VideoTopic: Content-based Video Recommendation Using a Topic Model

Qiusha Zhu, Mei-Ling Shyu
Department of Electrical and Computer Engineering
University of Miami
Coral Gables, FL 33124, USA
Email: q.zhu2@umiami.edu, shyu@miami.edu

Haohong Wang
TCL Research America
2700 Augustine Drive
Santa Clara, CA 95054, USA
Email: haohong.wang@tcl.com

***Abstract*—Most video recommender systems limit the content to the metadata associated with the videos, which could lead to poor results since metadata is not always available or correct. Meanwhile, the visual information of videos is typically not fully explored, which is especially important for recommending new items with limited metadata information. In this paper, a novel content-based video recommendation framework, called VideoTopic, that utilizes a topic model is proposed. It decomposes the recommendation process into video representation and recommendation generation. It aims to capture user interests in videos by using a topic model to represent the videos, and then generates recommendations by finding those videos that most fit to the topic distribution of the user interests. Experimental results on the MovieLens dataset validate the effectiveness of VideoTopic by evaluating each of its components and the whole framework.**

***Keywords*-VideoTopic; content-based video recommendation; topic model; video presentation**

I. Introduction

With the huge amount of video data uploaded to the Internet everyday, how to mine useful information from these videos is a big challenge [1]. In this paper, a content-based video recommendation framework called VideoTopic is proposed, which is particularly useful in the cold-start scenarios. First, both visual and textual features of videos are extracted. Using a topic model, each video is then represented as a mixture of a set of topics, and each topic is a mixture distribution of textual and visual words/content extracted from a video collection. User interests are then estimated based on users' previously watched videos, and can also be represented as a distribution over topics. A list of personalized videos is generated by finding videos with topic distributions as close as the topic distribution of user interests. The assumption is that recommending those videos most similar to user interests can maximize the accuracy. Hence, the contributions of this study lie in two folds:

- A novel content-based recommendation framework, VideoTopic, is proposed which uses a topic model to represent both visual and textual content of videos, and links user interests and video content by estimating user interests using the topic distributions of user watched videos.
- A new approach is proposed that maps the problem of recommending personalized videos to an optimization problem, which maximizes the recommendation accuracy by minimizing the topic distribution differences between user interests and the recommended videos.

This paper is organized as follows: Section II discusses related work in this area. Section III presents the framework of VideoTopic and each of its components, followed by the experimental results in Section IV. Conclusions are drawn in Section V.

II. Related Work

A few studies have attempted to bring visual content analysis into the scope of content-based video recommendations. Mei et al. [2] presented a contextual video recommender system, called VideoReach, which fuses three models based on textual, visual, and aural information respectively. Video relevance scores from different models are calculated using different distance functions. Attention Fusion Function is applied, which first filters out the videos with low textual relevance since textual information is usually more reliable than visual and aural information. Then the relevance scores from these three modalities are combined using linear weight fusion. Online evaluation is performed with 20 subjects using about 6000 videos from MSN Video[1]. A similar framework was presented in [3] where audio, textual, and visual information are first synchronized to detect the predefined topics in news videos. The recommendation strategy is to recommend the top 5 ranked videos for a given topic as well as the videos of related topics in the topic network. The ranking is done by considering time factor, visiting times, and qualities. Their evaluation shows that the results of topic detection using the combined information sources are better than the results using a single source, but no concrete experiment was conducted to evaluate the recommendation strategy.

Compared to the aforementioned two approaches in video recommendation, we limit our content analysis to visual content and metadata. An advantage of our framework is

[1]http://video.msn.com/video.aspx?mkt=en-us&tab=soapbox/

that a topic model is used to represent the video content as well as user interests, which naturally links them and enables the representation of user interests using the watched videos.

III. The Framework of VideoTopic

The proposed recommendation framework first represents the video content using a topic model from user interests' point of view, and then captures the interests from a user's behavior history. A personalized recommendation list is then generated to fit the user's interests. The proposed recommendation framework is not limited to recommend videos. It can be applied to general items, even if only visual or textual information is available. The whole process is performed by two key components in the framework which are video representation and recommendation generation.

A. Video Topic Model

The "Bag of words" (BoW) model is a very popular model used in information retrieval (IR). It models a document as a collection or a bag of words regardless of grammar and word order. Thus a document can be represented by a sparse histogram over the vocabulary. If treating images as documents and image features/patches as words, an image can be represented by a bag of visual words, which is a sparse histogram over a vocabulary of image patches. As a result, a combined vocabulary $\boldsymbol{V} = (w_1, ..., w_V)$ can be generated from a video collection, which contains both textual words from metadata and visual words from raw video frames (or images).

Generally speaking, an image usually contains several different scenes, analog to multiple topics of a document. Hence, it is natural to apply topic models [4] in text mining to tackle the multiple scene problem in images. As one of the most widely used topic models, LDA represents a document as random mixtures over latent topics, denoted as $\boldsymbol{Z} = (z_1, ..., z_k, ..., z_K)$, where K is the total number of topics, and each topic z_k is characterized by a distribution over words. In [5], LDA has shown very promising results in categorizing 13 natural scenes. If using a topic in general to stand for both scenes of keyframes and topics of metadata, and using a word to represent a visual word as well, LDA can model each video as a mixture of topics while each topic is a mixture of words in the combined vocabulary. The topic distribution of a video and the word distribution of a topic can therefore be estimated.

B. Problem Formalization

Suppose we define K independent topics for a video collection $\boldsymbol{D} = (d_1, ..., d_i, ..., d_M)$ of size M, and each video d_i in $\boldsymbol{D}$ is independent from each other, the goal is to calculate the topic distribution of a video, denoted as the probability of the topic set $\boldsymbol{Z}$ given d_i, $P(\boldsymbol{Z}|d_i)$. This can typically be solved by Gibbs sampling [6] or variational Bayes approximation [4]. As mentioned before, we use keyframes to represent the visual content of a video, so the topic distribution calculated is frame based, and the average topic distributions of the keyframes extracted from a video is used to represent the topic distribution of the video.

If $\boldsymbol{H} = (d_1, ..., d_i, ..., d_G)$ denotes the video set containing G videos that have been watched by a user, the user's interests can be computed by the average topic distributions of the videos in $\boldsymbol{H}$. Equation (1) shows how to estimate a user's interests based on his or her video history $\boldsymbol{H}$.

Given the reasonable assumption that a user would like to watch videos having contents consistent with his or her interests, the problem of recommending videos can be formalized into an optimization problem which finds videos having topic distributions as close as the topic distribution of the user's interests, as expressed by Equation (2), where d_r denotes the recommended video. ℓ_1-norm or Manhattan distance is adopted to measure the difference between two distributions. For top-N recommendation, the top N ranked videos generated by Equation (2) are recommended to the user, and the time complexity of generating the recommendations is O(M*K).

$$P(\boldsymbol{Z}|\boldsymbol{H}) = \frac{1}{G}\sum_{i=1}^{i=G} P(\boldsymbol{Z}|d_i). \tag{1}$$

$$\begin{aligned} \arg\min_{d_r} \quad & \|P(\boldsymbol{Z}|d_r) - P(\boldsymbol{Z}|\boldsymbol{H})\|_1 \\ = \quad & \textstyle\sum_{k=1}^{k=K} |P(z_k|d_r) - P(z_k|\boldsymbol{H})|. \end{aligned} \tag{2}$$

C. A Practical Framework

Figure 1 presents a practical framework for VideoTopic, with the two key components highlighted in bold lines. The video representation can be further divided into three sub-modules: visual feature extraction, textual feature extraction, and topic model. All these tasks can be done offline to compute the topic distribution of each video using LDA, that is $P(\boldsymbol{Z}|d_i)$. Then the recommendation generation component can calculate the topic distribution of a user's interests $P(\boldsymbol{Z}|\boldsymbol{H})$ according to Equation (1) in the user interests estimation sub-module. This module allows online updating. That is, for a new user, the interests are learned on the fly as he or she watches the videos; while for an existing user, the current interests can be calculated based on the old interests and the current watched videos. After knowing the user's interests, the topic distribution distance calculation sub-module can generate a personalized recommendation list by solving Equation (2).

IV. Experiments

In the experiments, the performance evaluation of the proposed VideoTopic framework is conducted by first validating the usefulness of the topic representation of videos, and then comparing with other approaches which also utilize visual information for content-based video recommendation.

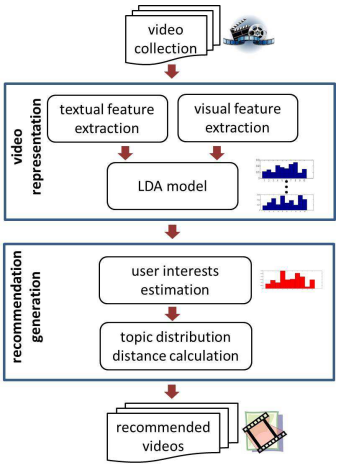

Figure 1. A practical framework of VideoTopic

A. Dataset

The MovieLens 1M dataset[2] is chosen to evaluate our proposed framework because it is widely used and publicly available. The textual information we used is movie genre information included in the dataset. In addition, we crawl movie metadata from the Internet Movie Database (IMDB) using My Movie API[3]. The crawled information includes plot, actors, directors, and writers, but only actors and directors are used considering the quality and the importance of these features as reported in [7]. For the visual information, we analyze 3475 movie trailers downloaded from YouTube[4] since trails can usually well represent the important visual content of a movie. Then 3 keyframes are extracted from each video [8] and SIFT [9] features are extracted from these keyframes. Similar to the experiment done in [7], we randomly split the movies into 5 folds in roughly equal sizes, and assign ratings to each fold accordingly to perform 5-fold cross-validation. Therefore, the items in the test set are new items which do not have any behaviors in the training set.

For evaluation metrics, we adopt common metrics - precison and recall, denoted as prec@n and recall@n respectively - where n is set to 5 in the experiments. Other metrics adopted are the area under the ROC curve (AUC), mean average precision (MAP), and normalized discounted cumulative gain (NDCG). Since these metrics are all well known, we do not further elaborate them in details considering the space limit.

B. Results and Discussion

To evaluate the performance of VideoTopic, we conduct experiments in two parts. The first part is to verify the usefulness of the topic representation on new items. For general purposes, a popular model k-nearest neighbor (kNN) is used as the recommendation algorithm for the recommendation generation component in Figure 1. The input of kNN is the topic distribution of each video, which can be viewed as features. Cosine similarity is adopted to calculate item similarities using the topic features. The kNN model feeded with pure visual features extracted from videos is denoted as V-kNN, and T-kNN is the kNN model feeded with pure textual features. Rule-based late fusion is employed to combine visual similarity and textual similarity. More specifically, the linear weighted sum approach is used and denoted as Fusion-kNN. The weights of V-kNN and T-kNN in Fusion-kNN are empirically decided, which are 0.1 for V-kNN and 0.9 for T-kNN. These weights also indicate that the scores from T-kNN are more reliable than the scores from V-kNN. We compare the performance of T-kNN, V-kNN, Fusion-kNN as well as the randomly generated results which correspond to the collaborative filtering based methods when dealing with new items. Their performance results are presented in Figure 2. The results of V-kNN on all four metrics are much better than the randomly generated results, which proves that visual features can provide some useful information. However, T-kNN still outperforms V-kNN by a large margin. This confirms the weights that are empirically decided for T-kNN and V-kNN, which is also consistent with the observation found in [2]. When combining these two information sources, the performance of Fusion-kNN is better than those of V-kNN and T-kNN. We also see that the number of topics affects the model performance. Comparing with V-kNN, the performance of T-kNN stays relatively stable as the number of topics increases. However, the results from V-kNN drop quickly. The reason is that the visual features we extracted from videos have low quality, which brings more noise when the number of topics increases. On average, the best performance of T-kNN and V-kNN is achieved when the numbers of topics are 50 and 20, respectively. For Fusion-kNN, the optimal number of topics is 50, and an equal number of topics is given to T-kNN and V-kNN, which is 25, to prevent one model from overshadowing the other.

The second part is to compare the whole framework with the two methods discussed in Section II. The recommendation generation component is also evaluated by comparing to Fusion-kNN which uses the same topic representation of videos as the input. Fusion-kNN is set as a baseline method, and as mentioned before, the number of topics is set to 50. The same number of topics is used for VideoTopic. VideoReach introduced in [2] is chosen as a comparison method, where only visual and textual features are feeded into the model. According to the properties required by the Attention Fusion Function, VideoReach first filters out videos with low textual similarity to assure all videos are more or less relevant with the query video, and then it only calculates the visual similarity of the filtered videos. We use the grid search to find this filtering threshold, and the reported results are

[2]http://www.grouplens.org
[3]http://imdbapi.org/
[4]http://www.youtube.com

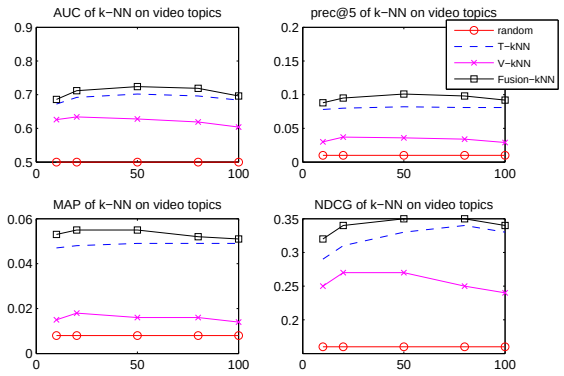

Figure 2. AUC, prec@5, MAP and NDCG of k-NN on video topics

	prec@5	AUC	MAP	NDCG
Fusion-kNN	0.10	0.69	0.062	0.39
Filter-Fusion-kNN	0.11	0.70	**0.072**	0.41
OneTopic	0.08	0.65	0.058	0.39
VideoTopic	**0.14**	**0.75**	0.071	**0.45**

generated by the best threshold when the textual similarity equals 0.6. The method used in VideoReach is essentially filtering plus Fusion-kNN, denoted as Filter-Fusion-kNN. Another comparison method is the work presented in [10]. We ignore the part of recommending videos of related topics and only recommend the top ranked videos of the majority topic as identified from the watched videos. This method is called OneTopic. Table IV-B shows the comparison results of the four methods. VideoTopic performs the best in prec@5, AUC, and NDCG, but it is slightly lower than Filter-Fusion-kNN in MAP. On average, Filter-Fusion-kNN achieves the second best results followed by Fusion-kNN. OneTopic gives the worst performance, which is because it only considers the most important topic and discards the information from the rest of the topics. The relatively high precision of Filter-Fusion-kNN is due to the effect of filtering using textual similarity to remove some noisy irrelevant videos. The fact that VideoTopic outperforms Fusion-kNN validates the effectiveness of the recommendation generation component.

V. Conclusions

This paper presents a recommendation framework that focuses on using a topic model to represent the textual and visual content of the videos in an integrated manner. Topics are used to link video content and user interests which are estimated from the topics of users' previous watched videos. Based on each user's interests, recommending a personalized list of videos is formulated into an optimization problem which maps the problem of maximizing the recommendation accuracy to minimize the topic difference between user interests and the recommended videos. The evaluation on MovieLens 1M dataset confirms that for new items, visual information does help and VideoTopic outperforms the other three comparison methods. In the future work, we need to consider the topics used in Equation (1) and Equation (2), and how to automatically identify the important topics. Another limitation of our current work is that the topic distributions of videos are based on the average of the topic distributions of keyframes. Hence, we can further take the temporal information into consideration.

References

[1] C.-D. Zhang, X. Wu, M.-L. Shyu, and Q. Peng, "A novel web video event mining framework with the integration of correlation and co-occurrence information," *JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY*, vol. 28, no. 5, pp. 788–796, 2013.

[2] T. Mei, B. Yang, X.-S. Hua, and S. Li, "Contextual video recommendation by multimodal relevance and user feedback," *ACM Transaction on Information System*, vol. 29, no. 2, pp. 1–24, apr 2011.

[3] H. Luo, J. Fan, and D. A. Keim, "Personalized news video recommendation," in *Proceedings of the 16th ACM international conference on Multimedia*, 2008, pp. 1001–1002.

[4] D. M. Blei, "Probabilistic topic models," *Communications of the ACM*, vol. 55, no. 4, pp. 77–84, apr 2012.

[5] F.-F. Li and P. Perona, "A bayesian hierarchical model for learning natural scene categories," in *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2005, pp. 524–531.

[6] I. Porteous, D. Newman, A. Ihler, A. Asuncion, P. Smyth, and M. Welling, "Fast collapsed gibbs sampling for latent dirichlet allocation," in *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2008, pp. 569–577.

[7] Z. Gantner, L. Drumond, C. Freudenthaler, S. Rendle, and L. Schmidt-Thieme, "Learning attribute-to-feature mappings for cold-start recommendations," in *Proceedings of the 2010 IEEE International Conference on Data Mining*, 2010, pp. 176–185.

[8] D. Liu, M.-L. Shyu, C. Chen, and S.-C. Chen, "Within and between shot information utilisation in video key frame extraction," *Journal of Information & Knowledge Management*, vol. 10, no. 03, pp. 247–259, 2011.

[9] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the International Conference on Computer Vision*, 1999, pp. 1150–1157.

[10] J. Liu, P. Dolan, and E. R. Pedersen, "Personalized news recommendation based on click behavior," in *Proceedings of the 15th international conference on Intelligent user interfaces*, 2010, pp. 31–40.