Improving User Topic Interest Profiles by Behavior Factorization

Zhe Zhao[1*], Zhiyuan Cheng[2], Lichan Hong[2], Ed H. Chi[2]
[1]Department of EECS, University of Michigan, Ann Arbor, MI, USA
[2]Google Inc, Mountain View, CA, USA
zhezhao@umich.edu, zcheng@google.com, lichan@google.com, edchi@google.com

ABSTRACT

Many recommenders aim to provide relevant recommendations to users by building personal topic interest profiles and then using these profiles to find interesting contents for the user. In social media, recommender systems build user profiles by directly combining users' topic interest signals from a wide variety of consumption and publishing behaviors, such as social media posts they authored, commented on, +1'd or liked. Here we propose to separately model users' topical interests that come from these various behavioral signals in order to construct better user profiles.

Intuitively, since publishing a post requires more effort, the topic interests coming from publishing signals should be more accurate of a user's central interest than, say, a simple gesture such as a +1. By separating a single user's interest profile into several behavioral profiles, we obtain better and cleaner topic interest signals, as well as enabling topic prediction for different types of behavior, such as topics that the user might +1 or comment on, but might never write a post on that topic.

To do this at large scales in Google+, we employed matrix factorization techniques to model each user's behaviors as a separate example entry in the input user-by-topic matrix. Using this technique, which we call "behavioral factorization", we implemented and built a topic recommender predicting user's topical interests using their actions within Google+. We experimentally showed that we obtained better and cleaner signals than baseline methods, and are able to more accurately predict topic interests as well as achieve better coverage.

Categories and Subject Descriptors

H.2.8 [**Database applications**]: Data Mining

Keywords

Personalization, Behavior Factorization, User Profiles

1. INTRODUCTION

An important aspect of building content recommenders is the construction of personalized user profiles, which consists of two important sub-problems. One is feature engineering, which is the gathering of signals that indicate users' long-term and short-term interests. The other is the proper utilization of these signals using data mining and machine learning approaches. Due to the scale of today's social media platforms, we have a great number of users as well as topics/items. To build a user profile, signals are generally gathered and then aggregated directly to feed into topic modeling algorithms such as Latent Dirichlet Allocation (LDA) or matrix factorization.

Intuitively, different behavioral signals should be weighed differently in building user profiles. For example, users typically only publish posts or comment on topics that are more important to them, while a +1 on a post might also involve topics that are only somewhat interesting. As a concrete example, an academic research user might typically post about her research interests, and also comment on the latest politics and current events. However, she might tend to only +1 posts about bicycling, gardening, and parenting, preferring to keep her hobbies somewhat private. In fact, we might think of her as having multiple personalities, expressed through these actions.

Perhaps more importantly, different behavioral actions represent users' different publication and consumption interests. Since writing a post and resharing posts means the user is distributing information to her followers, these publishing acts can also be thought of as actions that curate a particular image to her audiences. Commenting and +1 actions are much more like reactions to others in a conversation, so these actions more likely represent topics that the user is interested in consuming from others. Obviously, there should be overlaps between the two sets of topic interests. These actions parallel ideas from a well-known social scientist Erving Goffman, whose seminal book called "*The Presentation of Self in Everyday Life*[5]" emphasizes how people are 'performing' for others like on a stage when they are in contact with others. Applied to social media, one can think of users as putting on a performance for her followers in order to control or guide the impression that others will form of her as a person.

Motivated by these ideas, we first performed an analysis on behavioral actions on Google+ social network. We will describe this analysis in more detail later in the paper, but in short, for each user, we gathered the two set of topics that she publishes and consumes separately. We found that the average Jaccard Index between the two sets is only 0.122, which suggests that users do tend to publish and consume information on somewhat different topics.

Because user have different topic interests depending on the behaviors, we wish to build different user profiles and make different topic interest predictions for different behaviors. As usual, we care most about recommending content for user to consume, but there are instances where we might want to recommend content that the user would want to reshare instead, for instance.

Here the main task is to predict the preferences of a user for a particular topical item, using observed implicit or explicit topical preferences. We model this recommendation task as a matrix completion problem, which can be solved using matrix factorization

*Work done while interning at Google

Copyright is held by the International World Wide Web Conference Committee (IW3C2). IW3C2 reserves the right to provide a hyperlink to the author's site if the Material is used in electronic media.
WWW 2015, May 18–22, 2015, Florence, Italy.
ACM 978-1-4503-3469-3/15/05.
http://dx.doi.org/10.1145/2736277.2741656 .

([16]) in three steps: 1. Learn a latent embedding space from the user-item matrix; 2. Represent both user and items in embedding space; 3. Compute the similarity of user to items as the preferences. Current state-of-the-art recommenders typically model user-item preferences using a single rating score for each observed user-item pair, due to scalability reasons.

Instead, in our approach, for each user, we use the embeddings built from factorization techniques to separately model the topic interests for different behaviors. We then use the embeddings to predict user preferences for different topics under different behavioral contexts. We call our approach "Behavioral Factorization".

To be precise, we first build a embedding model by separating each user's preferences into several preferences in different behavior contexts. This separation is key to getting a clean topic interest signals to train the embedding model. We then combine a user's topic preferences across different behaviors to make predictions of topical interests for, as an example, consumption. That is, our approach provides different recommendations for different behavioral engagement types. For example, given a user's activities on content topics such as creating post about "data mining", or +1 post about "video games", our prediction framework will recommend topics for different behavioral actions, e.g., recommend "minecraft" posts to consume, but "machine learning" posts to reshare.

The contributions of this work are:

- We introduce the general notion of separating the behavioral engagement types in the construction of an latent embedding model for user interest profiles.
- We develop a method to perform "behavioral factorization", in which we apply matrix factorization to the user-behavior by item matrix directly to construct an embedding space, which is used in the prediction of future topic interests.
- We evaluate behavioral factorization on a large-scale data set and show the amount of improvement obtained in building user profiles.

2. RELATED WORK

2.1 Building User Profiles

The diversity and volume of information shared on social media is overwhelming for many users. Therefore, the construction of topic interest profiles is an important part of personalized recommenders in social media systems. We are inspired by a wide variety of past personalization research that utilize behavioral signals.

Search engine researchers have utilized user profiles to provide personalized search results [9], [11], [4], [30], [35]. Usually user profiles are represented as vectors in a high dimensional space [1], [23], with vectors denoting users' preferences on different items, (e.g., web pages, movies, or social media posts.) or users' preferences on various topics (e.g., keywords representing topics, or topic categories from a taxonomy.)

2.1.1 Matrix Factorization & Embedding Models

In one family of recommender approach called Collaborative Filtering (CF) [26], systems typically model user preferences using a user-by-item matrix, with each entry representing a user's rating on a corresponding item. Therefore, a row in the input matrix is a particular user's expressed preferences of the items in the system. A user's unknown preference on a certain item is inferred using matrix completion and researchers have made great progress in using matrix factorization methods effectively for this problem [16], [17].

In our paper, instead of representing user profile by preferences of items (posts), we focus on inferring user's topical interests, and topics are represented as entities in Google Knowledge Graph [29], such as 'basketball' or 'video games'. Researchers have utilized matrix factorization to create embedding models, as well as generative models such as Latent Dirichlet Allocation (LDA) [2] to build user profiles [21]. Matrix factorization as well as generative models learn latent embedding spaces, where preferences can be calculated by similarity between user and item's latent embedding factors.

Compared to item-based approach, topic-based approach should be more scalable for applications in social media, where the number of actual items (posts) is large. Instead, we can make predictions of a user's interest in an item by calculating the relevance between the user's topic interests and the post's topics.

2.2 Personalized User Profiles in Social Media

Just as in other recommendation problem, social media researcher often treat building a user profile as the first task in building a personalized recommender. Researchers have applied matrix factorization and generative models such as LDA to modeling the user-topic matrix in social media and to building user profiles in particular [8], [7], [34], [3], [6], [15], [33], [27]. For example, Guy et al. built user profiles based on content and item preferences, and then provided personalized recommendations for social media items such as bookmarks and social software [7], [8]. Chen et al. built user topic profiles and provided personalized recommendation of conversations on Twitter. User profiles are also used to provide recommendations of friends [6], communities [33], and activities such as mentioning [33] and commenting [27], etc.

For a user profile, user preferences can be inferred using implicit feedback such as user's activities [8]. In contrast, in traditional recommender systems, CF usually requires user to have some explicit input on rating some items, e.g., movies and books, etc., which brings extra burden for users. For example, Hu et al. proposed a matrix factorization approach that leverages implicit feedback and was shown to be efficient with large-scale sparse data sets [13]. Extending on this idea, Noel et al. proposed a novel objective function in matrix factorization that considers feature-based similarity as well as user-user information in social media [24].

2.3 Contextual Personalization

Social media platforms also provide us rich contextual information such as who comments on who's post on what topic and when. Many recent works discussed how to make use of the rich context to learn better user profiles. Collective Matrix Factorization has been proposed by Singh et al. to provide recommendation in heterogeneous network where context information is used [28], [19]. Probably closest to our work are: (a) Liu et al. propose a social-aided context-aware recommender systems for books and movies, which makes use of rich context to partition user-item matrix into multiple matrices [20]; (b) Jamali et al. propose a context-dependent matrix factorization model to create user profiles for recommendation in social network [14].

Beyond matrix factorization techniques, context-aware generative models have been proposed by researchers to help creating user profiles and latent semantic models in social media platforms such as Twitter [25], [32], [36]. For example, Zhang et al. proposed a two-step framework that first discover different topic domains using generative models, and then provide recommendation within each domain using matrix factorization methods [37]. Their idea that different users may be interested in different domains is relevant to our work in differentiating user's behaviors. But we focus instead on how each user's topical interests are separated by different types of behaviors.

Researchers have also used content from a two or more different social media platform to build improved user profiles. Li et al. proposed a transfer learning approach that can factorize two matrices from two domains together using information from each other [18]. Hu et al. propose a triadic-factorization-based approach to factorize user-item-domain tensor to provide personalized recommendation across domains [12].

2.4 Behavioral Factorization

> "If we see perception as a form of contact and communion, then control over what is perceived is control over contact that is made, and the limitation and regulation of what is shown is a limitation and regulation of contact." — Erving Goffman, *The Presentation of Self in Everyday Life* [5].

Recent works have shown us that differentiating various contexts can improve the quality of user profiles. In our paper, we show that because social media users interact with different topics using different types of behaviors, we should use behavior types as an important context. We should also build multiple user profiles for different behavior types, then use these different profiles flexibly in different behavioral-dependent recommendation, e.g., recommend content to read, or recommend content to reshare, etc.

Sociologists have shown that people present different images to others in their everyday life and their everyday conversations engage in different topics with different audiences [5]. The emergence of social media has also drawn sociologists' interests to study this phenomenon in online communities. For instance, sociologists theorize that, because users do not have a clear idea of the exact audiences in public social media, they end up with blurred context boundaries [22]. However, because different types of behaviors, such as posting or commenting, affect very different audiences, our analysis below suggest that users still show different 'identities', exhibiting different types of behaviors around different topics in social media. By conducting qualitative study, Zhao et al. point out that users experience social media platforms such as Facebook as multiple different functional regions, similar to their multiple identities in real life [38]. To the best of our knowledge, our paper is the first work that utilizes users' different online presentations on a real-world social media platform.

3. GOOGLE+ BEHAVIORAL ANALYSIS

To motivate our work further, here we analyze users' online behaviors in Google+ using an anonymized data set. We first extract the topic entities in posts as our features to construct feature vectors for each post. For each users' behavioral actions on posts, we aggregate the corresponding post feature vectors to build an entity vector for each user-behavior combination. Then we coarsely measure the differences of topical interests represented by these user-behavior entity vectors. We show that there exists significant differences between these vectors, which motivates our approach to utilize behavioral factorization to model different behavioral types.

3.1 Dataset Description

We use anonymized Google+ users' public behaviors in May 2014 to conduct our analysis. We analyzed all user actions on all public posts, and each record is represented as a tuple: (u, b, E), where a user u (with an anonymized id) used behavior b to engage with a post containing E set of entities. There are four types of behaviors in our data: Create Post, Reshare, Comment, and +1.

Instead of using low-level features such as word tokens, we extract higher-level semantic concept from the post in the form of entities using Google's Knowledge Graph [29], which contains entities that represent concepts such as computer algorithms, landmarks, celebrities, cities, or movies. It currently contains more than 500 million entities, which provides both wide and deep coverage of topics.

Behavior	Comment	Plus One	Reshare	Create Post
Comment	1	0.092	0.050	0.102
+1	0.092	1	0.048	0.071
Reshare	0.050	0.048	1	0.012
Create Post	0.102	0.071	0.012	1

Table 1: Average Jaccard similarity between pairs of behavior types

Entity extraction is an open research problem and not a focus of our work here, but in a nutshell, we utilized an entity extractor based on standard entity recognition approaches that utilize prior co-occurrences between entities, likelihood of relatedness between entities, entities' positions within the text, and then finally ranking the topicality of the entity for the text.

Given a post, we use its corresponding Knowledge Graph entities as features to represent its topics. Therefore, each E in input tuple (u, b, E) is a set of Knowledge Graph entities. For example, if a user u_1 created a post with his dog's picture, this behavior might correspond to $(u_1, CreatePost, \{\text{“}Dog'', \text{“}Pet'', \ldots\})$. If another user u_2 commented on a post with a YouTube video about Minecraft on Xbox, this behavior might correspond to the tuple $(u_2, Comment, \{\text{“}Minecraft'', \text{“}Xbox'', \ldots\})$.

3.2 Measuring differences among behaviors

For each user, we aggregate the entities from the posts she interacted with using a particular type of behavior. In the end, for each user, we obtain four sets of topic entities corresponding to the four behavior types mentioned above.

We then use the Jaccard similarity index to measure the differences between the sets. Jaccard similarity index is a common metric for measuring the similarity between two sets and is calculated as follows given sets A and B: $J(A, B) = \frac{A \cap B}{A \cup B}$.

After we calculate Jaccard similarity scores of different behaviors for each user, we then average the scores across all users. We filter out users who have less than 10 entities as non-active. Table 1 shows the results of the average Jaccard similarities. We can see that the Average Jaccard Index between any two types of behaviors is low. Take user's commenting and +1 behaviors as an example, only 9% of the topics overlapped between these two behaviors. We also measure the difference between user's publishing and consuming behaviors. We combine the entities of user commenting and +1 behaviors as a set of entities of consuming, and we combine the entities of user creating post and resharing behaviors as a set of entities of publishing. The average Jaccard Index is 0.122. The low overlap rate of these Jaccard scores suggests that user acts differently in different behaviors.

3.3 Discussion

The results of the analysis show that, for each user, she typically have different topic interests with each behavior. That is, she will often create posts on topics that are different from the topics she comments on. The results suggest that general non-behavior-specific user profiles might not perform well in applications that emphasize different behavior types.

Content recommenders usually targets predicting contents for user to consume, which might be better reflected by behaviors such

as commenting and +1. In other contexts, we might instead predict what topics users would create posts about. Therefore, by creating topic preferences for each behavior type separately, behavior-specific user profile might have better performance in different recommendation contexts.

In summary, users' various behaviors in social media contain important contextual information, which might help us improve the performance of user personalization profiles. We showed that users have significantly different topical interests reflected by their different behavior types in G+, and that building multiple profiles with separate behavior types allows us to tailor our content recommendation systems for various behavioral contexts.

4. PROBLEM DEFINITION

4.1 Input Behavioral Signals

Instead of building one single profile per user, we propose to build multiple profiles for a user to represent her different behavior types. Specifically, here we take users' behaviors on social media posts as input, and output a set of topic interest vectors to represent each user's different types of profiles.

Given a set of users $\mathfrak{U}$, a set of different behaviors types $\mathfrak{B}$, and a set of features that can represent social media content $\mathfrak{E}$, the input data can be represented as a set of tuples:

$$\mathfrak{T} = \{t_i = (u_i, b_i, E_i), i = 1, \ldots, N\}$$

where $u_i \in \mathfrak{U}$, $b_i \in \mathfrak{B}$, and $E_i \subset \mathfrak{E}$. Each t_i represents a user's action on a particular piece of social media content. For example, a t_i can be creating a post, or commenting on one. E_i is the set of features of that post. Here since we are building user topic profiles, we use entities from Google Knowledge Graph as our feature set. However, in general, $\mathfrak{E}$ can be any low-level (i.e., words) or high-level features (i.e., other entities, or even demographic features).

4.2 User Profiles

We define user profiles as sets of vectors in the feature space $\mathfrak{E}$:

$$\mathfrak{P} = \{P_u = \{\mathbf{V_{u_B}}\}\}$$

where $u \in \mathfrak{U}$, $B \subset \mathfrak{B}$, and P_u is the user profile for user u, and $\mathbf{V_{u_B}}$ is a vector of user u's preferences on features corresponding to her behavior types B. P_u can be thought of as a user tensor.

B can be either a single behavior type (e.g., creating a post), or a combination of behavior types (e.g., both creating a post and resharing a post). To be precise:

$$\mathbf{V_{u_B}} = (p_{u_B}^{e_1}, p_{u_B}^{e_2}, \ldots, p_{u_B}^{e_k}), e_j \in \mathfrak{E}$$

where $p_{u_B}^{e_j}$ is user u's behavior types B's preference on feature e_j, for $j = 1, \ldots, k$.

In the following sections, we propose our behavioral factorization approach to build user profiles, and compare the quality of the profiles with profiles built using the traditional matrix factorization technique.

5. OUR APPROACH

Here we introduce our behavior factorization approach to build user profiles for personalized recommendation, which includes three steps, as shown in Figure 1 and 2.

- Step 1: Given input user action tuples $\mathfrak{T}$ defined in Section 4, we first build matrices of different behavior types. This corresponds to the left part in Figure 1.
- Step 2: We factorize the matrices generated in step 1 to learn the latent embedding space. This corresponds to the right part in Figure 1
- Step 3: At last we build user profiles by making predictions of topics of interest utilizing the learned latent space. This creates profiles $P_u = \{\mathbf{V_{u_B}}\}$ for each user u. This corresponds to Figure 2.

We introduce each of these steps in turn below.

5.1 Step 1: Building matrices of different behavior types

In typical matrix factorization techniques, input user-item matrix R is represented as a $N \times K$ matrix, in which N is the number of users and K is the number of items. R is factorized into the product of two matrices, matrix X of $N \times L$, and matrix Y of $K \times L$. In other words, both row vector and column vector in R have been mapped into a L-dimensional latent embedding space. With this learned latent space, for any observed row vector in user-item matrix, the learned embedding space can be used to help complete the particular row vector to get complete estimated preferences of a user on items.

Since we are building user-topic-based profiles, instead users' interests on *items* ($N \times K$ user-item matrix) as input, we use users' interests on *topics* ($N \times K$ user-topic matrix) as input.

In addition, instead of using only one $N \times K$ matrix as input, we build and factorize multiple matrices as described below, including: (a) The traditional $N \times K$ matrix – referred to as Behavior Non-specific User-topic Matrix (**BNUM**); (b) Single Behavior-Specific User-topic Matrix (**SBSUM**); (c) Combined Behavior-Specific User-topic Matrix (**CBSUM**).

5.1.1 *Behavior Non-specific User-topic Matrix* BNUM

Here each entry indicates a user's implicit interests on a particular topic. Given input user tuples $\mathfrak{T} = \{t_i = (u_i, b_i, E_i), i = 1, 2, \ldots\}$, we first pull out the tuples $\mathfrak{T}_u$ involving user u:

$$\mathfrak{T}_u = \{t_j = (u_j, b_j, E_j)\}, t_j \in \mathfrak{T} \wedge u_j = u$$

Then we generate observed value for each user and topic pair:

$$r_{ui} = r(\mathfrak{T}_u, i)$$

That is, we first extract out all tuples $\mathfrak{T}_u$ involving user u and apply the function r to calculate implicit interests, given user u's tuples involving topic i. There are many possible forms of this function, and different weights can be trained for different behaviors. We use the following equation here in baseline methods as well as in later sections to calculate implicit interests:

$$r_{ui} = \frac{(\sum_{\mathfrak{T}_u} \sum_{e \in E_j} \sigma_i(e)) + 1}{(\sum_{\mathfrak{T}_u} \|E_j\|) + (\| \cup_{\mathfrak{T}_u} E_j\|)} \quad (1)$$

where $\sigma_i(e)$ is 1 if $i = e$ and 0 otherwise. That is, the implicit interest of topic i from user u is calculated by the number of occurrences of i in all user u's behaviors, divided by the sum of occurrences of all items. We smooth the value using additive smoothing.

5.1.2 *Single Behavior-Specific User-topic Matrix* SBSUM

Both **SBSUM** and **CBSUM** separate behavior types to generate separate user-topic matrices. Given a specific set of behavior types $B \subset \mathfrak{B}$, we want to build matrix $R_B = \{r_{ui}^B\}$, in which each entry represents the implicit interest only from behavior types in B.

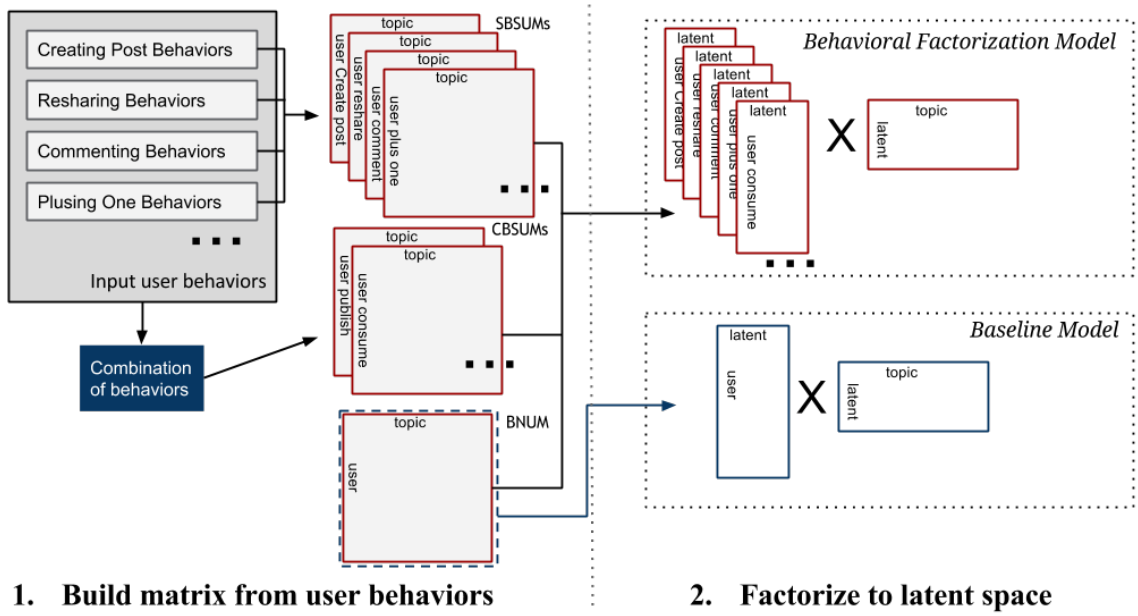


Figure 1: Framework of generating matrices and factorization.

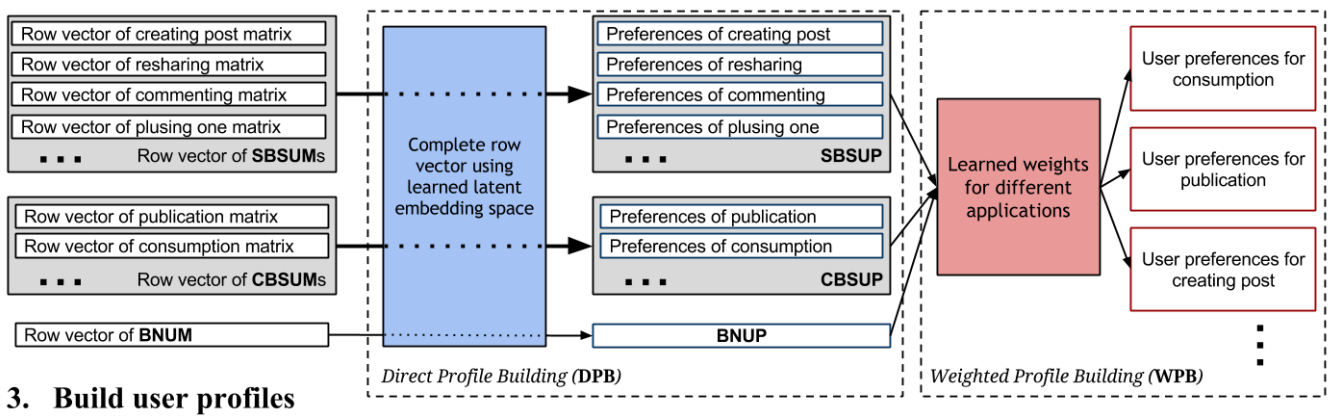


Figure 2: Building user profiles using latent embedding space.

We use the same method as in Equation 1, but add constraints to filter out behavior types not in B:

$$r_{ui}^{B} = \frac{(\sum_{\mathfrak{T}_u \wedge b_j \in B} \sum_{e \in E_j} \sigma_i(e)) + 1}{(\sum_{\mathfrak{T}_u \wedge b_j \in B} \|E_j\|) + (\| \cup_{\mathfrak{T}_u \wedge b_j \in B} E_j \|)} \tag{2}$$

Using this equation, for each B, we can build a matrix that represents users' observed implicit feedback with behavior types in B, which can be set as either a single behavior type, or a set of multiple behavior types. Therefore, based on the choices of B, we can build two types of behavior-specific user-topic matrices: Single Behavior-Specific User-topic Matrix **SBSUM**, and Combined Behavior Specific User-topic Matrix **CBSUM**.

First, we build one user-topic matrix for each behavior type, such as creating post, resharing, commenting or +1. The entry of each matrix is the observation value r_{ui}^{B} calculated by Equation 2, where B is a single behavior. Given $\mathfrak{B} = \{b_1, b_2, \ldots, b_M\}$ as a set of all behavior types, we generate the following M single behavior-specific user-topic matrices (**SBSUM**):

$$\begin{cases} R_{b_1} = \{r_{ui}^{B}\}, \quad B = \{b_1, b_1 \in \mathfrak{B}\} \\ R_{b_2} = \{r_{ui}^{B}\}, \quad B = \{b_2, b_2 \in \mathfrak{B}\} \\ \ldots \\ R_{b_M} = \{r_{ui}^{B}\}, \quad B = \{b_M, b_M \in \mathfrak{B}\} \end{cases} \tag{3}$$

5.1.3 *Combined Behavior Specific User-topic Matrix* **CBSUM**

In building **SBSUM**, we create M matrices, each of which represents a single behavior type. However, we also want to capture topic interests of combinations of more than one related behavior types. For example, in G+, both creating and resharing posts generate content that is broadcast to followers, and these two behavior types can be combined together to represent the user's *publication*.

Meanwhile, commenting and +1'ing posts both indicate user's consumption of post. Combining them together can represent topics of interests in the user *consumption*. Therefore, given sets of behavior types, with each set being a subset of $\mathfrak{B}$, $\{B_1, B_2, \ldots, B_P\}$, we build P matrices, each of which represent user's combined be-

haviors in each set of behavior types.

$$\begin{cases} R_{B_1} = \{r_{ui}^B\}, \;\; B = \{B_1, B_1 \subset \mathfrak{B}\} \\ \dots \\ R_{B_P} = \{r_{ui}^B\}, \;\; B = \{B_P, B_P \subset \mathfrak{B}\} \end{cases} \tag{4}$$

5.2 Step 2: Learning latent embedding space

Here we introduce a matrix factorization technique for building user topic profile as the baseline method. In addition, we introduce our proposed method that extends the baseline algorithm to behavior factorization.

5.2.1 Baseline model: Matrix factorization

After building Behavior Non-specific User-topic Matrix (**BNUM**), we learn a latent embedding space that can be used to complete observed user-topic matrix to get predicted user-topic preferences. In recommender research, there are many efforts trying to improve matrix factorization techniques in both academia and industry. Here we use the factorization techniques as proposed in Hu et al. [13].

There is a very specific reason why we adopted Hu et al.'s approach. In social media platforms, implicit interest signals are easier to obtain for most users than explicit interest signals. There are just more implicit interest signals in the system. However, many recommender algorithms do not consider the potential differences between using explicit interest vs. implicit interest signals. Hu et al. [13] proposed a matrix factorization method that addressed this difference. It is worth noting that all other matrix factorization methods that work on user-item matrix can also be applied in our framework to build user profiles using behavior factorization. Note that 'topic' in user-topic matrix is the same as item in user-item matrix in the discussion below.

Given observation of the user-item matrix obtained from implicit interests from r_{ui}, Hu et al. splits observations into two variables: preference p_{ui} and confidence c_{ui}. Here p_{ui} is a binary variable that represents whether user u has interests in item i:

$$p_{ui} = \begin{cases} 1 & r_{ui} > 0 \\ 0 & r_{ui} = 0 \end{cases}$$

Confidence c_{ui} represents the confidence level of the preference p_{ui}. It indicates how confident we are in the interest value. It can be calculated in the following way: $c_{ui} = 1 + \alpha r_{ui}$.

Then the algorithm learns a latent embedding space and maps every user u and item i into that space (to $\mathbf{x_u}$ and $\mathbf{y_i}$ respectively). To learn that space, the algorithm tries to solve the following optimization equation:

$$\min_{x_*, y_*} \sum_{u,i} c_{ui}(p_{ui} - \mathbf{x_u^T y_i})^2 + \lambda(\sum_u \|\mathbf{x_u}\|^2 + \sum_i \|\mathbf{y_i}\|^2) \tag{5}$$

The results $\mathbf{x_u}$ and $\mathbf{y_i}$ will be used to complete the user-item matrix that estimates how likely a user will like an item. The proposed algorithm works well for implicit feedback/interest datasets as mentioned in Hu et al. [13].

At this point, we have built user-topic matrix using Equation 1, have adopted matrix factorization to learn a latent embedding space. Furthermore, we can model any user u's interests by estimating her preferences on all topics. For any new users who do not appear in the original user-topic matrix for training of the embedding space, we can still map them to the embedding space by using learned topic embedding vectors $\mathbf{y_i}$. We will discuss this in Section 5.3.

5.2.2 Behavior Factorization model (BF)

Different from the matrix factorization model introduced above, we want to separate user's different behavior types and generate topic preference for each user for different behaviors. So instead of factorizing one user-topic matrix, we factorize multiple user-topic matrices (**BNUM**, **SBSUM**, and **CBSUM**) generated in Step 1.

There are some early exploration on context-aware matrix factorization and tensor factorization techniques such as a social-network-aided context-aware recommender system proposed by Liu et al. [20], which creates multiple matrices and learns a latent space simultaneously. However, these techniques cannot be used directly in our behavior factorization problem, because we are building multiple user-topic matrices having the same column/topic space but having different rows/users. They build matrices having different items for different context, instead we use an implicit modeling approach, and also consider relations among behavioral contexts, i.e., such as combining publication behaviors and consumption behaviors.

Figure 1 shows the differences of our proposed Behavior Factorization (BF) approach as compared to the baseline model. In the first step of constructing matrix from user behaviors, instead of only constructing the Behavior Non-specific User-topic Matrix **BNUM**, we also construct two more types of matrices: Single Behavior Specific User-topic Matrix **SBSUM** and Combined Behavior Specific User-topic Matrix **CBSUM**.

Here in the second step, we factorize all generated matrices into the same latent embedding space. We learn a latent embedding space and map every user of each specific behavior types and every item into this space. Each entry in each matrix is the implicit interest value from user's behaviors, so we can extend the Baseline matrix factorization model as follows.

Here p_{ui}^B and c_{ui}^B represent the preference and confidence value for each matrix. Given all specific behavior types we used $\Gamma = \{B_1, B_2, \dots\}$ in Equation 3 and 4, we learn the embedding space by optimizing the following equation:

$$\min_{x_*, y_*} \sum_{B \in \Gamma} \sum_{u,i} c_{ui}^B (p_{ui}^B - \mathbf{x_u^{B\,T}} \mathbf{y_i})^2 + \lambda(\sum_{B \in \Gamma} \sum_u \|\mathbf{x_u^B}\|^2 + \sum_i \|\mathbf{y_i}\|^2) \tag{6}$$

By writing out the summation on Γ, we use a similar solution of the original Equation 5 to solve this optimization problem and learn embedding space for user-behavior and topics.

Compared to the original user-topic matrix, the embedding space learned by our approach might be better at measuring semantic similarity, because from previous analysis (Section 3), we know that the observed values in user-topic matrix are mixtures of multiple different interests from different behaviors. Separating the signals should therefore result in a cleaner topic model. This is also suggested by a recent paper studying how to learn generative graphical models such as LDA in social media [31]. In that work, they explored how to aggregate documents into corpus that represent a particular context. Since generative graphical model and matrix factorization both tries to learn latent space from data, this intuition can be shared in both techniques. Here our hypothesis is that building matrix at user-behavior level instead of user level can help us identify cleaner semantic alliances across topics, without increasing too much sparsity.

5.3 Step 3: Building user profiles

Finally, we introduce how we build user profiles using learned latent embedding spaces from previous steps. As shown in Figure 2, we introduce two methods: (i) direct profile building from input row vectors of profile matrices, and (ii) weighted profile building by merging different direct profiles using a set of weights learned from a regression model.

As defined in Section 4, for each user u, we will build $P_u = \{\mathbf{V_{u_B}}\}$. Each $\mathbf{V_{u_B}}$ is a vector of topic preferences of user u on

specific behavior types B. We build three types of user profiles, corresponding to the three types of input matrices:

- **Behavior Non-specific User Profile (BNUP)**: By setting $B = \mathfrak{B}$, we build a single $\mathbf{V_u} = \mathbf{V_{u_B}}$ for user u as her profile. This profile does not differentiate behavior types. This profile is used in existing approaches for building user profile and personalized recommendation. Each user will have one preference on each topic of user-topic matrix.

- **Single Behavior-Specific User Profile (SBSUP)**: Setting B to only one behavior type, $\mathbf{V_{u_B}}$ represents users' preferences of topics with only behavior types in B.

- **Combined Behavior-Specific User Profile (CBSUP)**: By setting B to contain more than one behavior types, $\mathbf{V_{u_B}}$ that represents users' preferences of topics with the behavior types in B. In our paper, for example, we construct both the Publishing and Consumption CBSUP.

5.3.1 Direct Profile Building (DPB)

We use a user's embedding factors (i.e., vector $\mathbf{x_u}$ for user u in learned latent embedding space) to generate her complete user profiles of $\mathbf{V_{u_B}} \in P_u$. In **DPB**, the input will be observed row vectors in matrix R_B for any B in Γ, and we build user profile for each B.

Given a user u and B, we obtain embedding factor $\mathbf{x_u^B}$ and then use this embedding factor and topic embedding factors $Y = \{y_i\}$ to generate preference list of user u's behavior B by computing the dot product: $\mathbf{x_u^{B^T}} Y$. Then for each user u, her output user profile can be represented as:

$$P_u = \{\mathbf{V_{u_B}} = \mathbf{x_u^{B^T}} Y\}, B \in \Gamma \tag{7}$$

Specifically, given any B that is a subset of $\mathfrak{B}$, we use the following equation to generate user's **SBSUP** and **CBSUP**:

$$V_{u_B} = (p_{u_B}^{e_1}, p_{u_B}^{e_2}, \ldots, p_{u_B}^{e_K}), \;\; p_{u_B}^{e_i} = \mathbf{x_u^{B^T}} \mathbf{y_i} \tag{8}$$

where x_u^B is the embedding factor of user u on behavior types B.

In summary, in **DPB**, different profiles are generated from different input row vectors, row vector of **BNUM** generates **BNUP**, row vector of **SBSUM** generates **SBSUP**, row vectors of **CBSUM** generates **CBSUP**. For example, to build **BNUP** for each user u, we set $B = \mathfrak{B}$, and use all her observed topic interest values to generate her embedding factor $\mathbf{x_u}$ in the embedding space. Then by calculating $\mathbf{x_u y_i}$ with every topic i, we get her **BNUP**.

$$\mathbf{V_u} = (p_u^{e_1}, p_u^{e_2}, \ldots, p_u^{e_K}), \;\; p_u^{e_i} = \mathbf{x_u^T} \mathbf{y_i} \tag{9}$$

For new users who are not in the learned embedding model, we can still generate their row input vectors using Equation 2, and then project the vector to an embedding factor.

5.3.2 Weighted Profile Building (WPB)

DPB generates a behavior profile for a user only if that user have exhibited that behavior in the past. By separating user's behavior types, we can generate profile for user u's behavior B using **DPB**, but this requires user u to have non-zero observed values with behavior B. For some users who do not have behaviors in B, $\mathbf{V_{u_B}}$ will be empty, which means that a user who do not exhibit the required behavior type actions will not have a user profile. This somewhat corresponds to the cold-start problem in recommender systems.

However, we can solve this by using user's profiles on other behavior types. We combine them to generate a combined preference vector on topics using weighted sum. This corresponds to a transfer learning problem.

Here, to generate preference vector $\mathbf{V_{u_B}}$ of user u's behavior types B, instead of directly using results of Equation 7, we use weighted sum of all preference vectors we generated for any behavior types in Γ using Equation 10:

$$\mathbf{V_{u_B}} = \sum_{B_t \in \Gamma} w_{B_t} \mathbf{x_u^{B_t}}^{\mathbf{T}} Y \tag{10}$$

The weights for different behavior types in Γ are model-level parameters, i.e., we learn one weight for each $B_t \in \Gamma$ for the entire dataset. Therefore, these weights can be learned using a supervised method from all users who have multiple types of behaviors in our dataset. Therefore, for users who have no B_t in their history, we can still build those profiles for them.

In our implementation, we use linear regression with stochastic gradient descent to learn these parameters. Therefore, **WPB** can be used to generate either **BNUP**, **SBSUP** or **CBSUP**, depending on particular applications. In most content recommendation applications, usually information consumption behaviors are most important, and thus, we use users' observed consumption behaviors to learn weights to build consumption profile.

Having described the steps in our modeling method, we now turn our attention to an evaluation of this method.

6. EVALUATION

In previous sections, we proposed the behavior factorization approach which can both learn a powerful latent space and build user profiles for multiple behavior types. In this experimental study section, we want to verify the following two hypotheses:

H1: The latent embedding model learned from our behavior factorization approach is better in building user profiles than the baseline matrix factorization model.

H2: By combining preference vectors from multiple behavior types, we improve the coverage of user profiles on specific behavior types.

In the rest of this section, we first describe how we set up our experiment, i.e., what datasets we use, and how we evaluate the performance of the output user profiles. Then we compare our Behavior Factorization model with baseline model. We also compare our two proposed methods in building user profiles to show that, by combining behavior types, we can improve user coverage with good quality.

6.1 Experiment Setup

To evaluate the performance of building user profiles, we examine how well different approaches do in predicting users' topical interests. We separate our dataset into two parts: training set and testing set. We train and build user profiles using only training set, and compare the performance of different models with the testing set. We consider an approach to be a better one if it completes the user-behavior-topic matrix more accurately.

6.1.1 Dataset

Our dataset contains public Google+ user behaviors in May and June 2014. How we generate the dataset is described in section 3.1. We train both baseline and our approach's matrix factorization model using Google+ user behaviors in May 2014. We use 20% randomly sampled behaviors in June 2014 to learn weights used in

Weighted Profile Building (**WPB**) of Section 5.3.2. We use the remaining 80% behaviors in June 2014 to evaluate the performances of different methods.

Input Matrices: In our dataset, we include all public posts created in May and June 2014. There are four types of behaviors on those posts: Creating post, Resharing, Commenting, and +1. We build different user-behavior-topic matrices by making sets of behavior types Γ contain the following behavior types:

- We directly use four behavior types in our dataset to build Single Behavior Specific User-topic Matrix (**SBSUM**).
- We want to capture users' interests in publishing information, so we group creating post and resharing together to build a publication Combined Behavior Specific User-topic Matrix (**Publication CBSUM**).
- We also want to understand users' interests in consuming information, so we group commenting and +1 to build a **Consumption CBSUM**.
- We also group all four behavior types together to generate a Behavior Non-specific User-topic Matrix (**BNUM**).

6.1.2 Evaluation Metrics

The user profile we built for a given behavior type B_t is a vector of preferences on topics, $\mathbf{V_{u_{B_t}}}$. The values in this vector estimate whether user u would like each topic with behavior B_t. This can be evaluated using implicit interests of $R_u^{B_t} = \{r_{ui}^{B_t},\ i \in \mathfrak{E}\}$ in testing set calculated using Equation 2.

Although the actual values in $\mathbf{V_{u_{B_t}}}$ and $R_u^{B_t}$ do not need to be identical, a good user profile of B_t should have the ranking order of topics in $\mathbf{V_{u_{B_t}}}$ similar to what we have observed in testing set.

To compare the orders of these two vectors, we transfer vectors $\mathbf{V_{u_{B_t}}}$ and $R_u^{B_t}$ into two ranked lists of topics in $\mathfrak{E}$: $L_{method} = (e_{r_1}, e_{r_2}, \ldots, e_{r_N})$ is the rank list of top N topics generated by profile building method, and $L_{observed} = (e_{o_1}, e_{o_2}, \ldots, e_{o_{N'}})$ is a rank list of all the observed topics. We use the following as evaluation metrics:

- **Recall@N**: indicates how many topics appearing in top N of L_{method} also appear in $L_{observed}$, divided by the actual number of observed topics.
- **NDCG@N**: Normalized Discounted Cumulative Gain (NDCG) is a widely used metric to evaluate two ranked lists. It is the discounted cumulative gain of current ranked list normalized by the ideal discounted cumulative gain[1].
- **Average Percentile@N**: Percentile denotes where each e_{r_j} in L_{method} appears in $L_{observed}$. Similar metric has been used in evaluating matrix completion tasks [13]. Here we use 100% to represent when e_{r_j} is at the top of $L_{observed}$ and 0% to represent when e_{r_j} is at the bottom of $L_{observed}$. Average percentile@N is the averaged position across all e_{r_j} in L_{method}.

6.1.3 Comparison Methods

Next we show our experiment results to verify our two hypotheses at the beginning of this section. The methods used are shown in Table 2.

For our first hypothesis H1, to evaluate the performance of the latent embedding model learned from our Behavior Factorization model (**BF**) against the one learned with traditional matrix factorization model (**Baseline**), we compare them on building user profiles using the Direct Profile Building (DPB) method. To do this, we generate user profiles (BNUP, SBSUP, or CBSUP) using Direct Profile Building (**DPB**) with input matrices of BNUM, SBSUM, or CBSUM, respectively. We then evaluate the profiles' performances with three evaluation metrics introduced above.

Subsequently, to verify our second hypothesis H2, we evaluate our proposed **WPB** by comparing its coverage and quality of user profiles against **DPB**. To do this, we use latent space of **BF** to build user profile for the specific behavior type of consumption (i.e., Commenting & +1), using both the Direct Profile Building **DPB** and Weighted Profile Building **WPB**.

Methods	Latent Model Training	Profile Building
Baseline + DPB	Baseline	Direct (DPB)
BF + DPB	Behavior Factorization	Direct (DPB)
BF + WPB	Behavior Factorization	Weighted (WPB)

Table 2: Method combinations used in our experiments.

6.2 Evaluation Results

6.2.1 H1: Baseline v.s. Behavior Factorization

To compare **BF** with Baseline matrix factorization approach, we first use the two latent embedding model learned from these two methods to build different user profiles (**BNUP**, **SBSUP** and **CBSUP**). Then, we compare their performances using our evaluation metrics. Since we are only comparing learned latent embedding models, we use **DPB** to build behavior-specific user profiles.

Performance of building **BNUP**.

Here we compare two approaches in building **BNUP**. Baseline model is learned using the method discussed in Section 5.2.1, and **BF** model is learned using the method discussed in Section 5.2.2. The comparison results are shown in Figure 3. From the figure we can see that our approach achieves significant improvement on all evaluation metrics. Compared to Baseline, Our **BNUP** has 89% improvement on NDCG, 93% improvement on Average Percentile and 82% improvement on Recall.

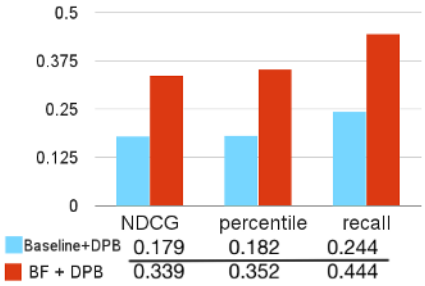
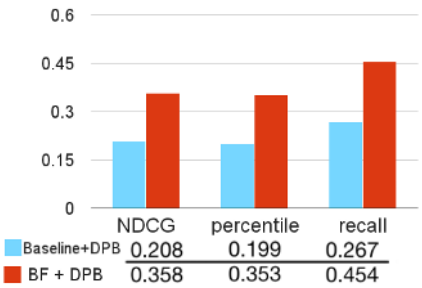


Figure 3: Comparison of BNUP outputs between Baseline and BF methods, given BNUM as input.

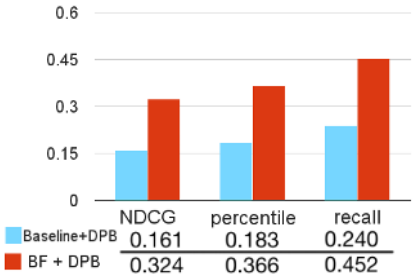
Performance of building **CBSUP**.

Next we evaluate the performance in building user's behavior specific profile, i.e., **CBSUP** for combined behavior types of Consumption and Publication. Results are shown in Figure 4(a) and Figure 4(b). We see significant improvement of our approach. We see that more improvement (about 100%) is on publication pro-

[1] http://en.wikipedia.org/wiki/Discounted_cumulative_gain



(a) **Consumption profile** output w/ **CBSUM** as input.



(b) **Publication profile** output w/ **CBSUM** as input.

Figure 4: Comparison of CBSUP between Baseline and BF.

file than consumption profile. This aligns somewhat well with Goffman's work [5], because users usually have clearer idea about their audiences when they are publishing information. As a result, clearer personal topic interest representation appears in publication behaviors than consumption behaviors.

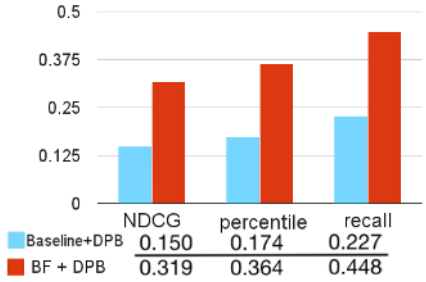
Performance of building **SBSUP**.
We build users' **SBSUP** for four single behavior types, and compare the performance between Baseline and our BF method. The results are shown in Figure 5. On average, our approach is about 80% better than Baseline. The greatest improvement is on creating post, followed by commenting. In these two behaviors, users usually know better about their audiences and have clearer goals, i.e., talking to his followers or creators of posts, as compared to resharing and +1. We can see that our approach learns better latent embedding models to build Single Behavior Specific User-topic Profiles. The improvement of using BF compared to using Baseline method passed paired-sample t-test with significant level $p < 0.01$.

6.2.2 *H2: Direct Profile Building (***DPB***) v.s. Weighted Profile Building (***WPB***)*

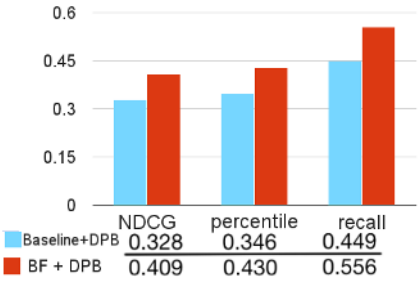
Next we verify our second hypothesis that Weighted Profile Building (**WPB**) improves coverage. We show that, compared to Direct Profile Building (**DPB**), generating user profile using WPB over all behavior types in Γ improves coverage for users without specific behavior types and with reasonable performance.

The left side of Figure 6 shows the user coverage of **Consumption CBSUP** built by **DPB** vs. **WPB**. Using **DPB** , we can only provide consumption profile for 30% of users in the testing set, because it requires users in the testing set to have comment and +1 histories in our training set. By using **WPB**, we improve the coverage to 49.7%, which is a 65.8% improvement.

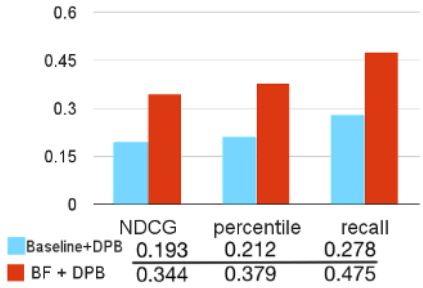
The right side of Figure 6 shows the performance of user consumption profile of **DPB** and **WPB** for new and existing users: **Existing users:** For users who have comment and +1 behavior histories, **WPB** further improves the quality of their user consumption profiles by 6.7% on average on all three evaluation metrics, suggesting the interests of one types of behaviors can transfer to interests of another using behavioral factorization.



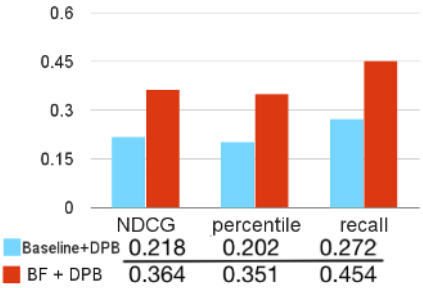
(a) **Create post profile** given **SBSUM**.



(b) **Reshare profile** given **SBSUM**.



(c) **Comment profile** given **SBSUM**.



(d) **+1 profile** given **SBSUM**.

Figure 5: Comparison of SBSUP between Baseline and BF.

New Users: For users who have none of those behaviors in history, **WPB** still provides consumption profiles with reasonable performance, i.e., 79.2% of the performance of **DPB** on existing users, and 74.1% of the performance of **WPB** on existing users.

All Covered Users: We also calculated the average quality of profiles built by **WPB** on all 49.7% covered users who have or do

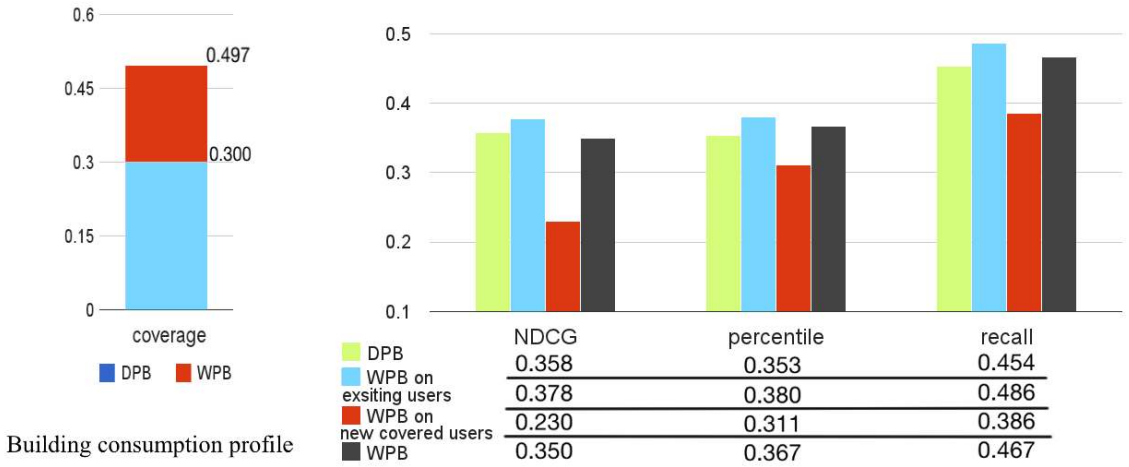


Figure 6: Comparison of consumption profile between DPB and WPB

not have consumption behaviors. The average percentile and recall are still slightly higher than profiles built by **DPB** on existing users.

Due to the limited space, here we only show our results on building profile for one specific set of behavior types—consumption of information, because, compared to other behavior types, the consumption topical interests of users is the most important.

However, besides consumption profile, we also tested **WPB** on building user profiles for other behavior types that we have examined in above experiments, such as **SBSUP** on commenting and +1, and we observed similar results. We can improve user coverage more than 70% for commenting and +1, while slightly improving the profile quality on average. The improvement of using **WPB** on existing users compared to DPB passed paired-sample t-test with significant level $p < 0.01$.

7. DISCUSSION

We have conducted experiments to evaluate our proposed Behavior Factorization approach to build either Behavior Non-specific user topic profiles or Behavior Specific user topic profiles. We demonstrated that it is important to build user profiles by separating types of behaviors. We also showed how this enables applications that target recommendations for different behaviors types.

7.1 Potential Applications

There are many applications can use the user profiles built by our method. Since we can map user's behaviors as well as different sets of items, e.g., posts, communities, or other users, into the same embedding model, the similarity between user's behavior and items can be used to generate rank lists for recommendation. Compared to conventional user profiles which don't separate user behaviors, our user profiles not only considers the content similarity between users and items, but also considers the context of different recommendation tasks. For example, consumption profile can be used to recommend relevant posts when a user is reading a post, publication profile can be used to recommend new friends after a user creates a post, etc.

7.2 Limitations and Future Work

As our results show, the proposed behavior factorization framework indeed improves the performance building users' interest profiles, however there are still some limitations. One limitation is that the framework depends on the fact that users' various behavior types inherently reveal users' different interests. It works well in building users' interest profiles in social media platforms (e.g., Google+), but may not generalize well for other domains where different behavior types do not necessarily reflect different user interests.

In addition, the results have shown that our method does not work as well when users have very sparse or no data for the target behavior type we are trying to predict. One reason is that these users may be less active than other users. The other reason is that our method optimizes multiple matrices which could lose the correlation across behavior types for the same user. To solve this problem, we are interested in applying tensor factorization techniques such as PARAFAC [10] on behavioral matrices. Our method can be thought of as an extension of an unfolding-based approach to tensor factorization, but a full Tucker3 decomposition might bring improvements in some modeling situations.

Furthermore, we would like to directly deploy the behavior factorization framework in real world recommendation systems and evaluate it via online live experiments.

8. CONCLUSION

In this paper, we proposed Behavior Factorization (**BF**) as a way to build user topic interest profiles in social media. To motivate our work, by analyzing a large quantity of behavior data from users in Google+, we showed that, for each user, topic interests exhibited by one type of behavior is different from other types.

To model users, we separate users' topic interests by their behaviors around those topics, and then constructing various combinations of novel user-behavior by topic matrices. Behavior Factorization first learns a latent embedding model by factorizing matrices separated by behaviors, then builds user topic profiles for different types of behaviors using this embedding model.

We verified our approach using large-scale Google+ data. Experiment results showed that BF improved our latent embedding model by about 80% in predicting user topic preferences, and by using Weighted Profile Building (**WPB**), we can improve coverage of consumption profile by 60%. Finally, it is our hope that our behavioral factorization approach might inspire other researchers to deeply think about how to model user context in topic modeling and recommender systems.

9. REFERENCES

[1] F. Abel, Q. Gao, G.-J. Houben, and K. Tao. Analyzing user modeling on twitter for personalized news recommendations. In *User Modeling, Adaption and Personalization*, pages 1–12. Springer, 2011.

[2] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022, 2003.

[3] J. Chen, R. Nairn, and E. Chi. Speak little and well: recommending conversations in online social streams. In *SIGCHI 2011*, pages 217–226.

[4] Z. Dou, R. Song, and J.-R. Wen. A large-scale evaluation and analysis of personalized search strategies. In *WWW 2007*, pages 581–590.

[5] E. Goffman. The presentation of self in everyday life. 1959.

[6] P. Gupta, A. Goel, J. Lin, A. Sharma, D. Wang, and R. Zadeh. Wtf: The who to follow service at twitter. In *WWW 2013*, pages 505–514.

[7] I. Guy, N. Zwerdling, D. Carmel, I. Ronen, E. Uziel, S. Yogev, and S. Ofek-Koifman. Personalized recommendation of social software items based on social relations. In *RecSys*, pages 53–60. ACM, 2009.

[8] I. Guy, N. Zwerdling, I. Ronen, D. Carmel, and E. Uziel. Social media recommendation based on people and tags. In *SIGIR 2010*, pages 194–201.

[9] A. Hannak, P. Sapiezynski, A. Molavi Kakhki, B. Krishnamurthy, D. Lazer, A. Mislove, and C. Wilson. Measuring personalization of web search. In *WWW 2013*, pages 527–538.

[10] R. A. Harshman. Foundations of the parafac procedure: Models and conditions for an" explanatory" multi-modal factor analysis. 1970.

[11] M. Hines. Google takes searching personally. cnet, 2004.

[12] L. Hu, J. Cao, G. Xu, L. Cao, Z. Gu, and C. Zhu. Personalized recommendation via cross-domain triadic factorization. In *WWW 2013*, pages 595–606.

[13] Y. Hu, Y. Koren, and C. Volinsky. Collaborative filtering for implicit feedback datasets. In *ICDM 2008*, pages 263–272.

[14] M. Jamali and L. Lakshmanan. Heteromf: recommendation in heterogeneous information networks using context dependent factor models. In *WWW 2013*, pages 643–654.

[15] Y. Kim, Y. Park, and K. Shim. Digtobi: a recommendation system for digg articles using probabilistic modeling. In *WWW 2013*, pages 691–702.

[16] Y. Koren. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *SIGKDD 2008*, pages 426–434.

[17] Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *Computer*, (8):30–37, 2009.

[18] C.-Y. Li and S.-D. Lin. Matching users and items across domains to improve the recommendation quality. In *SIGKDD 2014*, pages 801–810.

[19] C. Lippert, S. H. Weber, Y. Huang, V. Tresp, M. Schubert, and H.-P. Kriegel. Relation prediction in multi-relational domains using matrix factorization. In *Proceedings of the NIPS 2008 Workshop: Structured Input-Structured Output, Vancouver, Canada*. Citeseer, 2008.

[20] X. Liu and K. Aberer. Soco: a social network aided context-aware recommender system. In *WWW 2013*, pages 781–802.

[21] A. Majumder and N. Shrivastava. Know your personalization: learning topic level personalization in online services. In *WWW 2013*, pages 873–884.

[22] A. E. Marwick et al. I tweet honestly, i tweet passionately: Twitter users, context collapse, and the imagined audience. *New media & society*, 13(1):114–133, 2011.

[23] N. Matthijs and F. Radlinski. Personalizing web search using long term browsing history. In *WSDM 2011*, pages 25–34.

[24] J. Noel, S. Sanner, K.-N. Tran, P. Christen, L. Xie, E. V. Bonilla, E. Abbasnejad, and N. Della Penna. New objective functions for social collaborative filtering. In *WWW 2012*, pages 859–868.

[25] M. Qiu, F. Zhu, and J. Jiang. It is not just what we say, but how we say them: LDA-based behavior-topic model. In *SDM*, pages 794–802. SIAM, 2013.

[26] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl. Grouplens: an open architecture for collaborative filtering of netnews. In *CSCW 1994*, pages 175–186.

[27] E. Shmueli, A. Kagian, Y. Koren, and R. Lempel. Care to comment?: recommendations for commenting on news stories. In *WWW 2012*, pages 429–438.

[28] A. P. Singh and G. J. Gordon. Relational learning via collective matrix factorization. In *SIGKDD 2008*, pages 650–658.

[29] A. Singhal. Google blog: Introducing the knowledge graph: Things, not strings, May 2012.

[30] B. Tan, X. Shen, and C. Zhai. Mining long-term search history to improve search accuracy. In *SIGKDD 2006*, pages 718–723.

[31] J. Tang, Z. Meng, X. Nguyen, Q. Mei, and M. Zhang. Understanding the limiting factors of topic modeling via posterior contraction analysis. In *ICML 2014*, pages 190–198.

[32] J. Tang, M. Zhang, and Q. Mei. One theme in all views: modeling consensus topics in multiple contexts. In *SIGKDD 2013*, pages 5–13.

[33] B. Wang, C. Wang, J. Bu, C. Chen, W. V. Zhang, D. Cai, and X. He. Whom to mention: expand the diffusion of tweets by@ recommendation on micro-blogging systems. In *WWW 2013*, pages 1331–1340.

[34] J. Wang, Z. Zhao, J. Zhou, H. Wang, B. Cui, and G. Qi. Recommending flickr groups with social topic model. *Information retrieval*, 15(3-4):278–295, 2012.

[35] R. W. White, W. Chu, A. Hassan, X. He, Y. Song, and H. Wang. Enhancing personalized search by mining and modeling task behavior. In *WWW 2013*, pages 1411–1420.

[36] H. Yin, B. Cui, L. Chen, Z. Hu, and Z. Huang. A temporal context-aware model for user behavior modeling in social media systems. In *SIGMOD 2014*, pages 1543–1554.

[37] X. Zhang, J. Cheng, T. Yuan, B. Niu, and H. Lu. TopRec: domain-specific recommendation through community topic mining in social network. In *WWW 2013*, pages 1501–1510.

[38] X. Zhao, N. Salehi, S. Naranjit, S. Alwaalan, S. Voida, and D. Cosley. The many faces of facebook: Experiencing social media as performance, exhibition, and personal archive. In *SIGCHI 2013*, pages 1–10.