Personalized Video Recommendation Based on Latent Community

Han Yan, Ye Tian, Shunyao Wang, Xiangyang Gong, Xirong Que and Wendong Wang
State Lab of Networking and Switching Technologies
Beijing University of Posts and Telecommunications, China
{hanyan, yetian, wangshunyao, xygong, rongqx, wdwang}@bupt.edu.cn

***Abstract*—Facing with information overload, recommender system has been employed in many fields, from news, e-commerce to videos and musics. However, the traditional recommendation method that focuses on single individual may not has good performance because of the data sparsity and the curse of large dimensionality. Although group recommendation has been raised recently to utilize users' social information, many of them just simply aggregate users' rating information without analyzing the latent relations among users. In this paper, we proposed a latent community based video recommendation model (LCB-Rec). This method does not need explicit user preference information, it discovers the latent topics of each video with Latent Dirichlet Allocation (LDA) and finds latent user relations with Personalized PageRank. Then, latent community's profile is generated by cluster method and merge strategy. LCB-Rec focuses on giving recommendation to latent community rather than single user. We make comparative experiments with Matrix Factorization (MF) and Random Walk with Restart (RWR) based on the real-world datasets. The experiment results demonstrate that our proposed method has a better performance.**

***Keywords*—video recommendation, latent community detection, topic model**

I. Introduction

With the development of network transmission and data processing, people can spend more time interacting with the Internet. Recently, watching online video has become a popular entertainment among people. For instance, at YouTube, the world's most popular online video website, millions of users will request millions of videos in a single day. Besides, users will upload videos continuously to the YouTube with the speed of more than 24 hours of video per minute [5]. With such a tremendous video repositories, offering videos that match users' interest is a critical problem to be solved. This is why so many video websites adopt recommender system.

In general, the method for recommendation could be classified into three categories: collaborative filtering, content based method and hybrid recommendation [9]. These traditional recommendation methods focus on providing services for single user. Evidences that support this kind of recommendation are users' feedback to items, characteristic of each item and users' profile. However, facing large quantities of unregistered users and cold-start problem, the recommendation methods mentioned above may not have a good performance.

DOI reference number: 10.18293/SEKE2020-023

Since it is hard to recommend for single user, how about offering recommendation for a group of similar users to decrease dimensionality. In fact, it is reasonable to offer recommendation to a group of users. Since in real life, people with similar interest tend to like the same things [1]. However, in most circumstances [10], raw data does not contain enough information about users' social relations. Hence, it is necessary to find out the "latent social network" of users.

Recently, bullet comments are very popular in many video websites. As a form of socialized application, bullet comments give a real-time interaction between video contents and users' inside idea. Bullet comments can reflect topics of video, feelings of users, and what users may be interested in. Such vast amounts of information would be helpful to find the "latent community", and with these latent communities we can give a better recommendation for community users.

The main contributions and solved problems of this paper are as follows:

- We treat bullet comments as a "corpus" and build topic model with this "corpus" and LDA method, which returns the topic distribution for each video.
- We build a directed tripartite graph and apply Personalized PageRank to find similar users. we employ cluster method and merge strategy to generate the topic distribution of latent community.
- We compute the pearson correlations between new videos and each community and rank these videos with this correlations. The top k videos will be recommended.

II. Related Work

A. Traditional Recommendation

Generally speaking, traditional recommendation methods may be divided into three classes: collaborative filtering, content based and hybrid of the two [9]. Content based method provides recommendation by analyzing item's similarity or user's preference [6]. Collaborative Filtering (CF) recommendation is based on users' past behaviors. It assumes that users with similar behavior history tend to have same interests. It uses the past item-rating matrix to build a model for the purpose of measuring similarity between users.

B. Group Recommendation

To utilize users' social information, latent information based recommendation has been proposed. Christakopoulou et al. [4]

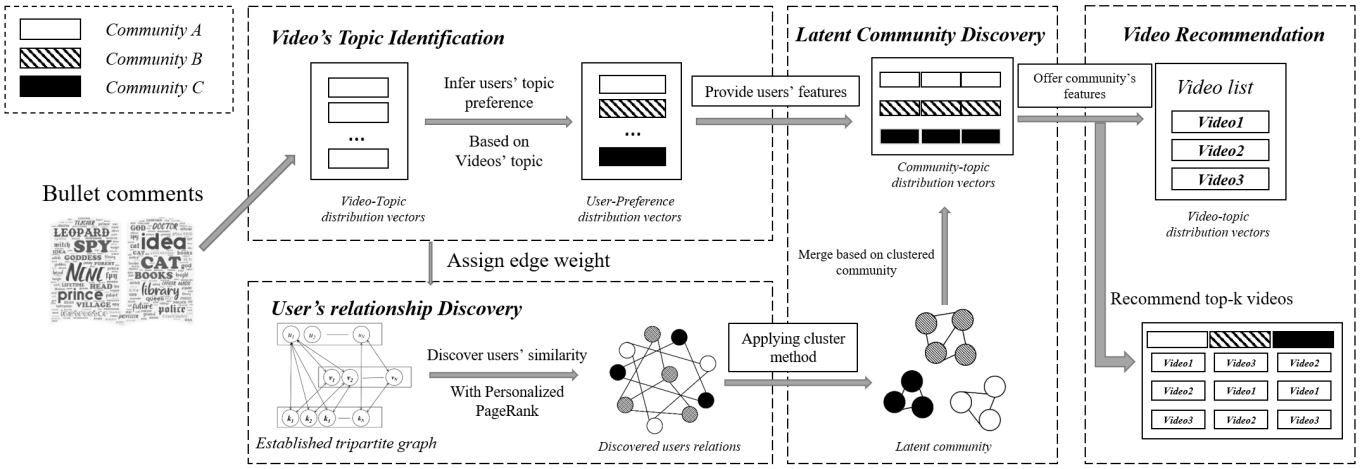


Fig. 1: Framework of the recommendation model

proposed SVD based model to learn latent user relations from rating patterns. What's more, some recommendation method tends to detect latent community. Cao et al. [2] proposed an improved CF algorithm, which predicts rating scores based on communities. Cheng et al. [3] proposed a method to detect overlapping community in complex network. Lin et al. [7] proposed a recommendation model with implicit communities from user ratings and social connections.

However, methods in [2][4][7] need explicit rating information such as "like" or "dislike", our proposed method focuses on offering recommendation by mining data without explicit ratings. Our method utilizes users' watching records and their bullet comments to discover latent factors.

III. Framework of Model

As Figure.1 shows, our recommendation model consists of four parts: video's topic identification, user relationship discovery, latent community discovery and recommendation for community.

A. Video's Topic Identification

We focus on analyzing the similarity between community's topic distribution and video's topic distribution. Thus, it is critical to dig out latent topics from video's bullet comments.

In this part, we discover the latent probabilistic topic distribution from each video. Topic models such as LDA is employed to extract the abstract topics from documents as a form of document-topic distribution and topic-word distribution. Although, LDA cannot be applied to videos directly, the bullet comments of the videos contain detailed information of each video. Therefore, our proposed method heuristically treats each video as a document, and all bullet comments will be regard as the contents of document sets.

B. User relationship discovery

In this part, we utilize the existing user-video relations in the raw data and latent video-topic distribution found with LDA to identify latent user relations. We build a directed tripartite graph indicating relations among users, videos and topics. The weight η of each edge in graph is defined as follows:

1) Weight $\eta_{i,j}$ of the edge pointing from user u_i to video v_j : $\eta_{i,j} = \frac{C_{i,j}}{C_i}$, where $C_{i,j}$ is the number of bullet comments that user u_i made in video v_j, C_i is the number of all bullet comments made by user u_i.
2) Weight $\eta_{i,j}$ of the edge pointing from video v_i to topic k_j: $\eta_{i,j} = \theta_{i,j}$, where $\theta_{i,j}$ denotes the probability that video v_i belongs to topic k_j.
3) For the edge pointing from user u_i to topic k_j: $\eta_{i,j} = \frac{1}{C_i}\sum_{v=1}^{V}\sum_{c\in v}\theta_{v,j}^{\langle c\rangle}$, where C_i is the number of bullet comments made by user u_i, V is the number of videos. v identifies a single video. c indicates a single bullet comment. $\theta_{v,j}^{\langle c\rangle}$ is the probability that video v belongs to topic k_j and comment c is from user u_i to video v.
4) For the edges of other directions, the weight is calculated as below: $\eta_{i,j} = \frac{1}{|out(i)|}$, where the $|out(i)|$ is the out degree of the node i in the graph.

In this part, we make a matrix implementation of Personalized PageRank and describe the graph in a form of transition matrix M. The final matrix R that describe the degree of user similarity could be calculated as:

$$R = (E - dM^T)^{-1}(1-d) \tag{1}$$

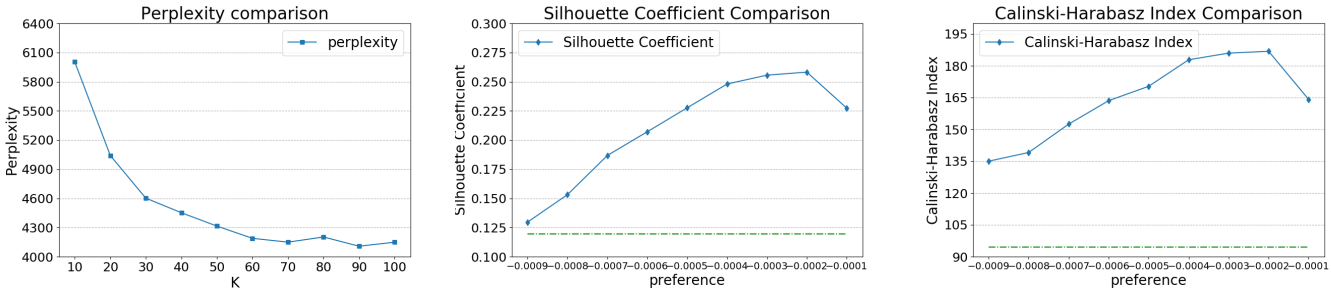
where E is diagonal matrix, d is the damping factor.

C. Latent community discovery

The essence of latent community discovery is to explore users' relationships and gather similar users. Affinity Propagation is a cluster method taking similarity matrix of sample nodes as input, which make it suitable for this problem.

The latent community profile is produced from user's profile. Three merge strategies (average strategy, least misery strategy and most pleasure strategy) are employed to generate community profile $\vec{g}_c$ for community c.

Average merging strategy (AMS) is a synthesize consideration of all users, $\vec{g}_c(k) = \sum_{x\in c}\frac{\vec{u}_x(k)}{N}$.



(a) Perplexity comparison with different K (b) Silhouette Coefficient comparison (c) Calinski-Harabasz index comparison

Fig. 2: Metrics comparison to select appropriate parameter K and preference

Least misery strategy (LeMS) represents lower bound of all users, $\vec{g}_c(k) = \min_{x \in c} \vec{u}_x(k)$

Most pleasure strategy (MoPS) represents upper bound of all users, $\vec{g}_c(k) = \max_{x \in c} \vec{u}_x(k)$.

$\vec{g}_c(k)$ is a vector denoting community c's preference for topic k, $\vec{u}_x(k)$ is a vector denoting user x's preference for topic k, N is the number of users in community c.

D. Video Recommendation

The recommendation is executed by analyzing correlations between latent community's topic distribution and video's topic distribution. As there is no information about new coming video's topic distribution, for each new coming video, we load the topic-word matrix Φ trained with LDA, transpose and normalize the matrix to get word-topic matrix Φ^T. For each word w_i in the new video's bullet comments, we sample word's topic k_j with probability Φ^T_{ij}. After that, we get a statistics vector $\vec{n} = \{n_1, n_2, \cdots, n_K\}$. Each element n_j denotes the number of words in each topic k_j. Finally, we calculate the topic distribution $\vec{r}_v = \{r_1, r_2, \cdots, r_K\}$ of video v , where $r_j = \frac{n_j}{\sum_{i=1}^{K} n_i}$.

With the topic distribution $\vec{r}_v = \{r_1, r_2, \cdots, r_K\}$ of video v and topic distribution $\vec{g}_c = \{g_1, g_2, \cdots, g_K\}$ of community c, we can use the Pearson correlation coefficient to measure the similarity between latent community c and video v:

$$corr(c, v) = \frac{\sum_{k=1}^{K}(r_k - \overline{r}) \times (g_k - \overline{g})}{\sqrt{\sum_{k=1}^{K}(r_k - \overline{r})^2} \times \sqrt{\sum_{k=1}^{K}(g_k - \overline{g})^2}} \quad (2)$$

where $\overline{r} = \frac{1}{K}\sum_{k=1}^{K} r_k$, $\overline{g} = \frac{1}{K}\sum_{k=1}^{K} g_k$.

We rank videos according to the Pearson correlation coefficient and select the top-k videos for recommending.

IV. Experiment

In this section, we perform experiments to answer the following questions: (1) What is the proper parameter K (the number of topics) to be set in the Latent Dirichlet Allocation. (2) What is the proper parameter $preference$ to be set in the Affinity Propagation cluster procedure. (3) Does our proposed method (LCB-Rec) have a better performance than the other recommendation methods.

A. Dataset

We obtained 3,847 video's bullet comments from video web sites (https://www.bilibili.com). To make sure that LDA have enough training data, we select top 120 videos and each video contains at least 8000 comments.

B. Comparative Methods

To evaluate the performance of LCB-Rec, Matrix Factorization (MF) and Random Walk with Restart (RWR) [8] are used for comparative experiments. LCB-Rec with different merging strategies and individual recommendation (Indi-Rec) without community profile are also comparative experiment.

C. Evaluation Metrics

1) Perplexity: Perplexity measures how well a probability distribution predicts a sample. Model with lower perplexity owns better performance. The definition of perplexity is:

$$perplexity = exp\{-\frac{\sum_{v=1}^{V} log(p(w_v))}{\sum_{v=1}^{V} N_v}\} \quad (3)$$

where V denotes the number of videos, N_v indicates the number of words without repetition in video v. $p(w_v)$ indicates word w's distribution in the video v's comments.

2) Cluster Performance Metrics: As the latent community is generated without the ground truth labels, it is indeed to use some metrics for evaluation.

We use Silhouette Coefficient (SC) and Calinski-Harabasz index (CH) to evaluate cluster performance. The definitions about the two metrics could be found within scikit-learn. A higher value of SC or CH relates to a better model.

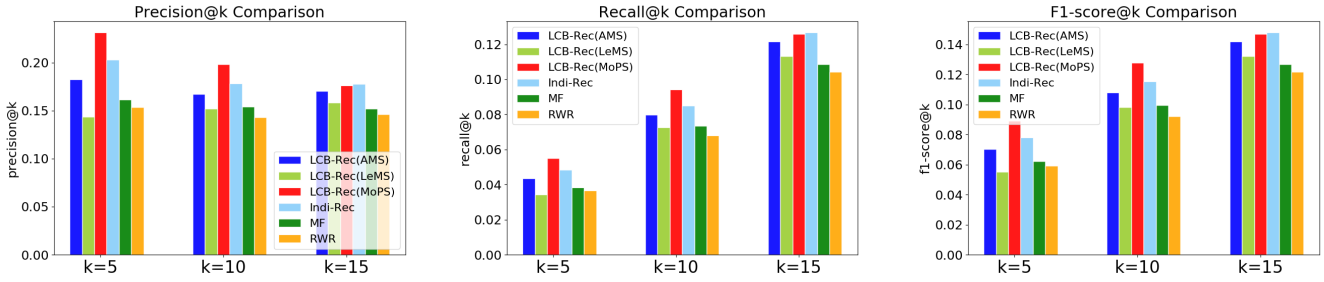
3) Top-k Metrics: We measure $precision@k$, $recall@k$ and $f_1score@k$ of each method to evaluate the performance. Definitions of these metrics are:

$$precision@k = \frac{|Actual(k) \cap Predicted(k)|}{Predicted(k)} \quad (4)$$

$$recall@k = \frac{|Actual(k) \cap Predicted(k)|}{Actual(k)} \quad (5)$$

$$f_1score@k = \frac{2 \times precision@k \times recall@k}{precision@k + recall@k} \quad (6)$$

where $Actual(k)$ is the top k actual videos' set, $Predicted(k)$ is the top k predicted videos' set.



(a) $precision@k$ comparison (b) $recall@k$ comparison (c) $f_1score@k$ comparison

Fig. 3: Performance comparison with MF, RWR, Indi-Rec

D. Experiments Results and Analysis

Figure.2a shows the variation of perplexity with 10 different K. With a synthetic consideration of the training set's size and the variation of perplexity, we choose the value of K equals 60, since the perplexity decrease drastically with value of K from 10 to 60, and decrease slowly from 60 to 100.

To determine the proper value of $preference$ in Affinity Propagation, we measure Silhouette Coefficient (SC) and Calinski-Harabasz index (CH) of each cluster result. We select 9 values of $preference$ from -0.0009 to -0.0001 with a step of 0.0001 and use the default $preference$ value as the baseline (Shown as a horizontal green line in the figure).

As Figure.2b and 2c shows, both metrics (SC and CH) indicate that the best cluster result is when the $preference$ equals -0.0002. This cluster result ($preference$ equals -0.0002) will be used to generate latent community.

We compare the performance of LCB-Rec, MF, RWR and Indi-Rec. As Figure.3 shows, LCB-Rec method with "MoPS" or "AMS" strategy and Indi-Rec method has a better performance than MF and RWR methods. The performance improvement owes to following reasons: First, LCB-Rec method and Indi-Rec method both discover latent topic distribution with users' bullet comments, while MF and RWR simply use users' rating information. Second, LCB-Rec method generates latent community, which reduces the dimensions of latent relation matrix. While Indi-Rec method lacks information about similar users. Thus, LCB-Rec method with "MoPS" strategy has better impact than Indi-Rec.

V. Conclusion

In this paper, we focus on recommending videos to a latent community rather than a single user. Despite simply aggregating users' rating information, we try to discover the latent information among users and build a latent community with the help of topic model and cluster method. Compared to other recommending method like MF and RWR, our proposed method shows better performance.

Discovering latent information from user generated content (UGC), like bullet comments, does help to boost recommendation performance. In the future, more latent information could be excavated from UGC data to strengthen algorithm.

Acknowledgment

This work was supported by National Natural Science Foundation of China (Grant No.61602051), and the Key Laboratory of Embedded System and Service Computing, China Ministry of Education (ESSCKF 2019-09).

References

[1] Irfan Ali and Sang-Wook Kim. Group recommendations: approaches and evaluation. In *Proceedings of the 9th International Conference on Ubiquitous Information Management and Communication*, page 105. ACM, 2015.

[2] Cen Cao, Qingjian Ni, and Yuqing Zhai. An improved collaborative filtering recommendation algorithm based on community detection in social networks. In *Proceedings of the 2015 annual conference on genetic and evolutionary computation*, pages 1–8. ACM, 2015.

[3] J. Cheng, X. Wu, M. Zhou, S. Gao, Z. Huang, and C. Liu. A novel method for detecting new overlapping community in complex evolving networks. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 49(9):1832–1844, 2019.

[4] Evangelia Christakopoulou and George Karypis. Local latent space models for top-n recommendation. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1235–1243. ACM, 2018.

[5] James Davidson, Benjamin Liebald, Junning Liu, Palash Nandy, Taylor Van Vleet, Ullas Gargi, Sujoy Gupta, Yu He, Mike Lambert, Blake Livingston, et al. The youtube video recommendation system. In *Proceedings of the fourth ACM conference on Recommender systems*, pages 293–296. ACM, 2010.

[6] Shanshan Feng, Huaxiang Zhang, Jian Cao, and Yan Yao. Merging user social network into the random walk model for better group recommendation. *Applied Intelligence*, 49(6):2046–2058, 2019.

[7] Xiao Lin, Min Zhang, Yiqun Liu, and Shaoping Ma. Enhancing personalized recommendation by implicit preference communities modeling. *ACM Transactions on Information Systems (TOIS)*, 37(4):48, 2019.

[8] Haekyu Park, Jinhong Jung, and U Kang. A comparative study of matrix factorization and random walk with restart in recommender systems. In *2017 IEEE International Conference on Big Data (Big Data)*, pages 756–765. IEEE, 2017.

[9] Lalita Sharma and Anju Gera. A survey of recommendation system: Research challenges. *International Journal of Engineering Trends and Technology (IJETT)*, 4(5):1989–1992, 2013.

[10] Jing Shi, Bin Wu, and Xiuqin Lin. A latent group model for group recommendation. In *2015 IEEE International conference on mobile services*, pages 233–238. IEEE, 2015.