

Which Tumblr Post Should I Read Next?

Zornitsa Kozareva

Yahoo!
701 First Avenue
Sunnyvale, CA 94089
zornitsa@kozareva.com

Makoto Yamada

Institute of Chemical Research
Kyoto University
Gokasho, Uji, 611-0011, Japan
myamada@kuicr.kyoto-u.ac.jp

Abstract

Microblogging sites have emerged as major platforms for bloggers to create and consume posts as well as to follow other bloggers and get informed of their updates. Due to the large number of users, and the huge amount of posts they create, it becomes extremely difficult to identify relevant and interesting blog posts.

In this paper, we propose a novel convex collective matrix completion (CCMC) method that effectively utilizes user-item matrix and incorporates additional user activity and topic-based signals to recommend relevant content. The key advantage of CCMC over existing methods is that it can obtain a globally optimal solution and can easily scale to large-scale matrices using Hazan's algorithm. To the best of our knowledge, this is the first work which applies and studies CCMC as a recommendation method in social media. We conduct a large scale study and show significant improvement over existing state-of-the-art approaches.

1 Introduction

The usage of social media sites has significantly increased over the years. Every minute people upload thousands of new videos on YouTube, write blogs on Tumblr¹, take pictures on Flickr and Instagram, and send messages on Twitter and Facebook. This has led to an information overload that makes it hard for people to search and discover relevant information.

Social media sites have attempted to mitigate this problem by allowing users to follow, or subscribe to updates from specific users. However, as

¹www.tumblr.com

the number of followers grows over time, the information overload problem returns. One possible solution to this problem is the usage of recommendation systems, which can display to users items and followers that are related to their interests and past activities.

Over time recommender methods have significantly evolved. By observing the history of user-item interactions, the systems learn the preferences of the users and use this information to accurately filter through vast amount of items and allowing the user to quickly discover new, interesting and relevant items such as movies, clothes, books and posts. There is a substantial body of work on building recommendation systems for discovering new items, following people in social media platforms, predicting what people like (Purushotham et al., 2012; Chua et al., 2013; Kim et al., 2013). However, these models either do not consider the characteristics of user-item adoption behaviors or cannot scale to the magnitude of data.

It is important to note that the problem of recommending blog posts differs from the traditional collaborative filtering settings, such as the Netflix rating prediction problem in two main aspects. First, the interactions between the users and blogs are *binary* in the form of follows and there is no explicit rating information available about the user's preference. The follow information can be represented as an unidirectional unweighted graph and popular proximity measures based on the structural properties of the graph have been applied to the problem (Yin et al., 2011). Second, the blog recommendation inherently has richer *side information* additional to the conventional user-item matrix (i.e. follower graph).

In Tumblr, text data includes a lot of information, since posts have no limitation in length, compared to other microblogging sites such as Twitter. While such user generated content charac-

terizes various blogs, user activity is a more direct and informative signal of user preference as users can explicitly express their interests by liking and reblogging a post. This implies that users who liked or reblogged the same posts are likely to follow similar blogs. The challenge is how to combine multiple sources of information (text and activity) at the same time. For the purpose, we propose a novel convex collective matrix completion (CCMC) social media recommender model, which can scale to million by million matrix using Hazan’s algorithm (Gunasekar et al., 2015).

Our contributions are as follows:

- We propose a novel CCMC based Tumblr blog post recommendation model.
- We represent users and blogs with an extensive set of side information sources such as the user/blog activity and text/tags.
- We conduct extensive experimental evaluations on Tumblr data and show that our approach significantly outperforms existing methods.

2 Convex Collective Matrix Completion

In this section, we formulate the Tumblr blog post recommendation task as collective matrix factorization problem and we describe our large-scale convex collective matrix completion method with Hazan’s algorithm (Gunasekar et al., 2015).

2.1 Model Description

Let $\mathbf{X}_1 \in \{0, 1\}^{n_{r_1} \times n_{c_1}}$ denote the user-blog (follower) matrix, where n_{r_1} is the number of users and n_{c_1} is the number of blogs. In this matrix, if the user i likes blog j , the (i, j) th element is set to 1. In addition to the user-blog matrix, we have other auxiliary matrices denoted by $\mathbf{X}_2 \in \mathbb{R}^{n_{r_2} \times n_{c_2}}$ and $\mathbf{X}_3 \in \mathbb{R}^{n_{r_3} \times n_{c_3}}$. For example, if we have an user activity matrix, we can use it as \mathbf{X}_2 , where $n_{r_2} = n_{r_1}$ and n_{c_2} is the number of activities. Moreover, if we have the content information of articles, we can use them as \mathbf{X}_3 . In this case, $n_{c_1} = n_{c_3}$ is the number of blogs, and n_{r_3} is the number of topics in LDA. Note that, \mathbf{X}_1 tends to be a sparse matrix, while \mathbf{X}_2 and \mathbf{X}_3 tend to be denser matrices. The final goal is to factorize \mathbf{X}_1 with the help of the auxiliary matrices \mathbf{X}_2 and/or \mathbf{X}_3 . First, we form a large matrix \mathbf{M} by concatenating all matrices $[\mathbf{X}_v]_{v=1}^V$ and then factorizing \mathbf{M} together with the regularizations.

In this paper, we adopt a convex approach (Bouchard et al., 2013; Gunasekar et al., 2015).

For example, for $V = 3$, the matrix \mathbf{M} is given as

$$\mathbf{M} = \begin{bmatrix} \cdot & \mathbf{X}_1 & \mathbf{X}_2 & \cdot \\ \mathbf{X}_1^\top & \cdot & \cdot & \mathbf{X}_3 \\ \mathbf{X}_2^\top & \cdot & \cdot & \cdot \\ \cdot & \mathbf{X}_3^\top & \cdot & \cdot \end{bmatrix}. \quad (1)$$

This framework is called convex collective matrix completion (CMC) (Singh and Gordon, 2008). The key advantage of the CCMC approach is that the sparse user-blog matrix \mathbf{X}_1 is factorized precisely with the help of the dense matrices \mathbf{X}_2 and/or \mathbf{X}_3 . Moreover, it has been recently shown that the sample complexity of the CCMC algorithm can be smaller than that of the simple matrix factorization approach (i.e., only factorize \mathbf{X}_1) (Gunasekar et al., 2015). Finally, the CCMC method can easily incorporate multiple sources of information. Over time if Tumblr provides new signals or if we decide to incorporate new features, CCMC can easily adopt them. Therefore, we believe that CCMC is very suitable for solving the Tumblr recommendation task.

2.2 CCMC-Hazan Algorithm

One of the key challenges of CCMC for Tumblr data is the scalability, since Tumblr has more than million users and hundred millions of blog posts. The original CCMC approach adopts Singular Value Thresholding (SVT) to solve the problem, and it works for small scale problems. However, SVT needs to solve $N \times N$ dimensional eigenvalue decomposition on each iteration, and thus it is not feasible to deal directly with the Tumblr data. Recently, Gunasekar et al. proposed an Atomic norm minimization algorithm for CCMC (Gunasekar et al., 2015) using the approximate SDP solver of Hazan (Hazan, 2008; Jaggi and Suvovsky, 2010). The optimization problem is given as

$$\begin{aligned} \min_{\mathbf{Z} \succeq 0} & \sum_{v=1}^V \|P_{\Omega_v}(\mathbf{X}_v - P_v(\mathbf{Z}))\|_F^2 \\ \text{s.t.} & \text{tr}(\mathbf{Z}) \leq \eta, \end{aligned} \quad (2)$$

where $\|\mathbf{X}\|_F$ is the Frobenius norm of matrix \mathbf{X} , P_{Ω_v} , which extracts the elements in the set, Ω_v is the set of non-zero indexes of \mathbf{X}_v , $P_v(\mathbf{Z}) = \mathbf{Z}_v \in \mathbb{R}^{n_{r_v} \times n_{c_v}}$, and $\eta \geq 0$ is a regularization parameter. The Hazan’s algorithm for CMF is summarized in Algorithm 1.

Algorithm 1 CCMC with Hazan’s Algorithm of (2)

Parameters: T (Number of iterations)
Rescale loss: $\hat{f}_\eta(\mathbf{Z}) = \sum_v \|P_{\Omega_v}(\mathbf{X}_v - P_v(\eta\mathbf{Z}))\|_F^2$
Initialize $\mathbf{Z}^{(1)}$
for all $t = 1, 2, \dots, T = \frac{4}{\epsilon}$ **do**
 Compute $\mathbf{u}^{(t)} = \text{approxEV}(-\nabla \hat{f}_\eta(\mathbf{Z}^{(t)}), \frac{1}{t^2})^2$
 $\alpha_t := \frac{2}{2+t}$
 $\mathbf{Z}^{(t+1)} = \mathbf{Z}^{(t)} + \alpha_t \mathbf{u}^{(t)} \mathbf{u}^{(t)\top}$
end for**return** $[P_v(\mathbf{Z}^{(T)})]_{v=1}^V$

The advantage of CCMC-Hazan is that it needs to compute only a top eigenvector on each iteration. Practically, on each iteration t in Algorithm 1, we just need to compute an $\frac{1}{t^2}$ -approximate largest eigenvalue of the sparse matrix with $|\Omega|$ non-zero elements, which needs $O(\frac{|\Omega|}{t})$ computation using Lanczos algorithm. On the other hand, the original CCMC algorithms adopt Singular Value Thresholding (SVT) method, which converges much faster than CCMC-Hazan. However, the SVT approach has to compute all eigenvalues in each iteration. Thus, CCMC-Hazan is more suited for large-scale dataset than CCMC-SVT. The details of CMC with Hazan’s algorithm, please refer to (Gunasekar et al., 2015).

3 Task Definition

We define our task as given a set of users and their Tumblr post adoption behavior over a period of time, the goal is to build a model that can discover and recommend relevant Tumblr posts to the users.

3.1 Evaluation Setup and Data

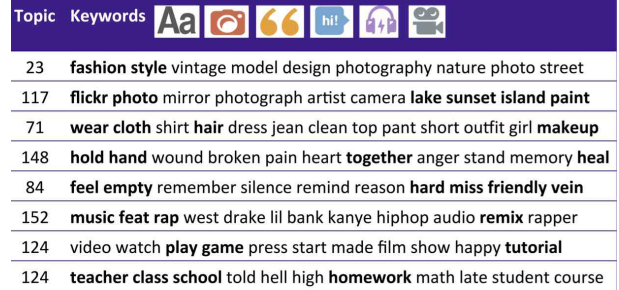
We set up our Tumblr post evaluation framework by considering the posting or reblogging of an item j by a user i as an adopted item, and otherwise as unadopted. We present each user with top k items sorted by their predicted adoption score and evaluate how many of the recommended items (posts) were actually adopted by the users.

For our post recommendation study, we used Tumblr data from July until September. We used the data from July to August for training, and tested on the data from September. This experimental set up simulates A/B testing.

From the derived data, we sampled 15,000 active users and 35,000 posts resulting in 5 million user-item adoptions for training and 8.6 million user-item adoptions for testing.

²approxEV(X, ϵ) computes the approximate top eigenvector of X upto ϵ error.

In post recommendation our CCMC-Hazan method uses an user-item matrix $\mathbf{X}_1 \in \{0, 1\}^{15000 \times 35000}$ and an item-topic matrix $\mathbf{X}_2 \in \mathbb{R}^{35000 \times 1000}$. To learn the topics we use Latent Dirichlet Allocation (LDA) (Blei et al., 2003). We represent a document as a collection of post description, captions and hashtags. We use 1000 topics for our experiments. Figure 1 shows some examples of the learned topics from the Tumblr posts.



Topic	Keywords	Aa	📷	🗨️	hit	👤	🎥
23	fashion style vintage model design photography nature photo street						
117	flickr photo mirror photograph artist camera lake sunset island paint						
71	wear cloth shirt hair dress jean clean top pant short outfit girl makeup						
148	hold hand wound broken pain heart together anger stand memory heal						
84	feel empty remember silence remind reason hard miss friendly vein						
152	music feat rap west drake lil bank kanye hiphop audio remix rapper						
124	video watch play game press start made film show happy tutorial						
124	teacher class school told hell high homework math late student course						

Figure 1: LDA.

3.2 Evaluation Metrics

To evaluate the performance of our collaborative matrix factorization approach for Tumblr post recommendation, we calculate precision (P), recall (R) and normalized discounted cumulative gain (nDCG) for top- k recommended posts.

- $\mathbf{P}@k$ as the fraction of adopted items by each user in top- k items in the list. We average precision@ k across all users.

- $\mathbf{R}@k$ as the fraction of adopted items that are successfully discovered in top- k ranked list out of all adopted items by each user. We average recall@ k across all users.

- $\mathbf{nDCG}@k$ computes the weighted score of adopted items based on the position in the top- k list. We average nDCG@ k of all users.

We set k to 10 since recommending too many posts is unrealistic. While nDCG@ k uses the position of correct answer in the top- k ranked list, it does not penalize for unadopted posts or missing adopted posts in the top- k ranked list. Therefore, to judge the performance of the algorithms, one has to consider all three metrics together. Intuitively a good performing model is the one that has high $\mathbf{P}@k$, $\mathbf{R}@k$ and $\mathbf{nDCG}@k$.

3.3 Comparison Against State-of-art Models

In addition to evaluating the performance of our algorithm on Tumblr post recommendation, we

also conducted a comparative study against existing state-of-the-art models.

Item-based³ The item-based model recommends items that are similar to what the users have already adopted (Karypis, 2001). The model does not use textual information and only uses adopted items to compute the similarity between the items. The similarity metric is the Tanimoto Coefficient, which is used to handle binary ratings.

User-based The user-based model recommends items that are adopted by other users with similar taste (Herlocker et al., 1999). The model does not use textual information and only uses adopted items to compute the similarity between the users. Similar to the item-based recommendation, we use the Tanimoto Coefficient. We choose top k items using k-Nearest Neighbor of similar users.

MC⁴ Alternating least squares (ALS) is matrix completion (MC) based collaborative filtering model, which was originally introduced to model user-movie rating prediction using mean-square loss function with weighted λ regularization (Zhou et al., 2008). The model does not use textual information or signals for adopted items.

PMC⁵ Probabilistic Matrix Completion (Salakhutdinov and Mnih, 2008) is a probabilistic linear model with Gaussian observation noise that handles very large data sets and is robust to sparse user-item matrix. Similar to MC, PMC models the user-item adoption as the product of two K -dimensional lower-rank user and item hidden variables. The model does not use textual information, but unlike the previous methods it uses information on unadopted items.

CF Collaborative Filtering model with softmax function (Guadagni and Little, 1983; Manski, 1975; McFadden, 1974) captures the adoption and un-adoption behavior of users on items in social media. The model does not use textual information, but it uses signals on unadopted items. CF allows us to study the gain of performance in post recommendation when softmax function is used instead of the objective functions used in MC and PMC.

CTR Collaborative Topic Regression (Wang and Blei, 2011) was originally introduced to recommend scientific articles. It combines collaborative filtering PMC and probabilistic topic model-

Method	PRC@10	RCL@20	AUC
Item-based	0.24	0.08	0.42
User-based	0.32	0.11	0.51
MC	0.31	0.11	0.52
PMC	0.35	0.12	0.55
CF	0.36	0.13	0.56
CTR	0.39	0.14	0.59
CCMC-Hazan	0.41	0.16	0.61

Table 1: Tumblr Post Recommendation Results

ing LDA. It captures two K -dimensional lower-rank user and item hidden variables from user-item adoption matrix and the content of the items. This model uses textual information and signal for un-adopted items.

3.4 Results

Table 1 shows the obtained results of the proposed CCMC-Hazan method against the remaining recommendation models. The simple user and item based recommendations have the lowest performance. This shows that for accurate post recommendation using direct post and user information is insufficient and one needs stronger context driven signals. This is shown in the performance of the CF and CTR methods, which model context information with LDA and perform better than the rest of the models.

However, when we compare the performance of our collaborative matrix completion method, we can see that the rest of the models have significantly lower performance. The main reasons are due to the dense information of CCMC-Hazan method and the fact that our method optimizes a convex function whereas the MC, CF and CTF models can get stuck in local optima.

4 Conclusions

Recommending blog posts is one of the major tasks for user engagement and revenue generation in online microblogging sites such as Tumblr. In this paper, we propose a convex collective matrix completion based recommendation method that effectively utilizes the user-item matrix as well as rich side information from users and/or items. We evaluate the proposed method on real-world dataset collected from Tumblr. Extensive experiments demonstrate the effectiveness of the proposed method in comparison to existing state-of-the-art approaches.

³<https://mahout.apache.org>

⁴www.graphlab.org

⁵<http://www.cs.cmu.edu/~chongw/citeulike/>

Acknowledgement

We would like to thank the anonymous reviewers for their valuable feedback and grant MEXT KAKENHI #16K16114.

References

- David Blei, Andrew Ng, and Michael Jordan. 2003. Latent dirichlet allocation. *The Journal of Machine Learning Research*, 3:993–1022.
- Guillaume Bouchard, Shengbo Guo, and Dawei Yin. 2013. Convex collective matrix factorization. In *AISTATS*.
- Freddy Chong Tat Chua, Hady W. Lauw, and Ee-Peng Lim. 2013. Generative models for item adoptions using social correlation. *IEEE Trans. on Knowl. and Data Eng.*, 25:2036–2048.
- Peter M Guadagni and John DC Little. 1983. A logit model of brand choice calibrated on scanner data. *Marketing science*, 2(3):203–238.
- Suriya Gunasekar, Makoto Yamada, Dawei Yin, and Yi Chang. 2015. Consistent collective matrix completion under joint low rank structure. In *AISTATS*.
- Elad Hazan, 2008. *Sparse Approximate Solutions to Semidefinite Programs*, pages 306–316. Springer Berlin Heidelberg.
- Jonathan L. Herlocker, Joseph A. Konstan, Al Borchers, and John Riedl. 1999. An algorithmic framework for performing collaborative filtering. In *SIGIR*.
- Martin Jaggi and Marek Sulovsky. 2010. A simple algorithm for nuclear norm regularized problems. In *ICML*.
- George Karypis. 2001. Evaluation of item-based top-n recommendation algorithms. In *CIKM*.
- Jihie Kim, Jaebong Yoo, Ho Lim, Huida Qiu, Zornitsa Kozareva, and Aram Galstyan. 2013. Sentiment prediction using collaborative filtering. In *ICWSM*.
- Charles F Manski. 1975. Maximum score estimation of the stochastic utility model of choice. *Journal of Econometrics*, 3(3):205–228.
- Daniel McFadden. 1974. Conditional logit analysis of qualitative choice behavior. In *Frontiers in Econometrics*, pages 105–142.
- Sanjay Purushotham, Yan Liu, and C.-C. Jay Kuo. 2012. Collaborative topic regression with social matrix factorization for recommendation systems. *CoRR*.
- Ruslan Salakhutdinov and Andriy Mnih. 2008. Bayesian probabilistic matrix factorization using markov chain monte carlo. In *ICML*.
- Ajit P. Singh and Geoffrey J. Gordon. 2008. Relational learning via collective matrix factorization. In *KDD*.
- Chong Wang and David M. Blei. 2011. Collaborative topic modeling for recommending scientific articles. In *KDD*.
- Dawei Yin, Liangjie Hong, and Brian D Davison. 2011. Structural link analysis and prediction in microblogs. In *CIKM*.
- Yunhong Zhou, Dennis Wilkinson, Robert Schreiber, and Rong Pan. 2008. Large-scale parallel collaborative filtering for the netflix prize. In *AAIM*.