User Profiling by Combining Topic Modeling and Pointwise Mutual Information (TM-PMI)

Lifang Wu[1(✉)], Dan Wang[1], Cheng Guo[1], Jianan Zhang[1], and Chang wen Chen[2]

[1] School of Electronic Information and Control Engineering, Beijing University of Technology, Beijing 100124, China
lfwu@bjut.edu.cn, anncheng.student@sina.com, {wangdan2013, jnzhang}@emails.bjut.edu.cn

[2] Department of Computer Science and Engineering, State University of New York at Buffalo, 316 Davis Hall, Buffalo, NY 14260-2500, USA
chencw@buffalo.edu

Abstract. User profiling is one of the key issues in personalized recommendation systems. A content curation social network is a content-centric network; it encourages users to repin items from other users and other websites. It further permits users to arrange the pins according to their interests. It is therefore possible to estimate user interest from the pins. In this paper, we propose a user profiling approach to combining topic model and pointwise mutual information (TM-PMI). We first extract a pin's description, and then apply latent Dirichlet allocation (LDA, one of the topic modeling schemes). A three-layer hierarchical Bayesian model of user-topic-word is thus obtained. Then, a personal model is obtained by selecting a set of correlated words with constraints of word probability and PMI. The experimental results confirm the efficiency of the proposed approach.

Keywords: Topic modeling · Latent dirichlet allocation · Pointwise mutual information · User profile

1 Introduction

With the development of Web 2.0 technology, social network services (SNSs), such as blogs, Facebook, Twitter, Weibo, and Flickr are being widely used and play an important role in human lives. In 2009, Pinterest, one of the early content curation social networks (CCSNs), was launched. Pinterest is a photo sharing social networking site. With new CCSNs such as Huaban.com and Snip.it, CCSNs have attracted many social network users [1, 2].

The user-generated media on SNSs has brought about an explosion of information on SNSs. The generated information is much more than one person can deal with, leading to information overload. This has led to an increasing amount of research in the area of social recommendation. User profiling is one of the key issues in social recommendation.

Q. Tian et al. (Eds.): MMM 2016, Part II, LNCS 9517, pp. 152–161, 2016.
DOI: 10.1007/978-3-319-27674-8_14

The conventional SNSs are user-centric networks [3], which are not optimized to create comprehensive user profiles based on user-generated content. Rather, CCSNs permit users to arrange and categorize data from social media for the purpose of further consumption. For example, on Pinterest, users (called "pinners") find photos (called "pins") which are interesting in them, and manually organize these photos into boards of their own. Therefore, it is possible to extract reliable social cues based on user preference. In particular, user curation through multimedia content helps to encode multi-level content-content connections, which help to pinpoint user preference in terms of the content generated by the user. Such connections between two items include user-level connection, where the strength indicates how many users have pinned both items. For example, if two images are shared by only a few users (or bundles), the connection between them rarely suggests similar user interest. However, when shared by many users they likely indicate the same interest.

Motivated by this knowledge, we propose a scheme for modeling the user profile based on all the items repined by the user. A pin includes significant information [4], such as a description written by the user, and a link to the original item, as shown in Fig. 1. In this paper, the description information is used for user profiling. The description is text-based.

Fig. 1. A pin in Pinterest

Topic modeling [5] is an effective scheme to extract a document's hidden topics. The current popular models include Probabilistic Latent Semantic Analysis (PLSA) and Latent Dirichlet allocation (LDA). PLSA, first proposed by Hofmann [6], is a method by which a document composed of multiple topics is presented in terms of the probability in the vocabulary. Each word in the vocabulary is generated from a topic. However, PLSA is only efficient for the documents in the training set. Blei [7] extended PLSA into LDA by introducing the Dirichlet distribution. It works with both the training set and the testing set. From that point, LDA has been a popular and promising scheme of topic modeling. The improved LDA algorithms [8–10] are proposed for user profiling in SNSs.

The traditional LDA scheme combines users and terms together; users are mixtures of multiple topics. A topic has a probability distribution over a vocabulary of terms. The advantage is that it picks out coherent and semantic properties of correlated terms. However, the terms of probabilistic topics are unable to give an accurate representation

of users. Accurate estimation of user preference is a challenging problem. To address this problem, we introduce pointwise mutual information (PMI).

PMI [14] is used to measure the connection between two items. If the PMI value is greater than zero, then these two items are considered related; otherwise, they are considered unrelated. In this paper, PMI is introduced to measure the correlation between a word from the topics and a specific user. The correlated words are then used to represent the user.

A user profiling scheme is proposed by combining LDA and PMI. First, LDA is applied to the descriptions of all pins from all users. A three-layer hierarchical Bayesian model of user-topic-word is then obtained. The personal model is then obtained by selecting a set of correlated words with the constraints of word probability and PMI. The results of user profiling are tested using pin recommendation results from Levenshtein distance of user profiles. The similarities between the recommended pins and the target user are then evaluated by a user study.

2 The Proposed Approach

2.1 Framework of the Proposed Approach

The framework of the proposed approach is shown in Fig. 2. First, the descriptions of all pins from all users are collected. The description is then preprocessed, including word segmentation and the building of the dictionary. Second, LDA is used to extract the probability of the words, and the word frequency vector for each user is generated. Then, the PMI between each user and a topic word is computed. Finally, user topic words are generated from LDA, using PMI and word frequency constraints. The user profile is represented as the user topic words and their probabilities. The user profiles are then tested using pin recommendation.

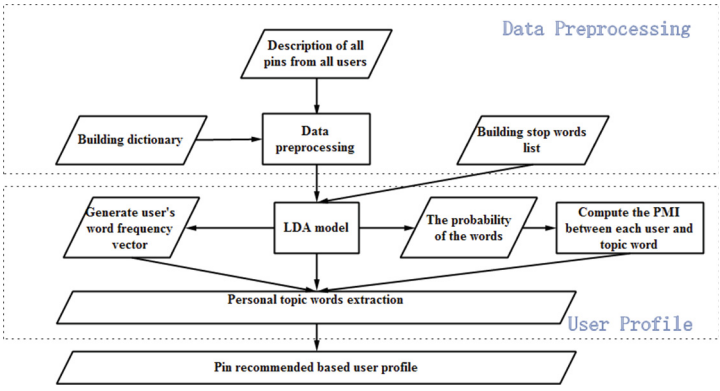

Fig. 2. Framework of the proposed approach

2.2 Data Preprocessing

In this paper, pin descriptions from Huaban.com are used, which are composed of Chinese-language documents. The first task is word segmentation. The popular Chinese word segmentation tool ICTCLAS [11] (Institute of Computing Technology, Chinese Lexical Analysis System) is employed. ICTCLAS includes word segmentation, part-of-speech tagging, and unknown word recognition; it also supports multiclass Chinese code such as GB2312, GBK, and UTF8. Following this, we set up a complementary dictionary, which includes approximately 300,000 new words. These words are frequently used in the network and are not included in the ICTCLAS dictionary. Third, generate a complementary dictionary, which includes 1433 stop words, which are filtered out.

After preprocessing, all user descriptions are represented as isolated words.

2.3 LDA from Description of User Pins

An LDA model can be represented as a three-layer hierarchical Bayesian model of user-topic-word:

- Word layer: the set of words $V = \{w_1, w_2, \ldots, w_V\}$ with frequency greater than or equal to 10.
- Topic layer: the set of topics $Z = \{z_1, z_2, \ldots, z_k, \ldots, z_K\}$, where each component is a probability distribution on the word set V, presented as $\varphi_{\mathbf{k}} = \{p_{k,1}, p_{k,2}, \ldots p_{k,i}, \ldots, p_{k,V}\}$, where $p_{k,i}$ is the probability that word w_i is generated from topic z_k.
- User layer: at the word layer, a user can be represented as a word frequency vector $\boldsymbol{u} = \{tf_{u,1} tf_{u,2}, \ldots, tf_{u,w_j}, \ldots, tf_{u,w_V}\}$, where, tf_{u,w_j} is the frequency of word w_j in user $\boldsymbol{u}$; at the topic layer, a user can be represented as $\boldsymbol{\theta_u} = \{p_{u,1}, p_{u,2}, \ldots, p_{u,z_k}, \ldots, p_{u,z_K}\}, p_{u,z_k}$, the probability of topic z_k for that user.

Figure 3 illustrates generation of the LDA model; we now introduce the generation process in detail:

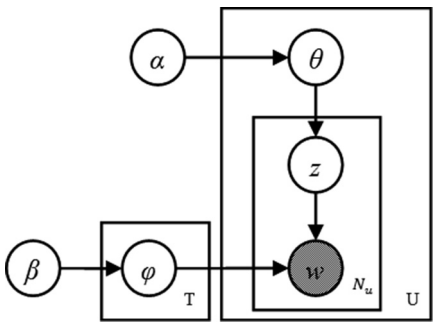

Fig. 3. Graphical model representation of the LDA model

1. For each topic $z_k, k \in \{1, 2, \ldots, K\}$:

   Draw a distribution over words $\varphi_k \sim \mathrm{Dir}(\beta)$

2. For each user $\boldsymbol{u} \in (1, 2, \ldots, U)$:
   (1) Draw a vector of topic proportions $\theta_u \sim \mathrm{Dir}(\alpha)$
   (2) For each word in user $\boldsymbol{u}$:
      (a) Draw a topic assignment $z_{u,n} \sim \mathrm{Multi}(\theta_d), z_{u,n} \in \{k = 1, 2, \ldots, K\}$
      (b) Draw a word $w_{u,n} \sim \mathrm{Multi}\left(\varphi_{z_{u,n}}\right), w_{u,n} \in \{w_1, w_2, \ldots, w_V\}$

Markov chain Monte Carlo (MCMC) is a popular probability reasoning algorithm. It is an iterative approximation algorithm to extract samples from a complex probability distribution. Gibbs sampling [12, 13] is an algorithm for obtaining a sequence of observations using MCMC. In this paper, Gibbs sampling is used to obtain the probability distribution of user-topic θ and topic-word φ.

$$P(z_{-u} = k|\{z_{-u}, w\}) \propto \exp\{\log \theta_{u,k} + \log \varphi_{k,w_u}\} \tag{1}$$

$$\theta_{u,k} = \frac{\alpha_k + n_{k|u}}{K * \alpha_k + n_{\cdot|u}} \tag{2}$$

$$\varphi_{k,w_u} = \frac{\beta + n_{w_u|k}}{\beta * V + n_{\cdot|k}} \tag{3}$$

2.4 Pointwise Mutual Information (PMI)

PMI [14] is proposed based on mutual information. It is generally used to measure the connection between two items. PMI is obtained by extracting the probability by which two events happen simultaneously. For two items x and y, their PMI could be computed as follows:

$$\mathrm{PMI}(x,y) = \log \frac{p(x,y)}{p(x)p(y)} = \log \frac{p(x|y)}{p(x)} = \log \frac{p(y|x)}{p(y)} \tag{4}$$

From Eq. (4), we can infer the following:

- If $\mathrm{PMI}(x, y) > 0$, then item x and item y are correlated with high probability. Larger PMI values indicate higher probability.
- If $\mathrm{PMI}(x, y) = 0$, then x and y are isolated from each other.
- If $\mathrm{PMI}(x, y) < 0$, then x and y are complementary to each other.

2.5 Personal Topic Words Extraction

For a specific user, personal topic words are extracted by combining the PMI and LDA models.

For a specific user $\boldsymbol{u}$, we first order the topics by decreasing probability $\mathbf{p} = \{p_1, p_2, \ldots, p_K\}$. We select the first j topics corresponding to the first 80 % of probability [16]. Then, we sample $\boldsymbol{N}$ topic words from these topics according to the word frequency to get the word sets $\boldsymbol{w}_{\mathbf{LDA}} = \{w_1, w_2, \ldots, w_N\}$. Then we compute PMI between each word and the user.

$$\begin{aligned} \mathrm{PMI}(\mathrm{user}, \mathrm{word}) &= \log_2 \frac{\mathrm{p}(\mathrm{user}, \mathrm{word})}{\mathrm{p}(\mathrm{user})\mathrm{p}(\mathrm{word})} \\ &= \log_2 \frac{\mathrm{p}(\mathrm{word}|\mathrm{user})}{\mathrm{p}(\mathrm{word})} \\ &= \log_2 \frac{\sum_{z=k}^{K} \mathrm{p}(\mathrm{word}|\mathrm{topic})\mathrm{p}(\mathrm{topic}|\mathrm{user})}{\mathrm{p}(\mathrm{word})} \end{aligned} \tag{5}$$

If PMI for a given word is greater than or equal to 0, we assume this word is correlated to the user, and the word is preserved. Otherwise, the word is discarded.

Using this method, we generate the personal topic word set $\mathbf{tw}_{\boldsymbol{u}}$ for all users, u = 1,2,…,U. These topic words represent user preference.

2.6 Pins Recommended Based User Profile

In order to test the performance of our method, we recommend pins to a user based on the user profile model. The consistency between the recommended results and user's actual preference measures the accuracy of the user profile.

For a specific user $\boldsymbol{u}$, we first select the related pins based on his/her topic word set $\mathbf{tw}_{\boldsymbol{u}}$ and obtain the related pin set $\boldsymbol{p}_{-\boldsymbol{u}}$. We compute the normalized Levenshtein distance [15] between a pin that does not belong to user $\boldsymbol{u}$. The minimum distance is used as the distance between the pin and user $\boldsymbol{u}$. If the minimum distance is smaller than 0.8, we assume this pin is related to user $\boldsymbol{u}$, and we can recommend this pin to user $\boldsymbol{u}$.

3 Experiments

3.1 Dataset

The experimental dataset was crawled from a typical Chinese content curation social network, Huaban.com, which includes 34 categories (as show in Fig. 4.). In our experiments, three categories: "home", "food_drink" and "design" were randomly selected. A total of 100 users (35/30/35 for three categories, respectively) were randomly selected. From there, 633,337 pins (179,081 pins for users in "home", 225,273 pins for users in "food_drink" and 228,983 pins for users in "design") were selected.

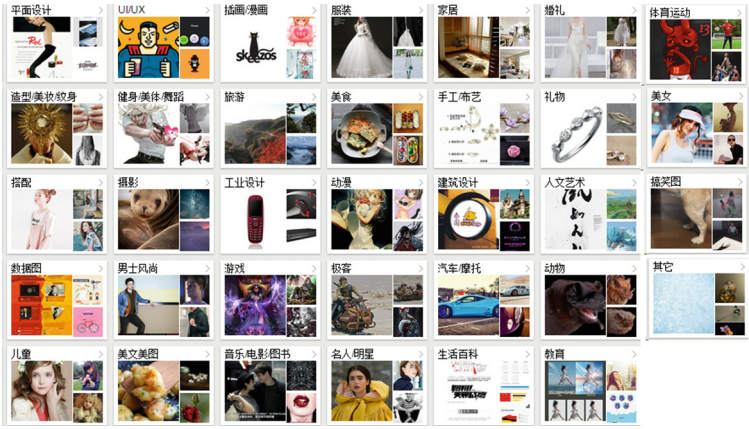

Fig. 4. Total 34 categories in Huaban.com

3.2 Perplexity

Perplexity is a metric to measure the language generation model. It measures the prediction ability of the model with respect to a new document. The smaller the perplexity value, the higher the prediction ability. In this paper, perplexity is used to measure the performance of user profiling:

$$Perplexity(U_{test}) = exp\left\{-\frac{\sum_{u=1}^{U_t} \log p(w_u)}{\sum_{u=1}^{U_t} N_u}\right\} \tag{6}$$

$$p(w_u) = \prod_{n=1}^{N_m} \sum_{k=1}^{K} p(w_n|z_k)p(z_k|u) \tag{7}$$

Where U_{test} is the user in the testing set, U_t is the user number in the testing set, w_u is the word set of pin description from user $\boldsymbol{u}$, $p(w_u)$ is the generated user model from pin description of user $\boldsymbol{u}$, and N_u is the topic word number. In our experiment, 10 % of the data is used for testing. The results are shown in Fig. 5.

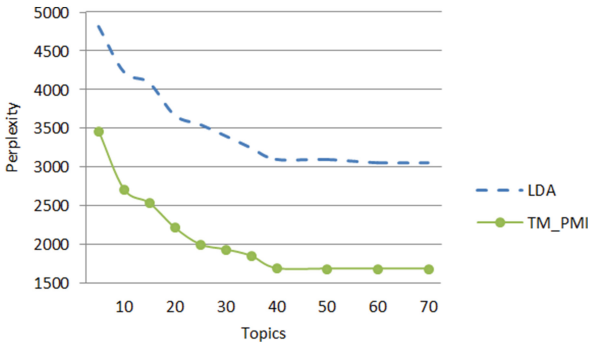

Fig. 5. Perplexity results

From Fig. 5 we can observe that the performance of user profiling is related to the topic number. Perplexity decreases as topic number increases, and Perplexity is stable when T >= 40. In the following experiment, the topic number is selected as T = 40.

3.3 User Study

This experiment is used to test the performance of algorithm-based recommendation by measuring the correlation between recommended pins and the target user.

For the 100 users in the training set, we recommended 10 pins to each user from LDA and TM-PMI, for a total of 2000 recommended pins. For each recommended pin, we selected three related pins from the corresponding user as shown in Fig. 6. The correlation between the recommended pins and the three pins from the user were tested by a user study.

Fig. 6. The recommended pin and related pins of the target user

A total of 40 subjects were selected for the user study. Each subject was randomly assigned 100 pins and required to give the correlation value (0–5) between the recommended pins and each related pin. The maximum correlation value was considered to be the correlation of the test set.

For each set, we assumed that the ground truth correlation is 5, and then based on the user study measurement, we computed the root mean squared error (RMSE) and mean absolute error (MAE) values as follows.

$$\mathrm{MAE} = \frac{\sum_{(u,\alpha)\in T} \left| r_{u\alpha} - r'_{u\alpha} \right|}{|T|} \tag{8}$$

$$\mathrm{RMSE} = \frac{\sqrt{\sum_{(u,\alpha)\in T} \left(r_{u\alpha} - r'_{u\alpha} \right)^2}}{|T|} \tag{9}$$

Where $r_{u\alpha}$ is the ground truth, which is generally 5; $r'_{u\alpha}$ is the rating from the user study; $|T|$ is the testing set.

3.4 Influence of the Number of Topic Words on the Result

In Sect. 2.5, some words are obtained from the former 80 % topics to represent the user. The number of the topic words is N. In order to test the influence of N on the recommendation results, the compared experiments are conducted with N = 20, 30, and 40 respectively. We also compare our algorithm with LDA. The experimental results are shown in Fig. 7. From Fig. 7, we can see that our algorithm is better than LDA with N = 20, 30, and 40. And our algorithm gets the best results with N = 40.

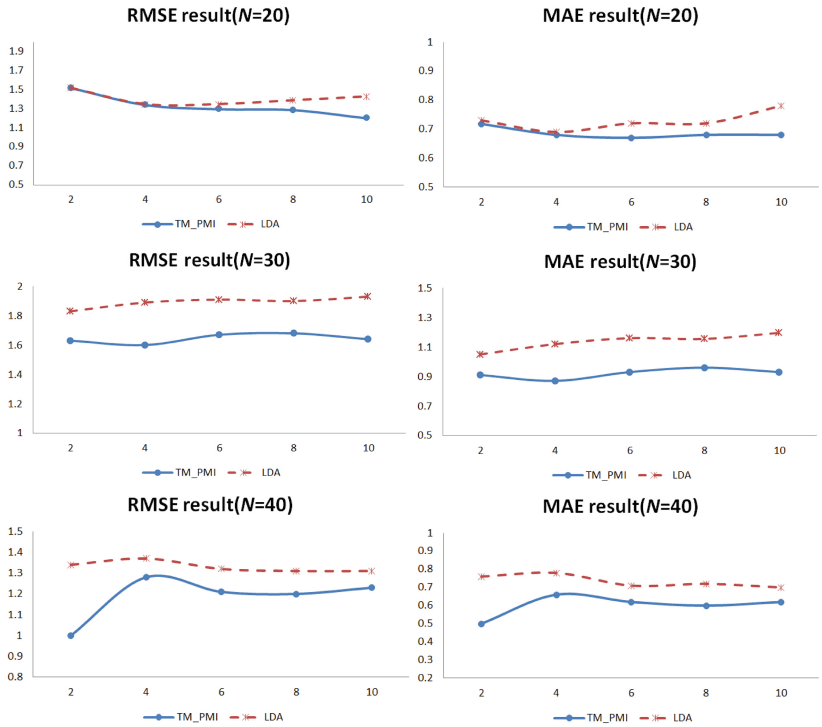

Fig. 7. Comparison of the experimental results (N = 20, 30 and 40)

4 Conclusion and Future Work

In this paper, we proposed a user profiling approach, combining LDA, for obtaining a three-layer hierarchical Bayesian model of user-topic-word, and pointwise mutual information to model a user profile. The model was tested using perplexity and user study of the recommended pins. The experimental results showed that the proposed

algorithm performs slightly better than the traditional LDA. In this paper, only user description was used for user profiling. In future work, multimodal information will be introduced for user profiling.

References

1. Hall, C., Zarro, M.: Social curation on the website Pinterest.com. Proc. Am. Soc. Inf. Sci. Technol. **49**(1), 1–9 (2012)
2. Gilbert, E., Bakhshi, S., Chang, S., Terveen, L.: "I need to try this"? a statistical overview of pinterest. In: CHI, pp. 2427–2436. ACM (2013)
3. Geng, X., Zhang, H., Song, Z., Yang, Y., Luan, H., Chua, T.: One of a kind: user profiling by social curation. In: Proceedings of the ACM International Conference on Multimedia, Orlando, pp. 567–576. ACM (2014)
4. Bernardini, C., Silverston, T., Festor, O.: A Pin is worth a thousand words: characterization of publications in Pinterest, pp. 322–327. IEEE (2014)
5. Blei, D., Carin, L., Dunson, D.: Probabilistic topic models: a focus on graphical model design and applications to document and image analysis. IEEE Signal Process. Mag. **27**(6), 55–65 (2010)
6. Hofmann, T.: Probabilistic latent semantic analysis. In: Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence, Stockholm, pp. 289–296. Morgan Kaufmann Publishers Inc., (1999)
7. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet allocation. J. Mach. Learn. Res. **3**(4–5), 993–1022 (2003). doi:10.1162/jmlr.2003.3.4-5.993
8. Rosen-Zvi, M., Griffiths, T., Steyvers, M., Smyth, P.: The author-topic model for authors and documents. In: Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence, Banff, pp. 487–494. AUAI Press (2004)
9. McCallum, A., Wang, X., Corrada-Emmanuel, A.: Topic and role discovery in social networks with experiments on enron and academic email. J. Artif. Intell. Res. **30**, 249–272 (2007)
10. Weng, J., Lim, E., Jiang, J., He, Q.: TwitterRank: finding topic-sensitive influential twitterers. In: Proceedings of the Third ACM International Conference on Web Search and Data Mining, New York, pp. 261–270. ACM (2010)
11. Zhang, H., Yu, H., Xiong, D., Liu, Q.: HHMM-based Chinese lexical analyzer ICTCLAS. In: Proceedings of the Second SIGHAN Workshop on Chinese Language Processing, Vol. 17, Sapporo, Association for Computational Linguistics, pp. 184–187 (2003)
12. Griffiths, T.L., Steyvers, M.: Finding scientific topics. Proc. Natl. Acad. Sci. USA **1011**, 5228–5235 (2004). doi:10.1073/pnas.0307752101
13. Heinrich, G.: Parameter estimation for text analysis. Technical Note Version 2.4, vsonix. Technical Report (2008)
14. https://en.wikipedia.org/wiki/Pointwise_mutual_information
15. https://en.wikipedia.org/wiki/Levenshtein_distance
16. https://en.wikipedia.org/wiki/Pareto_principle