Co-learning Multiple Browsing Tendencies of a User by Matrix Factorization-based Multitask Learning

Guo-Jhen Bai, Cheng-You Lien, Hung-Hsuan Chen
Computer Science and Information Engineering
National Central University
ivy2350442@gmail.com,littlelien.peanut@gmail.com,hhchen@g.ncu.edu.tw

ABSTRACT

Predicting an online user's future behavior is beneficial for many applications. For example, online retailers may utilize such information to customize the marketing strategy and maximize profit. This paper aims to predict the types of webpages a user is going to click on. We observe that instead of building independent models to predict each individual type of web page, it is more effective to use a unified model to predict a user's future clicks on *different* types of web pages *simultaneously*. The proposed model makes predictions based on the latent variables that represent possible interactions among the multiple targets and among the features. The experimental results show that this method outperforms the carefully tuned single-target training models most of the time. If the size of the training data is limited, the model shows a significant improvement over the baseline models, likely because the hidden relationship among different targets can be discovered by our model.

ACM Reference Format:
Guo-Jhen Bai, Cheng-You Lien, Hung-Hsuan Chen. 2019. Co-learning Multiple Browsing Tendencies of a User by Matrix Factorization-based Multitask Learning. In *IEEE/WIC/ACM International Conference on Web Intelligence (WI '19), October 14–17, 2019, Thessaloniki, Greece.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3350546.3352526

1 INTRODUCTION

It has been reported that users' online behavior may change before and during holidays (e.g., Christmas or Singles' Day [16]) or special events (e.g., immediately before a purchase [20]). Discovering and predicting such changes beforehand helps online retailers target the right users at the right times and can, therefore, provide a customized strategy of marketing to individuals.

This study proposes a new method – matrix factorization-based multitask learning (MFMT) – to *simultaneously* predict changes in users' browsing behaviors on different types of webpages. The proposed method learns the relationship between all the features and all the target variables to make predictions in one unified framework. As a result, MFMT can capture the hidden correlation among the target variables and encode such information in the model. Compared to most of the supervised learning algorithms, which typically learn one target at a time, MFMT considers the distribution of the features and all the target variables. We believe that such a model is especially useful if there are interrelated correlations between various target variables. In our application scenario – predicting a user's future online browsing behavior – we believe that a user's clicks on one type of webpage may indicate her/his preferences regarding other types of pages.

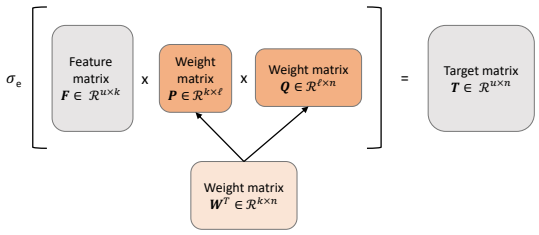


Figure 1: Multitask learning through matrix factorization.

We compare MFMT with several supervised learning approaches, including the k-nearest neighbors classifier (KNN), the logistic regression classifier, and support vector machine (SVM). We observe that our MFMT model better predicts user behavior changes on several types of webpages. Perhaps more importantly, we observe that if the size of the training data was limited, the advantage of the proposed method is more apparent. We believe the reason is that MFMT may utilize the hidden relationship among the multiple target variables that cannot be discovered by the baseline methods since they treat each target variable independently.

2 RELATED WORK

Online users' collective behaviors are recorded digitally and can be used to for various studies, such as identifying friendship [23], social influence [2], browsing habits [6], search intention [24], personality prediction [17], etc. Most studies have leveraged mathematical formulas to model users' behaviors. Popular models include Markov chains [1, 28], matrix factorization (MF) [5, 12, 14], sequential pattern mining [22] and, recently, deep learning-based approaches [19, 32]. Our proposed MFMT model can be integrated with many of the abovementioned methods as long as there are several target variables that need to be predicted. Later we will show how the MFMT model can be integrated with the logistic regression classifier. However, it is straightforward to apply other supervised learners, e.g., support vector machines, factorization machines (FM) [25, 26], or field-aware factorization machines (FFM) [12].

MF and its variants are widely used by recommender systems. MF decomposes the user-to-item interaction matrix into small matrices [13, 14, 21]. MF is also applied to various tasks that can model the

WI '19, October 14–17, 2019, Thessaloniki, Greece

ACM ISBN 978-1-4503-6934-3/19/10…$15.00
https://doi.org/10.1145/3350546.3352526

data in the matrix form with unknowns in the matrix that need to be predicted, such as coauthorship network [7, 29] and genetic disease network [8, 10]. MF is shown to be a special form of FM [25, 26], which incorporate both the features and the interactions between each pair of features into the model. FM and its variants won several click prediction competitions recently [11, 12, 31]. Our proposed method is relevant to MF and FM because we decompose a large matrix into small ones. However, FM models the hidden relationship between features and predict only one target variable; in contrast, our model captures the hidden relationship between the features and between the multiple target variables.

Multitask learning is aimed at learning multiple tasks simultaneously. It has been observed that in many cases, letting the models focus on learning one target variable is less effective than learning multiple targets simultaneously [3, 4, 30]. Traditional multitask learning algorithms commonly encourage sparse parameters through regularizations, of which the ℓ_1/ℓ_q norm has probably attracted the most interest [15, 18]. However, if these tasks do not share many common features, the performance of such a regularization is unsatisfactory [27]. MFMT is different because we do not encourage sparsity to reduce the number of parameters. MFMT can also be viewed as a variation of neural network-based multitask learning, which has been studied recently [9, 27]. However, previous models mostly connect the shared layers with small independent layers for each task, whereas our model uses only the shared layers. Additionally, as far as we know, we are the first to predict a user's future browsing tendencies based on multitask learning.

3 METHODOLOGY

3.1 Preliminary notes

Matrix factorization is widely used by recommender systems to generate the low-dimensional latent factors for the users and the items. Given m users, n items, and users' ratings of the items, MF creates a large but sparse matrix $\boldsymbol{C} = \left[c_{ij}\right] \in \mathcal{R}^{m\times n}$ to record all the known ratings c_{ij} from user i on item j. MF attempts to decompose matrix $\boldsymbol{C}$ into two small matrices $\boldsymbol{A} = [a_{i\ell}] \in \mathcal{R}^{m\times k}$ and $\boldsymbol{B} = \left[b_{\ell j}\right] \in \mathcal{R}^{k\times n}$ ($k \ll m$ and $k \ll n$) such that the sum of the squared errors between the known ratings and the predicted ratings is minimized, as shown in Equation 1.

$$loss = \sum_{\forall (i,j)\in\mathcal{K}} \left(c_{ij} - \sum_{\ell=1}^{k} a_{i\ell}\cdot b_{\ell j}\right)^2, \tag{1}$$

where $\mathcal{K}$ is the set of all known pairs (i, j) (i.e., user i rated item j).

3.2 MF-based multitask learning

Given an instance's k features, a multitask learning algorithm predicts the n targets based on these features. A naïve implementation of a multitask learning model is to build n *independent* learners for each of the targets. Let $\boldsymbol{F} = \left[f_{ij}\right] \in \mathcal{R}^{u\times k}$ be the feature matrix (for u training instances, each with k features) and $\boldsymbol{T} = \left[t_{ij}\right] \in \mathcal{R}^{u\times n}$ be the target matrix for u training instances, each with n target variables; the naïve implementation trains n classifiers $c_p()$ ($p = 1, 2, \ldots, n$) such that, for the ath training instance, $c_p(\boldsymbol{F}_a) = \hat{t}_{ap} \approx t_{ap}$ (where $\boldsymbol{F}_a$ denotes the ath row of matrix $\boldsymbol{F}$). Here, we consider the generalized linear model as our predicting function and let $\sigma^{-1}()$ be the link function for the generalized linear model. For example, if we use the logistic regression classifier, the link function $\sigma^{-1}()$ is a logit function (therefore, $\sigma()$ is a logistic function), and the predicting function c_p is shown in Equation 2.

$$c_p(\boldsymbol{F}_a) = \sigma\left(\boldsymbol{W}_p\cdot\boldsymbol{F}_a^T\right) = \frac{1}{1+\exp\left(-(\boldsymbol{W}_p\cdot\boldsymbol{F}_a^T)\right)}, \tag{2}$$

where $\boldsymbol{W}_p = \left[w_{p1}, w_{p2}, \ldots, w_{pk}\right]$ are the parameters to be learned for the pth classifier, and $\boldsymbol{W} = [\boldsymbol{W}_1, \boldsymbol{W}_2, \ldots, \boldsymbol{W}_n]^T = \left[w_{ij}\right] \in \mathcal{R}^{n\times k}$ are the parameters to be learned for all the n independent generalized linear models.

The prediction of the targets $\hat{\boldsymbol{T}}$ are obtained using Equation 3.

$$\hat{\boldsymbol{T}} = \begin{bmatrix} c_1(\boldsymbol{F}_1) & \cdots & c_n(\boldsymbol{F}_1) \\ \vdots & \vdots & \vdots \\ c_1(\boldsymbol{F}_u) & \cdots & c_n(\boldsymbol{F}_u) \end{bmatrix} = \sigma_{\mathrm{e}}\left(\boldsymbol{F}\cdot\boldsymbol{W}^T\right), \tag{3}$$

where the subscript e denotes that function $\sigma()$ is applied element-wise to all the entries in the matrix.

The objective is to find the parameters such that the following objective function is minimized:

$$O = \left\|\boldsymbol{T} - \hat{\boldsymbol{T}}\right\|^2 + \frac{\lambda}{2}\|\boldsymbol{\Theta}\|^2, \tag{4}$$

where $\|\cdot\|$ is the Frobenius norm of the given matrix or vector, $\boldsymbol{\Theta}$ is the vector of parameters to be learned, and λ is a hyperparameter that determines the relative importance of the training error and the Frobenius norm of the learnable parameters.

However, such an approach may fail to capture the hidden relationship among the target variables. The reason is that the entries in the ith column of matrix $\boldsymbol{W}^T$ in Equation 3 only influence the prediction of the ith target variable for all the instances. This means that all the target variables are conditionally independent given the training features. Unfortunately, the target variables may have a certain dependency. For example, a tourist who has reserved a ticket to Tokyo may also be interested in visiting nearby cities, e.g., Hakone. If we simply build two independent classifiers to predict a user's intention to visit Tokyo and Hakone, we may not be able to elicit the hidden relationship between the two targets.

We propose utilizing MF to learn multiple tasks simultaneously in one model so the hidden relationship among the target variables is likely to be captured. In other words, if the targets are indeed conditionally dependent given the features of an instance, our model may capture such dependency and make better predictions. Specifically, we propose decomposing matrix $\boldsymbol{W}^T$ into two small matrices $\boldsymbol{P} = [p_{sv}] \in \mathcal{R}^{k\times\ell}$ and $\boldsymbol{Q} = \left[q_{vj}\right] \in \mathcal{R}^{\ell\times n}$. As shown in Figure 1, instead of searching for a matrix $\boldsymbol{W}$ to minimize the objective function (Equation 4), we want to find $\boldsymbol{P}$ and $\boldsymbol{Q}$ to minimize the objective function, so the predictive function can be written as in Equation 5.

$$\hat{\boldsymbol{T}} = \left[t_{ij}\right] = \left[\sigma(f_{is}\sum_{v=1}^{\ell} p_{sv}q_{vj})_{ij}\right]_{i=1,\ldots,u;j=1,\ldots,n}, \tag{5}$$

where $\boldsymbol{P}$ and $\boldsymbol{Q}$ are the parameters to be learned.

We apply gradient-based optimization to obtain the learnable parameters $\boldsymbol{P}$ and $\boldsymbol{Q}$. Equation 6 and Equation 7 show the derivatives of the objective function with respect to p_{sv} and q_{vj}, respectively.

$$\frac{\partial O}{\partial p_{sv}} = -2d\sigma'\left(f_{is}\sum_{v=1}^{\ell} p_{sv}q_{vj}\right) f_{is}q_{vj} + \lambda p_{sv}, \quad (6)$$

$$\frac{\partial O}{\partial q_{vj}} = -2d\sigma'\left(f_{is}\sum_{v=1}^{\ell} p_{sv}q_{vj}\right) f_{is}p_{sv} + \lambda q_{vj}, \quad (7)$$

where $d = \sum(t_{ij} - \hat{t}_{ij})$, and $\sigma'()$ denotes the derivative of the respective function. For example, if we apply the logistic regression classifier, $\sigma'(x) = \sigma(x)(1 - \sigma(x))$.

3.3 Model analysis

If we use the naïve implementation of multitask learning, i.e., training n independent models for the n targets, we will need to estimate all the entries in $\boldsymbol{W}$; thus, the total number of parameters to be learned will be kn. Our proposed MFMT model, on the other hand, requires estimating matrices $\boldsymbol{P}$ and $\boldsymbol{Q}$ that have $k\ell$ and ℓn entries, respectively. Thus, the total number of parameters to be learned becomes $\ell(k + n)$. If $\ell \ll k$ and $\ell \ll n$, our proposed method involves a much lower number of parameters. Therefore, it is less likely to overfit the training data (especially when the size of the training data is small) and requires a shorter training time.

The two matrices $\boldsymbol{P}$ and $\boldsymbol{Q}^{-1}$ can be regarded as the functions that encode the features and the target variables, respectively. Specifically, matrix $\boldsymbol{P}$ encodes an instance's k features into a shorter vector of length ℓ, and matrix $\boldsymbol{Q}^{-1}$ encodes an instance's n target variables $t_{i1}, \ldots, t_{in}$ (or, more precisely, $\sigma^{-1}(t_{i1}), \ldots, \sigma^{-1}(t_{in})$) into a shorter vector of length ℓ. Therefore, matrix $\boldsymbol{P}$ captures the hidden relationship among the features, whereas matrix $\boldsymbol{Q}$ obtains the hidden relationship among the target variables. As a result, the MFMT model captures the relationships among not only the features but also among the target variables simultaneously.

4 EXPERIMENTS

4.1 Future browsing tendency prediction

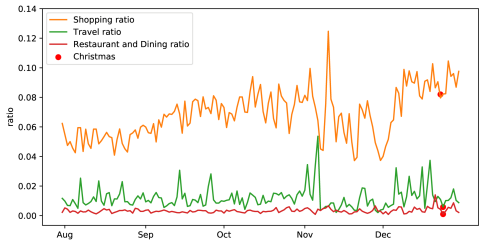
Table 1: Statistics for users' numbers of page views

min	Q1	Q2	mean	Q3	max
44	4,239	13,335	19,103	26,698	130,992

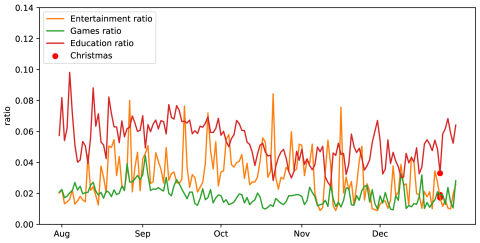
4.1.1 Dataset. A recent study showed that users' online shopping tendencies varied before and during the shopping holidays [16]. Here, we would like to observe users' browsing tendencies for other types of pages. We model such a problem as a multitask learning problem and apply the proposed MFMT model to it.

We recruited 672 users as target users. We recorded each user's online browsing history using a plug-in in Google Chrome. Eventually, we accumulated 12,837,216 browsing records. Table 1 shows the statistics for the page views of these users.

We selected 6 types of pages as our target categories: shopping, travel, restaurants and dining, entertainment, games, and education. The users' click ratio for each type of pages from August to September 2016 is shown in Figure 2, where Figure 2(a) shows the click ratios for shopping, travel, and restaurants and dining, and Figure 2(b) shows the click ratios for entertainment, games, and education. We use two subfigures for better visualization.



(a) Click ratio of shopping, travel, and restaurant and dining pages.



(b) Click ratio of entertainment, game, and education pages.

Figure 2: Click ratio of various types of pages on different dates.

4.1.2 Features and targets. We use users' demographic information and regular browsing habits to generate the features. The demographic information includes user gender, age, and relationship status. Browsing habits are represented by the user's browsing ratios for various types of pages. We define the browsing ratio of page category c for user i during period D based on Equation 8.

$$br(i, c, D) = \frac{\#\text{ user } i\text{'s visited pages of type } c \text{ within } D}{\#\text{ user } i\text{'s visited pages within } D}. \quad (8)$$

We use the page category instead of the URL or the domain name to define the browsing ratio because the distribution of users' visited URLs is highly imbalanced. The most popular page, that of Facebook (https://www.facebook.com/), accounts for 27.6% of visits, which makes the browsing ratio of Facebook a nondiscriminative feature. On the other hand, most URLs receive very little visits, which makes these features almost useless because only a small number of users visited these URLs. We mapped each URL to the corresponding category based on a webpage classification service.[1] For example, Facebook (https://facebook.com) is classified as "Social Networking", Gmail (https://gmail.com) is classified as "Web-based Email", and Amazon.com (https://amazon.com) is classified as "Shopping". Eventually, we obtained 88 classes of webpages, among which "Social Networking", "Search Engines and Portals", and "Web-based Email" were those with the highest browsing ratios.

We define a user's browsing behavior before Dec. 12, 2016, as the regular browsing behavior and the behavior from Dec. 12 to Dec. 25, 2016, as the holiday behavior. The regular browsing behavior is included as part of the features. We generated 6 target variables to indicate a user's behavior change during the holiday seasons according to 6 categories: shopping, travel, restaurants and dining, entertainment, games, and education. For a certain page type c, if a user i's browsing ratio in the holiday period is larger than that in

[1] http://www.fortiguard.com/webfilter

Table 2: F_1 score of various models on different page categories

model	shopping	travel	restaurants and dining	entertainment	games	education
KNN	0.574	**0.615**	0.528	0.440	0.492	0.484
Logreg	0.578	0.489	0.501	0.402	0.441	0.437
SVM	0.576	0.391	0.410	0.399	0.409	0.385
MFMT	**0.584**	0.570	**0.561**	**0.479**	**0.531**	**0.515**

Table 3: A comparison of logistic regression and MFMT

category	proportion in log	r_c
shopping	7%	1.04%
education	5.4%	17.85%
entertainment	3.3%	19.15%
games	1.9%	20.41%
travel	1.2%	13.91%
restaurants and dining	0.3%	11.98%

the regular period, we define it as a positive instance (i.e., $y_{ic} = 1$); otherwise, it is a negative instance (i.e., $y_{ic} = -1$).

4.1.3 Results. We compared the proposed MFMT model with 3 baseline models: k-nearest neighbors, the logistic regression classifier (Logreg), and support vector machine (SVM). For each method, we carefully fine-tuned the important hyperparameters of each model using the grid search. We compared the performance of the baseline models and our proposed method based on the F_1 score, which integrates both the precision and the recall in one number.

Table 2 shows the F_1 scores of these models. Our proposed MFMT better predicts a user's future behavior in 5 out of 6 categories. MFMT performs better likely because the hidden relationship among the targets are captured by MFMT but not by the baseline models, which treat each target as an independent output.

Since both the logistic regression classifier and MFMT are based on the generalized linear model, it is worth investigating the performance of the two methods. In Table 3, we list 6 target categories and the proportions of pages belonging to each of these categories in the log. We also list the improvement ratio for each category. The improvement ratio is defined by Equation 9.

$$r_c = \frac{F_1(\text{MFMT}, c) - F_1(\text{Logreg}, c)}{F_1(\text{Logreg}, c)}, \tag{9}$$

where $F_1(m, c)$ is the F_1 score of method m on target category c.

From Table 3, we observe that if the number of training instances is limited, MFMT performs significantly better than the logistic regression classifier. If the number of training instances increases, MFMT still performs slightly better than the logistic regression classifier. It appears that MFMT should be used especially if the size of training data is small.

4.2 Prediction results for various training sizes

We use a simulated dataset to compare the predictions of MFMT and the general linear approach, as the training size varies. We generated 20 features for each instance; for each instance, we generated 12 target variables based on a linear combination of the features together with some random factors.

Figure 3 shows the test RMSE (averaged over 12 target variables) of the two methods (MFMT and general linear model) as the size

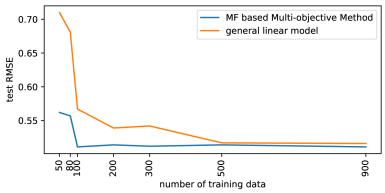


Figure 3: Relationship between the training size and the test RMSE of our proposed model and the general linear model based on the simulated dataset.

of the training data increases. With such an increase, the RMSE scores of both methods decrease (i.e., the predictions become more accurate). However, if the size of the training data is limited, our proposed MFMT algorithm performs much better than the baseline approach. Specifically, compared to the baseline approach, MFMT requires only 1/5 of the training data instances to reach the same level of test RMSE. Therefore, if the available training data is limited, our method is a better choice.

5 DISCUSSION

A common rule-of-thumb for many e-commerce companies is that during the special holidays, such as the Black Friday and Christmas Day, the number of sales increases. However, our collected dataset indicates that the increase in sales may be due to a few individuals. As a result, identifying the appropriate users correctly beforehand may provide a very large advantage to these companies.

Our proposed method can identify a user's tendency to visit the shopping websites based on her/his demographical information and the usual browsing behavior. Thus, e-commerce companies may use different marketing strategies to better advertise to different types of users. Compared with most supervised learning approaches, MFMT is excellent in simultaneously predicting users' future browsing trends for various types of pages. Additionally, when hidden correlations exist among the target variables, the MFMT model may capture such relationship and therefore much more effective if the size of the training dataset is limited.

ACKNOWLEDGMENTS

We acknowledge partial support by the Ministry of Science and Technology under Grant No.: MOST 107-2221-E-008-077-MY3. We are grateful to the National Center for High-performance Computing for computer time and facilities.

REFERENCES

[1] Fabrício Benevenuto, Tiago Rodrigues, Meeyoung Cha, and Virgílio Almeida. 2009. Characterizing user behavior in online social networks. In *Proceedings of the 9th ACM SIGCOMM Conference on Internet Measurement*. ACM, 49–62.

[2] Robert M Bond, Christopher J Fariss, Jason J Jones, Adam DI Kramer, Cameron Marlow, Jaime E Settle, and James H Fowler. 2012. A 61-million-person experiment in social influence and political mobilization. *Nature* 489, 7415 (2012),

295.
[3] Rich Caruana. 1997. Multitask learning. *Machine learning* 28, 1 (1997), 41–75.
[4] Olivier Chapelle, Pannagadatta Shivaswamy, Srinivas Vadrevu, Kilian Weinberger, Ya Zhang, and Belle Tseng. 2010. Multi-task learning for boosting with application to web search ranking. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 1189–1198.
[5] Hung-Hsuan Chen. 2017. Weighted-SVD: Matrix Factorization with Weights on the Latent Factors. *arXiv preprint arXiv:1710.00482* (2017).
[6] Hung-Hsuan Chen. 2018. Behavior2Vec: Generating Distributed Representations of Users' Behaviors on Products for Recommender Systems. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 12, 4 (2018), 43.
[7] Hung-Hsuan Chen, Liang Gou, Xiaolong Zhang, and Clyde Lee Giles. 2011. CollabSeer: a search engine for collaboration discovery. In *Proceedings of the 11th Annual International ACM/IEEE Joint Conference on Digital Libraries*. ACM, 231–240.
[8] Hung-Hsuan Chen, Liang Gou, Xiaolong Luke Zhang, and C Lee Giles. 2013. Towards the discovery of diseases related by genes using vertex similarity measures. In *Healthcare Informatics (ICHI), 2013 IEEE International Conference on*. IEEE, 505–510.
[9] Daxiang Dong, Hua Wu, Wei He, Dianhai Yu, and Haifeng Wang. 2015. Multi-task learning for multiple language translation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Vol. 1. 1723–1732.
[10] Kwang-Il Goh and In-Geol Choi. 2012. Exploring the human diseasome: the human disease network. *Briefings in Functional Genomics* 11, 6 (2012), 533–542.
[11] Yuchin Juan, Damien Lefortier, and Olivier Chapelle. 2017. Field-aware factorization machines in a real-world online advertising system. In *Proceedings of the 26th International Conference on World Wide Web Companion*. International World Wide Web Conferences Steering Committee, 680–688.
[12] Yuchin Juan, Yong Zhuang, Wei-Sheng Chin, and Chih-Jen Lin. 2016. Field-aware factorization machines for CTR prediction. In *Proceedings of the 10th ACM Conference on Recommender Systems*. ACM, 43–50.
[13] Yehuda Koren. 2008. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 426–434.
[14] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* 8 (2009), 30–37.
[15] Matthieu Kowalski. 2009. Sparse regression using mixed norms. *Applied and Computational Harmonic Analysis* 27, 3 (2009), 303–324.
[16] Cheng-You Lien, Guo-Jhen Bai, Ting-Rui Chen, and Hung-Hsuan Chen. 2018. Predicting user's online shopping tendency during shopping holidays. In *Conference on Technologies and Applications of Artificial Intelligence*.
[17] Cheng-You Lien Lien, Guo-Jhen Bai Bai, and Hung-Hsuan Chen. 2019. Visited Websites May Reveal Users' Demographic Information and Personality. In *2019 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*. IEEE.
[18] Jun Liu and Jieping Ye. 2010. Efficient l1/lq norm regularization. *arXiv preprint arXiv:1009.4766* (2010).
[19] Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. 2016. Predicting the Next Location: A Recurrent Model with Spatial and Temporal Contexts.. In *AAAI*. 194–200.
[20] Caroline Lo, Dan Frankowski, and Jure Leskovec. 2016. Understanding behaviors that lead to purchasing: a case study of pinterest. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 531–540.
[21] Andriy Mnih and Ruslan R Salakhutdinov. 2008. Probabilistic matrix factorization. In *Advances in Neural Information Processing Systems*. 1257–1264.
[22] Bamshad Mobasher, Honghua Dai, Tao Luo, and Miki Nakagawa. 2001. Effective personalization based on association rule discovery from web usage data. In *Proceedings of the 3rd international workshop on Web information and data management*. ACM, 9–15.
[23] Michael Moricz, Yerbolat Dosbayev, and Mikhail Berlyant. 2010. PYMK: friend recommendation at myspace. In *Proceedings of the 2010 ACM SIGMOD International Conference on Management of data*. ACM, 999–1002.
[24] Jaimie Y Park, Neil O'Hare, Rossano Schifanella, Alejandro Jaimes, and Chin-Wan Chung. 2015. A large-scale study of user image search behavior on the web. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. ACM, 985–994.
[25] Steffen Rendle. 2010. Factorization machines. In *Data Mining (ICDM), 2010 IEEE 10th International Conference on*. IEEE, 995–1000.
[26] Steffen Rendle. 2012. Factorization machines with libfm. *ACM Transactions on Intelligent Systems and Technology (TIST)* 3, 3 (2012), 57.
[27] Sebastian Ruder. 2017. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098* (2017).
[28] Narayanan Sadagopan and Jie Li. 2008. Characterizing typical and atypical user sessions in clickstreams. In *Proceedings of the 17th international conference on World Wide Web*. ACM, 885–894.
[29] Jie Tang, Sen Wu, Jimeng Sun, and Hang Su. 2012. Cross-domain collaboration recommendation. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 1285–1293.
[30] Ya Xue, Xuejun Liao, Lawrence Carin, and Balaji Krishnapuram. 2007. Multi-task learning for classification with dirichlet process priors. *Journal of Machine Learning Research* 8, Jan (2007), 35–63.
[31] Peng Yan, Xiaocong Zhou, and Yitao Duan. 2015. E-commerce item recommendation based on field-aware factorization machine. In *Proceedings of the 2015 International ACM Recommender Systems Challenge*. ACM, 2.
[32] Yuyu Zhang, Hanjun Dai, Chang Xu, Jun Feng, Taifeng Wang, Jiang Bian, Bin Wang, and Tie-Yan Liu. 2014. Sequential Click Prediction for Sponsored Search with Recurrent Neural Networks.. In *AAAI*, Vol. 14. 1369–1375.