:22,Issue:5) Sept.1978

[11] Gregory J.Chaitin. Randomness and Mathematical Proof. Scientific American 232,No 5(May 1975) pp. 47-52

[12] Sultanullah Jadoon,Salman Faiz Solehria. Optimized Selection Sort Algorithm is faster than Insertion Sort Algorithm:a Comparative Study.
International Journal of Electrical & Computer Sciences IJECS-IJENS
Vol:11 No:02.