Recommend Social Network Users Favorite Brands

He Feng and Xueming Qian

SMILES LAB, Xi'an Jiaotong University, China
fenghe7658@stu.xjtu.edu.cn, qianxm@mail.xjtu.edu.cn

Abstract. With the development of social network and image sharing websites, users are willing to upload their favorite photos on the websites and assign them some texts to describe the image content. Thus we can capture their interest by these photos and corresponding texts, and recommend relevant brands based on user's interest. This paper proposes a novel brands recommendation approach for social network users based on their browsing images and labeled texts. Firstly, we enrich the uploaded image's texts by image annotation approach. Secondly, we build brand tree from the collected datasets. And then, we recommend brands by scalable brand mining based on tree structure. Finally, we conduct a series of experiments on real Flickr users. The experiment results show the effectiveness of our approach.

Keywords: User's Interest, Brands Recommendation, Brand Tree, Scalable, Social Network.

1 Introduction

Nowadays, we are moving forward into a new era with the rapid development of network. It almost becomes an essential part of daily life, and an important way to obtain and disseminate information. Especially, the birth of web2.0 improves user's initiative in Internet greatly, and then brings volume of social network such as Flickr, Facebook, Twitter etc. The booming of information and user's invaluable resource on the Internet can bring huge economic benefits. Advertising is one of the important ways. It is not only a way of businessmen to release goods information but also an approach of users to get services in network economy.

If ads can meet people's need, then they become useful information to them and can facilitate their life. For example, for the social user shared photos, Qian et al., proposed to model users' tagging behavior and then recommend user preferred vocabularies for user by fusing the photo taken time, visual information, and photo taken locations [8]. How to make ads meet user's need? We think that it is necessary to advertise from user's interest. It is the first step to attract user's attention. Furthermore, some users have their favorite brands, which are the more accurate interests. If we can mine user's favorite brands information from their browsed images and labeled texts, then we will recommend more user-targeted ads to get the favor of users. So, to make ads meet user's need and clicked, we should solve brand recommendation problem firstly.

B. Huet et al. (Eds.): PCM 2013, LNCS 8294, pp. 730–739, 2013.

Previous works [5],[6],[9] paid more attention on produces advertising based on user's interest. They learnt and matched the topics of user's photos and products based on ODP tree, which did well in semantic gap and vocabulary impedance problem of the vocabulary of users' photos and products, but little attention on brands recommendation, because both products' description and ODP's webs content have less relevance to brand. We think user's favorite brand is more accurate user interest. So we build brand tree including more brand information offline to handle semantic gap and vocabulary impedance problem like ODP tree of Argo [6]. However, Argo mapped user photos tags (user labeled and annotated) to all leaf nodes of ODP tree to get user photos' topic distribution to represent user's interest. It did solve textual ambiguities and semantic mismatch problem, but make user's interest less prominent and susceptible to noise cause of the large number of leaf nodes. Some of the leaf nodes' topic has little relevance to user photos' topic and little effect on textual ambiguities and semantic mismatch problem. And different users correspond to different leaf nodes. So, to make user's interest prominent and robust, we need get candidate leaf nodes relevant to user photos' topic, which will also shorten match time.

Motivated by the necessity of user favorite brand mining and the potential improvement of Argo, we propose a novel approach scalable brands recommendation to match user interested brand products from one picture and corresponding texts of user browsed. The contribution of this paper is twofold: Firstly, we could get user's more accurate interest by brand mining. Secondly, our scalable approach recommends brands layer by layer, which firstly detect user interested categories to get candidate brands (leaf nodes) relevant to user photos' topic and then rank them. We believe that user's interest is relevant to several categories usually. For example, user's interested in sports, and he may be interested in clothes of Adidas.

2 The Approach

With the rage of the social network, users' browsed photos and corresponding texts reflect their interest and even brand preference.

To recommend more user-targeted ads, we need to mine user's favorite brand from their browsed photos and corresponding texts. So we should get keywords to represent user's interest and then match brands effectively. The system overview of scalable brand recommendation for social network users is shown as Figure 1. Firstly, one of user's browsing photo and corresponding texts are system input. Secondly, we enrich picture's texts with image annotation approach in [7], [10]. Here we denote user's label texts and generated annotations as user tags (UT). Thirdly, we use scalable brand mining to get brand distribution to represent user's brand preference based on brand tree, which can be seen as the constraint condition to brand recommendation online. To express our scalable approach better, we detail the brand tree in next part firstly, and then scalable brand mining. Fourth, we rank brands based on brand mining. At last, the targeted brands are the real results from our approach.

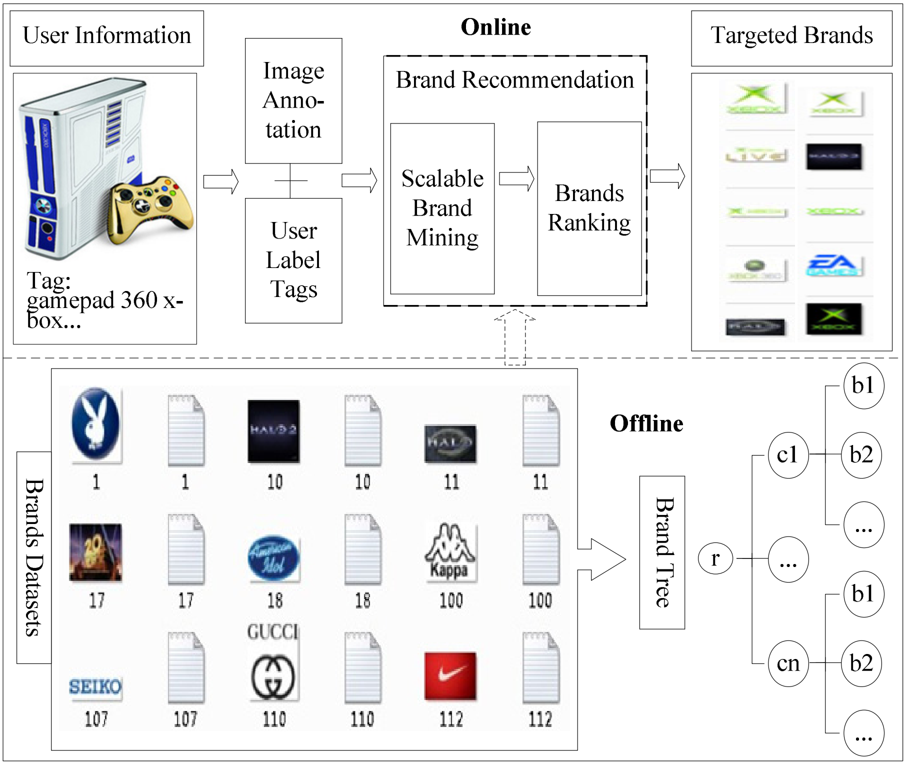

Fig. 1. System overview of scalable brands recommendation for social network users

2.1 Brand Tree

Argo's topic distribution model solved vocabulary impedance problem well, but little attention on brands mining, because both products' description and ODP's webs content have less relevance to brand. So, we build brand tree (Figure 2) from brands datasets to get brand distribution like ODP tree of Argo. However, there are two main difference: 1) brand tree is build based on volumes of brands' description [4] including brand category, brand name, products' description of the brand, and some keywords of the products' description as shown in Table 1. This information can be mined from Internet; and 2) Argo recommends products based on ODP tree without consideration of brand, while ours recommends user interested brands from brand tree. There are two layers of brand tree structure. The first layer is products category layer, and the second is brand layer. We put the same brand category of brands datasets into one big category, which build the first layer of brand tree. In our test brand datasets, we get 20 big categories which almost cover brand category of all products. They are Arts & Entertainment, Automobile, Clothes & Fashion & Beauty product, Community & Government, Computers & Electronics, Education, Food &

Dining, Health & Medicine, Home & Family, Hotel, Industry & Energy, Legal & Financial, Media & Communications, Miscellaneous, Office Products, Other, Pets, Sports, Tools & Hardware, Travel & Transportation.

Different categories of the first layer have different brand nodes of the second layer. For example, the Sports category has brand nodes of ADIDAS, NIKE, and YONEX etc.

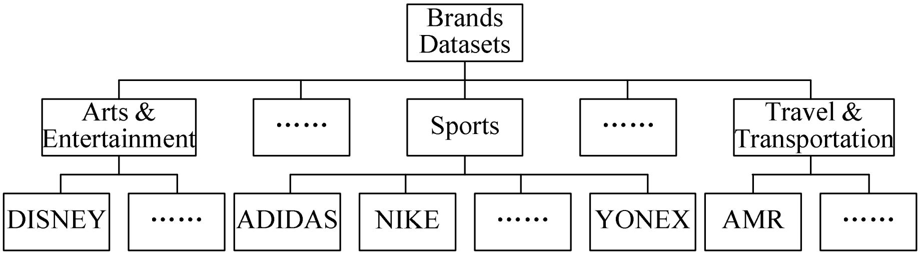

Fig. 2. Brand tree

Table 1. An example of brand description

Nike	
brand category	Sports
brand name	Nike
products' description	Nike is a major American supplier of athletic shoes, apparel and sports equipment.
products' keywords	athletic shoes, sport

2.2 Scalable Brand Mining

To learn efficient user interest, we remove the high-noise term of UT. And then, we detect user interested categories (UIC) to filter irrelevant brand nodes. Finally, we score candidate brands' similarity to user's interest to get brand distribution.

Noise Tag Filter

Usually, the labeled texts for user shared photo and even the annotations are not all relevant to photo topic. Some of texts are noise which will give negative results for detecting user interested category.

Many previous works used TF-IDF to evaluate term's importance of a document in set of files. It can be also used to reduce noise terms of a document out set of files. We reduce noise of UT by IDF as Eq. (1). Here UT is the document out set of files including 20 categories' description of first layer in brand tree. Because noisy terms have global distribution (the term appears in most categories of first layer) while theme-clear terms have localized distribution (the term appears in a few categories of first layer) [6], we can discriminate them by IDF. Let $T=\{t_1,t_2,...,t_M\}$ denotes UT, and $\Gamma=\{\tau_1,\tau_2,...,\tau_m\}$ denotes the filtered texts, where M is the total number of terms in UT, m is the number of lower-noise terms in UT, and $m \le M$, $\tau_i \in T$.

$$IDF_i = \sum_{j=1}^{n} D(d_j, t_i), i \in \{1, \cdots, M\} \quad (1)$$

where $D(x, y)$ denotes whether term y appears in document x or not. We have $D(x, y) = 1$ if y appears in x, and $D(x, y) = 0$ otherwise. Here $y = t_i$ is the i-th term of T, $x = d_j$ is the document of all brands product description of the j-th category of first layer. And n=20 is the number of categories in first layer.

Thus IDF_i is the times that t_i appears in all categories of first layer. The higher the IDF_i, the more likely the term t_i is a noise term. If IDF_i is large enough then we consider the term t_i has less contribution for detecting user interested category. So, it is settled as user interest irrelevant term.

Detect UIC

We use α_j to denote the possibility of the j-th category to be user's interest as Eq. (2).

$$\alpha_j = \sum_{i=1}^{m} F(d_j, \tau_i), j \in \{1, ..., n\} \quad (2)$$

where $F(x, y)$ is frequencies that term y appears in document x. Here $y = \tau_i$ is the i-th term of Γ, $x = d_j$ is the document of all brands product description of the j-th category of the first layer.

Thus the higher the α_j, the more likely the category is user's interest, and the more likely the brands belonging to the category are user interested brands. Then we rank these categories by α_j to get top k user's interest relevant categories to filter irrelevant brand nodes. And we compare the performance with different k in Section 3.1.

We weight top k categories in the following two ways: 1) put $c_j = 1, j \in \{1,\ldots,k\}$. We call it the unweighted scale method; and 2) normalize α_j of the top k categories as Eq. (3):

$$c_j = \alpha_j / \sum_{j=1}^{k} \alpha_j, j \in \{1, ..., k\} \quad (3)$$

And we compare the performance of these two weighted ways in Section 3.2.

Brand Distribution

After detecting UIC, we consider the k categories' brands as candidate. We use β_j^l to denote the possibility of the l-th brand of j-th category to be user interested brand as Eq. (4).

$$\beta_j^l = \sum_{i=1}^{m} F(b_j^l, \tau_i), j \in \{1, ..., k\}, l \in \{1, ..., N\} \quad (4)$$

where τ_i is the i-th term of Γ, b_j^l is the document of the l-th brand node description of j-th category. And we will normalize β_j^l. It's worth noting that different category has different brand node number N of brand tree.

We use s_j^l to denote similarity between the l-th brand of j-th category to user's interest as Eq. (5), and then we get brand distribution to represent user's brand preference.

$$s_j^l = c_j \cdot \beta_j^l, j \in \{1, ..., k\}, l \in \{1, ..., N\} \quad (5)$$

We can see that not all of brand nodes are taken into consideration and we add a new factor (category weight) to make suggested brands more prominent.

2.3 Brands Ranking

We rank brands by the value of s_j^l. Then, We have two ranking results (unweighted scale and scalable approach) corresponding to two category weight ways. From Eq. (5), we can see that Argo is a special case of our approach when k equals n (the number of categories of first layer) and categories are unweighted.

3 Experiments

Experiments are conducted to evaluate the performance of our proposed scalable brands recommendation based on tree structure approach.

We use 125 users' pictures and corresponding labeled text from Flickr [1] to test our scalable approach. And the test brand datasets contains 7,284 brands logo and corresponding brands description [3][4].

We invite three volunteers to evaluate top 10 targeted brands for each of the Flickr user as irrelevant, relevant or perfect like Argo [6], by browsing the shared photos and the textual descriptions. "irrelevant" means the recommended brand is a false-alarm. "relevant" means the recommended brand is somewhat relevant, and "perfect" means strong relevance. And then, AP (average precision) and WAP (weighted average precision) [6] defined as Eq. (6) and Eq. (7) were used for the evaluation.

$$\mathrm{AP} = (p + r) / (p + r + i) \tag{6}$$

$$\mathrm{WAP} = (p + 0.5 * r) / (p + r + i) \tag{7}$$

where p, r, i denote the number of "perfect", "relevant", and "irrelevant" brands respectively.

3.1 Performance Comparison with Different *k* Categories

Here we conducted experiment to compare our approach scalable brands recommendation based on tree structure with different k categories, and the categories are unweighted.

Figure 3 illustrates the AP and the WAP of top 10 recommended brands to the 125 users of three different k values. We can see that the performance of k=2 has the highest AP and WAP at almost all top 10 results, because user's interest is usually relevant to a few categories.

3.2 Performance Comparison between Scalable Brand Ranking and Argo

Here we conducted experiment to compare our scalable approach with non-scale approach Argo [6] based on brand tree. And we make k=3 in detect UIC part.

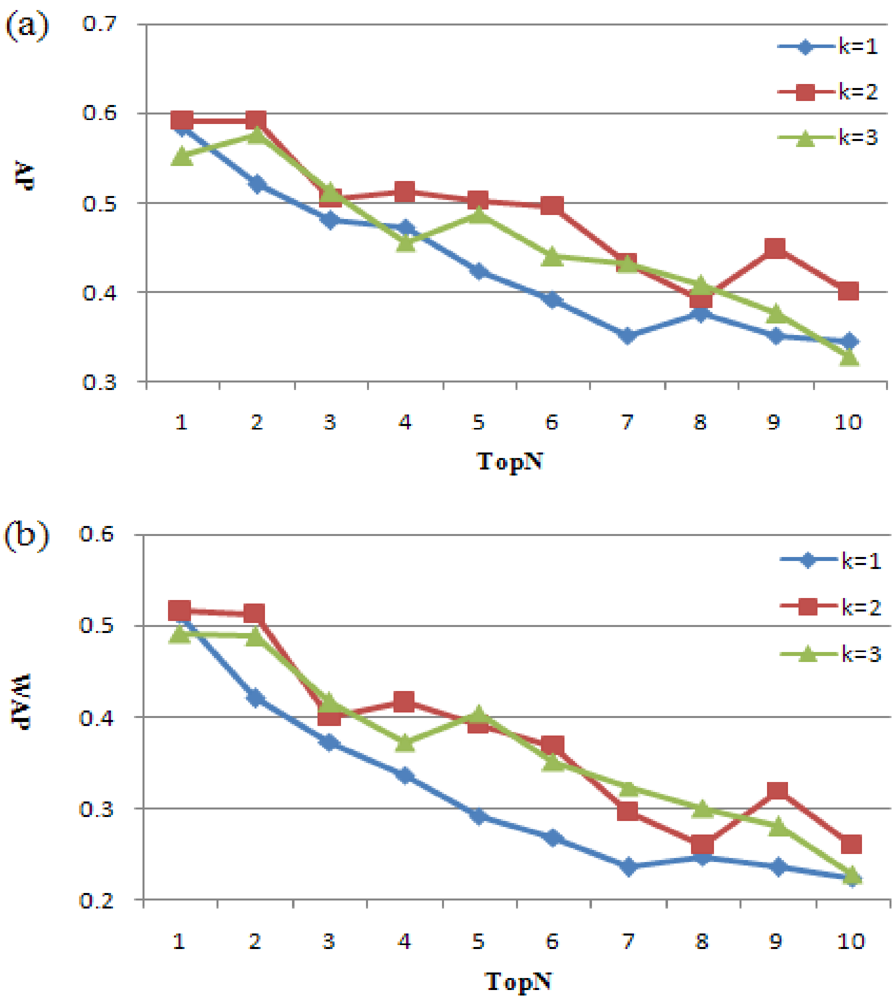

Fig. 3. Performance of scalable brands recommendation with different *k* categories

Figure 4 illustrates the AP and the WAP of top 10 recommended brands to the 125 users of three approaches (Scalable brand ranking, unweighted scalable approach and Argo). From Figure 4, we can see that the AP and WAP value are declining overall from the top1 to top10 result. The weighted scalable approach has the highest AP and WAP values. The AP value ranged from 0.648 to 0.376, the WAP value ranged from 0.576 to 0.3 for the top ranked 10 products. And all of the WAP value is under the AP value from the top1 to top10 result. Because not all of the recommended brands are perfect result, some of them are only relevant.

Table 2 shows the execution time and average AP of top 5 results of Argo, unweighted scale and scalable approach. We can see that execution time of scalable approach is lower than Argo, and average performance is higher than Argo.

Table 2. Execution time and average performance of Argo and scalable approach

Comparison	Time(ms)	AP
Argo	225.128	0.402
Unweighted Scale	17.683	0.517
Scalable	17.128	0.587

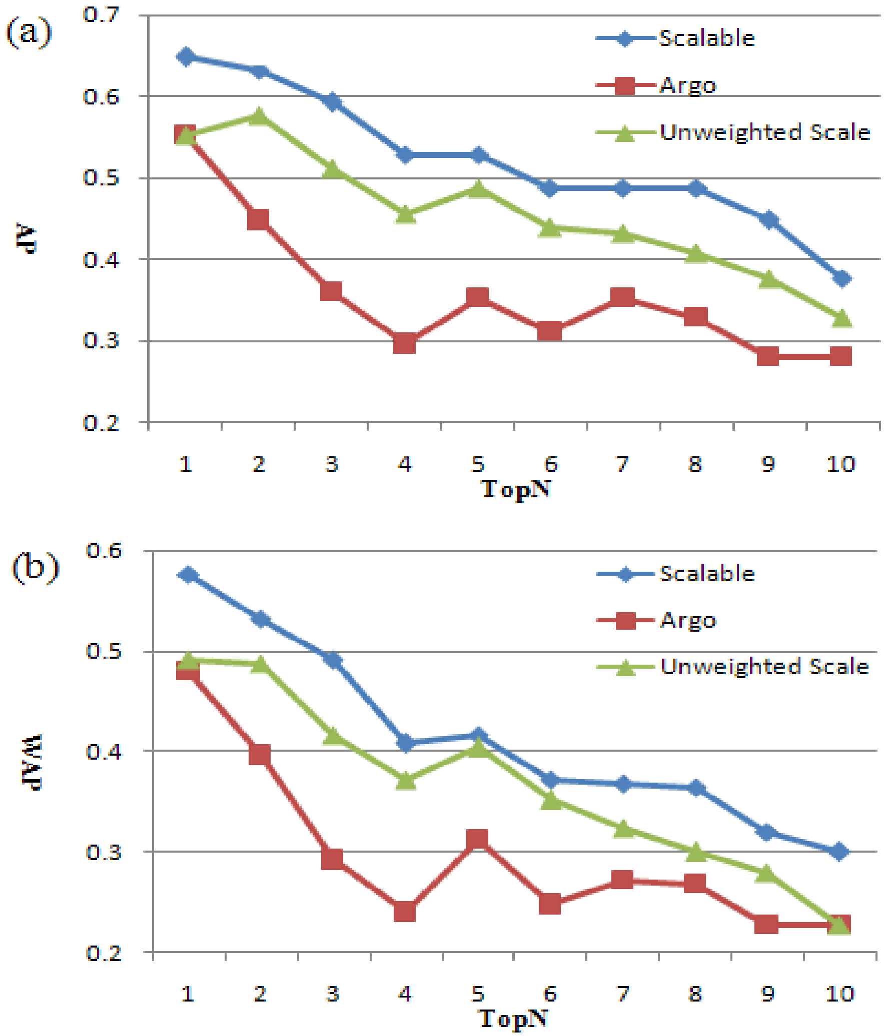

Fig. 4. Performance of scalable brand ranking with weighted and unweighted categories and Argo

3.3 Example Cases

Figure 5 illustrates a few examples of the system output. Each row corresponds to a Flickr user and recommended brands. In the first row, top 6 results are high relevant to Sony products. In the second and third row, only the top 1 or 2 results are relevant to user interest because it's the only relevant brands in brand datasets. In the fourth and fifth row, although users didn't label their favorite brand, our approach still recommends high relevant brands to their interest.

We consider that some Flickr users' photos are mainly products, and it is unreasonable to suggest relevant products to them. Thus, we assured that the 125 tested users' photos could reflect their interest rather than commercial intentions.

Fig. 5. Examples of flickr users and recommended brands

4 Conclusions

We expand brand recommendation for social network users by this work. Firstly, we built the two layer brand tree from brands datasets to match relevant brand efficiently. Further, we recommend brands layer by layer to filter irrelevant brand nodes. And, we noticed the importance of the connection between the two layers by scalable brand mining. At last, real Flickr users' experiments proved the efficiency of our approach.

At present, our approach only recommends the relevant brands from user's submission but not takes the advertiser into consideration. Our approach can be expanded in following aspects: 1) Competitive bid from advertiser when we recommend brands. 2) Recommend products based on user's favorite brand.

Acknowledgments. This work is supported partly by NSFC No. 60903121, No.61173109, Microsoft Research Asia.

References

[1] Flickr, `http://www.flickr.com/`
[2] ODP, `http://www.dmoz.org/`
[3] Mei, T., Hua, X.S., Li, S.: Contextual in-image advertising. In: ACM Multimedia, pp. 439–448 (2008)
[4] Mei, T., Hua, X.-S.: Contextual Internet Multimedia Advertising. In: Proceedings of the IEEE, pp. 1416–1433 (2010)
[5] Wang, X.-J., Yu, M., Zhang, L., Ma, W.-Y.: Advertising based on users' photos. In: ICME 2009, pp. 1640–1643 (2009)
[6] Wang, X.-J., Yu, M., Zhang, L., Cai, R., Ma, W.-Y.: Argo: Intelligent Advertising by Mining a User's Interest from His Photo Collections. In: ACM Data Mining and Audience Intelligence for Advertising, pp. 18–26 (2009)
[7] Wang, X.-J., Zhang, L., Li, X.-R., Ma, W.-Y.: Annotating Images by Mining Image Search Results. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1919–1932 (2008)
[8] Qian, X., Liu, X., Zheng, C., Du, Y., Hou, X.: Tagging photos using users' vocabularies. Neurocomputing 111, 144–153 (2013)
[9] Feng, H., Qian, X.: Recommendation via user's personality and social contextual. In: ACM CIKM 2013 (2013)
[10] Li, Q., Gu, Y., Qian, X.: LCMKL: Latent-community and multi-kernel learning based image annotation. In: ACM CIKM 2013 (2013)