NMF-based DCG Optimization for Collaborative Ranking on Recommendation Systems

Noor Ifada, Dziyaur Rohman Miftah Alim, Mochammad Kautsar Sophan
Informatics Department
University of Trunojoyo Madura, Bangkalan, Indonesia
noor.ifada@trunojoyo.ac.id, 27dziyaur@gmail.com, kautsar@trunojoyo.ac.id

ABSTRACT
A recommendation system predicts a top-N list of items that a target user might like by considering the user's previous rating history. In this paper, we solve the task of recommendation by developing a method that implements an NMF-based DCG optimization for collaborative ranking. Three main processes are applied to calculate the rating prediction for making the list of top-N item recommendations: constructing the user profile, initialising the latent-factor models using NMF (Non-Negative Matrix Factorization), and further optimising the models based on the DCG (Discounted Cumulative Gain). Extensive evaluations show that our proposed method beats all baseline methods on both the Precision and NDCG metrics. This fact confirms that NMF-based DCG optimization is an effective approach to enhance the recommendation performance and to deal with the sparsity problem.

Keywords
Collaborative Filtering; DCG; NMF; Optimization; Ranking; Recommendation.

1. INTRODUCTION
Recommendation systems (RS) help users to find items suit their preference by generating a set of a personalized list of recommendations that might be of interest its users by learning through their previous rating activities [1, 2]. A very popular learning approach in RS is Collaborative Filtering (CF) [1-3], in which the latent factor models perform the best [2]. One of the well-known methods of the latent factor models is the NMF (Non-Negative Matrix Factorization) [4] that implements a low-rank approximation technique. Meanwhile, CF is known to commonly suffer from a sparsity issue that impacts the recommendation performance [3, 5-7].

Given the list of item recommendations, users tend to select fewer top-N items than those further down the list [8]. For that reason, the task of recommendation can be formulated as a ranking task and employing an optimization approach based on the evaluation metric would generate a quality top-N list of recommendations [8, 9]. One of the metrics commonly used for evaluating the quality of a ranking model is the DCG (Discounted Cumulative Gain) [9].

In this paper, we develop a top-N item recommendation method that implements the NMF-based DCG optimization for collaborative ranking. Experiment results using a rating dataset show that our method beats all of the baseline methods. This finding confirms that our proposed optimization is effective to enhance the performance of recommendation and to tackle the sparsity issue.

The summary of our contributions is as follows: (1) a novel top-N item recommendation method that implements NMF-based DCG Optimization for collaborative ranking, and (2) a concise observation that details how does our method perform matched to the baseline CF methods for generating top- N item recommendations.

The remains of this paper are presented as follows. Section 2 reviews the works related to this study. Section 3 presents the proposed NMF-based DCG optimization for collaborative ranking on recommendation systems. Section 4 shows the series of experimentations and discussion. Lastly, Section 5 describes the conclusion.

2. RELATED WORK
The Collaborative Filtering (CF) approach can be modelled as the memory-based or model-based approach [2]. The memory-based is a CF approach that generates the list of recommendations by employing the similarities of users or items [2]. Meanwhile, the model-based is a CF approach that generates recommendations by developing a model for learning a pattern based on the training data [2]. Examples of techniques for developing the model-based methods are the Bayesian [10], clustering [11-13], and latent factor models [14-18]. The latent factor model is superior compared to others [2], and therefore, we apply the technique in this paper.

Well-known approaches of the latent factor models are the PCA (Principal Component Analysis), SVD (Singular Value Decomposition), NMF (Non-Negative Matrix Factorization), and PMF (Probabilistic Matrix Factorization) [2]. The NMF-based methods are very popular and have been widely applied in various areas [4]. For this reason, our study applies the NMF-based approach for the initialization of the latent factor models.

The task of a recommendation system that generates a top-N list of recommendations can be considered as a ranking problem, in which employing an evaluation metric based optimization is beneficial to the RS method [8, 9]. This paper employs DCG, a ranking model evaluation metric [9], as the optimization criteria.

3. NMF-BASED DCG OPTIMIZATION FOR COLLABORATIVE RANKING METHOD
Our proposed NMF-based DCG Optimization for Collaborative Ranking method, i.e., *DO-NMF*, is focusing on a ranked item recommendation system from rating data. To yield the rating prediction calculation and eventually generating the list of top-N item recommendations, the framework of the *DO-NMF* is decomposed into three main processes: (1) constructing the user profile, (2) initializing the latent-factor matrices using NMF (Non-Negative Matrix Factorization), and (3) further optimizing the latent factor models based on the DCG (Discounted Cumulative Gain).

3.1 User Profile Construction

The user profile is constructed from the rating data. Let $U = \{u_1, u_2, u_3, \dots, u_m\}$ be the set of m users and $I = \{i_1, i_2, i_3, \dots, i_n\}$ be the set of n items. The rating data records the preference score of a user u towards item i that can be represented as a rating matrix $R \in \mathbb{R}^{m\times n}$ in which $r_{u,i}$ is the rating of item i set by user u. Table 1 shows the toy example of a rating matrix $R \in \mathbb{R}^{3\times 4}$ where $U = \{u_1, u_2, u_3\}$ and $I = \{i_1, i_2, i_3, i_4\}$.

Table 1. A toy example of rating matrix $R \in \mathbb{Z}^{3\times 4}$

		Item			
		i_1	i_2	i_3	i_4
User	u_1	5	0	3	0
	u_2	1	4	2	0
	u_3	0	0	4	1

The NMF is a low-rank F approach that factorizes a non-negative matrix $A \in \mathbb{R}^{m\times n}$ into two latent factor matrices $P \in \mathbb{R}^{m\times F}$ and $Q \in \mathbb{R}^{n\times F}$ having only non-negative elements [4, 19] such that:

$$A \approx PQ^T \tag{1}$$

In this paper, as the initialization procedure [20] of *DO-NMF*, we apply NMF to the rating matrix R to generate the initial latent factor models of users $\widehat{U}$ and items $\hat{I}$. These models are later used in the DCG optimization presented in the following subsection.

3.2 DCG Optimization

Using DCG optimization for collaborative ranking means that the task of recommendation is to predict the finest list of items based on the DCG perspective. In this problem, the objective function is built based on the formulation of a DCG score. DCG score is calculated from a gain numerator function that weighs up the score based on the rating of an item given by the user and a discount denominator function that penalize the score based on the item's ranking position in top-N recommendations [9, 20]. The DCG score of a user u over all items $i \in I$ is formulated as:

$$DCG_u = \sum_{i\in I} \frac{2^{r_{u,i}} - 1}{\log_2(1 + p_{u,i})} \tag{2}$$

where $p_{u,i}$ is the item i ranking position for a user u. The DCG defines the score for all users as:

$$DCG = \frac{1}{m}\sum_{u\in U}\sum_{i\in I} \frac{2^{r_{u,i}} - 1}{\log_2(1 + p_{u,i})} \tag{3}$$

Following the approach of ranking model smoothed approximation [21, 22], we approximate the ranking position $p_{u,i}$ as:

$$p_{u,i} \approx 1 + \sum_{j\neq i} \sigma(\Delta\hat{r}) \tag{4}$$

where $\sigma(x)$ is the logistic function $\frac{1}{1+e^{-x}}$ and $\Delta\hat{r} = \hat{r}_{u,i} - \hat{r}_{u,j}$ is the difference of rating prediction of two items i and j calculated from a factorization technique predictor function model. In this paper, *DO-NMF* uses the CP model [23] as the rating matrix factorization technique.

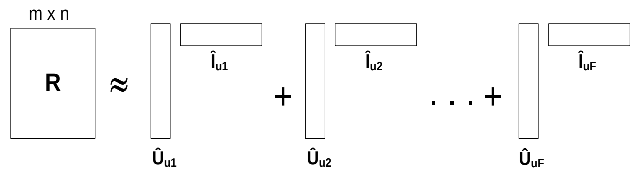

Figure 1. CP matrix factorization model.

As illustrated in Figure 1, CP factorizes a matrix $R \in \mathbb{R}^{m\times n}$ into a sum of latent factor rank-one of $\widehat{U}_{u_f} \in \mathbb{R}^m$ and $\hat{I}_{i_f} \in \mathbb{R}^n$ for $f = 1, \dots, F$, where F is the rank of the conforming latent factor models. The rating prediction is calculated as:

$$\hat{r}_{u,i} := \sum_{f=1}^{F} \widehat{U}_{u,f} \cdot \hat{I}_{i,f} \tag{5}$$

Based on Equation (4) and (5), the Smoothed DCG score for all users is now formulated as:

$$SDCG = \frac{1}{m}\sum_{u\in U}\sum_{i\in I} \frac{2^{r_{u,i}} - 1}{1 + \log_2\left(\sum_{j\neq i}\sigma(\Delta\hat{r})\right)} \tag{6}$$

The objective function is then formulated as:

$$L(\Theta) = \sum_{u\in U}\sum_{i\in I} \frac{2^{r_{u,i}} - 1}{1 + \log_2\left(\sum_{j\neq i}\sigma(\Delta\hat{r})\right)} - \lambda_\Theta \|\Theta\|_F^2 \tag{7}$$

where λ_Θ is the regularization coefficient that corresponds to σ_Θ as the model parameters used to control overfitting. We neglect the coefficient $\frac{1}{m}$ since it remains constant and, thus, it does not influence the optimization process.

The gradient descent is performed for objective function optimization:

$$\frac{\partial L}{\partial \theta} = \sum_{u\in U}\sum_{i\in I} \frac{-\varphi\left[\frac{1}{(\ln 2\cdot\beta)}\left(\sum_{j\neq i}(-\gamma+\gamma^2)\cdot\left(\frac{\partial(\Delta\hat{r})}{\partial\theta}\right)\right)\right]}{[1+\log_2\beta]^2} - \lambda_\theta\theta \tag{8}$$

where $\varphi = 2^{r_{u,i}} - 1$, $\gamma = \sigma(\Delta\hat{r})$, and $\beta = \sum_{j\neq i}\sigma(\Delta\hat{r})$.

Given a case (u, i, j) that corresponds to the model parameter $= \{\widehat{U}_u, \hat{I}_i, \hat{I}_j\}$, the gradients for the model based on their latent factor matrices are as follows:

$$\frac{\partial L}{\widehat{U}_u} = \sum_{u\in U}\sum_{i\in I} \frac{-\varphi\left[\frac{1}{(\ln 2\cdot\beta)}\left(\sum_{j\neq i}(-\gamma+\gamma^2)\cdot(\hat{I}_i - \hat{I}_j)\right)\right]}{[1+\log_2\beta]^2} - \lambda \tag{9}$$

$$\frac{\partial L}{\hat{I}_i} = \sum_{u\in U}\sum_{i\in I} \frac{-\varphi\left[\frac{1}{(\ln 2\cdot\beta)}\left(\sum_{j\neq i}(-\gamma+\gamma^2)\cdot(\widehat{U}_u)\right)\right]}{[1+\log_2\beta]^2} - \lambda \tag{10}$$

$$\frac{\partial L}{\hat{I}_j} = \sum_{u\in U}\sum_{i\in I} \frac{-\varphi\left[\frac{1}{(\ln 2\cdot\beta)}\left(\sum_{j\neq i}(-\gamma+\gamma^2)\cdot(-\widehat{U}_u)\right)\right]}{[1+\log_2\beta]^2} - \lambda \tag{11}$$

Figure 2 shows the complete algorithm of *DO-NMF* method.

```
Algorithm : DO-NMF Method
Input     : Rating data, rank of latent
            factor model F, regularization
            λ, learning rate α
Output    : top-N list of item recommendation
            for target user u TopN_u
1. Construct the rating matrix R ∈ ℝ^{m×n} from
   the rating data as the user profile
2. Generate the initial latent factor
   matrices for user Û ∈ ℝ^{m×f} and item Î ∈ ℝ^{n×f}
   using NMF
3. DCG Optimization:
   a. Based on Equation (9), update the user
      u latent factor model: Û_u ⟵ Û_u + α ∂L/∂Û_u
   b. Based on Equation (10), update the item
      i latent factor model: Î_i ⟵ Î_i + α ∂L/∂Î_i
   c. Based on Equation (11), update item j
      latent factor model: Î_j ⟵ Î_j + α ∂L/∂Î_j
4. Calculate the rating prediction r̂ using
   Equation (5)
5. Generate the top-N list of item
   recommendation for target user u TopN_u
   based on the rating prediction
```

Figure 2. Algorithm of *DO-NMF* method.

4. EXPERIMENTAL EVALUATION

This section presents a series of experiments conducted to evaluate the quality of our proposed *DO-NMF*, and to benchmark *DO-NMF* with the baseline CF methods in generating the top-N item recommendations.

4.1 Experiment Setup

4.1.1 Dataset and Experiment Design

The experiments are conducted using the MovieLens rating dataset (https://grouplens.org/datasets/movielens/). The data has 943 users, 1682 movies, and 100K rating data (scale 1 – 5), in which the sparsity is 93.6953% and that the users have rated at least 20 movies.

The 5-fold cross-validation is employed as the evaluation method that we arbitrarily distribute per fold into an 80% training data D_{train} and a 20% test data D_{test}. In this case, we report the evaluation scores based on the average results of all folds.

We empirically adjust the parameters in *DO-NMF* that the best performance is obtained when the regularization λ and learning rate α coefficients are respectively set to 0.00007 and 0.1. Whereas the rank of latent factor model F is set to 64.

4.1.2 Evaluation Metrics

The task of recommendation is to generate the top-N predicted items for the users in D_{test}. For each user u, we compare the top-N list of item recommendations $Top_u(N)$ to the ground truth items GT_u in D_{test}. This study uses two metrics commonly used for evaluating the recommendation performance of each target user u, i.e., Precision and Normalized Discounted Cumulative Gain (NDCG):

$$Precision_u(N) = 100 \cdot \frac{|Top_u(N) \cap GT_u|}{N} \qquad (12)$$

$$NDCG_u(N) = \frac{DCG_u(N)}{IDCG(N)} \qquad (13)$$

where $\mathbb{I}(\cdot)$ is the indicator function that sets the satisfied or unsatisfied condition as 1 or 0. The DCG and IDCG of target user u are calculated as:

$$DCG_u(N) = \sum_{x=1}^{N} \frac{1}{\log_2(1+x)} \cdot \mathbb{I}(Top_u(x) \in GT_u) \qquad (14)$$

$$IDCG(N) = \sum_{x=1}^{N} \frac{1}{\log_2(1+x)} \qquad (15)$$

The reported performance scores are the average results of all target users in D_{test}.

4.2 Performance Comparison

We benchmark the performance of *DO-NMF* with the four baseline methods:

- *UBCF* [24]: The user-based CF method that generates the list of recommendations by taking into account the users' similarities. To obtain its best performance, the tuned parameter for *UBCF* is $user\_neighborhood\_size = 50$.
- *IBCF* [3]: The item-based CF method that generates the recommendations list by taking into account the similarities of the items. To obtain its best performance, the tuned parameter for *IBCF* is $item\_neighborhood\_size = 10$.
- *SVD* [18]: The latent factor method that generates the list of recommendations by using its two orthonormal singular vector latent factor matrices controlled by a singular value diagonal matrix. To achieve its best performance, the tuned parameter for *SVD* is the rank of latent factor model $F = min(m,n) = 943$.
- *NMF* [19]: The latent factor method that generates the list of recommendations by employing its two non-negative latent factor matrices. To achieve its best performance, the tuned parameter for *NMF* is the rank of latent factor model $F = 64$.

The performance comparison of recommendation between the proposed *DO-NMF* and the baseline methods, in terms of Precision and NDCG, are respectively listed in Table 2 and Table 3. The results show that, in general, *DO-NMF* outperforms the baseline methods in terms of Precision and NDCG. This finding confirms that NMF-based DCG optimization is an effective approach to enhance the performance of recommendation. To detail, we present three observations based on these results.

A first observation is the outperforming reasoning of the *DO-NMF* over the baseline methods. Regarding the *UBCF* and *IBCF*, i.e., baseline methods categorized as the memory-based CF approach, the *DO-NMF* outperformance is again proving that a model-based approach results in more quality recommendation than that of the memory-based [2]. Regarding the *SVD* and *NMF*, i.e., the model-based methods that use the MSE (Mean Square Error) optimization as an objective function for generating the latent factor models, the implementation DCG optimization in *DO-NMF* confirm that using the evaluation metric as an objective function can improve the recommendation performance.

The second observation is regarding the *DO-NMF* outperformance based on the evaluation metrics. In terms of Precision, the performances of *IBCF*, *SVD*, and *NMF* are not too far behind the *DO-NMF*. On the other hand, the performances of all benchmarking methods (except SVD) are far away behind the *DO-*

NMF in terms of NDCG. Note that NDCG is a more suitable metric used to evaluate the recommendation ranking quality, i.e., top-*N* list of recommendations, than Precision [9]. These findings highlight further that *DO-NMF* can better solve the top- *N* recommendation task compared to all benchmarking methods.

The third observation is regarding the density of the dataset. Recall that the sparsity problem is a common issue in the collaborative filtering approach. Given the 93.6953% sparsity of the MovieLens dataset, the superiority of *DO-NMF* establishes that the method can solve the sparsity problem better in compared to all baseline methods.

Table 2. Comparison of performances in terms of precision

Method	**Top-*N***				
	@1	**@5**	**@10**	**@15**	**@20**
UBCF	0.81526	0.16681	0.08532	0.05829	0.04437
IBCF	**0.98934**	0.98594	0.98501	0.98478	0.98556
SVD	**0.96761**	0.97237	0.97212	0.97288	0.97410
NMF	0.89550	0.91503	0.92418	0.93024	0.93406
DO-NMF	**0.93229**	**0.98523**	**0.99148**	**0.99333**	**0.99439**

Table 3. Comparison of performances in terms of NDCG

Method	**Top-*N***				
	@1	**@5**	**@10**	**@15**	**@20**
UBCF	0.81526	0.27951	0.18275	0.14262	0.11917
IBCF	0.00258	0.00195	0.00202	0.00197	0.00214
SVD	**0.96761**	0.97161	0.97168	0.97230	0.97321
NMF	0.00459	0.07757	0.06161	0.04848	0.04071
DO-NMF	**0.93229**	**0.97598**	**0.98363**	**0.98664**	**0.98847**

5. CONCLUSION AND FUTURE WORK

Our proposed *DO-NMF* is a top-*N* item recommendation method that implements an NMF-based DCG optimization for collaborative ranking. The experimental results on a rating dataset show that *DO-NMF* outperforms all baseline methods on both the Precision and NDCG metrics. This fact conjectures that the NMF-based DCG optimization is an effective approach to enhance the recommendation performance and to deal with the sparsity problem. For the future works, we plan to conduct extensive experiments of *DO-NMF* on various datasets and also to study the implementation of NMF-based using the other metrics optimization.

6. ACKNOWLEDGEMENT

This work is supported by the Ministry of Research, Technology and Higher Education (Indonesia) under the “Penelitian Dasar” grant scheme, for the financial year 2019.

7. REFERENCES

[1] Konstan, J. A. and Riedl, J., "Recommender systems: from algorithms to user experience," *User Modeling and User-Adapted Interaction,* vol. 22, pp. 101-123, 2012.

[2] Aggarwal, C. C., *Recommender Systems: The Textbook*: Springer, 2016.

[3] Sarwar, B., Karypis, G., Konstan, J., and Riedl, J., "Item-based Collaborative Filtering Recommendation Algorithms," in Proceeding of *The 10th International Conference on World Wide Web*, Hong Kong, 2001, pp. 285-295.

[4] Kim, J., He, Y., and Park, H., "Algorithms for Nonnegative Matrix and Tensor Factorizations: A Unified View based on Block Coordinate Descent Framework," *Journal of Global Optimization,* vol. 58, pp. 285-319, 2014.

[5] Sarwar, B. M., Karypis, G., Konstan, J., and Riedl, J., "Recommender Systems for Large-scale E-Commerce: Scalable Neighborhood Formation Using Clustering," in Proceeding of *The 5th International Conference on Computer and Information Technology*, Dhaka, Bangladesh, 2002, pp. 23-28.

[6] Kumar, N. P. and Fan, Z., "Hybrid User-Item Based Collaborative Filtering," *Procedia Computer Science,* vol. 60, pp. 1453-1461, 2015.

[7] Phorasim, P. and Yu, L., "Movies recommendation system using collaborative filtering and k-means," *International Journal of Advanced Computer Research,* vol. 7, pp. 52-59, 2017.

[8] Cremonesi, P., Koren, Y., and Turrin, R., "Performance of Recommender Algorithms on Top-N Recommendation Tasks," in Proceeding of *The 4th ACM Conference on Recommender Systems*, Barcelona, Spain, 2010, pp. 39-46.

[9] Chapelle, O. and Wu, M., "Gradient Descent Optimization of Smoothed Information Retrieval Metrics," *Information Retrieval,* vol. 13, pp. 216-235, 2010.

[10] Alper, M. E. and Ögüdücü, S. G., "Personalized Recommendation in Folksonomies using a Joint Probabilistic Model of Users, Resources and Tags," in Proceeding of *The 11th International Conference on Machine Learning and Applications (ICMLA)*, Boca Raton, Florida, 2012, pp. 368-373.

[11] Begelman, G., Keller, P., and Smadja, F., "Automated Tag Clustering: Improving Search and Exploration in the Tag Space," in Proceeding of *Collaborative Web Tagging Workshop at WWW2006*, Edinburgh, Scotland, 2006, pp. 15-33.

[12] Pan, R., Dolog, P., and Xu, G., "KNN-Based Clustering for Improving Social Recommender Systems," in *Agents and Data Mining Interaction*, ed: Springer, 2013, pp. 115-125.

[13] Shepitsen, A., Gemmell, J., Mobasher, B., and Burke, R., "Personalized Recommendation in Social Tagging Systems using Hierarchical Clustering," in Proceeding of *The 2008 ACM Conference on Recommender Systems*, 2008, pp. 259-266.

[14] Koren, Y., "Factorization Meets The Neighborhood: A Multifaceted Collaborative Filtering Model," in Proceeding of *The 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Las Vegas, Nevada, USA, 2008, pp. 426-434.

[15] Koren, Y., Bell, R., and Volinsky, C., "Matrix Factorization Techniques for Recommender Systems," *Computer,* vol. 42, pp. 30-37, 2009.

[16] Koren, Y. and Sill, J., "OrdRec: An Ordinal Model for Predicting Personalized Item Rating Distributions," in Proceeding of *The 5th ACM conference on Recommender Systems*, Chicago, Illinois, USA, 2011, pp. 117-124.

[17] Bokde, D., Girase, S., and Mukhopadhyay, D., "Matrix factorization model in collaborative filtering algorithms: A survey," *Procedia Computer Science*, vol. 49, pp. 136-146, 2015.

[18] Sarwar, B., Karypis, G., Konstan, J., and Riedl, J., "Application of dimensionality reduction in recommender system - A case study," Minnesota University Minneapolis Department of Computer Science2000.

[19] Lee, D. D. and Seung, H. S., "Algorithms for non-negative matrix factorization," in *Advances in Neural Information Processing Systems 13*, ed: MIT Press 2001, pp. 556–562.

[20] Balakrishnan, S. and Chopra, S., "Collaborative Ranking," in Proceeding of *The 5th ACM International Conference on Web Search and Data Mining*, Seattle, Washington, USA, 2012, pp. 143-152.

[21] Ifada, N. and Nayak, R., "Do-Rank: DCG Optimization for Learning-to-Rank in Tag-Based Item Recommendation Systems," in *Advances in Knowledge Discovery and Data Mining*, ed: Springer, 2015, pp. 510-521.

[22] Rendle, S., Freudenthaler, C., Gantner, Z., and Schmidt-Thieme, L., "BPR: Bayesian Personalized Ranking from Implicit Feedback," in Proceeding of *The 25th Conference on Uncertainty in Artificial Intelligence*, Montreal, Quebec, Canada, 2009, pp. 452-461.

[23] Acar, E., Plopper, G., and Yener, B., "Coupled analysis of in vitro and histology tissue samples to quantify structure-function relationship," *PloS ONE*, vol. 7, p. e32227, 2012.

[24] Ekstrand, M. D., Riedl, J. T., and Konstan, J. A., "Collaborative Filtering Recommender Systems," *Human–Computer Interaction*, vol. 4, pp. 81-173, 2010.