Social Image Tag Recommendation by Concept Matching*

Aixin Sun
axsun@ntu.edu.sg

Sourav S. Bhowmick
assourav@ntu.edu.sg

Jun-An Chong
chon0113@e.ntu.edu.sg

School of Computer Engineering, Nanyang Technological University, Singapore 639798

ABSTRACT

Tags associated with social images are valuable information source for superior image search and retrieval experiences. In this paper, we propose a novel tag recommendation technique that exploits the user-given tags associated with images. Each *candidate tag* to be recommended is described by a few *tag concepts* derived from the collective knowledge embedded in the *tag co-occurrence* pairs. Each concept, represented by a few tags with high co-occurrences among themselves, is indexed as a textual document. Then user-given tags of an image is represented as a text query and the matching concepts are retrieved from the index. The candidate tags associated with the matching concepts are then recommended. Leverages on the well studied Information Retrieval (IR) techniques, the proposed approach leads to superior tag recommendation accuracy and lower execution time compared to the state-of-the-art.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Information filtering*

General Terms

Algorithms, Experimentation

Keywords

Flickr, social image, tag recommendation, tag concept

1. INTRODUCTION

The availability of user-given tags enables novel and superior tag-based techniques for social image retrieval. In this paper, we propose a novel approach that exploits the notion of *tag concepts* (or concepts for brevity) to recommend tags.

First, a set of *candidate tags* is selected from all tags in the social image collection. For each candidate tag t_c, a *tag relationship graph* (TRG) is constructed from the *tag co-occurrence* pairs. Each node in a TRG is a tag and an edge between two nodes indicates that the two tags have high co-occurrence measured by an *association measure* (e.g., Jaccard coefficient). Figure 1 depicts an example of TRG for the candidate tag rock. Next, a set of *concepts* associated with t_c through the collective knowledge embedded in the tag co-occurrence pairs is detected from the TRG using Modularity Clustering [7]. For example, two different concepts (*rock stone* and *rock music*) are evident in Figure 1 on the left and right hand sides of the `rock` node, respectively. Each concept is represented by a set of high co-occurrence tags. Lastly, we leverage on well-studied information retrieval techniques to recommend tags from the detected concepts. Specifically, the detected concepts are first indexed as textual documents, where a tag in the concept is treated as a word (or an index term). Then, the user-given tags of an image is represented as a text query and the matching concepts are retrieved from the index by exploiting cosine similarity. The candidate tags associated with the matching concepts are then recommended.

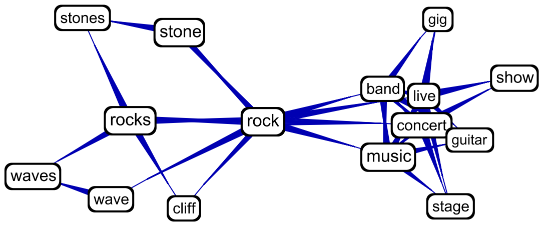


Figure 1: Tag relationship graph of rock.

We have evaluated the impact of various issues on our proposed tag recommendation technique using NUS-WIDE dataset [1]. Specifically, we evaluated more than 190K test cases involving over 99K distinct images. To the best of our knowledge, this is the largest experiments conducted for social image tag recommendation with ground-truth labeling. Our experimental results suggest that the proposed concept-based strategy leads to superior tag recommendation accuracy as well as lower execution time compared to a state-of-the-art tag recommendation technique [8].

2. RELATED WORK

The efforts in social image tag recommendation can be broadly categorized into three types, namely, *model-based*, *example-based*, and *knowledge-based* approaches [2, 3, 4, 8, 9].

The *model-based* approach typically exploits the visual content component of images for tag recommendation. Methods based on this approach learn models from labeled examples of a predefined set of visual concepts. The learned models are then used to annotate new images according to their relevance to the concepts [2]. The *example-based* approach assumes that visually similar images are annotated by a similar set of tags. For a given image, tags are recommended among those associated with its nearest neighbors based on visual content similarity [3, 4, 9]. Nevertheless, computing nearest neighbors from a large image collection is often expensive. On the other hand, both approaches are particularly useful

*Area chair: Kiyoharu Aizawa.
This work is supported by Singapore MOE AcRF Tier-1 Grant RG13/10.

MM'11, November 28–December 1, 2011, Scottsdale, Arizona, USA.

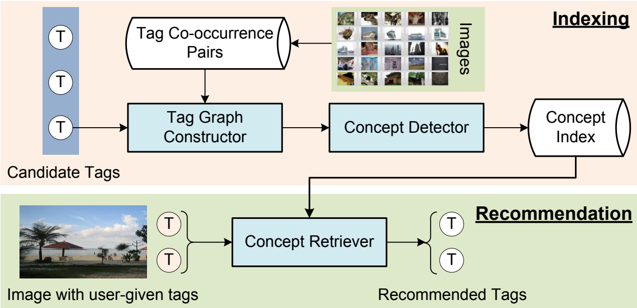


Figure 2: Architecture of the tag recommendation system.

Algorithm 1: Concept detection for candidate tag t_c

Input: TRG G_c of candidate tag t_c
Output: A set of concepts C_c for t_c

```
1 begin
2     obtain first hop tags H_c = firstHop(G_c, t_c);
3     remove t_c from G_c and get G'_c = removeTag(G_c, t_c);
4     detect concepts C_c = detectConcept(G'_c);
5     foreach concept c_p ∈ C_c do
6         foreach tag t ∈ c_p do
7             if t ∉ H_c then
8                 remove tag t from c_p
```

when user-given tags are unavailable. The *knowledge-based* approach does not consider the visual content of images. Instead, it relies on the relationships (e.g., co-occurrence) among tags [8]. For instance, for each user-given tag, the method in [8] first selects its top-m co-occurring tags as candidate tags. Then, the candidate tags are re-ranked for recommendation based on strength of *co-occurrence*, *stability*, and *descriptiveness*.

In this paper, our proposed tag recommendation framework is also based on tag co-occurrence. In contrast to [8], we exploit the collective knowledge of *concepts* embedded in tag co-occurrence pairs to recommend tags. Also, given an association measure, our approach needs the user to define a single parameter (*hop threshold*) to facilitate tag recommendation. However, in [8] four parameters need to be tuned, which may not be practical for end users.

3. TAG RECOMMENDATION

The procedure of our *concepts-based* tag recommendation consists of three phases, namely the *tag relationship graph construction* phase, the *concept detection* phase, and the *tag recommendation* phase. Figure 2 depicts the architecture of our proposed tag recommendation system that realizes these phases.

3.1 Tag Relationship Graph (TRG)

First, we select a set of *candidate* tags from all tags in the social image collection. The selection criteria is application-dependent. For each selected candidate tag t_c, a *tag relationship graph* (TRG) is constructed from the *tag co-occurrence* pairs involving t_c and the pairs involving t_c's co-occurring tags. A *tag co-occurrence pair* of tags t_a and t_b is a 5-tuple $\langle t_a, t_b, f(t_a), f(t_b), f(t_a, t_b)\rangle$ where $f(t_a)$ (resp. $f(t_b)$) and $f(t_a, t_b)$ denote the numbers of images annotated by t_a (resp. t_b) and by both t_a and t_b, respectively. Given a collection of socially tagged images, the set of tag co-occurrence pairs forms the collective knowledge base. The TRG of t_c therefore captures the collective knowledge about t_c.

The construction of TRG requires two user-defined parameters, namely, *association measure* (denoted by α) and *hop threshold* (denoted by φ). The *association measure* specifies how to compute tag co-occurrence strength. Equation 1 gives three example association measures, namely, Jaccard coefficient, co-occurrence probability (*CoProb*), and Interest. The *hop threshold* φ specifies the number of co-occurring tags we need to consider for the TRG construction.

$$assoc(t_a, t_b) : \begin{cases} Jaccard(t_a, t_b) = \frac{f(t_a, t_b)}{f(t_a)+f(t_b)-f(t_a, t_b)} \\ CoProb(t_b|t_a) = \frac{f(t_a, t_b)}{f(t_a)} \\ Interest(t_b|t_a) = CoProb(t_b|t_a) - \frac{f(t_b)}{N} \end{cases} \quad (1)$$

In Equation 1, $assoc(t_a, t_b)$ denotes the association between two tags t_a and t_b, computed by one of the association measures on the right hand side, and N denotes the number of images in the collection. We now elaborate on the construction process.

Given a candidate tag t_c, α, and φ, the TRG construction is initiated by creating t_c as the *central* node. Then the top-φ most co-occurring tags with t_c are included as the *first-hop* tags of t_c. Next, from the top-φ most co-occurring tags of each first-hop tag we select those tags that are associated with at least two first-hop tags and add them in the TRG as *second-hop* tags of t_c. Note that the association strength of the second-hop tags with the first-hop ones must be above the median of the association strength derived from all tag co-occurrence pairs. This constraint allows us to avoid adding too many tags that are not strongly associated. For example, Figure 1 depicts the TRG of candidate tag rock when $\alpha = Jaccard$ and $\varphi = 8$. Observe that rock is associated with 8 first-hop tags. Among the second-hop tags, stones is included in the graph as it is one of the top-8 most co-occurring tags of stone as well as of rocks (stones is associated with two first-hop tags rocks and stone).

3.2 Concept Detection

Concept detection is based on the intuition that tags often co-occur with other tags within the same concept but not with tags in a different concept. Hence, concept detection can be naturally formulated into a community detection or graph-cut problem. However, these techniques cannot be directly adopted as the central node (*i.e.,* candidate tag t_c) has strong associations with all the first-hop tags which often represent multiple concepts. The procedure to detect concepts is outlined in Algorithm 1. Observe that the removal of the central node from the TRG (line 3) is a key step for concept detection. This step ensures that associations between t_c and the first-hop tags do not affect the detection of concepts. For instance, the removal of tag rock leads to two disconnected components in Figure 1 and each component naturally forms a concept. The concept detection is then achieved by using a community detection or graph cut algorithm (line 4). In our implementation, we adopted Modularity Clustering, which has shown effectiveness in many community detection tasks and does not require to specify as input the number of clusters to be detected [7]. From the detected concepts (or subgraphs), the second-hop tags are removed as only the first-hop tags are strongly associated with t_c and hence can be used for recommendation (lines 5-8).

A concept is a 3-tuple $\langle t_c, assoc(t_c, c_p), c_p\rangle$ where $assoc(t_c, c_p)$ is the *association strength* between the candidate tag t_c and the concept c_p detected from its TRG (see Equation 2), and c_p is represented by a set of first-hop tags.

$$assoc(t_c, c_p) \quad = \quad \sum_{t \in c_p} assoc(t, t_c) \quad (2)$$

In the above equation $assoc(t, t_c)$ is computed based on one of the association measures defined in Equation 1. Table 1 illustrates representation of concepts. The first two records are the concepts for candidate tag rock whose TRG is depicted in Figure 1. The remain-

Table 1: Example concepts for candidate tags rock **and** sunset

⟨rock,	0.265,	{wave, cliff, rocks, stone}⟩
⟨rock,	0.247,	{concert, band, music, live}⟩
⟨sunset,	0.483,	{clouds, sun, sky, silhouette}⟩
⟨sunset,	0.341,	{beach, sea, ocean, water}⟩

ing two records represent the concepts for candidate tag sunset (the TRG is obtained using α = *Jaccard* and $\varphi = 8$). Due to space constraints, we do not display the TRG of sunset. The two concepts of sunset show two different aspects of the sunset scenes, *landscape* and *seascape*.

We index a concept as a textual document. Specifically, the set of tags in c_p is indexed as the content of the document in an inverted index where each tag is a word (or term) in the index [6]. Both the candidate tag t_c and the association strength $assoc(t_c, c_p)$ are indexed as attributes of the document (similar to *author*, *publication date* attributes of any textual document).

3.3 Tag Recommendation

The tag recommendation is modeled as a textual search problem and consists of two steps. First, it computes the *recommendation score* of each candidate tag with respect to the user-given tags of an image. Then, it sorts the candidate tags in descending order of their scores and recommends top-k tags to the user. As the second step is trivial, we elaborate on the first step further.

Let T_u be the set of user-given tags for a social image whose tags are to be recommended. All tags in T_u form a textual query, where each tag $t_u \in T_u$ is a term in the query and all terms are equally weighted. The *recommendation score* $score(t_c)$ of a candidate tag t_c is computed as follows.

$$score(t_c) \quad = \quad \sum_{c_p \in C} match(T_u, c_p) \cdot assoc(t_c, c_p) \qquad (3)$$

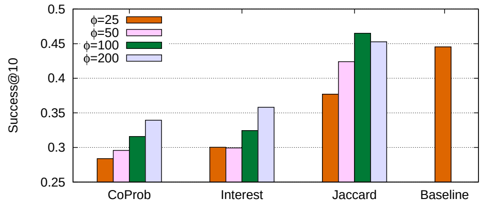
In this equation, $match(T_u, c_p)$ is the *match score* between a concept of t_c and T_u, computed using widely adopted cosine similarity between the textual query and the document content. We use Lucene as the indexer and its default similarity setting.

4. EXPERIMENTS

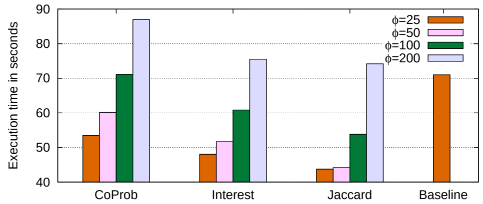
Evaluation of social image tag recommendation is a challenging task partially due to the lack of a benchmark dataset. Existing studies have largely engaged in user study for evaluation. However, this impose practical limits to number of test cases. Recommending tags that are syntactically removed from existing user-tagged images may not truly reflect the accuracy of a tag recommendation method. This is because these syntactically removed tags may not be strongly related to the image due to the noisy nature of user-assigned tags. In our evaluation, we use the NUS-WIDE dataset, which is the largest publicly available human-annotated dataset [1].

4.1 Dataset

In the NUS-WIDE dataset, ground-truth images of 81 categories were manually labeled. These categories were carefully selected such that they (i) correspond to some frequent tags in Flickr, (ii) cover both general concepts like "animal" and specific ones like "dog" and "flowers", and (iii) belong to different genres including scene, object, event, program, people and graphics [1]. All the three aforementioned criteria and the ground-truth labeling make NUS-WIDE a fairly good dataset for tag recommendation evaluation. Importantly, although many images in the dataset have been manually labeled to one or more categories, they may not be tagged by the corresponding category labels. For instance, an image is manu-



(a) Accuracy using Success@10 for all methods.



(b) Execution time of all methods

Figure 3: Time and accuracy (Success@10).

ally labeled under tiger category because it contains a tiger object. However, it may not contain tiger as a tag due to incompleteness of user tagging. Fortunately, this creates a good *test case* for our tag recommendation method as we just need to evaluate whether the method can effectively recommend tag tiger to the image.

In our experiments, we used the 81 category labels as candidate tags and their corresponding ground-truth for evaluation. To generate the tag co-occurrence pairs, we select 5981 frequent tags, each of which has been used to tag at least 100 images, among the 420K distinct tags appeared in the dataset. A tag co-occurrence pair is recorded if two tags co-occurred for at least 10 times, resulting in more than 527K pairs[1]. The infrequent tags are not considered in the knowledge base (*i.e.,* tag co-occurrence pairs).

For each of the 81 categories, all images that belong to the category but are not tagged by the category label are selected as test cases. Among them, we further select images containing *at most* 10 frequent tags[2] for recommendation leading to 190,288 test cases. These test cases involve 99,106 distinct images as an image may belong to more than one category.

4.2 Methods

We evaluate three association measures, namely, Jaccard, CoProb, and Interest (defined in Equation 1) for TRG construction. For each association measure, we use $\varphi = \{25, 50, 100, 200\}$. Hence, 12 methods are evaluated based on the proposed tag recommendation framework. In the sequel, for a given φ we denote a method using association measure α as $\alpha\varphi$ (*e.g., Jaccard100*).

We compare our proposed technique with the state-of-the-art tag recommendation algorithm for Flickr reported in [8]. In the sequel, we refer to this algorithm as the *baseline* approach. For fair comparison, the tags recommended by the baseline method that are not among the 81 category labels (*i.e.,* candidate tags in our methods) are excluded as there is no ground-truth labeling for these tags.

4.3 Results

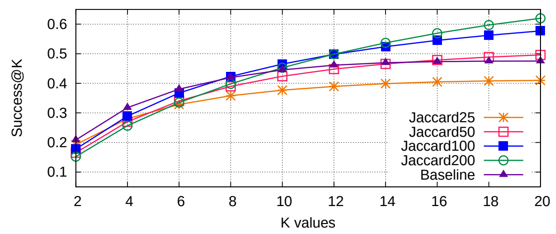
S@10. We evaluate the recommendation accuracy of our approach using *Success@K* (or simply *S@K*), which is the probability of finding a labeled category among the top K recommended tags.

[1]Tag pairs $\langle t_a, t_b\rangle$ and $\langle t_b, t_a\rangle$ are considered the same as co-occurrence is unidirectional.

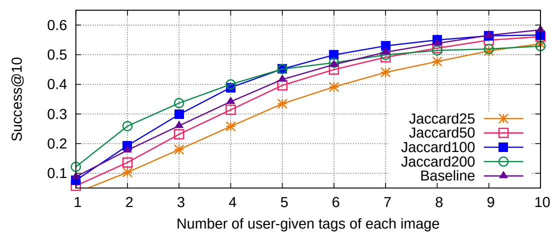
[2]The infrequent tags have no impact on tag recommendation as they are not included in the knowledge base.

Table 2: Number of concepts indexed.

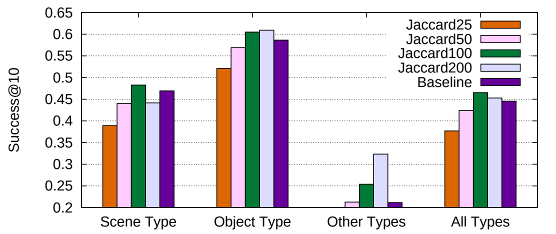
Assoc measure α	$\varphi = 25$	$\varphi = 50$	$\varphi = 100$	$\varphi = 200$
CoProb	309	356	418	506
Interest	353	438	520	600
Jaccard	387	503	596	709



(a) Effect of K on Success@K.



(b) S@10 vs number of user-given tags.



(c) Effect of candidate tags types on S@10.

Figure 4: Tag recommendation performance comparison.

Note that it is also used in [8]. Figure 3(a) plots the $S@10$ values of the baseline and our proposed method based on the three association measures. We can make the following observations. Firstly, Jaccard-based method outperforms its counterparts based on CoProb or Interest by a large margin. In other words, symmetric association measure performs better than asymmetric association measure in tag recommendation, consistent with that reported in [8]. Secondly, *Jaccard100* and *Jaccard200* both outperform the baseline method. Particularly, *Jaccard100* is the winner among all methods.

Execution time. Figure 3(b) reports the execution times for processing 190,288 test cases for all methods executed on the same computer. Observe that CoProb-based methods are the slowest and methods based on Jaccard are the most efficient ones among the 12 proposed methods. Larger φ leads to larger TRG and consequently more concepts need to be indexed (Table 2). Obviously, potentially more concepts are then matched for a given image, leading to a longer execution time. Observe that *Jaccard100* not only achieves better tag recommendation accuracy ($S@10$) compared to the baseline method, its execution time is also 24% lower than the latter. In the following experiments we shall compare the 4 methods with Jaccard association measure (*Jaccard25*, *Jaccard50*, *Jaccard100*, and *Jaccard200*) and the baseline.

Effect of K in $S@K$. Figure 4(a) reports the $S@K$ values for K ranging from 2 to 20. The baseline method achieves the best $S@K$ for $K \leq 8$ and then reaches a plateau for $K \geq 10$. Both *Jaccard100* and *Jaccard200* achieve slightly better $S@10$ and then significant better $S@K$ values for $K > 10$ compared to the baseline. Note that for a tag recommendation system, a larger K indicates that its users might be overloaded with too many recommended tags. Therefore, in the sequel we discuss our results for $K = 10$, which is a reasonable number for tag recommendation and has been used in several recent works such as [5].

Number of user-given tags. Figure 4(b) reports the effect of number of user-given tags an image already has on $S@10$. Clearly, more user-given tags means that a tag recommendation method can leverage on more information about the image for better recommendation, leading to higher $S@10$ values. Observe that *Jaccard100* outperforms the baseline when the number user-given tags is between 2 and 8. *Jaccard200* is the best performing method when the number of tags is fewer than 5. However, it did not benefit as much as other methods when more user-given tags are available.

Tag types. The 81 candidate tags include 33 tags apiece in *scene* and *object* categories. These two major categories cover 63% and 19% of the test cases, respectively. Figure 4(c) reports the values of $S@10$ for these categories of objects. Observe that tags in *object* category have highest $S@10$ values for all methods.

5. CONCLUSIONS

In this paper, we propose a novel knowledge-based approach for image tag recommendation that exploits tag concepts, which are derived based on the collective knowledge embedded in tag co-occurrence pairs. Our proposed tag recommendation framework is generic in nature. In particular, the candidate tags maybe selected based on various criteria most important to the target application; the TRG may be constructed based on different association measures and hop threshold values; the tag concepts can be detected by alternative graph-cut algorithms; and the matching score between the user-given tags and the concepts are based on well-studied techniques in Information Retrieval. This enables not only customized matching score computation (*e.g.,* query term weighting and similarity definitions between the user-given tags and the matching concepts), but also boost efficiency and scalability of the tag recommendation process. It is therefore part of our future work to investigate techniques along these dimensions to further improve tag recommendation accuracy.

6. REFERENCES

[1] T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y.-T. Zheng. NUS-WIDE: A real-world web image database from National University of Singapore. In *Proc. CIVR*, 2009.

[2] J. Li and J. Z. Wang. Real-time computerized annotation of pictures. *IEEE TPAMI*, 30(6):985–1002, 2008.

[3] X. Li, C. G. M. Snoek, and M. Worring. Learning social tag relevance by neighbor voting. *IEEE TMM*, 11(7):1310–1322, 2009.

[4] D. Liu, X.-S. Hua, L. Yang, M. Wang, and H.-J. Zhang. Tag ranking. In *Proc. WWW*, 2009.

[5] Y.-T. Lu, S.-I. Yu, T.-C. Chang, and J. Y.-J. Hsu. A content-based method to enhance tag recommendation. In *Proc. IJCAI*, 2009.

[6] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[7] M. E. J. Newman. Modularity and community structure in networks. *Proc. of Nat. Aca. of Sci. of USA*, 103:8577–8582, 2006.

[8] B. Sigurbjörnsson and R. van Zwol. Flickr tag recommendation based on collective knowledge. In *Proc. WWW*, 2008.

[9] G. Zhu, S. Yan, and Y. Ma. Image tag refinement towards low-rank, content-tag prior and error sparsity. In *Proc. MM*, 2010.