Joint Social and Content Recommendation for User-Generated Videos in Online Social Network

Zhi Wang, *Student Member, IEEE,* Lifeng Sun, *Member, IEEE,* Wenwu Zhu, *Fellow, IEEE,* Shiqiang Yang, *Senior Member, IEEE,* Hongzhi Li, and Dapeng Wu, *Senior Member, IEEE*

Abstract—Online social network is emerging as a promising alternative for users to directly access video contents. By allowing users to import videos and re-share them through the social connections, a large number of videos are available to users in the online social network. The rapid growth of the user-generated videos provides enormous potential for users to find the ones that interest them; while the convergence of online social network service and online video sharing service makes it possible to perform recommendation using social factors and content factors jointly. In this paper, we design a joint social-content recommendation framework to suggest users which videos to import or re-share in the online social network. In this framework, we first propose a user-content matrix update approach which updates and fills in cold user-video entries to provide the foundations for the recommendation. Then, based on the updated user-content matrix, we construct a joint social-content space to measure the relevance between users and videos, which can provide a high accuracy for video importing and re-sharing recommendation. We conduct experiments using real traces from Tencent Weibo and Youku to verify our algorithm and evaluate its performance. The results demonstrate the effectiveness of our approach and show that our approach can substantially improve the recommendation accuracy.

Index Terms—Video recommendation, online social network, social propagation

I. INTRODUCTION

Online social network service (*e.g.*, Facebook, Twitter) and online video sharing service (*e.g.*, YouTube) have emerged as the two most important services in today's Internet. The rapid convergence of the two services makes the market and industry — Internet Social TV — very promising [1], [2], [3]. In the context of online social network, video contents are generated by individuals, instead of the centralized content providers [4], *e.g.*, more than 60 hours' worth of videos are uploaded

Z. Wang, L. Sun W. Zhu and S. Yang are with the Department of Computer Science and Technology, Beijing Key Laboratory of Networked Multimedia, Tsinghua University, Beijing 100084, China (e-mail: wangzhi04@mails.tsinghua.edu.cn, {sunlf, wwzhu, yangshq}@tsinghua.edu.cn).

H. Li is with the Department of Computer Science, Columbia University, New York, NY 10027, U.S. (e-mail: hongzhili@cs.columbia.edu).

D. Wu is with the Department of Electrical & Computer Engineering, University of Florida, FL 32611-6130, U.S. (e-mail: wu@ece.ufl.edu).

by users per minute on YouTube [5]. A large portion of the videos originally hosted in the video sharing system are being "imported" by individuals to the online social network [6], and "re-shared" among users through the social connections.

The online social network is reshaping not only the way how videos are generated, but also the way how users consume the video contents. The massive number of videos available makes it possible for users to directly find the videos that interest them in the online social network. In the online social network, users are psychologically willing to import and re-share videos [7]; however, it can be difficult for them to choose videos to import or re-share among the large number of videos available, making recommendation for both activities in demand. To realize the potential of such social video sharing, we are interested in the recommendation of videos for two important social activities in the online social network [8] as follows.

- *Importing recommendation* answers the question "what videos to import?" In the popular online social network systems such as Facebook and Twitter, most of the videos are not hosted by the social network systems, instead, they are imported from other external video sharing systems, *e.g.*, users can import videos from YouTube to Twitter by simply posting the links to the videos. The importing recommendation helps users to discover interesting videos from the video sharing websites.

- *Re-sharing recommendation* answers the question "what videos to re-share?" After users have imported videos to the online social network, such videos will be distributed through the social connections, *e.g.*, users on Twitter see the videos shared by people they follow, and will further re-share the videos to people who follow them, making videos propagate in a cascade way [9]. The re-sharing recommendation helps users discover the videos among the large number of videos shared by people they follow.

Existing video recommendation includes both *content-based recommendation* [10] and *social-based recommendation* [11]. In the content-based recommendation, content-based filtering and collaborative filtering [12] have been widely studied. They are based on the content similarity from either the content analysis, or the user's historical ratings of the contents. Such recommendation approaches can provide a user with contents similar to the ones he has viewed before, or the contents that users similar to him have viewed before, as illustrated in Fig. 1(a). On the other hand, in the social-based recommendation, social relationship (*e.g.*, friending, following, *etc.*) is used to filter the contents distributed through the social connections,

(a) Content based recommendation. (b) Social based recommendation.

Fig. 1.   Content-based recommendation and social-based recommendation.
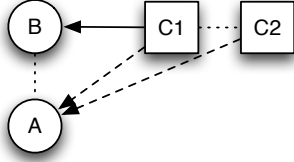

Fig. 2.   Joint social and content recommendation.

so that contents that one likes can be suggested to their social connections [11]. Such recommendation approach is able to provide users with the contents that have previously interested their friends, as illustrated in Fig. 1(b).

In the context of online social network, existing video recommendations are facing the following problem and challenges. (1) They are lack of consideration of how contents propagate through the social connections, which is a unique feature of the online social network. (2) Importing and sharing videos in the online social network are implicit, *i.e.*, users hardly give explicit ratings to videos they have imported/re-shared, which are required in many existing recommendation approaches. (3) Cold-start is even more challenging in the recommendation for today's user-generated contents for two reasons: (a) For users who have just joined the system, they have hardly imported or shared any videos in the system, making it difficult to recommend any contents for them, since existing recommendation systems rely on users' historical preferences. (b) For the videos imported by users who have few friends, a large fraction of them have almost no viewer. It is difficult to determine which users these contents should be recommended to as well.

To address the above problem and challenges, we propose a joint social and content recommendation framework. Fig. 2 illustrates the concept design, where videos can be recommended to users according to both the social relation and the content similarity. In particular, a *user-user matrix* (how users are socially connected), a *content-content matrix* (how videos are similar to each other) and a *user-content matrix* (how users import and re-share these videos) are utilized as inputs in our recommendation. More specifically, first, we propose a user-content matrix update algorithm by incorporating both social propagation and content similarity. We describe how socially connected users influence each other and how similar videos can interest the same user, so as to predict which videos are to be imported/re-shared by which users. In the update, entries for cold users and cold contents in the user-content matrix are updated and filled in so as to improve the recommendation. Second, based on the updated user-content matrix, we build

a joint *user-content space* to measure the relevance between users and contents. Dynamical adjustment of the weights of user space and content space is employed to improve the recommendation for both importing and re-sharing.

The remainder of this paper is organized as follows. We summarize the related works in Sec. II. We introduce the framework of our recommendation in Sec. III. We present the detailed recommendation algorithms in Sec. IV and Sec. V. We discuss the efficiency issues and potential extensions of our recommendation algorithms in Sec. VII. We evaluate the performance of our design in Sec. VI. Finally, we conclude the paper in Sec. VIII.

II. RELATED WORK

In this section, we survey literature on online social networks, video recommendation technologies, and recommendation in the context of online social network.

A. Online Social Network

Online social network has become a popular research topic in recent years. Mislove *et al.* [13] use real world traces to study the topology of the social graph, and confirm the power-law, small-world, and scale-free properties of the online social network. Krishnamurthy *et al.* [14] investigate Twitter and identify the distinct classes of users and their behaviors, as well as geographic growth patterns of the social network. Information in a online social network spreads among users in a "word-of-mouth" manner. A number of research efforts have been devoted to studying the propagation of information. Kwak *et al.* [15] investigate the impact of users' retweets on information diffusion in Twitter. Dodds *et al.* [16] use the contagion model to study information propagation where a piece of information is regarded as an infective disease which spreads via the social connections. Domingos *et al.* [17] explore the value of social networks in estimating potential buyers of a product or a service, which can be influenced by an existing customer. Kempe *et al.* [9] investigate how to maximize the spread of influence in an online social network, and Hartline *et al.* [18] utilize such maximum spread to achieve revenue maximization. Recommendation in the online social network not only provides users with the contents that can interest them, but also provides information that can potential improve the service quality for the content sharing systems, which host the socialized contents [19].

In this paper, we will explore the potential to use social propagation for user-generated video recommendation.

B. Video Recommendation

For general video recommendation, content collaboration and collaborative filtering have been widely used in the existing recommender systems [20] [12]. The basic idea is that when recommending contents for a user, the contents that are similar to the ones he has viewed before will be suggested. On one hand, such similar contents can be found out by content analysis technologies; on the other hand, contents that are accessed by similar users can also be regarded as similar to

each other. However, individually they fail to provide good recommendation: (1) content-based approaches only measure the similarity using multimedia analysis, which cannot directly reflect users' interests; and (2) collaborative filtering suffers from the sparsity of users' preference database [21].

Due to the above drawbacks of pure content-based and collaborative filtering approaches, there have been some studies on combining the two to achieve better recommendation performance. Melville *et al.* [22] have proposed to incorporate components from both content-based and collaboration filtering approaches to generate a hybrid recommender system, in which a content-based predictor is used to enhance the existing user data, where the pseudo user-ratings are generated based on the content analysis and used in the content-boosted collaborative filtering. Basilico *et al.* [23] have designed a kernel function between user-item pairs that allows simultaneous generalization across the user and item dimensions. There are also other recommendation frameworks for user-generated video recommendation. Baluja *et al.* [24] have proposed to use a random walk through a co-view graph in YouTube to recommend the videos.

In this paper, we study how videos in the online social network can be suggested to users using the information from both the online social network and the online video sharing network.

C. Video Recommendation in the Context of Online Social Network

Due to the massive number of user-generated videos available in the online social network, recommendation is essential to realize the potential of social media in the online social network [25]. In order to keep users entertained and engaged, it is imperative that these recommendations are updated regularly and reflect a users recent activity on the site [26]. Social connections and users' social activities are important records that can be used in video recommendation. Debnath *et al.* [27] have proposed to improve the recommendation performance using online social network, where attributes used for content based recommendations are assigned weights depending on their importance to users. Walter *et al.* [11] have proposed a trust-based model to perform recommendation, where users leverage their social connections to reach interesting information and make use of the trust relationship to filter unwanted information. Since recommendation generally relies on users' private information (*e.g.*, video ratings), it is challenging to perform content suggestion when users will not contribute their rating information. Isaacman *et al.* [28] have proposed to use matrix factorization for recommendation for user-generated contents when the rating information is only shared between content producer and consumer pairs, which is a common privacy demand by users. Wang *et al.* [29] have studied recommendation for social groups using instead of individuals in the online social network, using the followees' information of the users inside the groups.

To the best of our knowledge, these is no study on using social propagation, content similarity analysis and users' social activities jointly to perform recommendations for user-generated videos. In this paper, we explore how information from the online social network and online video sharing network can be used jointly to improve recommendation performance.

III. FRAMEWORK

In this section, we first show the difference between importing recommendation and re-sharing recommendation, then we present the framework for both types of recommendations.

A. Recommendation for Importing and Re-Sharing

Our recommendation is for users' imports and re-shares, *i.e.*, we suggest videos that users like to import to or re-share in the online social network. The two social activities are different as follows. (1) *Different purposes*. When importing videos to the online social network, users act as sources of the video contents, and they are willing to contribute videos that will interest their friends in the online social network. When re-sharing videos, they are helping the diffusion of the videos that are already in the online social network. (2) *Different scales*. In our measurement of Tencent Weibo, we observe that the number of imports issued by users is 6 times larger than the number of re-shares issued by users. People are more likely to generate videos than re-share videos that are already in Weibo. (3) *Different recommendation factors*. When importing a video, a user mainly considers the video itself, *e.g.*, determining whether the video can interest his friends; when re-sharing a video, besides the video itself, the user also considers the user who shares the video in the first place, *e.g.*, the user may be willing to help their friends to distribute a video. To perform recommendation for imports and re-shares, respectively, we first design a general recommendation model for both activities, then we suggest videos for imports and re-shares using different factors from the online social network and online content sharing network.

B. Recommendation Input: Matrices from Social Network and Content Network

We first introduce the information used in our design from both the online social network and the online video sharing network as follows: (1) the user matrix which represents how users follow each other in the online social network; (2) The content matrix which represents the similarity between the videos according to content analysis; and (3) the initial user-content matrix which represents how users import and re-share videos.

1) User-User Matrix: The microblogging system allows users to maintain "weak" social connections between each other [30], *i.e.*, a user can generally follow anyone without their explicit permission. Thus, the social connections in a microblogging system can well reflect users' interests. From the technical team of Tencent Weibo, one of the largest microblogging systems in China [31], we have obtained traces recording how users are socially connected to each other. Let **A** denote the user-user matrix as follows.

$$\mathbf{A}_{ij} = \begin{cases} 1, & i \text{ follows } j, \text{ or } i = j \\ 0, & \text{otherwise} \end{cases}.$$

2) Content-Content Matrix: In traditional recommender systems, videos can be recommended only when they have been rated by some users before, *i.e.*, after the system learns which users are interested in the videos [32], [33], [34], [35]. However, in the online social network, many user-generated videos are very cold with almost no viewer. In our design, we perform recommendation for these videos by using the content similarity analysis. Particularly, we use a tag-based approach to construct the content similarity matrix as follows.

• *Collecting keywords from videos' tag lists.* The similarity between two videos is evaluated using their common keywords of their tags. The videos in Tencent Weibo are generally imported by users from other video sharing sites such as Youku, where each video i is given a list of tags before it is published, containing several short sentences describing content of the video. In our design, we first segment each sentence into several keywords in set W_i, then we use the common keywords of two videos to evaluate their similarity.

• *Weighting the keywords.* One problem of the similarity calculation is the highly skewed frequency of the keywords, *i.e.*, some keywords appear much more frequently than others. *e.g.*, the word "fun" can easily be chosen as a tag keyword. To resolve this problem, we adjust the weight of different keywords according to their appearance frequency. The weight z_w of a keyword w is defined as $z_w = \psi_1 + \psi_2 \frac{1}{P_w}$, where ψ_1, ψ_2 are two control parameters, and P_w is the appearance frequency of w in all the videos. The rationale lies as that (1) if two videos have more common keywords, the similarity between them should larger; and (2) if a keyword is more common, the weight of that keyword should be smaller. Finally, by adjusting the similarity according to their importance weights, we have the similarity between two videos as follows,

$$\mathbf{C}_{ij} \begin{cases} 1, & i = j \\ \frac{\sum_{w \in W_i \cap W_j} z_w}{\sum_{w \in W_i \cup W_j} z_w}, & \text{otherwise} \end{cases}.$$

Larger $\mathbf{C}_{ij}$ indicates that content i is more similar to content j.

3) Initial User-Content Matrix: The user-content matrix contains information about how users import and re-share videos in Tencent Weibo. This information is also included in the traces we obtained. To improve the recommendation performance for cold users and cold videos, we can update some missing entries later. Let $\mathbf{B}$ denote the initial user-content matrix as follows,

$$\mathbf{B}_{ij} = \begin{cases} 1, & \text{user } i \text{ has imported/re-shared video } j \\ 0, & \text{otherwise} \end{cases}.$$

In our study, we only use the above "2D" matrices; however, they can be extended to have "3D" tensors, *e.g.*, in the user matrix, the social connection between two users can be a vector indicating the different influences of a user to the other one; and in the content matrix, more context can be included to measure the similarity (*e.g.*, location). We will discuss the details of the matrix extension in Sec. VII-B.

C. Framework of our Recommendation

Next, we present the framework of our recommendation, which includes the following key technologies: (1) the user-content matrix update based on the social propagation and content similarity and (2) the activity-aware user-content space construction.

1) User-Content Matrix Update: As discussed above, the user-content matrix can be very sparse [36]. It is difficult for traditional recombination algorithms to deal with users who have imported/re-shared no or fewer videos, since personalized recommendation is based on users' historical preferences. In our design, we update the user-content matrix by making use of both the social matrix and content matrix to enable recommendation for cold users/videos, that is, we "predict" the items which are likely to be imported/re-shared by a user who has little historical importing/re-sharing information available.

On one hand, we use the social propagation model to connect users and videos, that is, videos imported/re-shared by a user's friends are likely to be imported/re-shared by the user. On the other hand, we connect videos to users who have imported/re-shared similar videos. Fig. 3 illustrates the user-content update framework in our design. When updating the user-content matrix $\mathbf{B}'$, besides the initial user-content matrix $\mathbf{B}$, the user-user matrix $\mathbf{A}$ and the content-content matrix $\mathbf{C}$ are also used. In particular, $\mathbf{A}$ and $\mathbf{B}$ will be used for the social-propagation-based update, while $\mathbf{C}$ and $\mathbf{B}$ will be used for the content-similarity-based update. We will discuss the details in Sec. IV.

2) Construction of the Joint User-Content Space: Our recommendation is based on constructing a joint user-content space and measuring the relevance between users and contents. Fig. 4 illustrates how the relevance between a user and a video is measured. The joint user-content space is based on combining a user space and a content space. The user space is constructed using the user-user matrix, and the content space is constructed using the content-content matrix. A user or a video can be represented by two *description vectors* in both spaces, which have the same dimension. The *relevance* between a user and a video is then measured in both spaces jointly. Finally, videos can be recommended to a user according to their relevance with the user. We will discuss the details in Sec. V.

Before we present the detailed design of the user-content matrix update and the user-content space construction, we list important notations in Table I.

IV. USER-CONTENT MATRIX UPDATE BASED ON SOCIAL PROPAGATION AND CONTENT SIMILARITY

We update the initial user-content matrix before performing the relevance-based recommendation.

A. Selecting the Entries to Update

When choosing the entries in the initial user-content matrix to update, there can be many different principles, *e.g.*, we can target at some important users and videos to perform better recommendation for them. In this work, we choose the entries to update as follows. (1) We choose the users/videos with no
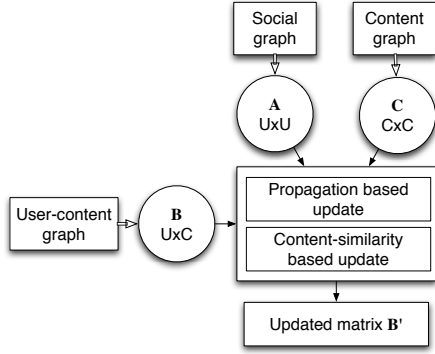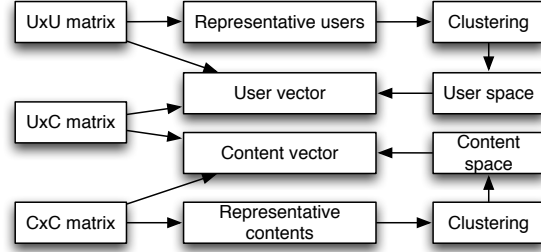
Fig. 3. Update of the user-content matrix.

Fig. 4. Construction of the user-content space.

TABLE I
NOTATIONS

Symbol	Definition
$\mathbf{A}$	The user-user matrix
$\mathbf{B}$	The user-content matrix
$\mathbf{C}$	The content-content matrix
$\mathbf{\Omega}(\mathbf{M})$	Matrix indicating which entries in $\mathbf{M}$ are missing
$\mathcal{G}(\mathbf{M})$	Updating gain matrix of $\mathbf{M}$
I_{uc}	The update value from social propagation
J_{uc}	The update value from content similarity
$\mathbf{E}$	The candidate set of entries in the user-content matrix for update
L_1	Maximum number of missing entries in the user-content matrix to be updated in each round
L_2	Maximum number of rounds for the update procedure
$\mathbf{H}$	The candidate representative items
$\mathbf{R}, r$	The representative item set and its size
K	The number of clustering groups
$sim(x, y)$	A similarity between x and y
$RE(u, c)$	A relevance index between user u and video c

or few existing entries, since these users/videos need more information for recommendation; (2) We choose the entries that can maximize the "accuracy" when being updated. We will discuss how the two objectives are achieved, respectively.

First, we choose the candidate entries that need the information for recommendation. The candidate users/videos to be updated in the user-content matrix are the ones with little information. The update is carried out round by round, we define the candidate set of entries to update in round T as $\mathbf{E}^{(T)}$. $\mathbf{E}^{(T)}$ is constructed in the following steps: (a) Rank users according to the number of videos they have imported/re-shared recently; (b) Select the users with the minimum imported/re-shared videos, and choose the videos randomly from a list of representative videos which will be discussed in Sec. V; (c) The selected users/videos will form the candidate set $\mathbf{E}^{(T)}, T = 1, 2, \ldots$.

Next, we choose the entries to update according to the accuracy when updating them. To evaluate the accuracy of updating an missing entry, which reflects the level of existing entries when updating it, we define an updating gain as in Eq. (1), where $\mathcal{G}(\mathbf{B}^{(T)})_{ij}$ is the updating gain of entry $\mathbf{B}_{ij}$, and

$\mathbf{\Omega}(\mathbf{B}^{(0)})_{ij} = 1$ indicates $\mathbf{B}_{ij}^{(0)}$ exists in the initial user-content matrix. For matrix $\mathbf{A}$ and $\mathbf{C}$, we have $\mathcal{G}(\mathbf{A})_{ij} = \mathbf{\Omega}(\mathbf{A})_{ij}$ and $\mathcal{G}(\mathbf{C})_{ij} = \mathbf{\Omega}(\mathbf{C})_{ij}$. The initial updating gain (when $T = 0$) is based on whether the entry exists in the initial user-content matrix, *i.e.*, 1 is assigned to an existing entry and 0 is assigned to an missing entry; when an entry is updated, the updating gain is evaluated according to how the entry is calculated. In our design, we select the entries in the candidate set $\mathbf{E}^{(T)}$ to maximize the matrix updating gain, by mathematically solving the following problem:

$$\max_{\mathbf{\Gamma}} \sum_{(i,j) \in \mathbf{\Gamma}} \mathcal{G}(\mathbf{B}^{(T)})_{ij}, \tag{2}$$

subject to

$$\mathbf{\Omega}(\mathbf{B}^{(T)})_{ij} = 0, (i,j) \in \mathbf{\Gamma},$$

$$\mathbf{\Gamma} \subset \mathbf{E}^{(T)},$$

$$|\mathbf{\Gamma}| \leq L_1,$$

where $\mathbf{\Gamma}$ is the set of entries selected for updating, and L_1 is the maximum number of entries to be updated in each round. Next, we discuss how the entries are updated.
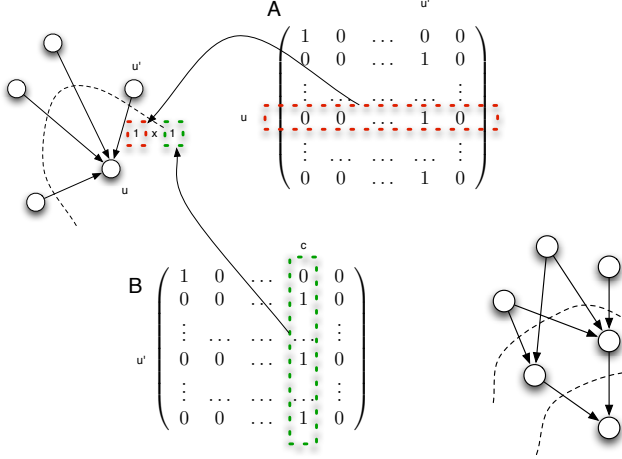
B. Updating the Missing Entries

Next, we will present the user-content matrix update based on the social propagation and content similarity.

First, we discuss the update using social propagation. Fig. 5(a) illustrates how social propagation can be used for the recommendation. A video propagates through social connections in a cascade way [9], where one's interests can influence users. In our design, a row vector $\mathbf{A}_{u,*}$ in the user-user matrix $\mathbf{A}$ indicates the people who are followed by user u, and a column vector $\mathbf{B}_{*,c}$ in the user-content matrix $\mathbf{B}$ indicates the users who have shown interest in video c by importing/re-sharing it. When more people followed by a user have shown interests in a particular video, the user himself can be influenced as well. Thus, according to the social propagation, the ability that content c can interest user u is evaluated by

$$I_{uc}^{(T)} = \sum_{k | \mathbf{\Omega}(\mathbf{B}^{(T-1)})_{kc} = 1} \mathbf{A}_{uk} \mathbf{B}_{kc}^{(T-1)}. \tag{3}$$

$$
\mathcal{G}(\mathbf{B}^{(T)})_{ij} = \begin{cases} 1, & \mathbf{\Omega}(\mathbf{B}^{(0)})_{ij} = 1 \\ 0, & \mathbf{\Omega}(\mathbf{B}^{(0)})_{ij} = 0 \\ \mathcal{G}(\mathbf{A})\mathcal{G}(\mathbf{B}^{(T-1)})_{ij} \cdot \mathcal{G}(\mathbf{B}^{(T-1)})\mathcal{G}(\mathbf{C})_{ij}, & \text{Updated in round } T \\ \mathcal{G}(\mathbf{B}^{(T-1)})_{ij}, & \text{otherwise} \end{cases} \quad \begin{matrix} T = 0 \\ \\ , \\ T > 0 \end{matrix} \tag{1}
$$

(a) Cascade propagation of contents.   (b) Simulating the propagation in multi-hop cascade.

Fig. 5.   Update based on social propagation prediction.

Algorithm 1 Update of the user-content matrix.

1: **procedure** MATRIX UPDATE($\mathbf{A}$, $\mathbf{B}^{(0)} = \mathbf{B}$, $\mathbf{C}$)
2:     $t \leftarrow 0$
3:     **while** $t < L_2$ **do**
4:         According to Eq. (1), calculate the updating gains for entries in $\mathbf{E}^{(t)}$
5:         Rank the entries in $\mathbf{E}^{(t)}$ in their updating gains' descending order
6:         **for** the top L_1 entries in the ranked entry list **do**
7:             Update the selected L_1 entries using Eq. (5)
8:         **end for**
9:         $t \leftarrow t + 1$
10:    **end while**
11: **end procedure**

We update the missing entries in the user-content matrix round by round, based on simulating the propagation of videos through the social connections. As illustrated in Fig. 5(b), after an entry in the user-content matrix is updated, indicating that the user will potentially import or re-share the video, the user will be able to influence others who follow him. In our algorithm, parameter L_2 determines the maximum "depth" of the propagation.

Second, we discuss the content similarity based matrix update. Based on a user's historical preference, we can predict what videos the user will be more likely to import/re-share according to the similarity between the videos he has imported/re-shared and other videos. Thus, the ability that a content c can interest user u can be evaluated by

$$
J_{uc}^{(T)} = \sum_{k|\mathbf{\Omega}(\mathbf{B}^{(T-1)})_{uk}=1} \mathbf{B}_{uk}^{(T-1)} \mathbf{C}_{kc}. \tag{4}
$$

Finally, a missing entry is updated according to the combination of $I_{uc}^{(T)}$ and $J_{uc}^{(T)}$:

$$
\mathbf{B}_{uc}^{(T)} = \begin{cases} 1, & I_{uc}^{(T)} \cdot J_{uc}^{(T)} > \eta \\ 0, & \text{otherwise} \end{cases}, \tag{5}
$$

where η is the threshold to determine the updating result. In each update round, η will be dynamically adjusted so that at least half of the candidate entries will be updated to 1. The algorithm for the user-content matrix update is given in Algorithm 1.

V. USER-CONTENT SPACE CONSTRUCTION AND THE RECOMMENDATION

We construct the user-content space after the user-content matrix has been updated. In our design, we describe each user and each video by a vector in the user-content space, such that the *relevance* of a user and a video can be measured for the recommendation.

A. Construction of the Joint User-Content Space

We first construct the user space and content space separately, so that a user and a video can be mapped into either of the spaces. Then we combine the vectors in both spaces to measure the relevance between a user and a video according to the dot product of their vectors in the joint user-content space, so as to perform the recommendation.

1) A General Algorithm for User/Content Space Construction: Since the user space and content space are constructed using the similar procedure, we present the general algorithm which will be employed in both user space and content space construction.

(a) Choose r representative items from a candidate set. Let $sim(x, y)$ denote the similarity between item x and item y (an item can be a user or a video). Larger $sim(x, y)$ indicates that x is more similar to y. In our design, we choose r representative items from the candidate set by solving the following problem:

$$
\min_{\mathbf{R}} \sum_{x \neq y, x, y \in \mathbf{R}} sim(x, y), \tag{6}
$$

subject to

$$
\mathbf{R} \subset \mathbf{H},
$$

$$
|\mathbf{R}| = r,
$$

where $\mathbf{R}$ is the set of the selected representative items and $\mathbf{H}$ is the whole candidate set. The rationale of the optimization in Eq. (6) is that we select the representative items "different" enough from each other to maximize their ability to describe the diversity of other items. The problem can be solved by a

---

Algorithm 2 Selection of the representative items.

1: **procedure** REPRESENTATIVE ITEM SELECT($\mathbf{H}$)
2:     Select r items randomly from $\mathbf{H}$ to form a initial $\mathbf{R}$
3:     Find item $m \in \mathbf{R}$ that has the largest similarity cost $cost(m, \mathbf{R}, \mathbf{H})$
4:     Replace item m with item n selected from $\mathbf{H} - \mathbf{R}$, which has the smallest similarity cost
5:     Repeat 2-4 until the similarity cost change is smaller than threshold TH_{select}
6: **end procedure**

---

heuristic algorithm in Algorithm 2. First it randomly chooses r items from the candidate set, and then replaces the items to obtain a better set iteratively, by reducing a similarity cost, which is defined as follows,

$$cost(m, \mathbf{R}, \mathbf{H}) = \sum_{x \neq m, x \in \mathbf{R}} sim(m, x). \qquad (7)$$

We will discuss the candidate items and the number of representative items later for the user space and content space, respectively.

(b) Cluster the r representative items into K groups. We define *description index* for a group of several representative items, which is a similar concept used in the clustering algorithms, *e.g.*, K-Means and K-Medoids [37], to evaluate the similarity between items within a group. Larger description index indicates that items are more closely clustered inside the group. We maximize the description index within the same group as follows,

$$\max_{G} \sum_{g \in G} \sum_{x \in g} ra(x, g), \qquad (8)$$

subject to

$$|G| = K,$$

$$g_i \cap g_j = \Phi, g_i, g_j \in G, i \neq j,$$

$$\cup_i g_i = \mathbf{R},$$

where G is the set of groups we choose for the space construction, g is a group in G, and $ra(x, g)$ defines a representative index of item x in group g. The representative index is defined as

$$ra(x, g) = \sum_{y \neq x, y \in g} sim(x, y) / |g|. \qquad (9)$$

We use a K-Medoids like algorithm to solve the above optimization as illustrated in Algorithm 3. First, K initial medoids are randomly selected from the representative set $\mathbf{R}$; then we change the medoids to other items to improve the clustering gain, which is defined as follows:

$$gain(x, x_0) = \sum_{y \neq x, y \in g} sim(x, y) - \sum_{y \neq x_0, y \in g} sim(x_0, y), \qquad (10)$$

where x_0 is the original medoid of a group. By improving the clustering gain iteratively, we are able to increase the description index of the groups.

(c) Map any user/video to a K-dimension vector. In the user space, a user u will be described as a vector p_u and a

---

Algorithm 3 Clustering of the representative items.

1: **procedure** REPRESENTATIVE ITEM CLUSTER($\mathbf{R}$)
2:     Randomly select K items $\{e_1, e_2, \ldots, e_K\}$ from $\mathbf{R}$ as the initial medoids
3:     **for** $\forall x in \mathbf{R} - \{e_1, e_2, \ldots, e_K\}$ **do**
4:         Assign x to a group whose medoid is the most similar to x
5:     **end for**
6:     For each group, replace the medoid with an item in the same group such that the *clustering gain* can be maximized
7:     Repeat 2-6 until the clustering gain change is smaller than threshold $TH_{cluster}$
8: **end procedure**

---

video c will be described as a vector $\bar{p}_c$; while in the content space, a user u will be described as a vector q_u and a video c will be described as a vector $\bar{q}_c$. We define a relevance index between u and c as follows,

$$RE(u, c, m) = \alpha p_u \cdot \bar{p}_c + (1 - \alpha) q_u \cdot \bar{q}_c, \qquad (11)$$

where α is a dynamical weight to combine the relevance values in both the user space and content space. Larger α indicates that the user will be more likely to refer to the user space for the recommendation, while small α indicates that he is more likely to refer to the content space for the recommendation. We will further evaluate and discuss the impact of this parameter in Sec. VI.

Next, we will discuss in particular how the user space and content space are constructed, respectively.

2) Construction of the User Space: Based on the user-user matrix generated from the Weibo traces, we present how the user space is constructed using the general algorithms.

(a) Select the representative users. In Weibo, some users are so famous that they have millions of followers; besides, these users distributed in different categories, *e.g.*, IT, business, *etc*. They can be used as the representative users. Fig. 6 illustrates the number of followers of each user in the $5,000$ top-followed users (who have the largest numbers of followers in Tencent Weibo) ranked according to the numbers of their followers. We observe that the number of followers of the top-followed users is much larger than that of the average users.

To avoid the impact the most popular users who can be easily followed by anyone, we skip the first $1,000$ top-followed users. Fig. 7 illustrates the number of people covered by the top $1,000$–$4,0000$ followed users, *i.e.*, users have followed at least one of the top-followed users. We observe the $3,000$ top-followed users can cover about 20 million users in the system, which is about the number of all users in Tencent Weibo. Thus, we choose the $1,000-4,000$ top-followed users as the representative users.

In our design, the similarity $sim(u, v)$ between two users is calculated as the fraction of their common followers as follows,

$$sim(u, v) = \frac{|fans(u) \cap fans(v)|}{|fans(u) \cup fans(v)|}, \qquad (12)$$
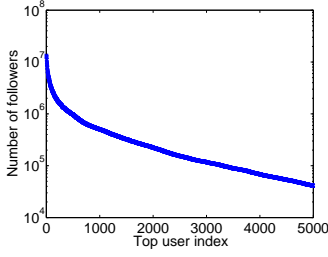
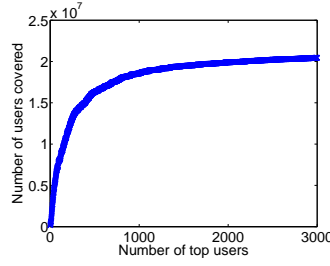Fig. 6. Number of followers of the top-followed users.

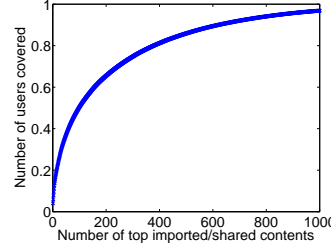Fig. 7. The number of covered users vs. the number of top-followed users.

Fig. 8. Fraction of users covered by the top-imported/re-shared videos.
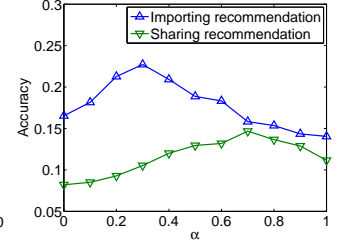
Fig. 9. Recommendation accuracy vs. α.

where $fans(u)$ is the set of users who follow u in Weibo. The rationale is as follows. (1) *Weak social relation*. Comparing with the strong social connections (friends) in social network systems like Facebook, the following relation in a microblogging system can be very weak, *i.e.*, users can follow anyone who interests them. (2) *Interest driven following connection between ordinary users and top-followed users*. Since users can follow users all according to their own interests, the fraction of common followers of two top-followed users can be used to represent their similarity. Facebook has also confirmed that weak relationship can represent users' interests and is important information source for users [38].

(b) Cluster the representative users. We need to solve the optimization given in Eq. (8) to obtain the groups used to construct the user space. We have already defined the similarity between users, and we are able to directly use Algorithm 3 to cluster the top-followed users. In our experiments, we choose K_U to be around 20.

(c) Construct user vector and content vector in the user space. The user vector and content vector are constructed as follows.

• User vector in user space (p_u). Let f_{iu}^1 denote the number of representative users in group i that user u follows. The user vector is the normalization of vector $\{f_{1u}^1, f_{2u}^1, \ldots, f_{K_U u}^1\}$, *i.e.*, entry p_{ui} in p_u is defined as $p_{ui} = \frac{f_{iu}^1}{\sum_k f_{ku}^1}$. The rationale is that when user u follows more representative users in a group, the corresponding entry in the user vector is larger to emphasize his interest in that particular group.

• Content vector in user space ($\bar{p}_c$). We use the updated user-content matrix to construct the content vector p_c. Let f_{ic}^2 denote the aggregate strength of users who have imported/re-shared video c, $f_{ic}^2 = \sum_{k \in U_c} p_{kc}$, where U_c is the set of users who have imported or re-shared video c. The content vector is then defined as the normalization of vector $\{f_{1c}^2, f_{2c}^2, \ldots, f_{K_U c}^2\}$, *i.e.*, entry $\bar{p}_{ci}$ in $\bar{p}_c$ is defined as $\bar{p}_{ci} = \frac{f_{ic}^2}{\sum_k f_{kc}^2}$. The rationale is that a large entry in $\bar{p}_c$ indicates that more users from the corresponding group likes that video.

3) Construction of the Content Space: Similarly, we present how we construct the content space using the general algorithm.

(a) Select the representative videos. We choose the representative contents from the top-imported/re-shared videos, which are the most popular ones which have the largest number of viewers. Fig. 8 illustrates the fraction of users covered by the top-imported/re-shared contents in Tencent Weibo, *i.e.*, the users have imported/re-shared at least one

video in the top-imported/re-shared videos. In our design, we use $r_C = 400$ candidate top-imported/re-shared videos, which can cover almost 80% of all the users. The content similarity is defined in Sec. III-B, *i.e.*, $sim(x, y) = \mathbf{C}_{xy}$. We choose the representative contents using Algorithm 2.

(b) Cluster the representative videos. We have selected the top-imported/re-shared videos that are the most popular ones in the system. To maximize the description index of the selected videos, we need to cluster these top-imported/re-shared videos into multiple groups. We use Algorithm 3 to cluster the representative videos to groups. In our experiments, the performance can be optimized when the number K_C of clusters is around 8.

(c) Construct content vector and user vector in the content space. The content vector and user vector are constructed as follows.

• Content vector in content space ($\bar{q}_c$). Let f_{ic}^3 denote the aggregate similarity of video c with all items in group i, *i.e.*, $f_{ic}^3 = \sum_{k \in g_i} \mathbf{C}_{kc}$. The content vector then is defined as the normalization of vector $\{f_{1c}^3, f_{2c}^3, \ldots, f_{K_C c}^3\}$, *i.e.*, entry $\bar{q}_{ci}$ in $\bar{q}_c$ is defined as $\bar{q}_{ci} = \frac{f_{ic}^3}{\sum_k f_{kc}^3}$. The rationale is that a larger entry in $\bar{q}_c$ indicates that video c is more similar to that group.

• User vector in content space (q_u). Let f_{ui}^4 denote the aggregate strength of contents which have been imported/re-shared by user u, $f_{ui}^4 = \sum_{k \in V_u} \bar{q}_{ck}$, where V_u is the set of videos that are imported or re-shared by u. The user vector q_u is the normalization of vector $\{f_{u1}^4, f_{u2}^4, \ldots, f_{uK_C}^4\}$, *i.e.*, entry q_{ui} in q_u is defined as $q_{ui} = \frac{f_{ui}^4}{\sum_k f_{uk}^4}$. The rationale is that if user u has imported/re-shared more videos similar to a group, the corresponding entry in the user vector should be larger to reflect his interest in that group.

B. Recommendation based on the User-Content Space

After the user space and content space have been constructed, the relevance between a user and a video can be measured according to Eq. (11). The recommendation is provided to a user as a list of videos, which have the largest relevance index with the user. In our design, we provide the importing list and re-sharing list as follows.

• *Importing list*. The importing list contains videos that are likely to be imported by users to the microblogging system. Since a user is able to view all the videos on the external video sharing sites like Youku, the candidate video pool contains all the recent popular videos on Youku, and the importing list generated by our algorithm is the ones with the largest

relevance index with the user. When importing a video, a user considers more about the video itself; thus, a smaller α will be used when performing the importing recommendation.

• *Re-sharing list.* The re-sharing list contains videos that are the most likely to be re-shared by a user further to his followers. Since the videos that the user can re-share are the ones imported or re-shared from people he follows. Thus, the candidate video pool for the re-sharing recommendation is the list of videos that can "reach" them, *i.e.*, the ones that have been re-shared by people they follow. When re-sharing a video, a user considers not only the video, but also the friends who imports/re-shares the video. Thus, a larger α will be used when performing the re-sharing recommendation.

VI. Experimental Results

In this section, we conduct experiments using real traces from Tencent Weibo and Youku to verify our algorithm and evaluate its performance.

A. Experiment Setup

We evaluate the performance of our joint social-content recommendation for both video importing and video re-sharing. We use traces from Tencent Weibo and Youku to run the experiments. We randomly choose 2200 videos from Youku, which are selected from 5 most popular categories on Youku. The videos were published between March 19 and June 20, 2011, and were the most popular ones on Youku's front pages on June 20, 2011. Around $600,000$ users have imported or re-shared these videos. In particular, the traces contain the following information. (1) The user-user matrix. The traces record the social connections, *i.e.*, which users are followed by which users. (2) The content-content matrix. The traces have recorded the original URLs of the videos, which can be used to crawl their text tags, which are used to generate the matrix as presented in Sec. III-B2. (3) The initial user-content matrix. The microblog traces have also recorded whether a video is imported or re-shared, and we generate the initial user-content matrix by setting the entry to 1 if the user has imported or re-shared the video.

In our experiments, we use the recommendation accuracy as the key performance metric, which is the fraction of videos correctly suggested by the recommender system over all the videos imported/re-shared by users [39]. After we provide a recommendation list to users, by comparing it with their true importing or re-sharing records, we can yield the recommendation accuracy. In the following experiments, we divide the user-content matrix into two parts: 60% of the records will be used as the input of the initial user-content matrix, while the rest will be used as the ground truth to evaluate the design.

B. Algorithm Verification

In Fig. 9, we first evaluate the impact of the user space and content space. The two curves in this figure illustrate the recommendation accuracy versus α for importing recommendation and re-sharing recommendation, respectively. We observe that for both recommendation, too large or too small α leads

to low recommendation accuracy. The reason is that when recommending importing and re-sharing videos to users, the relevance between users and videos in both user space (where users and videos are represented by representative user groups) and content space (where users and videos are represented by representative content groups) can assist the recommendation, and the recommendation based on a combination of the relevance in both spaces achieves the best performance. From this figure, we observe that importing recommendation relies more on the content space, *i.e.*, relatively small α (0.3) achieves the best accuracy; while re-sharing recommendation relies more on the user space, *i.e.*, relatively large α (0.7) achieves the best performance. The reason is that the importing activity is performed more according to users' interests in videos; while re-sharing is usually performed more according to the people they follow. The results verify the effectiveness of our joint social-content approach for both importing and re-sharing recommendation. In other experiments, we will use $\alpha = 0.3$ and $\alpha = 0.7$ for importing recommendation and re-sharing recommendation, respectively. $\alpha = 0$ will be referred to as using content space only, and $\alpha = 1$ will be referred to as using user space only.

Since personalized recommendation relies on users' historical preferences, we evaluate the performance by varying the number of users' historical importing and re-sharing records. Users with more importing/re-sharing records are referred to as more active users. The user-content matrix update based on social propagation and content similarity is expected to improve the recommendation for cold users/contents. In our experiments, we evaluate the recommendation for users with fewer re-sharing/importing records in the system. Specifically, we randomly choose $100,000$ users from the most un-active users for the evaluation. First, we evaluate the impact of L_1, which determines the number of missing entries to update in each round. Fig. 10 illustrates the recommendation accuracy versus the number of missing entries updated in each round. In Fig. 10(a), we observe that for the importing recommendation, when more missing entries are updated, higher accuracy can be achieved, and when the number is too large, the increase rate is getting smaller. In Fig. 10(b), similar results are observed for the re-sharing recommendation, but with a smaller increasing rate. Second, we evaluate the impact of the number of rounds used for the user-content matrix update. The number of rounds used in updating the missing entries determines the depth of the propagation simulation. Fig. 11(a) and (b) illustrate the recommendation accuracy versus the number of updating rounds used in the update. We observe that the best accuracy can be achieved when the number of updating rounds is between 2 and 3 for both importing and re-sharing recommendation, indicating that in the update a moderate propagation depth should be used.

C. Performance Evaluation

We first evaluate the performance of our own design under different combinations of the user space and content space. Fig. 12(a) and (b) illustrate the recommendation accuracy versus the number of top-active users used in the evaluation.
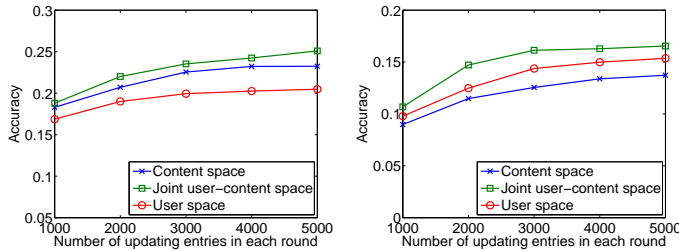
(a) Importing recommendation.

(b) Re-sharing recommendation.

Fig. 10. Accuracy vs. the number of entries filled (l_1) in the user-content update.
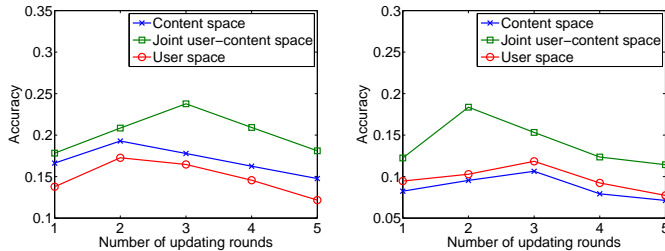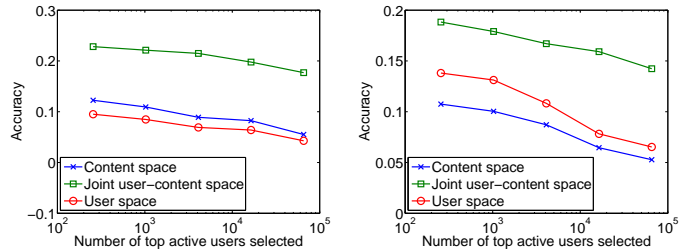

(a) Importing recommendation.

(b) Re-sharing recommendation.

Fig. 11. Accuracy vs. the updating rounds (L_2) in the user-content update.

(a) Importing recommendation.

(b) Re-sharing recommendation.

Fig. 12. Accuracy vs. the number of top-active users under different combinations of user space and content space.

(a) Importing recommendation.

(b) Re-sharing recommendation.

Fig. 13. Comparison of the joint social-content recommendation with content-based filtering and collaborative filtering approaches.

From the two figures we can see that when more un-active users are evaluated, the accuracy is decreased. Meanwhile, we observe that comparing with using only the user space ($\alpha = 1$) or only the content space ($\alpha = 0$), the user-content space achieves better performance. Similar results are also observe that the content space achieves better performance than the user space in importing recommendation, while the user space achieves better performance than the content space in re-sharing recommendation.

We also compare our approach with the following algorithms: content-based filtering approach [10] and the collaborative filtering approach [32]. Although the approaches are not directly designed to perform recommendation in our scenario, *i.e.*, the importing recommendation and re-sharing recommendation for user-generated videos in the online social network, we implement them as follows. (1) Content-based filtering approach. For each user, the videos he has imported/re-shared will be used to find the new ones that may interest him. Videos that are the most similar to the ones he has imported/re-shared before are recommended (according to the content-content matrix). The size of the recommendation list is the same as that used in our approach. (2) Collaborative filtering approach. In the collaborative filtering approach, "similar" users are discovered according to the videos they have imported or re-shared, *i.e.*, two users are more similar to each other if they have imported or re-shared more common videos before. The recommendation list for a user consists of the videos that are imported/re-shared by users who are the most similar to him.

Fig. 13 illustrates the recommendation accuracy of different recommendation algorithms with respect to the number of top-active users used in the evaluation. In general, we observe that for both importing and re-sharing recommendations, our approach achieves significantly higher recommendation accuracy, especially when more users with little importing/re-sharing history are involved in the evaluation. We also observe
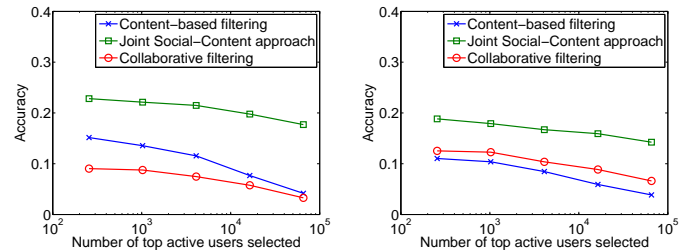
that content-based filtering works better in the importing recommendation while collaborative filtering works slightly better in the re-sharing recommendation.

VII. DISCUSSIONS

To further explore the potential of the joint social and content recommendation in a real large-scale online social network service, in this section, we discuss the efficiency of our recommendation algorithms and how they can be extended to satisfy more complicated social connections and video properties.

A. Efficiency of the Recommendation Algorithms

First, we discuss the efficiency of the user-content matrix update. The update algorithm is able to adapt to the Internet-scale social media recommendation thanks to the unique characteristics of the online social network: (1) *The number of a user's social connections is limited*. Dunbar's number [40] is suggested as a theoretical cognitive limit to the number (150) of people with whom one can maintain stable social relations. In our measurement of Tencent Weibo, we observe that the average number of people one follows is below 100. When performing the social-propagation-based update, the limited number of social connections of a user can largely reduce the computation. (2) *The number of video candidates for recommendation is limited*. In the online social network, users are more likely to exchange and consume recent contents. In our measurement study of Tencent Weibo, we observe that a dominant portion of videos that users import and re-share are published in the recent 24 hours [19]. Thus, in our recommendation, the size of the candidate videos can be quite limited. (3) *The number of entries to update is limited*. In our algorithm, the number of entries in the user-content matrix to

be updated can be dynamically adjusted according to the load of the recommendation system and the number of active users, *e.g.*, the update will be performed when a user is actively requesting the recommendation list and the system is able to handle the load; otherwise, the system will perform the recommendation with a lower quality when the update is not performed for some users.

Second, we discuss the efficiency of the space construction. (1) To maintain the set of the representative users or videos, a heap data structure can be utilized, so that the change of the representative users only requires $\log(r)$ comparisons and switches. (2) In our clustering algorithm, we use the threshold $TH_{cluster}$ to limit the rounds. (3) When calculating a user's vectors, only numbers instead of detailed contents are used, *i.e.*, the number of a user's idols and the number of videos he has viewed are utilized (the number of the videos one has viewed is limited since only the recent viewing history is referred to). Similarly, the space construction can scale for the large recommendation.

B. Extension of the Matrix

In this study, for each matrix, an entry is a numerical value to indicate the social connection, social activity and content similarity; however, when performing recommendation, more complicated information can be explored, *e.g.*, the location and gender of a user, the category and duration of a video, *etc*. To incorporate such information into our recommendation, we can extend the 2D matrices to 3D matrices. Let Z denote the number of these context items we can use; each entry in the three matrices will be replaced by a Z-dimension vector. In the user-user matrix, $\mathbf{a_{ij}} = \{e_1, e_2, \ldots, e_Z\}$ indicates the different aspects that user j can affect user i; in the user-content matrix, $\mathbf{b_{ij}} = \{e'_1, e'_2, \ldots, e'_Z\}$ indicates the different aspects that user i likes video j; and in the content-content matrix, $\mathbf{c_{ij}} = \{e''_1, e''_2, \ldots, e''_Z\}$ indicates the different aspects that content i is similar to j. In the user-content matrix update and the relevance space construction, the original operations of the matrix entries will be replaced by the operations of the vectors, *e.g.*, $\sum_k \mathbf{a_{ik}}^\mathsf{T} \cdot \mathbf{b_{kj}} \cdot \sum_k \mathbf{b_{ik}}^\mathsf{T} \cdot \mathbf{c_{kj}}$ will be the value to compare with the threshold η in the user-content matrix update. Since we are lack of such extensive information from our current traces from Tencent Weibo and Youku, we would like to explore the potential of the 3D tensors in our future work.

VIII. CONCLUSION

In this paper, we propose to use information from the online social network and online content sharing network jointly to perform recommendation for user-generated contents, *i.e.*, we suggest videos that users are likely to import or re-share in the online social network. We propose to use social propagation simulation and content similarity analysis to update the user-content matrix, so as to predict how cold videos can be imported/re-shared by cold users. Based on the user-content matrix update, we construct a joint user-content space using social relation, content similarity and user activity, to calculate the relevance between users and videos for the

recommendation. Our extended experimental results based on Weibo traces demonstrate the effectiveness of the joint social and content recommendation, which achieves a significantly higher recommendation accuracy than the existing content-based filtering and collaborative filtering approaches.
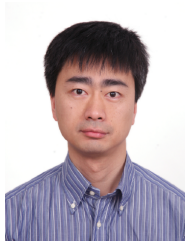
REFERENCES

[1] T. Coppens, L. Trappeniers, and M. Godon, "AmigoTV: Towards A Social TV Experience," in *Proc. of EuroITV*, 2004.

[2] Oehlberg, L. and Ducheneaut, N. and Thornton, J.D. and Moore, R.J. and Nickell, E., "Social TV: Designing for Distributed, Sociable Television Viewing," in *Proc. of EuroITV*, vol. 2006.

[3] R. Schatz, S. Wagner, S. Egger, and N. Jordan, "Mobile TV Becomes Social-Integrating Content with Communications," in *Proc. of 29th IEEE International Conference on Information Technology Interfaces*, 2007.

[4] M. Cha, H. Kwak, P. Rodriguez, Y. Ahn, and S. Moon, "I Tube, You Tube, Everybody Tubes: Analyzing the World's Largest User Generated Content Video System," in *Proc. of ACM SIGCOMM*, 2007.

[5] "http://www.telegraph.co.uk/technology/news/9033765/youtube-uploads-hit-60-hours-per-minute.html."

[6] Z. Wang, L. Sun, S. Yang, and W. Zhu, "Prefetching strategy in peer-assisted social video streaming," in *Proc. of ACM Multimedia*, 2011.

[7] M. Wasko and S. Faraj, "Why Should I Share? Examining Social Capital and Knowledge Contribution in Electronic Networks of Practice," *Mis Quarterly*, vol. 29, no. 1, pp. 35–57, 2005.

[8] F. Benevenuto, T. Rodrigues, M. Cha, and V. Almeida, "Characterizing User Behavior in Online Social Networks," in *Proc. of ACM IMC*, 2009.

[9] D. Kempe, J. Kleinberg, and É. Tardos, "Maximizing the Spread of Influence Through a Social Network," in *Proc. of ACM SIGKDD*, 2003.

[10] M. Pazzani and D. Billsus, *Content-Based Recommendation Systems*. Springer-Verlag, 2007, ch. The Adaptive Web, pp. 325–341.

[11] F. Walter, S. Battiston, and F. Schweitzer, "A Model of a Trust-Based Recommendation System on a Social Network," *Autonomous Agents and Multi-Agent Systems*, vol. 16, no. 1, pp. 57–74, 2008.

[12] J. Schafer, D. Frankowski, J. Herlocker, and S. Sen, "Collaborative filtering recommender systems," *The Adaptive Web*, pp. 291–324, 2007.

[13] A. Mislove, M. Marcon, K. Gummadi, P. Druschel, and B. Bhattacharjee, "Measurement and Analysis of Online Social Networks," in *Proc. of ACM IMC*, 2007.

[14] M. A. Balachander Krishnamurthy, Phillipa Gill, "A Few Chirps About Twitter," in *Proc. of ACM WOSN*, 2008.

[15] H. Kwak, C. Lee, H. Park, and S. Moon, "What Is Twitter, a Social Network or a News Media?" in *Proc. of ACM WWW*, 2010.

[16] P. Dodds and D. Watts, "A Generalized Model of Social and Biological Contagion," *Journal of Theoretical Biology*, vol. 232, no. 4, pp. 587–604, 2005.

[17] P. Domingos and M. Richardson, "Mining the Network Value of Customers," in *Proc. of ACM SIGKDD*, 2001.

[18] J. Hartline, V. Mirrokni, and M. Sundararajan, "Optimal Marketing Strategies over Social Networks," in *Proc. of ACM WWW*, 2008.

[19] Z. Wang, L. Sun, X. Chen, W. Zhu, J. Liu, M. Chen, and S. Yang, "Propagation-based Social-aware Replication for Social Video Contents," in *Proc. of ACM Multimedia*, 2012.

[20] M. Pazzani, "A Framework for Collaborative, Content-Based and Demographic Filtering," *Artificial Intelligence Review*, vol. 13, no. 5, pp. 393–408, 1999.

[21] "http://www.netflixprize.com."

[22] P. Melville, R. Mooney, and R. Nagarajan, "Content-Boosted Collaborative Filtering for Improved Recommendations," in *Proc. of the National Conference on Artificial Intelligence*, 2002.

[23] J. Basilico and T. Hofmann, "Unifying Collaborative and Content-Based Filtering," in *Proc. of ACM ICML*, 2004.

[24] S. Baluja, R. Seth, D. Sivakumar, Y. Jing, J. Yagnik, S. Kumar, D. Ravichandran, and M. Aly, "Video Suggestion and Discovery for YouTube: Taking Random Walks Through the View Graph," in *Proc. of ACM WWW*, 2008.

[25] R. Zhou, S. Khemmarat, and L. Gao, "The Impact of YouTube Recommendation System on Video Views," in *Proc. of ACM IMC*, 2010.

[26] J. Davidson, B. Liebald, J. Liu, P. Nandy, T. Van Vleet, U. Gargi, S. Gupta, Y. He, M. Lambert, B. Livingston *et al.*, "The YouTube Video Recommendation System," in *Proc. of ACM RecSys*, 2010.

[27] S. Debnath, N. Ganguly, and P. Mitra, "Feature Weighting in Content Based Recommendation System Using Social Network Analysis," in *Proc. of ACM WWW*, 2008.

[28] S. Isaacman, S. Ioannidis, A. Chaintreau, and M. Martonosi, "Distributed Rating Prediction in User Generated Content Streams," in *Proc. of ACM RecSys*, 2011.

[29] X. Wang, L. Sun, Z. Wang, and D. Meng, "Group Recommendation using External Followee for Social TV," in *Proc. of IEEE ICME*, 2012.

[30] "http://blog.hubspot.com/blog/tabid/6307/bid/4859/is-twitter-a-social-network.aspx."

[31] "http://t.qq.com."

[32] B. Sarwar, G. Karypis, J. Konstan, and J. Reidl, "Item-Based Collaborative Filtering Recommendation Algorithms," in *Proc. of ACM WWW*. ACM, 2001, pp. 285–295.

[33] G. Linden, B. Smith, and J. York, "Amazon.Com Recommendations: Item-To-Item Collaborative Filtering," *Internet Computing*, vol. 7, no. 1, pp. 76–80, 2003.

[34] J. Herlocker, J. Konstan, and J. Riedl, "Explaining Collaborative Filtering Recommendations," in *Proc. of Acm Conference on Computer Supported Cooperative Work*. ACM, 2000, pp. 241–250.

[35] J. Herlocker, J. Konstan, L. Terveen, and J. Riedl, "Evaluating collaborative filtering recommender systems," *ACM Transactions on Information Systems*, vol. 22, no. 1, pp. 5–53, 2004.

[36] C. Johnson, "Matrix completion problems: a survey," in *Matrix Theory and Applications*, vol. 40. Amer Mathematical Society, 1990, pp. 171–198.

[37] R. Xu, D. Wunsch *et al.*, "Survey of Clustering Algorithms," *IEEE Transactions on Neural Networks*, vol. 16, no. 3, pp. 645–678, 2005.

[38] "http://www.facebook.com/notes/facebook-data-team/rethinking-information-diversity-in-networks/10150503499618859."

[39] C. Basu, H. Hirsh, W. Cohen *et al.*, "Recommendation As Classification: Using Social and Content-Based Information in Recommendation," in *Proc. of the National Conference on Artificial Intelligence*, 1998.

[40] "http://en.wikipedia.org/wiki/dunbar's_number."

Wenwu Zhu is a "1000 People Plan" Professor of the Department of Computer Science and Technology at Tsinghua University. Prior to his current post, he was a Senior Researcher and Research Manager at Microsoft Research Asia. He was the Chief Scientist and Director at Intel Research China from 2004 to 2008. He worked at Bell Labs as a Member of Technical Staff during 19961999. He received the B.E. degree from National University of Science and Technology, China, in 1985, the M.S. degree from Illinois Institute of Technology, Chicago, and the Ph.D. degree from Polytechnic Institute of New York University, New York, in 1993 and 1996, respectively, in Electrical and Computer Engineering. He has published more than 200 refereed papers and filed 40 patents. His current research interests are in the area of multimedia computing, communications and networking, including multimedia cloud computing, social media computing, and wireless multimedia communications. He is a Fellow of IEEE.
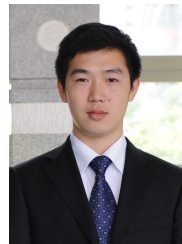
Hongzhi Li received the B.E. degree from Zhejiang University, Hangzhou, China, in 2010. He is currently pursuing the Ph.D. degree in the Department of Computer Science, Columbia University, New York, U.S. From 2009 to 2011, he was a Research Intern in Microsoft Research Asia. His research interests are in the areas of image/video retrieval and analysis, computer vision and machine learning.

Shiqiang Yang received the B.E. and M.E. degrees in Computer Science from Tsinghua University, Beijing, China in 1977 and 1983, respectively. From 1980 to 1992, he worked as an assistant professor at Tsinghua University. He served as the associate professor from 1994 to 1999 and then as the professor since 1999. From 1994 to 2011, he worked as the associate header of the Department of Computer Science and Technology at Tsinghua University. He is currently the President of Multimedia Committee of China Computer Federation, Beijing, China and the co-director of Microsoft-Tsinghua Multimedia Joint Lab, Tsinghua University, Beijing, China. His research interests mainly include multimedia procession, media streaming and online social network. He is a senior member of IEEE.

Zhi Wang received the B.E. degree in Computer Science in 2008 from Tsinghua University, Beijing, China. He is currently pursuing the Ph.D. degree in the Department of Computer Science and Technology, Tsinghua University. His research areas include online social network, cloud computing and large-scale multimedia systems. He is a student member of IEEE.


Dapeng Oliver Wu received B.E. in Electrical Engineering from Huazhong University of Science and Technology, Wuhan, China, in 1990, M.E. in Electrical Engineering from Beijing University of Posts and Telecommunications, Beijing, China, in 1997, and Ph.D. in Electrical and Computer Engineering from Carnegie Mellon University, Pittsburgh, PA, in 2003. Since 2003, he has been on the faculty of Electrical and Computer Engineering Department at University of Florida, Gainesville, FL, where he is a professor; previously, he was an assistant professor from 2003 to 2008, and an associate professor from 2008 to 2011. His research interests are in the areas of networking, communications, signal processing, computer vision, and machine learning. He is a senior member of IEEE.

Lifeng Sun received his B.S. and Ph.D. degrees in System Engineering in 1995 and 2000 from National University of Defense Technology, Changsha, Hunan, China; He was on the Post Doctor research of the Department of Computer Science and Technology at Tsinghua University from 2001 to 2003; He is currently an associate professor of the Department of Computer Science and Technology at Tsinghua University. His research interests lie in the areas of online social network, video streaming, interactive multi-view video, and distributed video coding. He is a member of IEEE.