HYBRID DEEP-SEMANTIC MATRIX FACTORIZATION FOR TAG-AWARE PERSONALIZED RECOMMENDATION

Zhenghua Xu[1,2,†], *Di Yuan*[1,⋆], *Thomas Lukasiewicz*[2], *Cheng Chen*[3], *Yishu Miao*[2], *Guizhi Xu*[1,†]

[1]State Key Laboratory of Reliability and Intelligence of Electrical Equipment,
Hebei University of Technology, China
[2]Department of Computer Science, University of Oxford, United Kingdom
[3]China Academy of Electronics and Information Technology, China

ABSTRACT

Matrix factorization has now become a dominant solution for personalized recommendation on the Social Web. To alleviate the cold start problem, previous approaches have incorporated various additional sources of information into traditional matrix factorization models. These upgraded models, however, achieve only "marginal" enhancements on the performance of personalized recommendation. Therefore, inspired by the recent development of deep-semantic modeling, we propose a hybrid deep-semantic matrix factorization (HDMF) model to further improve the performance of tag-aware personalized recommendation by integrating the techniques of deep-semantic modeling, hybrid learning, and matrix factorization. Experimental results show that HDMF significantly outperforms the state-of-the-art baselines in tag-aware personalized recommendation, in terms of all evaluation metrics.

Index Terms— Deep-Semantic Modeling, Matrix Factorization, Personalized Recommendation, Hybrid Learning

1. INTRODUCTION

In the Web 2.0, social tagging systems are introduced by many websites, where users can freely annotate online items using arbitrary tags (commonly known as *folksonomy* [1]). Since social tags are good summaries of the relevant items and the users' preferences, and since they also contain little sensitive information about their creators, they are valuable information for privacy-enhanced personalized recommendation. Consequently, many efforts have been put on tag-aware personalized recommendation using *content-based filtering* [2, 3, 4] or *collaborative filtering* [5, 6, 7, 8].

As users can freely choose their own vocabulary, social tags may contain many uncontrolled vocabularies. This usually results in sparse, redundant, and ambiguous tag information, and significantly weakens the performance of content-based recommendation systems. The common solution is to apply machine learning techniques, e.g., clustering [3] or autoencoders [9], to learn more abstract and representative features from raw tags. Our previous work in [4] proposes a deep-semantic model, DSPR, which utilizes deep neural networks to model abstract and recommendation-oriented representations for social tags. DSPR achieves a better performance than the clustering and autoencoder solutions.

Matrix factorization is a collaborative-filtering-based solution, which has become a dominant solution for personalized recommendation on the Social Web [5, 6, 7] and has been reported to be superior to memory-based techniques [10]. However, there exists a *cold start* problem in matrix factorization: many users only give very few ratings, resulting in a very sparse user-item rating matrix, and making it difficult to summarize users' preferences. A widely adopted solution is to incorporate additional sources of information about users, e.g., implicit feedback [10], social friendship [6], geographical neighborhood [11], or textual comments [7]. We call these upgraded models *additional-information-based matrix factorization (AMF)* models.

Although DSPR and AMF models have progressively improved tag-aware personalized recommendation, there are a few drawbacks: (i) DSPR does not utilize the idea of collaborative filtering; so the valuable correlation information between users and items is not being used to help recommendations. (ii) As a deep model, DSPR stacks many layers, making it difficult to optimize the model by gradient back-propagation. (iii) The existing AMF models generally incorporate the additional information as a regularization term of matrix factorization; this term's coefficient, as proved in [6], has to be very small; therefore, the additional information has very limited contribution on the optimizing gradient, resulting in only "marginal" improvements on the recommendation performance. (iv) The recommendation results of the existing AMF models are difficult to interpret, because latent factor matrices are used to represent users and items.

Consequently, to solve the above problems and to further improve the performance of tag-aware personalized recommendation, we propose a *hybrid deep-semantic matrix factorization (HDMF)* model, which integrates the techniques of deep-semantic modeling, hybrid learning, and matrix factorization. Generally, HDMF uses a *tag-based user matrix* and a *tag-based item matrix* as respective inputs of two deep autoencoders to generate *deep-semantic user and item matrices* at the code layers, and also *reconstructed user and item matrices* at the output layers. The deep model is then trained by using a *hybrid learning signal* to minimize both *reconstruction errors* and *deep-semantic matrix factorization errors*, i.e., the squared differences between the user-item rating matrix

⋆Co-first author.
†Corresponding authors: zhenghua.xu@hebut.edu.cn (Zhenghua Xu), gzxu@hebut.edu.cn (Guizhi Xu).

(seeing tags as positive ratings) and the dot product of deep-semantic user and item matrices (seeing deep-semantic matrices as the decomposed matrices in matrix factorization). The intuitions behind using the hybrid learning signal are: (i) minimizing reconstruction errors can learn better representations for both users and items; (ii) deep-semantic matrix factorization offers a learning signal that connects users and items to discover the underlying users' preferences; and (iii) two signals can complement each other to provide sufficient gradients for a better model optimization and escaping local minima.

HDMF has the following advantages. (i) It overcomes the drawback of DSPR by adding collaborative capabilities to the deep-semantic model. (ii) The hybrid learning signal helps HDMF to better optimize the model and escape local minima. (iii) Differently from AMF models, the additional tag information in HDMF is directly used to model the decomposed user and item matrices in matrix factorization; this thus maximizes the effect of the additional tag information on model optimization. (iv) HDMF remedies the non-interpretability problem in matrix factorization: considering deep-semantic matrices as the decomposed matrices and finding the most influential input tags for each dimension, the decomposed user and item matrices in HDMF become interpretable.

The main contributions of this paper are briefly as follows: (i) We briefly analyze the state-of-the-art personalized recommendation models that use content-based filtering or matrix factorization and identify their existing problems. (ii) We innovatively propose a hybrid deep-semantic matrix factorization (HDMF) model to tackle these problems and to further improve the performance of tag-aware personalized recommendation, by integrating the techniques of deep-semantic modeling, hybrid learning, and matrix factorization. (iii) Experimental results show that HDMF significantly outperforms the state-of-the-art baselines in tag-aware personalized recommendation, in terms of all evaluation metrics, e.g., its mean reciprocal rank (resp., mean average precision) is 1.52 (resp., 1.66) times as high as that of the best baseline.

2. PRELIMINARIES

A *folksonomy* is a tuple $\mathcal{F} = (U, T, D, A)$, where U, T, and D are sets of *users*, *tags*, and *items*, respectively, and $A \subseteq U \times T \times D$ is a set of *assignments* (u, t, d) of t to d by u [1].

A *tag-based user profile* is a feature vector $x = [g_i^u]_{i=1}^{|T|}$, where $|T|$ is the tag vocabulary's size, and $g_i^u = |\{(u, t_i, d) \in A \mid d \in D\}|$ is the number of times that user u annotates items with tag t_i; the *tag-based user matrix* is thus defined as $X = [x_i]_{i=1}^{|U|}$, where x_i is the profile vector of the ith user, and $|U|$ is the total number of users. Similarly, a *tag-based item profile* is a vector $y = [g_j^d]_{j=1}^{|T|}$, where $g_j^d = |\{(u, t_j, d) \in A \mid u \in U\}|$ is the number of times that item d is annotated with tag t_j; while the *tag-based item matrix* is defined as $Y = [y_j]_{j=1}^{|D|}$, where y_j is the profile vector of the jth item, and $|D|$ is the total number of items.

The *user-item rating matrix* is $R = [r_{i,j}]_{i=1,j=1}^{|U|,|D|}$, where $r_{i,j}$ is the number of tags annotated by user i to item j. Given R, traditional *matrix-factorization-based recommender systems* aim to approximate R using the decomposed latent matrices of users and items, i.e., X^l and Y^l, respectively, which are optimized by minimizing the squared differences between R and ${X^l}^T \cdot Y^l$ on a set of observed ratings; formally,

$$\min_{X^l, Y^l} \sum_{i=1}^{|U|} \sum_{j=1}^{|D|} I_{i,j} (r_{i,j} - {x_i^l}^T \cdot y_j^l)^2, \tag{1}$$

where $I_{i,j}$ is 1, if user i annotated item j, and 0, otherwise [7]. After optimization learning, the *predicted user-item rating matrix* $\hat{R} = {X^l}^T \cdot Y^l$ is used for recommendation.

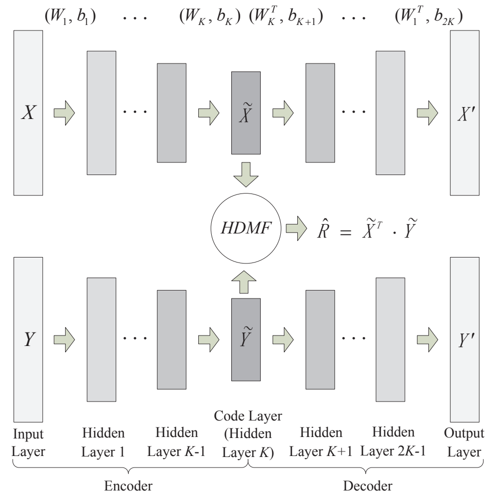

Fig. 1. Overview of HDMF

3. HDMF

To alleviate the cold start problem in traditional matrix factorization, a widely adopted solution is to incorporate additional sources of information about users to achieve additional-information-based matrix factorization (AMF) [11, 10, 6, 7]. However, as shown in our experiments and the results in [7], the existing AMF models achieve only "marginal" (around 5% in [7]) performance improvements. Therefore, we propose a hybrid deep-semantic matrix factorization (HDMF) model to further enhance the performance of tag-aware personalized recommendation, by integrating deep-semantic modeling, hybrid learning, and matrix factorization.

Figure 1 shows an overview of the HDMF model. Generally, HDMF takes the tag-based user and item matrices X and Y (defined in Section 2) as inputs of two deep autoencoders, consisting of *encoders* and *decoders*. These inputs are then passed through multiple hidden layers and projected to the deep-semantic user and item matrices $\widetilde{X}$ and $\widetilde{Y}$ at the code layers, and to the reconstructed user and item matrices X' and Y' at the output layers. The HDMF model is then trained by using a hybrid learning signal to minimize both deep-semantic matrix factorization errors and reconstruction errors. Finally, a predicted user-item rating matrix $\hat{R} = \widetilde{X}^T \cdot \widetilde{Y}$ is used for personalized recommendation.

3.1. Deep-Semantic Matrix Factorization

Deep-semantic matrix factorization is based on the encoder parts of autoencoders. Formally, given the tag-based user and item matrices X and Y, a weight matrix W_1, and a bias vector b_1, the intermediate outputs of the jth hidden layers $h_j(\cdot)$, $j \in \{2, \dots, K\}$, in the encoders are defined as follows:

$$h_j(X) = \tanh(W_j h_{j-1}(X) + b_j), \tag{2}$$

$$h_j(Y) = \tanh(W_j h_{j-1}(Y) + b_j), \tag{3}$$

where W_j and b_j are the weight matrix and the bias vector for the jth hidden layers in the encoders, respectively, and K is the total number of hidden layers in each encoder. Then, the outputs of the Kth hidden layers, i.e., the code layers, are the deep-semantic user and item matrices, denoted $\widetilde{X}$ and $\widetilde{Y}$:

$$\widetilde{X} = h_K(X), \qquad \widetilde{Y} = h_K(Y). \tag{4}$$

Consequently, by seeing deep-semantic matrices $\widetilde{X}$ and $\widetilde{Y}$ as the decomposed user and item matrices in matrix factorization, the parameters W_j and b_j can be optimized by minimizing the following deep-semantic matrix factorization errors:

$$L_{DMF}(\Theta) = (1-\lambda_\theta)\sum_{i=1}^{|U|}\sum_{j=1}^{|D|} I_{i,j}(r_{i,j} - \widetilde{x}_i^T \cdot \widetilde{y}_j)^2 + \lambda_\theta(\sum_{j=1}^{K}\|W_j\|^2 + \sum_{j=1}^{K}\|b_j\|^2), \tag{5}$$

where $r_{i,j}$ is an element in the user-item rating matrix R, indicating the number of tags assigned by user i to item j; $\widetilde{x}_i$ (resp., $\widetilde{y}_j$) is the vector at ith (resp., jth) column of $\widetilde{X}$ (resp., $\widetilde{Y}$), which is the deep-semantic representation of the ith user (resp., jth item); the second term is a regularization term used to prevent overfitting, and λ_θ is the regularization parameter.

3.2. Hybrid Deep-semantic Matrix Factorization

However, it is difficult to train the model using solely the learning signal from deep-semantic matrix factorization. This is because the model stacks many layers of non-linearities, and when learning signals are back-propagated to the first few layers, they become minuscule and insignificant to learn good representations for the users and items, which in turn results in poor local minima. A common solution is to first pretrain each layer using restricted Boltzmann machines (RBMs) [12, 13] or autoencoders [14] and then use back-propagation to fine-tune the entire deep neural network [15].

Motivated by our previous work [16], we directly incorporate autoencoders into deep-semantic matrix factorization model, and train the deep model using a hybrid learning signal that integrates reconstruction errors of autoencoders with deep-semantic matrix factorization errors. We call this model hybrid deep-semantic matrix factorization (HDMF). The intuition is as follows: (i) the reconstruction-error-based signal can learn better representations for both users and items; (ii) the collaborative learning signal from deep-semantic matrix factorization can connect users and items to discover underlying users' preferences; and (iii) furthermore, the reconstruction-error-based signal can complement deep-semantic matrix factorization to provide sufficient gradients for better optimizing the model and escaping local minima.

As in Figure 1, we adopt autoencoders with tied weights in HDMF, i.e., the weight matrices in the decoder are transposes of weight matrices in the encoder. The decoders take the deep-semantic user and item matrices $\widetilde{X}$ and $\widetilde{Y}$ at the code layer as the inputs and generate reconstructed user and item matrices X' and Y' at their output layers. Then, reconstruction errors are computed based on the squared differences between the original tag-based matrices (X and Y) and the reconstructed matrices (X' and Y'). Finally, the reconstruction-error-based learning signal will be used to first update W_1^T, then back-propagated to update W_2^T, W_3^T, and so on. As updating W_j^T is equivalent to updating W_j, this signal complements deep-semantic matrix factorization and offers sufficient gradients to the first few layers of the deep model.

Table 1. Dataset Information

Users (u)	Tags (t)	Items (i)	Assignments ((u,t,i))
1 843	3 508	65 877	339 744

The intermediate outputs of the $K+j$th hidden layers $h_{K+j}(\cdot)$, $j \in \{1,\ldots,K-1\}$, in the decoders are defined as:

$$h_{K+j}(X) = \tanh(W_{K-(j-1)}^T h_{K+(j-1)}(X) + b_{K+j}), \tag{6}$$

$$h_{K+j}(Y) = \tanh(W_{K-(j-1)}^T h_{K+(j-1)}(Y) + b_{K+j}), \tag{7}$$

where $W_{K-(j-1)}^T$ is the transpose of $W_{K-(j-1)}$, and b_{K+j} is the bias vector for the $K+j$th hidden layer. The outputs of the $2K-1$th hidden layers are used to generate reconstructed user and item profiles, denoted X' and Y', at the output layers:

$$X' = \tanh(W_1^T h_{2K\text{-}1}(X) + b_{2K}), \tag{8}$$

$$Y' = \tanh(W_1^T h_{2K\text{-}1}(Y) + b_{2K}). \tag{9}$$

Then, the reconstruction errors of the user (resp., item) matrix are computed as the sum of the Euclidean (i.e., L2) norms of the differences between the tag-based user (resp., item) profile x_i (resp., y_j) in X (resp., Y) and the reconstructed user (resp., item) profile x'_i (resp., y'_j) in X' (resp., Y'). By integrating the reconstruction errors with the deep-semantic matrix factorizations errors, the HDMF model is thus trained by minimizing the following hybrid signal:

$$L_{HDMF} = \lambda_\alpha \sum_{i=1}^{|U|}\sum_{j=1}^{|D|} I_{i,j}(r_{i,j} - \widetilde{x}_i^T \cdot \widetilde{y}_j)^2 + \lambda_e(\sum_{i=1}^{|U|}\|x'_i - x_i\| + \sum_{j=1}^{|D|}\|y'_j - y_j\|) + \lambda_\theta(\sum_{j=1}^{K}\|W_j\|^2 + \sum_{j=1}^{2K}\|b_j\|^2).$$

4. EXPERIMENTS

Extensive experiments are conducted to compare HDMF with the following state-of-the-art baselines: (i) Four content-based tag-aware recommendation models that utilize social tags as content information for tag-aware personalized recommendation are selected, where machine learning techniques are applied to model abstract and effective representations for users or/and items: the clustering-based models **CCS** and **CCF** [3], the autoencoder-based model **ACF** [9], and the deep-semantic similarity-based model **DSPR** [4]. (ii) Three matrix-factorization-based recommendation models are also selected: the traditional matrix factorization model **MF**, and the additional-information-based matrix factorization (AMF) models **MF**$_{sf}$ [6] and **MF**$_{tc}$ [7], which incorporate social friendships and textual comments of users as the additional sources of information for matrix factorization.

For fair comparison, experiments are performed on the same real-world social-tagging dataset as in [4, 9], which is gathered from Delicious.com and released in [17]. After using the same pre-processing to remove the infrequent tags

Table 2. Tag-Aware Personalized Recommendation Performances of Various Models (in %)

Models	$P@5$	$P@15$	$P@30$	$P@50$	$R@5$	$R@15$	$R@30$	$R@50$	$F@5$	$F@15$	$F@30$	$F@50$	MAP	MRR
CCF	0.913	0.757	0.597	0.454	0.439	1.051	1.499	1.803	0.593	0.880	0.854	0.726	0.437	0.200
ACF	1.120	0.909	0.736	0.595	0.590	1.209	1.917	2.364	0.791	1.038	1.064	0.950	0.637	0.252
CCS	2.397	1.903	1.564	1.273	0.938	2.271	3.739	4.774	1.349	2.070	2.205	2.010	1.319	0.523
DSPR	13.34	9.285	6.950	5.306	4.235	8.347	12.00	14.98	6.430	8.791	8.803	7.836	5.452	2.547
MF	9.157	7.467	6.784	6.302	1.302	2.851	4.988	7.587	2.280	4.127	5.749	6.899	6.757	1.682
MF_{sf}	10.16	8.063	7.302	6.736	1.457	3.109	5.407	8.132	2.549	4.487	6.213	7.368	6.920	1.798
MF_{tc}	10.06	8.032	7.282	6.741	1.436	3.066	5.388	8.101	2.513	4.438	6.197	7.359	6.908	1.790
HDMF	**18.20**	**15.96**	**13.61**	**11.37**	**5.510**	**13.05**	**21.13**	**28.70**	**8.458**	**14.36**	**16.56**	**16.29**	**11.50**	**3.870**

used less than 15 times, the resulting dataset is in Table 1. We randomly selected 80% of assignments as training set, 5% as validation set, and 15% as test set. All models were implemented using Python and Theano and run on a GPU server with an NVIDIA Tesla K40 GPU and 12GB GPU memory. The parameters of HDMF are selected by grid search, and the values are set as follows: (i) # of hidden layers is 5; (ii) # of neurons from 1st to 5th hidden layer are 2 000, 300, 128, 300, and 2 000, respectively; (iii) λ_θ and λ_e are set to 0.01 and 0.2; and (iv) the learning rate for model training is 0.002.

The most popular evaluation metrics for recommendation are precision, recall, and F1-score [18]. As users usually only browse the topmost recommended items, we apply these metrics at a given cut-off rank k, i.e., considering only the top-k results on the recommendation list, called *precision at k ($P@k$)*, *recall at k ($R@k$)*, and *F1-score at k ($F@k$)*. Since users always prefer to have their target items ranked in the front of the recommendation list, we also employ as evaluation metrics the *mean average precision (MAP)* and the *mean reciprocal rank (MRR)*, which take into account the order of items and give a greater importance to the ones ranked higher.

Table 2 depicts the performances of HDMF and the seven baselines on the Delicious dataset. Generally, the relative performances of the baselines reported in Table 2 are highly consistent with the results reported in [9], [4], and [7]; namely, (i) ACF outperforms CCF, (ii) DSPR outperforms CCF, ACF, and CCS, and (iii) MF_{sf} and MF_{ct} "slightly" outperform MF, respectively. More importantly, we note that our proposed model, HDMF, significantly outperforms all seven baselines in all metrics; e.g., the MRR (resp., MAP) of HDMF are 1.52 (resp., 1.66) times as high as that of the best baseline, DSPR (resp., MF_{sf}), while the relative performances in $P@k$, $R@k$, and $F@k$ are also similar. This finding strongly proves that by integrating the techniques of deep-semantic modeling, hybrid learning, and matrix factorization, HDMF overcomes the existing problems (as presented in Section 1) of state-of-the-art recommendation models and achieves a very superior performance in tag-aware personalized recommendation.

Specifically, the MRR and MAP of HDMF are 1.52 and 2.1 times, respectively, as high as those of the state-of-the-art deep-semantic model DSPR. In addition, the relative improvements of HDMF to DSPR, in terms of $P@k$, $R@k$, and $F@k$, all gradually enhance with the rise of the cut-off rank k, i.e., increasing from around 1.3 times at $k = 5$ to more than double at $k = 50$. This observation demonstrates that incorporating collaborative-based capabilities (i.e., using correlation information between users and items to help the recommendation) can greatly enhance the deep-semantic model's performance in tag-aware recommendation, especially for the one with relative long recommendation lists.

Furthermore, the AMF models, MF_{sf} and MF_{tc}, have close performances, and their relative improvements to MF are "marginal", e.g., their MAP and MRR are only 2.4% and 6.8% better than those of MF. This is consistent with the results in [7], where the improvement rates of MF_{sf} and MF_{tc} to MF are only 3.2% and 5.5%. The reason may be as follows: the AMF models incorporate the additional source of information as a regularization term with a small coefficient in matrix factorization, which greatly limits the additional information's contribution on the optimizing gradient and thus limits their capabilities in improving the recommendation performance. By contrast, as shown in Table 2, HDMF dramatically outperforms MF: the MAP and MRR of HDMF are about 70% and 130%, respectively, better than those of MF. This is mainly because the additional social tag information in HDMF is utilized to model the deep-semantic user and item matrices, which are then used directly as the decomposed user and item matrices in matrix factorization; since the decomposed matrices have a dominant contribution on the optimizing gradient, HDMF maximizes the effect of the additional social tag information on model optimization, making it possible to achieve significant improvements.

5. SUMMARY AND OUTLOOK

We analyzed existing problems of state-of-the-art tag-aware personalized recommendation models and proposed a hybrid deep-semantic matrix factorization (HDMF) model to tackle these problems. Extensive experimental studies were conducted and the results showed that, by integrating deep-semantic modeling, hybrid learning, and matrix factorization, HDMF greatly outperforms the state-of-the-art baselines in all evaluation metrics. In the future, further experiments will be conducted to compare the performances of HDMF on different kinds of Social Web datasets. We will also investigate methodologies to add spatial and temporal information into the HDMF model to capture the users' real-time preferences.

6. ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China under grants 61906063 and 51737003, by the Natural Science Foundation of Tianjin under the grant 19JCQNJC00400, by the "100 Talents Plan" of Hebei Province under the grant E2019050017 and by the UK EPSRC grants EP/J008346/1, EP/L012138/1, and EP/M025268/1.

7. REFERENCES

[1] Andreas Hotho, Robert Jäschke, Christoph Schmitz, and Gerd Stumme, "Information retrieval in folksonomies: Search and ranking," in *Proceedings of the European Semantic Web Conference*, 2006, pp. 411–426.

[2] Iván Cantador, Alejandro Bellogín, and David Vallet, "Content-based recommendation in social tagging systems," in *Proceedings of the ACM Conference on Recommender Systems*, 2010, pp. 237–240.

[3] Andriy Shepitsen, Jonathan Gemmell, Bamshad Mobasher, and Robin Burke, "Personalized recommendation in social tagging systems using hierarchical clustering," in *Proceedings of the ACM Conference on Recommender Systems*, 2008, pp. 259–266.

[4] Zhenghua Xu, Cheng Chen, Thomas Lukasiewicz, Yishu Miao, and Xiangwu Meng, "Tag-aware personalized recommendation using a deep-semantic similarity model with negative sampling," in *Proceedings of the International on Conference on Information and Knowledge Management*, 2016, pp. 1921–1924.

[5] Mohamed Reda Bouadjenek, Hakim Hacid, Mokrane Bouzeghoub, and Athena Vakali, "Using social annotations to enhance document representation for personalized search," in *Proceedings of the International ACM SIGIR Conference*, 2013, pp. 1049–1052.

[6] Hao Ma, Dengyong Zhou, Chao Liu, Michael R. Lyu, and Irwin King, "Recommender systems with social regularization," in *Proceedings of the International Conference on Web Search and Data Mining*, 2011, pp. 287–296.

[7] Jarana Manotumruksa, Craig Macdonal, and Iadh Ounis, "Regularising factorised models for venue recommendation using friends and their comments," in *Proceedings of the International on Conference on Information and Knowledge Management*, 2016, pp. 1981–1984.

[8] Karen H. L. Tso-Sutter, Leandro Balby Marinho, and Lars Schmidt-Thieme, "Tag-aware recommender systems by fusion of collaborative filtering algorithms," in *Proceedings of the ACM Symposium on Applied Computing*, 2008, pp. 1995–1999.

[9] Yi Zuo, Jiulin Zeng, Maoguo Gong, and Licheng Jiao, "Tag-aware recommender systems based on deep neural networks," *Neurocomputing*, vol. 204, pp. 51–60, 2016.

[10] Yehuda Koren, Robert Bell, and Chris Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, 2009.

[11] Longke Hu, Aixin Sun, and Yong Liu, "Your neighbors affect your ratings: on geographical neighborhood influence to rating prediction," in *Proceedings of the International ACM SIGIR conference*, 2014, pp. 345–354.

[12] Geoffrey E. Hinton, Simon Osindero, and Yee-Whye Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[13] Geoffrey E. Hinton and Ruslan Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[14] Yoshua Bengio, Pascal Lamblin, Dan Popovici, and Hugo Larochelle, "Greedy layer-wise training of deep networks," in *Advances in Neural Information Processing Systems*, 2007, vol. 19, pp. 153–160.

[15] Dumitru Erhan, Yoshua Bengio, Aaron Courville, Pierre-Antoine Manzagol, Pascal Vincent, and Samy Bengio, "Why does unsupervised pre-training help deep learning?," *Journal of Machine Learning Research*, vol. 11, pp. 625–660, 2010.

[16] Zhenghua Xu, Thomas Lukasiewicz, Chen Cheng, Yishu Miao, and Xiangwu Meng, "Tag-aware personalized recommendation using a hybrid deep model," in *Proceedings of the International Joint Conference on Artificial Intelligence*, 2017.

[17] Iván Cantador, Peter Brusilovsky, and Tsvi Kuflik, "Second Workshop on Information Heterogeneity and Fusion in Recommender Systems (HetRec2011).," in *Proceedings of the ACM Conference on Recommender Systems*, 2011, pp. 387–388.

[18] Jesús Bobadilla, Fernando Ortega, Antonio Hernando, and Abraham Gutiérrez, "Recommender systems survey," *Knowledge-Based Systems*, vol. 46, pp. 109–132, 2013.