Enhanced Higher Order Orthogonal Iteration Algorithm for Student Performance Prediction

Prema Nedungadi and T.K. Smruthy

Abstract Predicting Student Performance is the process that predicts the successful completion of a task by a student. Such systems may be modeled using a three-mode tensor where the three entities are user, skill, and task. Recommendation systems have been implemented using Dimensionality reduction techniques like Higher Order Singular Value Decomposition (HOSVD) combined with Kernel smoothing techniques to bring out good results. Higher Order Orthogonal Iteration (HOOI) algorithms have also been used in recommendation systems to bring out the relationship between the three entities, but the prediction results would be largely affected by the sparseness in the tensor model. In this paper, we propose a generic enhancement to HOOI algorithm by combining it with Kernel smoothing techniques. We perform an experimental comparison of the three techniques using an ITS dataset and show that our proposed method improves the prediction for larger datasets.

Keywords Recommendation systems · HOSVD · Higher order orthogonal iteration algorithm · Kernel smoothing · Tensors

1 Introduction

Task recommendation is the process by which new tasks are recommended to users by predicting student performance. There are three main entities in modeling a task recommendation system namely user, skill, and task. Traditional recommender systems do not consider these three features together [1]. They use techniques like

P. Nedungadi (✉)
Amrita CREATE, Amrita Vishwa Vidyapeetham, Kollam, Kerala, India
e-mail: prema@amrita.edu

T.K. Smruthy
Department of Computer Science, Amrita Vishwa Vidyapeetham, Kollam, Kerala, India
e-mail: smruthykrishnan@gmail.com

S.C. Satapathy et al. (eds.), *Proceedings of the Second International Conference on Computer and Communication Technologies*, Advances in Intelligent Systems and Computing 379, DOI 10.1007/978-81-322-2517-1_61

Collaborative Filtering (CF) applied to only two-dimensional data, users, and items. In systems modeled using all the three features, they were split into pair relations. Thus, the total interactions between the three features were missing. Later, recommendation systems were developed modeling the three entities using multi-way arrays called tensors [2]. Decomposition techniques like Higher Order Singular Value Decomposition (HOSVD) were applied on the tensors. Truncation of the factor matrices obtained as a result of HOSVD to find the low-rank approximation tensor proved sub-optimal in terms of the Frobenius norm [3]. HOSVD was combined with Kernel Smoothing Techniques, and these were shown to give more accurate results. Also, Higher Order Orthogonal Iteration (HOOI) algorithms were proposed to improvise on low-rank approximation [4] but sparseness affected their performance. Thus, Kernel smoothing was used which resulted in the mapping of the data into higher dimensions, and thereby, the data could be better separated and structured. In this paper, we aim to recommend tasks to users combining both the Higher Order Orthogonal Iterative procedure with the Kernel Smoothing Technique. The combination has proved to give out more accurate recommendations to users than the separate models. Experimental evaluation was performed using real-time datasets, and the proposed approach outperformed the previous models.

2 Recommendation Systems

2.1 Motivation

Students need to look for guidance from teachers and others when it comes to choosing tasks they are likely to solve. In order to cater to this need, many information and recommendation strategies have been developed. Task Recommendation systems are one among these that recommend tasks with high chances of successful completion to each student. There are three main entities in developing such systems: user, skill, and task. Recommendation system aims to derive a quadruplet (u, s, t, p) that provides a likeliness measure of interaction between the three features. 'p' measures the likeliness of the user 'u' with skill 's' solving task 't.' Thus, such a system needs to be modeled considering the interaction among the three dimensions together. Also, modeling such systems will result in sparseness. So we need to develop algorithms to alleviate the effect of sparseness on the approximations.

2.2 Related Work

Breese et al. proposed a model based on collaborative filtering (CF) but the model failed to capture the relationship between the entities [1]. Later many matrix factorization model was proposed that overcame the limitations of the previous model, but it did not address user personalization [5]. The interrelationship between entities and user personalization was together addressed using CubeSVD algorithm and division algorithm [6]. Also, multi-task learning [7] methods for collaborative filtering was proposed, which built a multi-task regression model for rating prediction. Systems were proposed to include a third attribute for user personalization to standard CF algorithms reducing three-dimensional relations to two-dimensional relations and then later applying fusion algorithms to associate correlations [8]. But this model lacked accuracy. A comparison of three classes of algorithms namely CF adaptations based on projections, page rank algorithm, and some simpler methods showed that FolkRank algorithm performs better [9]. However, we needed a model that preserves the three-dimensional relation as such and generates recommendation. Later, data was modeled using tensors—multi-way arrays. Various tensor decomposition algorithms were developed for analyzing the high-dimensional structures, and the approximations drawn from these decompositions were analyzed to reach conclusions.

There are a lot of models addressing the various tensor decomposition algorithms [10, 11]. Analysis proved that only HOSVD provided a unique solution [12]. Later, a research on the user web interest was performed by analyzing the click data and modeling it as a third-order tensor [13]. This used the concept of HOSVD to perform tensor decomposition. The data modeled for the above-mentioned researches were highly sparse due to the presence of huge number of user, item, and contexts, but the mapping between each of these combinations was very few. HOSVD with kernel functions was combined to address sparse data and the combination proved to improvise the traditional HOSVD approach especially on sparse data [14]. Truncation was done on each of the factor matrices that were obtained as a result of performing SVD to find the low-rank approximations [3]. This provided a solution that was sub-optimal in terms of the Frobenius norm. So we need a method to improvise on the rank approximations. An empirical analysis of four tensor decomposition algorithms was performed, and Higher Order Orthogonal Iteration algorithm was recommended. The Tensor Toolbox was introduced for efficient MATLAB computations [15]. This explained various functions for computations with both sparse and dense tensors. A unified model for social tagging was introduced giving out all three types of recommendation [2]. This used HOSVD with kernel functions and this approach was better than the state-of-the art methods. HOSVD is limited in terms of Frobenius norm. HOOI also provides a good orthogonal decomposition but the solution is affected by data sparseness. So we go for a combined model that alleviates the sparseness and is optimized in terms of Frobenius norm.

2.3 General Block Diagram with Different Approaches

Figure 1 represents a general block diagram of all the approaches discussed. The initial construction of the tensor and the unfolding are common for the three

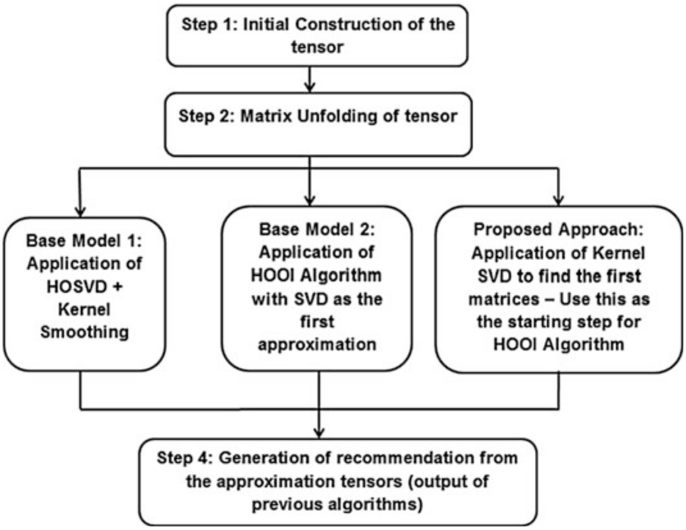


Fig. 1 General block diagram with different approaches

approaches. To these unfolding, algorithms are applied. The output of the algorithm is an approximation tensor from which recommendations can be drawn.

2.3.1 HOSVD with Kernel Smoothing Base Model 1

The Higher Order Singular Value Decomposition is a generalization of SVD computations to multidimensional data. Matrix unfolding is done to each of the modes of three-way tensor $\mathcal{A} \in R^{I_1 \times I_2 \times I_3}$, and SVD is applied to each of these modes to get the factor matrices U^1, U^2, U^3. HOSVD is performed by n-mode product of tensor A with the factor matrices.

$$A \approx \hat{A} = S \times_1 U^{(1)} \times_2 U^{(2)} \times_2 U^{(3)}. \tag{1}$$

where U^1, U^2, U^3 are the orthogonal vectors spanning the column space of the respective matrix unfoldings. S is called the core tensor and has the property of all orthogonality. The tensors modeled are sparse tensors, and this could affect the approximations performed. Hence, SVD is applied on Kernel-defined feature space. In order to avoid the explicit mapping of the data points of each tensor unfolding into higher dimensional space and thereafter applying Kernel functions, we follow the known process called the 'Kernel trick' and find the factor matrices of each unfolding using Eigen value decomposition.

2.3.2 Higher Order Orthogonal Iteration—Base Model 2

Higher Order Orthogonal Iteration is an iterative algorithm to compute better low-rank approximation to tensors than HOSVD. It successively solves the restricted optimization problem:

$$\min_{U^p} \| \mathcal{A} - \mathcal{B} \times_1 U^{(1)} \times_2 U^{(2)} \ldots\ldots \times_N U^{(N)} \|_F^2. \tag{2}$$

Optimization is done over the matrix U^p with the latest available values of other U^i.

3 Proposed Approach

3.1 *Outline*

Modeling the data using HOSVD suffers from a sub-optimal solution in terms of the Frobenius norm due to truncation of matrices to find the low-rank approximations. Frobenius norm is defined for a three-way tensor as $||\mathcal{A}||_F^2 = \Sigma |A_{ijk}|^2$. This can be improved by a Higher Order Orthogonal Iterative procedure by repeatedly optimizing the approximations. In the traditional HOOI procedure, the first step is to find the initial factor matrices by applying simple SVD over the different modes of the tensor followed by iteratively updating factor matrices. A sparse data can largely affect the accuracy of the iterative procedure. In the proposed approach, we combine the two models. Kernel functions are applied to find the initial factor matrices, and these are then updated iteratively using the usual known procedure. Combining the two models takes care of both the sparseness problem and the Frobenius norm. Figure 2 briefs the steps followed in the proposed approach. Step 3 is the modification step done to traditional HOOI Algorithm to alleviate the sparseness and improve accuracy.

3.2 *Example*

Figure 3a gives the information on tasks performed successfully by each user. Figure 3b details out information on the skillset needed to perform each task. The algorithm initially constructs the three-mode tensor from the available dataset considering the three entities: user, skill, and task. A ‘1’ in a cell A_{ijk} represents that user i with skill j can do task k. Figure 3 represents the tensor constructed for the example scenario represented by the table. Matrix unfolding is done as shown in Fig. 4.

We then move on to performing Higher Order Orthogonal Iteration Algorithm. The first step is to find the initial factor matrices by applying kernel functions that can nonlinearly map the data to higher dimension and thus lead to a better

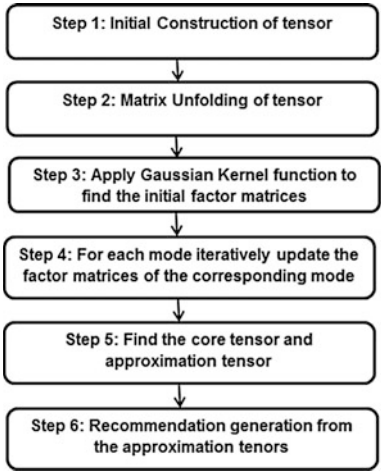


Fig. 2 Block diagram of proposed approach

(a)

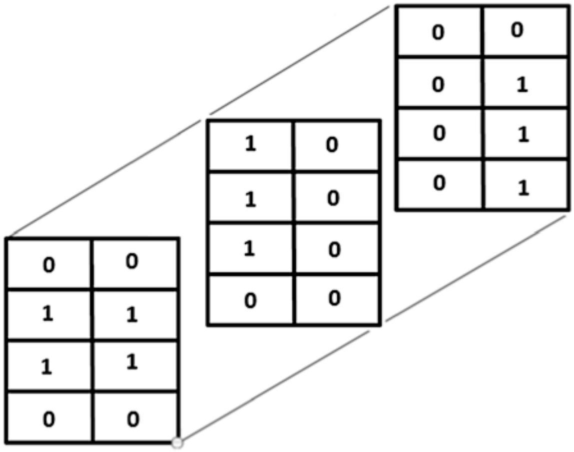
Users	Skills
U1	S1
U2	S1,S2
U3	S1,S2
U4	S2

(b)

Skill	Task
S1	T2
S2	T3
S1,S2	T1

Fig. 3 Tensor construction of the running example. **a** User-skill table. **b** Skill-task table

Fig. 4 Matrix unfolding of tensor



structuring of data. Thus, we alleviate the sparseness problem in the first step. With this as the input, we proceed to the iterative procedure to find the factor matrix of each mode using the latest available values of the other factor matrices while preserving the orthogonality as well. Gaussian Kernel functions were used taking the standard deviation of each matrix unfolding as the Gaussian parameter. A brief algorithm has been given in the Sect. 3.3.

The generation of approximation tensor marks the end of the training phase. The approximation tensor $\tilde{A}$ measures the interaction among the three features. In the testing phase, the final approximation tensor can be interpreted as a quadruplet (u, s, t, p) where p is the likeliness measure of user u with skill s performing task t. Thus, from the existing data, for a random user, skill and task checking the value of p in the approximation tensor will give the likeliness of the successful completion of the task by user.

3.3 Proposed Approach: Algorithm

```
Initialize a 3-mode tensor t with value 0.
for each activity a, in the dataset
  (u,s,t) = Indices in tensor corresponding to a.
  t[u][s][t] = 1
for n = 1 to N do
  Compute matrix unfolding $A_{(n)}$ for n = 1,2,3
for each matrix unfolding of tensor A
  Apply Gaussian Kernel Function
    $B_i = exp(- \| x - y \|^2 / 2\sigma^2)$
  $U^i = EigenVectorOf(B_i)$
for k = 0 to kMax do:
  for n = 1..3 do:
    $\tilde{U} = A \times_{-n} \{U_k^T\}$
    Let W of size $I_n \times R_n$ solve:
    max $||\tilde{U} \times_n W^T||$ subject to $WW^T = 1$
    $U_{k+1}^{(n)} = W$
Let $\{U\} = \{U_K\}$ where K is the final value of previous step.
Set $\tilde{S} = A \times \{U^T\}$
Set $\tilde{A} = \tilde{S} \times \{U\}$
end.
```

4 Testing and Analysis

The proposed approach has been implemented in the MATLAB, and the tensor toolbox version 2.5 [15, 17] has been used for efficient computation of algorithms. To test the proposed approach, we used three sets of data from the [16] Knowledge Discovery and Data Mining Challenge 2010. In all the three sets of dataset, only records with CFA = 1 have been considered as CFA = 0 can either indicate that the student has failed in the task or indicate a not attempted task.

4.1 Effectiveness of Kernel Smoothing Technique

To measure the effectiveness of using Kernel smoothing as the starting step of the HOOI algorithm, a small sample data was extracted from the dataset, and both the traditional HOOI algorithm with SVD as the starting step and our proposed approach were applied on it. Recommendation generations were drawn from this approximation tensor, and using different samples from the dataset, we have verified and proved that the accuracy of prediction of the proposed approach outperforms the other models. A comparison of the accuracy of the base models and the proposed approach has been detailed out in the Sect. 4.2. Although the dimensions of the factor matrices obtained as a result of applying SVD and kernel functions differ, the final structure of the approximation tensor obtained using both the approaches is the same. Also, the computational complexity of using kernel functions on one unfolded tensor is $O(n^3)$ where n is the number of training vectors and that of performing SVD is $m^2n + n^3$ for a $m \times n$ unfolded tensor. The time complexity of various approaches discussed in this paper has been analyzed in the following sections.

4.2 Results

The dataset considered was partitioned into 80 % as training data and 20 % as test data. The Root-Mean Square Error (RMSE) and accuracy were calculated for the proposed approach as well as the base models. The proposed approach proved better in terms of accuracy (Fig. 6). Also, the accuracy of the proposed approach was found to increase with increase in dataset size. The proposed approach proved optimal in terms of the RMSE (Fig. 5). Although there was an increase in training time for the proposed approach, the recommendation generation can be done in real time as is the case with other approaches.

Figures 7 and 8 give a comparison of the time taken for training and recommendation generation for both the base models and the proposed approach.

Fig. 5 RMSE

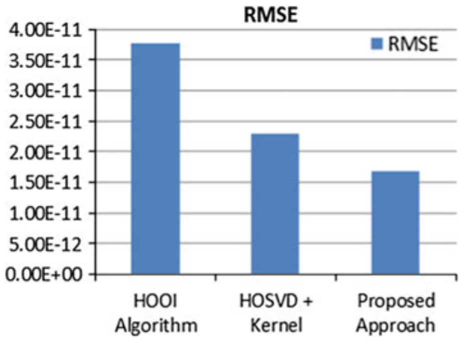


Fig. 6 Accuracy

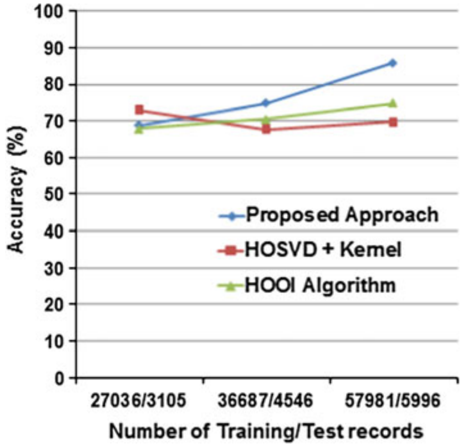
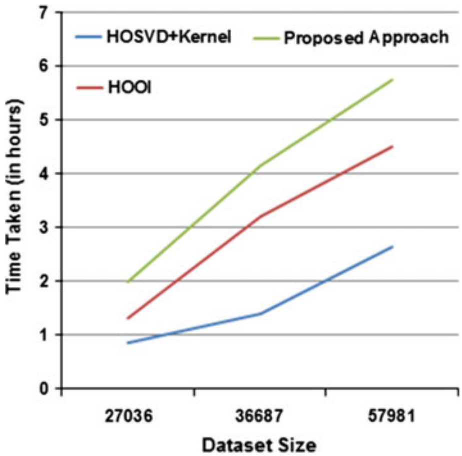


Fig. 7 Time taken for training



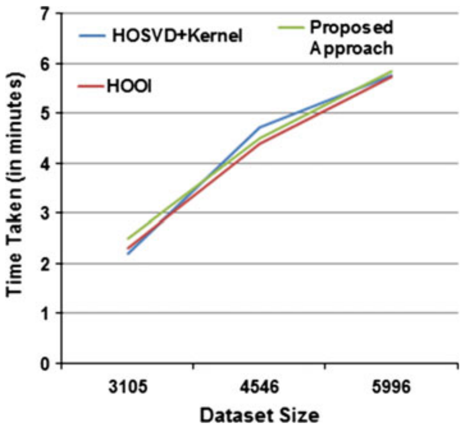


Fig. 8 Time taken for testing

5 Conclusion

The three main entities of a task recommendation system are user, skill, and task. The main idea was to consider the three entities together without splitting them. HOSVD proves sub-optimal in terms of the Frobenius norm due to the truncation of the factor matrices to find low-rank approximations. HOSVD also suffers from sparse data. Hence, it is difficult to separate the data and find a structuring of data. Thus, kernel smoothing was done to alleviate this sparseness. Higher Order Orthogonal Iterative procedure improved the accuracy of HOSVD by improvising on the low-rank approximations iteratively. But HOOI does not perform well in case of sparse tensors. Thus, we modified the starting step of the HOOI procedure in finding the first factor matrices by first projecting the data points into higher dimensional space and then iteratively finding the factor matrices. This gave an optimal solution in terms of the Frobenius norm as well as alleviated the sparseness in the original tensor constructed. Our test with over 50,000 log data records has proved that the proposed modification to HOOI algorithm is efficient than the traditional approaches.

Higher Order Orthogonal Iteration algorithm is an iterative SVD process where the resulting approximation tensors comprise negative values too. It is very difficult to interpret these negative values. Adopting a non-negative tensor factorization may lead to a better prediction.

Acknowledgment This work derives direction and inspiration from the Chancellor of Amrita University, Sri Mata Amritanandamayi Devi. We would like to thank Dr. M. Ramachandra Kaimal, Head, Department of Computer Science and Dr. Bhadrachalam Chitturi, Associate Professor, Department of Computer Science, Amrita University for their valuable feedback.

References

1. Breese, J.S., Heckerman, D., Kadie, C.: Empirical analysis of predictive algorithms for collaborative filtering. In: Proceedings of 14th Conference Uncertainty in Artificial Intelligence, p. 4352. Morgan Kaufmann (1998)
2. Symeonidis, P., Nanopoulos, A., Manolopoulos, Y.: A unified framework for providing recommendations in social tagging systems based on ternary semantic analysis. IEEE Trans. Knowl. Data Eng. **22**(2), (2010)
3. Ishteva, M., Absil, P.-A., Van Huffel, S., De Lathauwer, L.: Best Low multilinear rank approximation of higher-order tensors, based on the riemannian trust-region scheme. SIAM. J. Matrix Anal. Appl. **32**(1), 115–135
4. Turney, P.D.: Empirical evaluation of four tensor decomposition algorithms. In: Technical Report ERB-1152, NRC-49877, November 12 2007
5. Koren, Y., Bell, R., Volinsky, C.: Matrix factorization techniques for recommender systems. Computer **42**(8), 3037 (2009)
6. Xu, Y., Zhang, L., Liu, W.: Cubic analysis of social bookmarking for personalized recommendation. In: Frontiers of WWW Research and DevelopmentAPWeb 06, pp. 733–738. Springer (2006)
7. Ning, X., Karypis, G.: Multi-task Learning for recommender Systems. In: JMLR: Workshop and Conference Proceedings, vol. 13, pp. 269–284 2nd Asian Conference on Machine Learning (ACML 2010), Tokyo, Japan, Nov. 810 2010
8. Tso-Sutter, K., Marinho, B., Schmidt-Thieme, L.: Tag-aware recommender systems by fusion of collaborative filtering algorithms. In: Proceedings of ACM Symposium Applied Computing (SAC) Conference (2008)
9. Jäschke, R., Marinho, L., Hotho, A., Schmidt-Thieme, L., Stumme, G.: Tag recommendations in social bookmarking systems. In: AI Communications, p. 231247 (2008)
10. Kolda, T.G., Bader, B.W.: Tensor Decompositions Appl. SIAM Rev. **51**(3), 455500 (46 p)
11. Ishteva, M., De Lathauwer, L., Absil, P.: Dimensionality reduction for higher-order tensors: algorithms and applications. Int. J. Pure Appl. Math. **42**, 337–343 (2008)
12. Luo, D., Huang, H., Ding, C.: Are tensor decomposition solutions unique? In: On the Global Convergence of HOSVD and ParaFac Algorithms. CVPR2009
13. Sun, J.-T., Zeng, H.-J., Liu, H., Lu, Y., Chen, Z.: CubeSVD: a novel approach to personalized web search. In: Proceedings WWW '05 Proceedings of the 14th International Conference on World Wide Web, pp. 382–390 ACM New York, NY, USA (2005)
14. Nanopoulos, A., Krohn-Grimberghe, A.: Recommending in social tagging systems based on kernelized multiway analysis. In: Proceedings of the 11th IFCS Biennial Conference and 33rd Annual Conference of the Gesellschaft fÃ¼r Klassifikation e.V., Dresden, March 13–18 2009
15. Bader, B.W., Kolda, T.G.: Efficient MATLAB computations with sparse and factored tensors. SIAM J. Sci. Comput. **30**(1), 205231
16. Stamper, J., Niculescu-Mizil, A., Ritter, S., Gordon, G.J., Koedinger, K.R.: Algebra 2005–06. In: Challenge Dataset from KDD Cup 2010 Educational Data Mining Challenge (2010)
17. Bader, B.W., Kolda, T.G., et al.: MATLAB Tensor Toolbox Version 2.5, Available online, January 2012