A Unified Personalized Video Recommendation via Dynamic Recurrent Neural Networks

Junyu Gao[1,2], Tianzhu Zhang[1,2], Changsheng Xu[1,2]
[1]National Lab of Pattern Recognition, Institute of Automation, CAS, Beijing 100190, China
[2]University of Chinese Academy of Sciences
gaojunyu2015@ia.ac.cn,{tzzhang,csxu}@nlpr.ia.ac.cn

ABSTRACT

Personalized video recommender systems play an essential role in bridging users and videos. However, most existing video recommendation methods assume that user profiles (interests) are static. In fact, the static assumption is inadequate to reflect users' dynamic interests as time goes by, especially in the online video recommendation scenarios with dramatic changes of video contents and frequent drift of users' interests over different topics. To overcome the above issue, we propose a dynamic recurrent neural network to model users' dynamic interests over time in a unified framework for personalized video recommendation. Furthermore, to build a much more comprehensive recommendation system, the proposed model is designed to exploit video semantic embedding, user interest modeling, and user relevance mining jointly to model users' preferences. By considering these three factors, the RNN model becomes an interest network which can capture users' high level interests effectively. Extensive experimental results on both single-network and cross-network video recommendation scenarios demonstrate the superior performance of the proposed model compared with other state-of-the-art algorithms.

CCS CONCEPTS

• **Information systems** → *Social recommendation*;

KEYWORDS

social data science, personalized video recommendation, recurrent neural networks, user interest modeling

1 INTRODUCTION

With the rapid development of Internet, watching videos online has become one of the most indispensable entertainments in our daily life. According to Video Brewery[1], there are about 100 million Internet users who watch online videos every day. For example, the world's largest video sharing website, YouTube, has more than 1 billion unique users who watch hundreds of millions of hours of video every day and generate billions of views[2]. Generally, this leads to a rapid growth of demand for online videos, and many online video platforms hold millions of user-uploaded videos to meet such demand. However, the vast amount of videos increase the burden of users to find satisfactory information when they attempt to watch the unseen videos [52, 53]. To overcome this problem, video recommender systems are required to play an essential role in bridging users and videos. Despite much progress has been achieved in recent years [6–8, 17, 31], online video recommendation is still a difficult task due to the huge gap between the tremendous and multifarious online videos and users' personalized interests. In this paper, our goal is to design a robust and effective algorithm for the personalized video recommendation in Online Social Networks (OSNs).

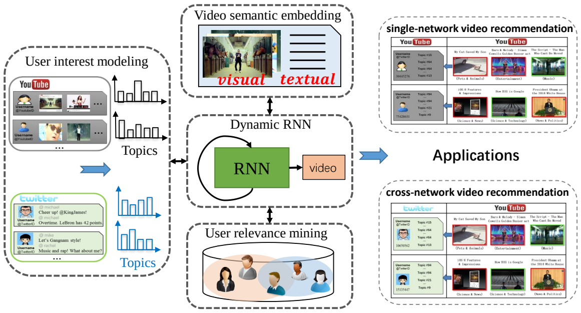

Figure 1: A simple illustration of our video recommendation strategy. We propose a dynamic RNN to model users' dynamic interests over time by considering video semantic embedding, user interest modeling, and user relevance mining in a unified framework for video recommendation, which can be applied in single-network and cross-network scenarios.

The personalized video recommendation is to exploit users' preferences and interests by analyzing various available information the users left in online social networks, such as multi-modal contents [31], social links [45, 57], explicit and implicit user ratings [16, 17, 21]. In single-network scenario, the recommendation often suffers from the cold-start problem when a new user registers on an online video website. To overcome this issue, video recommendation in cross-network scenario has been proposed [36, 48], which is to leverage users' rich cross-network activities to estimate their interests on the video sharing network, showing favorable performance, especially for new users. Generally, video recommendation algorithms for the above two scenarios can be categorized as *collaborative filtering* based methods [1, 17, 42], *content-based* methods [7, 9, 31, 35, 57, 58], or *hybrid* methods [4, 11, 48, 56]. The collaborative filtering based methods are to predict a user's ratings

[1] http://www.videobrewery.com/blog/18-video-marketing-statistics
[2] https://www.youtube.com/yt/press/statistics.html

MM'17, October 23–27, 2017, Mountain View, CA, USA.
© 2017 ACM. 978-1-4503-4906-2/17/10...$15.00
DOI: 10.1145/3123266.3123433

of videos based on the preference of other users explicitly or implicitly. The content-based methods are to calculate the similarity between videos and users' historically watched videos for recommendation. The hybrid methods usually combine the above two methods in a more complex framework. Although these algorithms show promising performance in personalized video recommendation field with various settings, most of these methods assume that user profiles (interests) are static. In fact, the static assumption is inadequate to reflect users' dynamic interests as time goes by, especially in the online video recommendation scenarios with dramatic changes of video contents and frequent drift of users' interests over different topics. Therefore, it is critical and important to exploit dynamic user interests over time for online video recommendation.

Recently, Recurrent Neural Networks (RNNs) have achieved much success in sequential modeling, such as sentence modeling [32], image captioning [20], click prediction [54], and recommendation [14, 39, 46, 51, 51]. In [23, 51], it shows that RNNs are more promising than other traditional sequential modeling methods, such as Markov Chain based methods [37]. Therefore, it is reasonable to exploit users' dynamic interests via dynamic recurrent neural networks for personalized video recommendation. Furthermore, to build a much more comprehensive recommendation system, it is important and necessary to consider three other factors including video semantic embedding, user interest modeling and user relevance mining. (1) Video semantic embedding aims to represent videos in a semantic space by using their content information. Video content is the most direct information to represent videos in recommendation scenarios. Typically, most of the content features in use are textual features [6, 8, 44]. In online video recommendation scenarios, it is far from enough to represent video context by only text. For example, many user-uploaded textual descriptions are noisy and incomplete, which makes these methods fail to achieve accurate performance in most cases. Therefore, many methods utilize multimodal features (e.g., visual and textual features) for video recommendation [7, 31, 58]. However, these methods mainly fuse these multimodal features with some manual defined weights, which may not adapt to the recommendation task with large amounts of videos. The video semantic embedding aims to map multimodal video contents into a common space to make them enhance and complement each other and learn video semantics. As a result, video contents can be represented comprehensively, which will be easily associated with user preference. (2) The user interest is essential in recommendation systems because it represents her/his preference in choosing videos. However, there always exists a gap between user interest and video semantics, which makes it difficult to determine specific videos to meet users' broad interests. Although Cui et al. [7] propose an algorithm to learn user and video representations by content attributes and social attributes, the social attributes are not often available in a single network such as YouTube. Therefore, we need to find a way to bridge video semantics and user interests in a common space with flexibility. (3) The user relevance is also important because it may provide additional constraints for improving recommendation performance. In fact, users who have large overlap in their watching history may have relatively high probability to watch similar videos in the future. As a result, we need to model these user relationships to emphasize similar interests among them for video recommendation. Although there are some methods [1, 31, 48, 56, 57] considering the above three factors more or less, little work formulates video recommendation using these factors in a unified framework which can be trained end-to-end. It is worth noting that all the three factors can be associated by users' interests. A robust video semantic representation can make it easier to bridge video content and user interest, and the user relevance is also measured by users' common interests. As a result, to make recommendation in a more reasonable and higher level by utilizing users' interests directly, these factors should be modeled jointly in recommendation framework with an end-to-end fashion.

To achieve the above goal, we propose a dynamic recurrent neural network to model users' dynamic interests over time by considering video semantic embedding, user interest modeling, and user relevance mining in a unified framework for personalized video recommendation. As shown in Figure 1, (1) We learn a common semantic space [20] by jointly embedding the visual and textual contents using semantic embedding loss to achieve robust multimodal representations for videos. Because of the relevance between both contents, we can make the visual embedding (or textual embedding) closer to its corresponding content than others. As a result, the visual and textual embedding can enhance and complement each other for better video representation, which will be associated with user interest more comprehensively. (2) To further alleviate the gap between video semantics and user interests, we derive a common interest space by jointly learning the relevance between video semantic embedding and user interested topics. The user interested topics can be discovered with online topic modeling approaches by utilizing user historical behaviors such as video watching streams and tweet steams. Through the interactions between user historical behaviors and video semantics, they are no longer isolated, i.e., the user historical behaviors can be mapped to the common interest space for representing the user's current interest, and video can also be mapped to the same space to denote the user's real-time interest. Thus we can obtain interpretable and reasonable representation for user-video recommendation in a recurrent neural network. Using this representation as input, the RNN will become an interest network, which can model users' dynamic interests over time. (3) We mine user relevance by capturing those relevant users' common interest to make the RNN model exploit the relationship among users' interest. In our implementation, the overlap between users' watching history is adopted to measure their relevance. The more similar two users are, the higher their relevance is. To achieve the relevance mining, we employ a ranking loss as an constraint in the last state of our RNN, which can remember a user's historical interests. (4) To express the contextual relationships among the recommended videos, a coherence loss is adopted in the RNN, which can make the sequentially recommended videos coherent and reasonable. Finally, our framework is applied to single-network and cross-network video recommendation. The main contributions are summarized as follows:

- We propose a dynamic recurrent neural network to model users' dynamic interests over time in a unified framework for personalized video recommendation.
- The proposed model can jointly exploit video semantic embedding, user interest modeling, and user relevance mining to

model users' preferences. By considering these three factors, the RNN model becomes an interest network which can capture users' high level interests effectively.

- Extensive experimental results on both single-network and cross-network video recommendation scenarios demonstrate the superior performance of the proposed model compared with other state-of-the-art algorithms.

The rest of the paper is organized as follows. In Section 2 we review related work. Section 3 introduces the proposed framework in details. Section 4 reports experimental results, and we conclude this work in Section 5.

2 RELATED WORK

In this section, we briefly review the existing methods on two problems related to our work, including video recommendation and recommendation based on RNNs.

Video Recommendation: The recommendation system is one of the most significant techniques for users to find videos, preceded by video searching [7, 25, 28]. In [8], YouTube introduces its video recommendation system, which recommends personalized sets of videos to users based on their activity on the site. The system is further improved by using a deep candidate generation model and a deep ranking model [6, 22, 26]. Most existing video recommendation methods commonly fall into three categories: collaborative filtering [1, 17, 42], content-based recommendation [7, 9, 31, 35, 57, 58], and hybrid recommendation [4, 11, 56]. For collaborative filtering, Baluja et al. [1] build a user-video graph to provide personalized video recommendations by propagating preference information through a variety of graphs. Huang et al. [17] propose a scalable online collaborative filtering method based on matrix factorization, with considering implicit feedback solution of different user actions for model update. However, collaborative filtering methods are hardly interpretable, which makes it difficult to generalize the learned representations to new data. Moreover, all these methods suffer from the cold start problem. To address this issue, content-based video recommendation methods suggest videos which have content characteristics similar to the ones a user liked in the past. Mei et al. [31] propose a contextual video recommendation system, VideoReach, based on multimodal content relevance and user click-through data. Deldjoo et al. [9] extract a set of stylistic features (lighting, color, and motion) for content-based video recommendation. The content-based methods are often limited by the features used for modeling content representation and users' interests. Hybrid approaches combine the above two approaches in a single framework. Zhao et al. [56] employ a multi-task ranking approach to integrate all the ranking lists generated from rich information sources. Ferracani et al. [11] exploit users' self-expression in user profiles and perception of visual saliency in videos to promote video recommendation. However, most of these methods model the user behaviors in a static way, which cannot capture uses' dynamic preference well. In this paper, we propose a unified video recommendation framework via a dynamic recurrent neural network, with considering video semantic embedding, user interest modeling, and user relevance mining jointly.

Most of the above methods focus on recommending videos on one single OSN. Recently, some preliminary work has also started to establish the association by directly observing users' collaborative behaviors in different OSNs [36, 48, 49]. This research line shows promising performance in recommendation field, especially for solving the cold-start problem. For example, Yan et al. [48] propose a unified video recommendation framework to embed the cross-OSN association in a transfer matrix. Our method can also be applied to cross-OSN video recommendation, and achieves superior performance.

Recommendation based on RNNs: Using recurrent neural networks for recommendation systems has recently received more and more attentions, such as next basket recommendation [51], shopping items recommendation [14, 24, 27], news recommendation [39], movie recommendation [46], etc. Yu et al. [51] represent a basket acquired by pooling operation as the input layer of RNN, which outperforms the state-of-the-art methods for next basket recommendation. Song et al. [39] propose a multi-rate Long Short-Term Memory (LSTM) with considering both long-term static and short-term temporal user preferences for commercial news recommendation. Compared with traditional sequential methods, the RNNs is more promising [23]. However, there is relatively little work using the RNNs for online video recommendation systems. Hidasi et al. [14, 15] utilize recurrent neural networks for session-based online video recommendation. However, the session based recommendation assumes that users' dynamics only exist in a short period (session) and each session should be treated independently. This may not be fitted for capturing a user's global interests in her/his watching history. Moreover, the session-based recommendation suffers from the cold-start problem. Despite Wu et al. [46] propose recurrent recommender networks (RRN) for modeling movie recommendation dynamically and achieve significant performance, they just employ the user rating information while ignoring the movie content. Different from existing work, we propose a unified framework which utilizes recurrent neural networks to capture users' dynamic interests over time by considering video semantic embedding, user interest modeling, and user relevance mining jointly.

3 OUR APPROACH

In this section, we introduce our proposed approach for the personalized video recommendation in details. We first introduce the video semantic embedding, user interest modeling, and user relevance mining. Then, we give the details about the unified recommendation framework. As shown in Figure 2, the proposed RNN model can be regarded as an interest network by considering the three factors jointly for learning users' dynamic preferences. Finally, our framework can be trained end-to-end by minimizing the loss in each factor as a whole objective. In the test phase, we feed the known information of a given user to the trained model to recommend the most relevant videos for her/him dynamically.

3.1 Video Semantic Embedding

In order to effectively represent videos in online social network, as shown in Figure 2 (a), we learn a common semantic space for visual and textual contents. For visual feature extraction of a given video, we follow [34] to use the output of 4096-way fc6 layer from the 19-layer VGG [38] and 4096-way fc6 layer from C3D [41] to represent sampled frames and clips. Theses frames and clips are

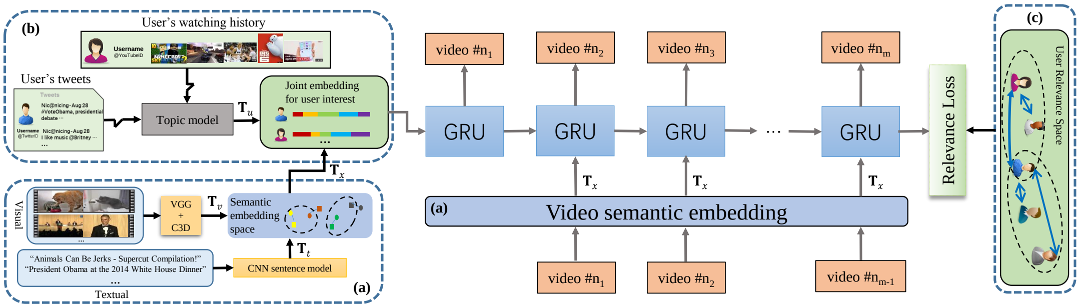

Figure 2: The unified framework for the personalized video recommendation via Dynamic Recurrent Neural Networks. In (a), the visual features (VGG and C3D) and textual features (CNN sentence model) are extracted to learn the embedding matrices T_v and T_t. In (b), we first utilize the topic model to capture users' interested topics, and then we map the users interested topics and their corresponding video semantic embedding into a common space by T_u and T_x. In (c), user relevance is mined by considering users' common interest and is adopted as an additional constraint in the RNN. Overall, the proposed RNN can effectively learn users' dynamic preferences over time.

then processed by mean pooling and concatenated together to generate a single 8192-dimensional feature vector $\mathbf{v}$. The textual description is a 300-dimensional feature vector $\mathbf{t}$ generated by the last convolutional layer of a pre-trained CNN sentence model [19]. This model takes the Word2Vecs [33] for word representation.

We assume both visual and textual cottents share similar semantics, which can be learned jointly in a common space. To project the visual and textual contents into the common semantic space, we design two transformation matrices, $\mathbf{T}_v \in \mathbb{R}^{D_e \times D_v}$ and $\mathbf{T}_t \in \mathbb{R}^{D_e \times D_t}$, where D_v, D_t and D_e are the dimensionality of the visual feature, textual feature and embedding respectively. Then the embedding of video v can be derived by $\mathbf{v}_e = \mathbf{T}_v\mathbf{v}$ and $\mathbf{t}_e = \mathbf{T}_t\mathbf{t}$, as shown in Figure 2 (a), we optimize the semantic embedding by minimizing a contrastive loss as defined in Eq. (1).

$$\begin{aligned}\mathcal{L}_{sem}(\mathcal{V}_e, \mathcal{T}_e) = &\sum_{\mathbf{v}_e \in \mathcal{V}_e, \mathbf{t}_e, \mathbf{t}'_e \in \mathcal{T}_e} \max(0, \tau_1 - \mathbf{v}_e\mathbf{t}_e + \mathbf{v}_e\mathbf{t}'_e) \\ &+ \sum_{\mathbf{v}_e, \mathbf{v}'_e \in \mathcal{V}_e, \mathbf{t}_e \in \mathcal{T}_e} \max(0, \tau_1 - \mathbf{v}_e\mathbf{t}_e + \mathbf{v}'_e\mathbf{t}_e)\end{aligned} \tag{1}$$

Where $\mathcal{V}_e$ and $\mathcal{T}_e$ are the visual and textual embedding vectors, which are first scaled to have unit norm. $\mathbf{t}'_e$ is a negative paired textual embedding for visual embedding $\mathbf{v}_e$, and vice-versa with $\mathbf{v}'_e$. τ_1 is the contrastive margin which enforces the distance between the positive paired visual and textual embedding smaller than the negative ones. In the training phase, the negative samples are randomly chosen from the training set and resampled each epoch.

Most textual contents of videos in online social networks are user-provided which may be noisy and incomplete. Additionally, visual contents are significantly diverse and hard to understand. Therefore, it is hard to project the visual and textual contents of a given video to the same point in the learned semantic space. However, due to the relevance between both contents, we can use the objective function Eq. (1) to make the visual embedding (or textual embedding) closer to its corresponding content than others. As a result, the visual and textual embedding can enhance and complement each other for better video representation, which is crucial to recommendation. Moreover, if we use more data for training, the learned semantic space will be more robust.

3.2 User Interest Modeling

Although videos have semantic information, they cannot reflect users' broad interest directly [7]. To represent videos and users' interest in a common space, as shown in Figure 2 (b), we jointly learn the relevance between video semantic embedding and user interested topics. Specifically, we first conduct some online topic modeling approach on all the users' historical behavior streams (e.g., video watching streams or tweet streams) to build a shared user topic space and learn the topical distribution for each user. Then we aggregate the topic distributions of each user's real-time behavior streams to derive the representation of user interested topics at the current time, where a time decay [10] is used to weight the behavior streams. Therefore, the user's interested topics can be defined as in Eq. (2).

$$\mathbf{u} = \frac{1}{N_u} \sum_{i \in \mathcal{B}_u} \mathbf{m}_i \cdot e^{-\lambda |t - t_i|} \tag{2}$$

Where $\mathbf{m}_i$ denotes a user's interested topics of the i^{th} behavior, $\mathcal{B}_u$ is the user's historical behaviors, $|t - t_i|$ indicates the time difference between the current time and the post time of user behavior i. N_u is a normalization parameter and λ is the time decay parameter.

To project the video semantic embedding and user interested topics into a common space, we adopt two transformation matrices, $\mathbf{T}_u \in \mathbb{R}^{D_e \times D_u}$ and $\mathbf{T}_x \in \mathbb{R}^{D_e \times 2D_e}$, where D_u is the dimensionality of the learned user topics representation. To measure the relevance between video semantic embedding and the user interested topics, one direct way is to calculate the distance between them. We integrate the video semantic embedding of a user's watching list in Eq. (3), and the distance loss is defined in Eq. (4):

$$\widetilde{\mathbf{u}} = \frac{1}{N_e} \sum_{\mathbf{v}_e \in \mathcal{U}_{\mathcal{V}_e}, \mathbf{t}_e \in \mathcal{U}_{\mathcal{T}_e}} (\mathbf{v}_e \oplus \mathbf{t}_e) e^{-\lambda |t - t_e|} \tag{3}$$

$$\mathcal{L}_{int}(\mathcal{U}, \widetilde{\mathcal{U}}) = \sum_{\mathbf{u} \in \mathcal{U}, \widetilde{\mathbf{u}} \in \widetilde{\mathcal{U}}} ||\mathbf{T}_u\mathbf{u} - \mathbf{T}_x\widetilde{\mathbf{u}}||_2^2 \tag{4}$$

Where $\mathcal{U}_{\mathcal{V}_e}$ and $\mathcal{U}_{\mathcal{T}_e}$ are the visual and textual embedding vectors of the viewed videos for user $\mathbf{u}$. $|t - t_e|$ indicates the time difference between the current time and the post time when the user watches the specific video. N_e is a normalization parameter and $\oplus$ denotes vector concatenation. It is worth noting that different from the image caption problem [20] which adopts one modal embedding to retrieve the other, we aim to represent a video by using both its visual and textual embedding, which can enhance and complement each other.

To learn a user's interested topics for her/ his historical behavior streams, we have two ways corresponding to the single-network and cross network scenarios: (1) For the single-network video recommendation, we represent a user's video watching behavior as a topical distribution of the corresponding video. Specifically, we follow [48] to utilize a multimodal topic model [2] to discover the YouTube video topics. After topic modeling, each video watching behavior can be represented as a topical distribution $m^Y \in \mathbb{R}^{D_u^Y}$, where D_u^Y is the dimension of the derived topic space. (2) For the cross-network video recommendation, we employ Twitter and Youtube as two networks, which are connected because of the overlapped user account linkage between Twitter and Youtube. When a new user registers on Youtube, the recommendation system knows nothing about her/his video preference thus cannot conduct recommendation. By leveraging users' tweeting behaviors, the proposed method can present a YouTube video recommendation application to deal with the cold-start problem. In our method, we use an incremental mode of the TwitterLDA model [55] for topic modeling, which shows favorable performance for coping with the short text characteristics of tweets. Specifically, we aggregate all the users' tweets at a daily level, and update the model with every user's tweets progressively. Therefore, the learned topic space is updated each day and can well track the recent focuses on Twitter. After topic modeling, each tweet is assigned a topic $z^T \in \{1, 2, ..., D_u^T\}$, where D_u^T is the number of topics in the generated Twitter topic space. We then represent the tweet as a delta distribution $m^T = (0, 0, ..., 1, ..., 0) \in \mathbb{R}^{D_u^T}$, where the only 1 denotes the learned topic z^t of this tweet.

In the training phase, we split the whole time space into L small sessions. At each session, we assume that all the user behaviors in and before this session can represent the current user interest. The proposed RNN is responsible for predicting the user behaviors after this session. Based on the session split, we can generate large scale of virtual user behavior sequences for model training, which will lead to a more robust user interest representation. In the test phase, given a specific user, we adopt all the user historical behaviors to model the current user interest, then we can generate a list of recommended videos step by step.

3.3 User Relevance Mining

As in [30], we use the overlap between users' watching history to denote their common interest (relevance). Given a user i, we utilize L_i to represent the set of the videos in her/his watching list. The relevance between two users, for example i and j, is defined as $Rel(i, j) = \frac{|(L_i \cap L_j)|}{|(L_i \cup L_j|)}$. Considering these relevance relationships, our model can achieve a more robust performance by capturing the relevant users' common interests. In our framework, as shown in Figure 2 (c), we use the last state of RNN to measure user relevance, where a contrastive loss is adopted for optimization. Specifically, we employ the classic GRU [5] as the basic unit of our RNN, which can capture long-range dependencies in sequence modeling. The vector formulas for a GRU layer forward pass are given as:

$$\begin{aligned} \mathbf{z}_t &= \sigma(\mathbf{U}_z \mathbf{x}_t + \mathbf{W}_z \mathbf{h}_{t-1}) \\ \mathbf{r}_t &= \sigma(\mathbf{U}_r \mathbf{x}_t + \mathbf{W}_r \mathbf{h}_{t-1}) \\ \widetilde{\mathbf{h}} &= tanh(\mathbf{U}_h \mathbf{x}_t + \mathbf{W}_h (\mathbf{h}_{t-1} \odot \mathbf{r}_t)) \\ \mathbf{h}_t &= (1 - \mathbf{z}_t)\mathbf{h}_{t-1} + \mathbf{z}_t \widetilde{\mathbf{h}} \end{aligned} \tag{5}$$

Where t is the current time, $\mathbf{z}_t$ and $\mathbf{r}_t$ are *update gate* and *reset gate*, respectively. $\mathbf{x}_t$ denotes the input. $\widetilde{\mathbf{h}}$ is the current hidden state, and $\mathbf{h}_t$ is the output of the GRU. For a given user i, we utilize the last GRU output $\mathbf{h}^i$ in the user's watching sequence as her/his global interest representation. The user relevance loss is defined as:

$$\mathcal{L}_{rel}(\mathcal{H}) = \sum_{\mathbf{h}^i, \mathbf{h}^p, \mathbf{h}^n \in \mathcal{H}} \max(0, \tau_2 - \mathbf{h}^i \mathbf{h}^p + \mathbf{h}^i \mathbf{h}^n) \tag{6}$$

Where $(\mathbf{h}^i, \mathbf{h}^p, \mathbf{h}^n)$ is a triplet, which is first scaled to have unit norm. $\mathbf{h}^i$ is a given user interest representation, $\mathbf{h}^p$ and $\mathbf{h}^n$ are the corresponding paired positive and negative representation, respectively. In the training phase, the negative samples are randomly chosen and resampled every epoch. It's worth noting that other ranking loss can also be applied to user relevance mining, such as deep relative loss[12, 50].

3.4 Unified Recommendation Framework

Illuminated by the recent successes of probabilistic sequential translation model [34, 43], given a user's current interested topics $\mathbf{u}$, we formulate our recommendation problem as a coherence loss, where the log probability of the recommendation is given by the sum of log probabilities over the watched videos as shown in Eq. (7).

$$\begin{aligned} \mathcal{L}_{rec}(\mathbf{u}, V) &= -\log \mathrm{P}(V|\mathbf{u}) \\ &= \sum_{t=1}^{N_r} -\log \mathrm{P}(v_t|\mathbf{u}, v_1, ..., v_{t-1}; \theta) \end{aligned} \tag{7}$$

Where $\{v_1, v_2, ..., v_{N_r}\}$ is the sequentially predicted videos. Here, v is corresponding to the concatenated visual and textual embedding, i.e., $\mathbf{v}_e \oplus \mathbf{t}_e$. By using the user interest embedding matrices $\mathbf{T}_u$ or $\mathbf{T}_x$, for each time step, we can get the interest embedding $\mathbf{T}_u\mathbf{u}$ or $\mathbf{T}_x(\mathbf{v}_e \oplus \mathbf{t}_e)$ as the RNN input, as shown in Figure 2, which makes our RNN become an interest network. θ are the parameters of our framework, including $\mathbf{T}_v$, $\mathbf{T}_t$, $\mathbf{T}_u$, $\mathbf{T}_x$, and the parameters of the RNN model. By minimizing the above loss, the user interest evolvement can be described dynamically, making the recommendation more coherent and reasonable. As in [6, 34], we simply use the softmax classification function to calculate the above probabilities:

$$\mathrm{P}(v_t = i|\mathbf{u}, v_1, ..., v_{t-1}; \theta) = \frac{\exp\{\mathbf{T}_p^{(i)} \mathbf{h}_t\}}{\sum_{j=1}^{N_v} \exp\{\mathbf{T}_p^{(j)} \mathbf{h}_t\}} \tag{8}$$

Where N_v is the number of videos in the video corpus, $\mathbf{T}_p$ is the parameter matrix of the softmax layer in our RNN.

Finally, we can obtain the objective function as in Eq. (9).

$$\begin{aligned} \mathcal{L} = &\sum_{\mathbf{u} \in \mathcal{U}} \mathcal{L}_{rec}(\mathbf{u}, V_u) + \lambda_1 \mathcal{L}_{sem}(\mathcal{V}_e, \mathcal{T}_e) \\ &+ \lambda_2 \mathcal{L}_{int}(\mathcal{U}, \widetilde{\mathcal{U}}) + \lambda_3 \mathcal{L}_{rel}(\mathcal{H}) + \lambda_4 ||\theta||_2^2 \end{aligned} \tag{9}$$

Where $\lambda_1, \lambda_2, \lambda_3$ are the trade-off parameters for these objectives. λ_4 is the coefficient of the weight decay term. By optimizing the above overall loss function in a unified framework, our proposed method achieves dynamic video recommendation with considering the video semantic embedding, user interest modeling, and user relevance mining jointly. In the test phase, we choose the video with the maximum probability at each time step and set it as the GRU input for the next time step.

4 EXPERIMENTS

In this section, we evaluate the performance of our proposed method on two online video recommendation applications: single-network video recommendation and cross-network video recommendation. The extensive results demonstrate the effectiveness of our method for online video recommendation. We also conduct model component analysis of our proposed framework.

4.1 Dataset Collection

To collect datasets for single-network and cross-network video recommendation, we started from Google+ website where users are willing to share their user accounts on different OSNs. For simplicity, we adopted the user accounts from cross-network dataset [47]. This dataset contains 143, 259 Google+ users, among which 38, 540 users provide YouTube account, and 11, 850 users provide both Youtube and Twitter accounts. The 38, 540 users and the 11, 850 users are recorded as the *single-network* users and *cross-network* users, respectively. We further downloaded the temporal user activities of these users from YouTube and/or Twitter, in a time range from June 1st, 2013 to June 1st, 2014. Specifically, (1) For single-network users, we downloaded her/his entire temporal video-watching behaviors. For each video, the video tags, categories, titles, and descriptions were also collected. As a result, we obtained 886, 885 video-related behaviors. To construct a dataset with appropriate user behaviors for model learning and performance evaluation, we filtered the raw single-network user set by keeping the ones who interacted with over 10 YouTube videos. The Youtube videos interacted by less than three users are also filtered out. This results in a single-network dataset of 3, 994 users and 6, 814 YouTube videos, and all the videos are downloaded via youtube-dl[3]. (2) For cross-network users, we additionally downloaded her/his generated tweets with timestamp and the user profile, which results in 3, 322, 807 tweet-related behaviors. Then we filtered the raw cross-network user set by keeping the ones who posted over 100 tweets. The rules for filtering YouTube users and videos in (1) are also conducted. As a result, we obtain 2, 522 cross-network users and 2, 859 Youtube videos.

4.2 Evaluation Metrics

For both single-network and cross-network recommendation, we adopt 80% of the users as the training set to learn our proposed model. The other 20% of users are used as the test set to evaluate the recommendation performance. Since our method is able to model user's dynamic preference, in the test set, we utilize the first n_f months for capturing user interest, and leave the remaining $12 - n_f$ months of user behaviors as the groundtruth for prediction. Specifically, (1) for single-network experiment, we design two evaluation modes: the *short* mode and the *long* mode. The n_f is set to 3 and 9 in both modes, respectively. (2) the cross-network experiment aims to tackle the cold-start problem. When a user newly registers to the system, the historical video-watching behaviors are not available. To simulate the new users in the cold-start scenario, we hide all the observed video-watching behaviors of each user in the test set, and only the tweet streams in the first $n_f (= 9)$ months can be used. Follow [36, 48], we view the online video recommendation as a Top-k recommendation task and employ *Top-k precision, recall, and F-score* as the evaluation metrics [13]. We set k to 5 in our experiments. For each test user, we utilize our dynamic RNN model to generate k videos sequentially with the highest probabilities. The evaluation metrics are computed by examining whether the test user has really watched the recommended videos in the last $12 - n_f$ months. The final results are averaged over all the test users.

4.3 Parameter Settings

For users' interested topics modeling, we resort to the standard perplexity [3] and choose the topic number that leads to small perplexity and fast convergence. Therefore, we obtain the topic numbers $D_u^Y = 70$ and $D_u^T = 100$ for YouTube and Twitter, respectively. The embedding dimension D_e is set to 300. For the time decay rate λ, we set a relatively slow decay $\lambda = 0.1$. In the model training phase, the trade-off parameters $\lambda_1, \lambda_2, \lambda_3$ are set to $0.2, 0.4, 0.2$ respectively by grid-search over $\{0.2, 0.4, 0.6, 0.8\}$ and three-fold cross validation. The coefficient λ_4 of weight decay term is set to $1e - 4$. The contrastive margin τ_1 and τ_2 are set to 0.3 and 0.5, respectively. We utilize stochastic gradient descent to optimize our model, and the learning rate is set to 0.001.

4.4 Single-network Video Recommendation

Compared Baselines: To evaluate the effectiveness of the proposed framework on single-network video recommendation problem, we compare our approach with other 5 baselines:

- Random: videos are randomly ranked for all users, which is the simplest baseline without using any information;
- Popularity: recommending popular videos with the most view count, which serves as a baseline without personalization;
- LFM: state-of-the-art Latent Factor Model [21], which mainly aims to address the sparsity problem;
- KNN: the typical item-based collaborative filtering recommendation method [18], which recommends videos to a user with the information of her/his most related users;
- FPMC: a sequential prediction algorithm based on markov chain [37], which is a baseline with considering dynamics.

Overall Recommendation Performance: The experimental results of the examined methods on both long mode and short mode are shown in Table 1. It can be observed that our method can achieve superior performance than all the other baselines. We also can obtain other observations: (1) Among these competitors, *Random* and *Popularity* get extremely bad results. The reason is that both methods only provide each user with random or the same popular videos, which fails to satisfy the user's personalized demands. (2) Two popular collaborative filtering based methods do not show superior performance in the experiments. This is because these methods cannot generalize the learned representations to new data, especially in the online video recommendation environment with large amounts of noisy and complicated videos. (3) Although the *FPMC* considers the dynamics of user behaviors and achieves favorable results in both evaluation modes, it still does not adopt the video content features and user interests. As a result, our proposed method outperforms it by (26.9%, 75.3%) with F-score in both

[3] http://rg3.github.io/youtube-dl/

Table 1: Comparisons with the baseline methods in terms of two modes by using Top-5 precision, recall and F-score.

Evaluation mode	Metrics	Random	Popularity	LFM	KNN	FPMC	Ours
long mode	*precision*	0.0007	0.0015	0.0039	0.0174	0.0216	**0.0296**
	recall	0.0005	0.0035	0.0053	0.0271	0.0357	**0.0399**
	F-score	0.0006	0.0021	0.0045	0.0212	0.0269	**0.0340**
short mode	*precision*	0.0019	0.0046	0.0028	0.0169	0.0235	**0.0350**
	recall	0.0009	0.0019	0.0023	0.0133	0.0134	**0.0259**
	F-score	0.0012	0.0027	0.0025	0.0149	0.0170	**0.0298**

modes, which also validates the effectiveness of the joint semantic embedding and user interest embedding. (4) Due to the decrease of historical behavior data, it is natural to find that all the methods degrade from the long mode to the short mode with respect to F-score. However, our method is only reduced by 12.4%, while the *LFM*, *KNN*, and *FPMC* are reduced by 44.4%, 29.7% and 36.8%, respectively. This also supports our motivation to capture users' dynamic interests in various users' video watching lists.

4.5 Cross-network Video Recommendation

Compared Baselines: To evaluate the effectiveness of the proposed method on the cross-network video recommendation for dealing with the cold-start problem, we compare our approach with four other baselines:

- Popularity: The same one as in single-network experiment;
- Modified K Nearest Neighbor (*MKNN*): instead of using users' rating behaviors in the typical user-based KNN method [40], we modify it by computing the similarity between YouTube users from their tweet behaviors, in order to handle the cold-start problem in cross-network recommendation scenario;
- Content-based Recommendation (*CB*): a typical keyword based vector space model [29], which directly utilizes the TF-IDF similarity between video's textual information and users' tweet profiles. This method severs as a content-based baseline;
- Static Cross-OSN Association (*SCA*): a typical cross-OSN video recommendation method [48], by transferring users' tweet profiles to conduct video recommendation on YouTube.

Overall Recommendation Performance: Table 2 presents comparisons with other four methods on the cold-start problem. The results are reported as Top-5 precision, recall and F-score over the test users. It is clear that our method performs noticeably well compared with all the other baselines. Moreover, we have other observations: (1) Although *Popularity* can handle the cold-start problem, it ignores the users' personalized interest. Therefore, its performance is not acceptable in the real online video recommendation scenario. (2) *MKNN* method directly utilizes the user twitter behaviors to calculate the similarities between users. However, there are huge gaps between Twitter topics and YouTube video contents. As a result, users who are similar on Twitter may not show similar video interests on YouTube. This also verifies our motivation to jointly learn an embedding between video content and user interested topics. (3) *CB* method considers the association between YouTube videos and users' twitter profile, and achieves better performance than the first two baselines. This demonstrates the necessity of using video content information. However, the naive use of video content may not meet the challenge of noisy online videos. Our method considers the semantic embedding for robust video representation, which is more reasonable. (4) *SCA* method presents a reasonable idea to bridge the gap between different OSNs, which shows favorable performance. However, *SCA* does not consider the dynamics of user interest, which may not enough to adapt to the online video recommendation scenario with rapid changes. Our method adopts the recurrent neural networks to capture users' dynamic interests, which is more reasonable and achieves better results.

Table 2: Comparisons with the baseline methods for the cold-start problem. We report Top-5 precision, recall and F-score.

Metrics	Popularity	MKNN	CB	SCA	Ours
precision	0.0015	0.0050	0.0132	0.0174	**0.0261**
recall	0.0021	0.0039	0.0111	0.0128	**0.0189**
F-score	0.0018	0.0043	0.0120	0.0147	**0.0219**

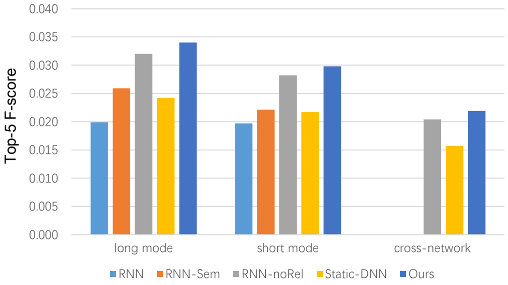

Figure 3: The Top-5 F-score of our method with different settings.

4.6 Model Component Analysis

Quantitative Analysis: Our method has four essential components including video semantic embedding, user interest modeling, user relevance mining, and the dynamic RNN. To verify the contribution of each component, we implement four variants of our approach:

- RNN: a standard RNN model using the GRU unit, which only adopts the 300-dimensional textual feature generated by [19] as model input. This baseline is a simple content-based video recommendation method;
- RNN-Sem: this variant adds the video semantic embedding into the RNN variant, and takes the concatenated visual and textual embedding as the model input;

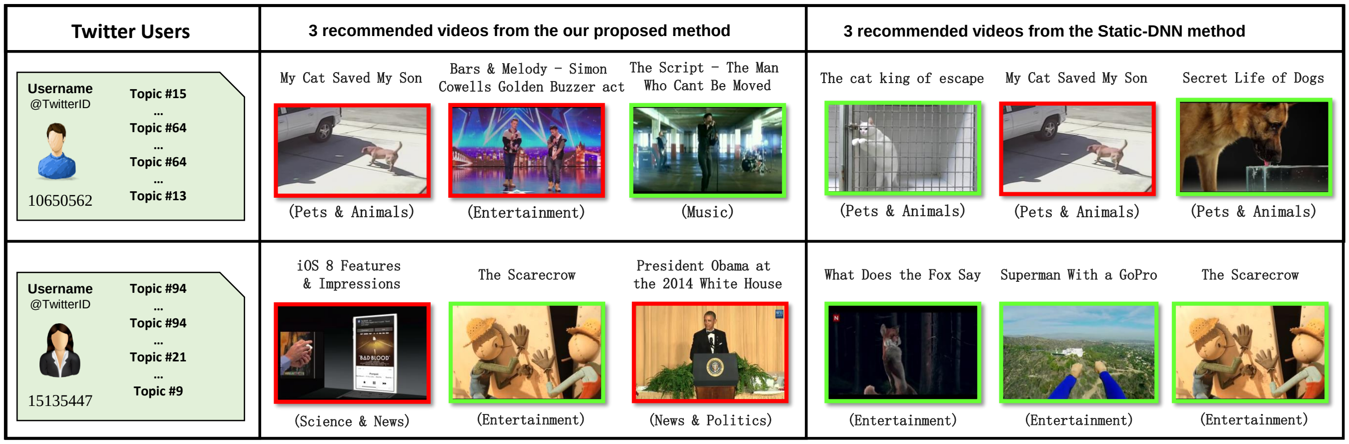

Figure 4: Comparisons of the top-3 recommended videos to two test users between our method and Static-DNN. The red and green boxes means the positive and negative predictions, respectively. See text for details.

- RNN-noRel: this variant only abandons user relevance modeling in our framework.
- Static-DNN: We replace the RNN with a three-layer deep neural network, where a ranking loss used in [6] is adopted for recommendation in a static way. The dimension of the hidden layer is set to 300. This baseline does not employ the time decay term in Eq. (2) and Eq. (3), but simply averages user's historical behaviors as the representation.

Results and Analysis: The model component analysis is conducted in both single-network and cross-network video recommendation. The comparison results of Top-5 F-score are shown in Figure 3. The results show all the components more or less contributes to the final recommendation performance. We take the results of long mode for illustration: (1) The simplest baseline RNN only adopts the original textual features as the model input, which results in inferior performance with a F-score 0.0199. We argue that the original textual feature cannot represent the video content comprehensively due to the noisy user uploaded descriptions. Therefore, we add the video semantic embedding into the RNN baseline to get RNN-Sem method. RNN-Sem achieves a F-score of 0.0259, which is significantly higher than RNN baseline by 30.2%. The result again validates that, with the video semantic embedding, the visual and textual content can enhance and complement each other for robust video representation. (2) The user interest modeling is also essential in our framework. To demonstrate this point, we improve the baseline RNN-Sem to RNN-noRel with the user interest modeling. The results show RNN-noRel gets a 23.6% promotion compared with RNN-Sem in terms of F-score. The result demonstrates that the user interest modeling can capture users' broad interests and improves the recommendation performance significantly. (3) Our method can obtain benefit from user relevance modeling. The performance of RNN-noRel suffers from a 6.3% decline when eliminating the user relevance loss of our method. (4) The result of Static-DNN demonstrates the significant importance of our dynamic modeling. Static-DNN gets a F-score of 0.0242, which suffers from a 28.8% decline. Without the dynamic modeling, the users' real-time interests cannot be captured accurately. As a result, Static-DNN may recommend some videos relevant to users' historical interest but not suitable for her/his current interests.

Qualitative Analysis: To interpret how our proposed method can dynamically recommend videos to a given user by capturing her/his real-time interest, we compare our method with Static-CNN qualitatively. In Figure 4, we show two new YouTube users with their twitter topic history information on Twitter and the corresponding recommended videos recommended by both methods. Take the test user "15135447" as an example. The Static-DNN method thinks the major interest of this user is entertainment related to topic #94 which frequently occurs in the user's historical tweet behaviors. As a result, the algorithm recommends three top trending videos about entertainment. However, the user's current interest is shifted to recent news such as the speech of Obama at the White House (topic #21) and some technological advance (topic #9). Our dynamic method can exploit the user's real-time interests using the memory unit of RNN, and gives more reasonable recommendations.

5 CONCLUSIONS AND FUTURE WORK

In this paper, we propose a dynamic model for online video recommendation via recurrent neural networks. To capture users' interests, we consider video semantic embedding, user interest modeling and user relevance mining in a unified framework, which can be trained end-to-end. The proposed method can effectively learn users' real-time preference and conduct personalized recommendation. The extensive experimental results on both single-network and cross-network video recommendation demonstrate the effectiveness of the proposed model. In the future, we will integrate other modal information, such as audio information, for recommendation. Moreover, we will consider context information in video recommendation for capturing external situations where user behaviors happen, such as time, location and so on.

6 ACKNOWLEDGEMENT

This work is supported in part by the National Natural Science Foundation of China under Grant 61432019, Grant 61572498, and Grant 61532009, the Key Research Program of Frontier Sciences, CAS, Grant NO. QYZDJ-SSW-JSC039, and the Beijing Natural Science Foundation 4172062.

REFERENCES

[1] Shumeet Baluja, Rohan Seth, D Sivakumar, Yushi Jing, Jay Yagnik, Shankar Kumar, Deepak Ravichandran, and Mohamed Aly. 2008. Video suggestion and discovery for youtube: taking random walks through the view graph. In *WWW*. 895–904.

[2] David M Blei and Michael I Jordan. 2003. Modeling annotated data. In *SIGIR*. 127–134.

[3] David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research* 3, Jan (2003), 993–1022.

[4] Bisheng Chen, Jingdong Wang, Qinghua Huang, and Tao Mei. 2012. Personalized video recommendation through tripartite graph propagation. In *MM*. 1133–1136.

[5] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *EMNLP*.

[6] Paul Covington, Jay Adams, and Emre Sargin. 2016. Deep neural networks for youtube recommendations. In *RecSys*. 191–198.

[7] Peng Cui, Zhiyu Wang, and Zhou Su. 2014. What videos are similar with you?: Learning a common attributed representation for video recommendation. In *MM*. 597–606.

[8] James Davidson, Benjamin Liebald, Junning Liu, Palash Nandy, Taylor Van Vleet, Ullas Gargi, Sujoy Gupta, Yu He, Mike Lambert, Blake Livingston, and others. 2010. The YouTube video recommendation system. In *RecSys*. 293–296.

[9] Yashar Deldjoo, Mehdi Elahi, Paolo Cremonesi, Franca Garzotto, Pietro Piazzolla, and Massimo Quadrana. 2016. Content-based video recommendation system based on stylistic visual features. *Journal on Data Semantics* 5, 2 (2016), 99–113.

[10] Yi Ding and Xue Li. 2005. Time weight collaborative filtering. In *CIKM*. 485–492.

[11] Andrea Ferracani, Daniele Pezzatini, Marco Bertini, and Alberto Del Bimbo. 2016. Item-Based Video Recommendation: An Hybrid Approach considering Human Factors. In *ICMR*. 351–354.

[12] Junyu Gao, Tianzhu Zhang, Xiaoshan Yang, and Changsheng Xu. 2017. Deep relative tracking. *IEEE Transactions on Image Processing* 26, 4 (2017), 1845–1858.

[13] Jonathan L Herlocker, Joseph A Konstan, Loren G Terveen, and John T Riedl. 2004. Evaluating collaborative filtering recommender systems. *TOIS* 22, 1 (2004), 5–53.

[14] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. 2016. Session-based recommendations with recurrent neural networks. In *ICLR*.

[15] Balázs Hidasi, Massimo Quadrana, Alexandros Karatzoglou, and Domonkos Tikk. 2016. Parallel recurrent neural network architectures for feature-rich session-based recommendations. In *RecSys*. 241–248.

[16] Yifan Hu, Yehuda Koren, and Chris Volinsky. 2008. Collaborative Filtering for Implicit Feedback Datasets. In *Eighth IEEE International Conference on Data Mining*. 263–272.

[17] Yanxiang Huang, Bin Cui, Jie Jiang, Kunqian Hong, Wenyu Zhang, and Yiran Xie. 2016. Real-time Video Recommendation Exploration. In *SIGMOD*. 35–46.

[18] George Karypis. 2001. Evaluation of item-based top-n recommendation algorithms. In *Proceedings of the tenth international conference on Information and knowledge management*. 247–254.

[19] Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *EMNLP*.

[20] Ryan Kiros, Ruslan Salakhutdinov, and Richard S. Zemel. 2014. Unifying Visual-Semantic Embeddings with Multimodal Neural Language Models. In *NIPS deep learning workshop*.

[21] Yehuda Koren. 2008. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *SIGKDD*. 426–434.

[22] Xiaodan Liang, Si Liu, Xiaohui Shen, Jianchao Yang, Luoqi Liu, Jian Dong, Liang Lin, and Shuicheng Yan. 2015. Deep human parsing with active template regression. *IEEE transactions on pattern analysis and machine intelligence* 37, 12 (2015), 2402–2414.

[23] Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. 2016. Predicting the Next Location: A Recurrent Model with Spatial and Temporal Contexts.. In *AAAI*. 194–200.

[24] Si Liu, Jiashi Feng, Csaba Domokos, Hui Xu, Junshi Huang, Zhenzhen Hu, and Shuicheng Yan. 2014. Fashion parsing with weak color-category labels. *IEEE Transactions on Multimedia* 16, 1 (2014), 253–265.

[25] Si Liu, Jiashi Feng, Zheng Song, Tianzhu Zhang, Hanqing Lu, Changsheng Xu, and Shuicheng Yan. 2012. Hi, magic closet, tell me what to wear!. In *MM*. 619–628.

[26] Si Liu, Xiaodan Liang, Luoqi Liu, Xiaohui Shen, Jianchao Yang, Changsheng Xu, Liang Lin, Xiaochun Cao, and Shuicheng Yan. 2015. Matching-cnn meets knn: Quasi-parametric human parsing. In *CVPR*. 1419–1427.

[27] Si Liu, Zheng Song, Guangcan Liu, Changsheng Xu, Hanqing Lu, and Shuicheng Yan. 2012. Street-to-shop: Cross-scenario clothing retrieval via parts alignment and auxiliary set. In *CVPR*. IEEE, 3330–3337.

[28] Si Liu, Changhu Wang, Ruihe Qian, Han Yu, and Renda Bao. 2016. Surveillance Video Parsing with Single Frame Supervision. *arXiv:1611.09587* (2016).

[29] Pasquale Lops, Marco De Gemmis, and Giovanni Semeraro. 2011. Content-based recommender systems: State of the art and trends. In *Recommender systems handbook*. Springer, 73–105.

[30] Xiaoqiang Ma, Haiyang Wang, Haitao Li, Jiangchuan Liu, and Hongbo Jiang. 2014. Exploring sharing patterns for video recommendation on YouTube-like social media. *Multimedia Systems* 20, 6 (2014), 675–691.

[31] Tao Mei, Bo Yang, Xian-Sheng Hua, and Shipeng Li. 2011. Contextual video recommendation by multimodal relevance and user feedback. *TOIS* 29, 2 (2011), 10.

[32] Tomáš Mikolov, Stefan Kombrink, Lukáš Burget, Jan Černockỳ, and Sanjeev Khudanpur. 2011. Extensions of recurrent neural network language model. In *ICASSP*. 5528–5531.

[33] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *NIPS*. 3111–3119.

[34] Yingwei Pan, Tao Mei, Ting Yao, Houqiang Li, and Yong Rui. 2016. Jointly modeling embedding and translation to bridge video and language. In *CVPR*. 4594–4602.

[35] Jonghun Park, Sang-Jin Lee, Sung-Jun Lee, Kwanho Kim, Beom-Suk Chung, and Yong-Ki Lee. 2010. An online video recommendation framework using view based tag cloud aggregation. *IEEE Multimedia* 99, 1 (2010).

[36] Shengsheng Qian, Tianzhu Zhang, Richang Hong, and Changsheng Xu. 2015. Cross-domain collaborative learning in social multimedia. In *MM*. 99–108.

[37] Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. 2010. Factorizing personalized markov chains for next-basket recommendation. In *WWW*. 811–820.

[38] Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. In *ICLR*.

[39] Yang Song, Ali Mamdouh Elkahky, and Xiaodong He. 2016. Multi-rate deep learning for temporal recommendation. In *SIGIR*. 909–912.

[40] Xiaoyuan Su and Taghi M Khoshgoftaar. 2009. A survey of collaborative filtering techniques. *Advances in artificial intelligence* 2009 (2009), 4.

[41] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. 2015. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*. 4489–4497.

[42] Mark Van Setten, Mettina Veenstra, Anton Nijholt, and Betsy van Dijk. 2003. Prediction Strategies in a TV Recommender System-Method and Experiments.. In *ICWI*. 203–210.

[43] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. 2015. Show and tell: A neural image caption generator. In *CVPR*. 3156–3164.

[44] Hao Wang, Naiyan Wang, and Dit-Yan Yeung. 2015. Collaborative deep learning for recommender systems. In *SIGKDD*. 1235–1244.

[45] Zhi Wang, Lifeng Sun, Wenwu Zhu, and Shiqiang Yang. 2013. Joint Social and Content Recommendation for User-Generated Videos in Online Social Network. *IEEE Transactions on Multimedia* 15, 3 (2013), 698–709.

[46] Chao-Yuan Wu, Amr Ahmed, Alex Beutel, Alexander J Smola, and How Jing. 2017. Recurrent Recommender Networks. In *WSDM*.

[47] Ming Yan, Jitao Sang, and Changsheng Xu. 2014. Mining cross-network association for youtube video promotion. In *MM*. 557–566.

[48] Ming Yan, Jitao Sang, and Changsheng Xu. 2015. Unified youtube video recommendation via cross-network collaboration. In *ICMR*. 19–26.

[49] Xitong Yang, Yuncheng Li, and Jiebo Luo. 2015. Pinterest board recommendation for twitter users. In *MM*. 963–966.

[50] Xiaoshan Yang, Tianzhu Zhang, Changsheng Xu, Shuicheng Yan, M Shamim Hossain, and Ahmed Ghoneim. 2016. Deep relative attributes. *IEEE Transactions on Multimedia* 18, 9 (2016), 1832–1842.

[51] Feng Yu, Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. 2016. A dynamic recurrent model for next basket recommendation. In *SIGIR*. 729–732.

[52] T Zhang, C Xu, G Zhu, S Liu, and H Lu. 2010. A generic framework for event detection in various video domains. In *MM*. 103–112.

[53] T Zhang, C Xu, G Zhu, S Liu, and H Lu. 2012. A Generic Framework for Video Annotation via Semi-supervised Learning. *IEEE Transactions on Multimedia* 14, 4 (2012), 1206–1219.

[54] Yuyu Zhang, Hanjun Dai, Chang Xu, Jun Feng, Taifeng Wang, Jiang Bian, Bin Wang, and Tie-Yan Liu. 2014. Sequential click prediction for sponsored search with recurrent neural networks. In *AAAI*.

[55] Wayne Xin Zhao, Jing Jiang, Jianshu Weng, Jing He, Ee-Peng Lim, Hongfei Yan, and Xiaoming Li. 2011. Comparing twitter and traditional media using topic models. In *European Conference on Information Retrieval*. 338–349.

[56] Xiaojian Zhao, Guangda Li, Meng Wang, Jin Yuan, Zheng-Jun Zha, Zhoujun Li, and Tat-Seng Chua. 2011. Integrating rich information for video recommendation with multi-task rank aggregation. In *MM*. 1521–1524.

[57] Xiangmin Zhou, Lei Chen, Yanchun Zhang, Longbing Cao, Guangyan Huang, and Chen Wang. 2015. Online video recommendation in sharing community. In *SIGMOD*. 1645–1656.

[58] Qiusha Zhu, Mei-Ling Shyu, and Haohong Wang. 2013. Videotopic: Content-based video recommendation using a topic model. In *Multimedia (ISM), 2013 IEEE International Symposium on*. IEEE, 219–222.