Annotating Web Image Using Parallel Graph Bipartition and Word Clustering

Zheng Liu
School of Computer Science and Technology, Shandong Economic University, Ji'nan Shandong 250014, China
Email: Lzh_48@126.com

***Abstract*—A novel web image annotation method by candidate annotations clustering and parallel graph bipartition is proposed in this paper. Firstly, surrounding texts and other textual information in the hosting pages are extracted as the candidate annotations. For Web images, the candidate annotation sets of which are usually fairly large. Therefore, we cluster candidate annotations to reduce computation complexity. Next, centroids of clustering results and the distance between them are used to construct a graph. Then a parallel 0.87856 heuristics MAX-CUT algorithm is applied to partition the graph. Finally, one part of the graph partition results is selected as final annotation results. Experimental results show that our method works more effectively than existing methods.**

***Index Terms*—Web Image annotation, graph bipartition, K-means, word clustering**

I. Introduction

With the popularization of Web and image devices, there are more and more digital images available on the Internet. How to effectively organize and manage these web images becomes a critical issue. Well-known commercial systems including Google, MSN and Yahoo! rely on surrounding descriptions of images embedded in the web pages for the image retrieval. Images without clear context descriptions will either be returned as false positives or be totally discarded during the retrieval. However, manual annotation is an expensive and tedious procedure. Image auto-annotation techniques provide an attainable way to associate the "visuality" of the images with their semantics, which can be used to search unlabeled image collections, and return more relevant images to the users[1]. Thus, Automatic annotation of Web image has great importance in improving the performance of web image retrieval. Annotation can facilitate image search through the use of text.

This paper addresses a novel graph-based method to automatically obtain Web image annotations. The rest of the paper is organized as follows. Section 2 introduces the related work on web image annotation, and image annotation refinement. Section 3 presents an overview of our Web image annotation framework. A parallel graph-based heuristics algorithm for Web image annotation is described in section 4 and 5. In section 6, an example is given to explain the algorithm proposed in this paper. Section 7 presents the experimental results to demonstrate the performance of our method. Section 8 concludes this paper and points out our future work.

Manuscript received January 1, 2009; revised June 1, 2009; accepted July 1, 2009.
corresponding author: Zheng Liu.
Email: Lzh_48@126.com

II. Related Work

In recent years, several Web image annotation methods have been proposed, which use the surrounding texts in Web pages for image annotating.

Rui et al.[12] proposed a bipartite graph reinforcement model for web image annotation that uses common sources of information like filemane, ALT text, URL and surrounding text as initial annotation. To achive a better extended annotation a search engine with access to about 2.4 million manually annotated images is queried with the initial textual and visual terms. The resulting annotation is a combination of the retrieved results.

Wang et al.[13] proposed a web image annotation method AnnoSearch using search and data mining techniques. However, in their framework, at least one accurate keyword is required in advance.

Hua et al. [14] proposed a system which can automatically acquire semantic knowledge such as description, people, temporal and geographic information for web images. Nevertheless, they did not explicitly exploit the visual similarity to label new images.

Diogenes[15] is a person photograph search engine which mainly tries to extract people names from the text surrounding the images and accociate it with the faces on images detected by a face recognition module.

There have been many pioneering works on image annotation refinement which select related annotations from candidate annotation set as the final annotations.

Jin et al. have developed a method using a generic knowledge-based WordNet[2]. From the small candidate annotation set obtained by an annotation method, the irrelevant annotations are pruned using WordNet without image content analyzing.

In [3], an algorithm using random walk with restarts was proposed to re-rank the candidate annotations. However, it was still implicitly based on the assumption

doi:10.4304/jcp.5.8.1185-1192

majority should win and the refinement process was still independent of the original query image.

Wang et al. [4] proposed a novel algorithm which formulates the annotation refinement process as a Markov process and defines the candidate annotations as the states of a Markov chain.

Recently, Jin et al. proposed a graph-based image annotations refinement method[5] to prune noisy annotations by graph partition.

III. Overview of our Web image annotation method

The testing images are crawled from web pages. Each image is initially annotated by candidate keywords, which are extracted from the surrounding texts and the tag information by the VIPS algorithm [16] and standard text processing techniques.

The surrounding texts, which have been successfully used by commercial image search engines such as Google, and Yahoo, can be treated as approximate annotations of web images. However, these annotations are very noisy, as many irrelative words in the hosting web pages, such as advertisement words. Therefore, it is necessary to refine the candidate annotations and then obtain more precise annotations. This paper formulates the problem of removing erroneous keywords as weighted MAX-CUT problem.

A. Relationship between Web image annotation problem and weighted Max-Cut problem

It is well known that MAX-CUT problem was proved to be NP-complete by Karp [10], Max-cut problem is a classical combinatorial optimization problem that has a wide range of applications in different domains. We solve MAX-CUT problem by a parallel algorithm with approximation ration of 0.87856 proposed by Goemans and Williamson in 1995[11].

Let G be an undirected graph with nodes $N=\{1,...,n\}$, and edge set E. Let $W_{ij}=W_{ji}$ be the weight on edge $E(i,j)$, for $E(i,j)\in E$. The MAX-CUT problem is to determine a subset S of the nodes N for which the sum of the weights of the edges that cross from S to its complement $\overline{S}$ is maximized, where $N=S\bigcup\overline{S}$.

We can formulate MAX-CUT as an integer quadratic programming (IQP) as follows. Let $x_j=1$ for $j\in S$ and $x_j=-1$ for $j\in\overline{S}$. Then our formulation is:

$$\text{MAX CUT: Maximize}_x\ \frac{1}{4}\sum_{i=1}^{n}\sum_{j=1}^{n}w_{ij}(1-x_ix_j) \quad (1)$$

$$\text{s.t. } x_j\in\{-1,1\},\ \ j=1,...,n.$$

Now let $Y=xx^T$, where $Y_{ij}=x_ix_j$ $i=1,...,n$, $j=1,...,n$.

B. System Framework

In our Web image annotation process, each centroid of candidate annotation cluster is considered as a vertex of a graph G. All vertices of G are connected with proper weights.

We transform each cluster centroid to a vertex of the graph and regard the distance between two centroids as edge weight. Thus, we can reduce an instance of Web image annotation problem into an instance of weighted MAX-CUT problem in polynomial time. Therefore, it is possible to solve weighted MAX-CUT problem for getting the solution of Web image annotation problem. The framework of our Web image annotation approach is shown in Fig.1.

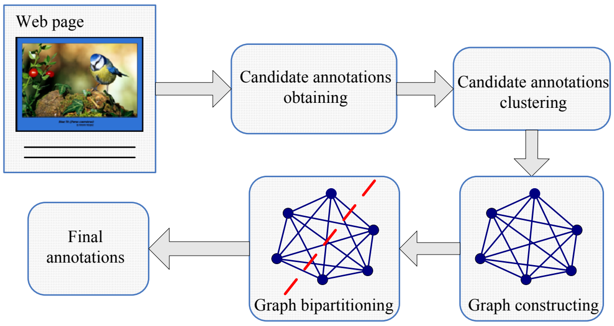

Figure 1. Framework of our Web image annotation system.

IV. Graph establishing

To reduce computation cost, we cluster candidate annotations in advance, and then construct graph.

A. Candidate Annotation Clustering

Following [17], we represent each word by a feature vector. Each feature corresponds to a context in which the word occurs. The feature vector of word w is denoted as V^w.

The value of the feature is the pointwise mutual information between the feature and the word. Let c be a context and f_c^w be the frequency count of a word w occurring in context c. The pointwise mutual information between c and w is defined as:

$$\widehat{V_c^w}=\frac{\frac{f_c^w}{N}}{\frac{\sum_{m=1}^{|C|}f_{C_m}^w}{N}\times\frac{\sum_{n=1}^{|W|}f_c^{W_n}}{N}} \quad (2)$$

$$N=\sum_{m=1}^{|C|}\sum_{n=1}^{|W|}f_{C_m}^{W_n} \quad (3)$$

where C is the set of all contexts and W is the set of all words.

A well-known problem with mutual information is that it is biased towards infrequent words/features. We therefore multiplied $\widehat{V_c^w}$ with a discounting factor:

$$V_c^w = \frac{f_c^w}{f_c^w + 1} \times \frac{\min(\sum_{m=1}^{|C|} f_{C_m}^w, \sum_{n=1}^{|W|} f_c^{W_n})}{\min(\sum_{m=1}^{|C|} f_{C_m}^w, \sum_{n=1}^{|W|} f_c^{W_n}) + 1} \times \widehat{V_c^w} \quad (4)$$

We compute the similarity between two words w_i and w_j using the cosine coefficient of their mutual information vectors:

$$Sim(w_i, w_j) = \frac{\sum_{m=1}^{|C|} (V_{C_m}^{w_i} \times V_{C_m}^{w_j})}{\sqrt{(\sum_{m=1}^{|C|} V_{C_m}^{w_i})^2 \times (\sum_{m=1}^{|C|} V_{C_m}^{w_j})^2}} \quad (5)$$

Given a set of candidate annotation $\{A_1, A_2, \cdots\cdots, A_t\}$, every candidate annotation can be represented by a $|C|$-dimensional feature vector, and we use K-means clustering algorithm to partition t candidate annotations into k sets($k < t$) $S = \{S_1, S_2, \cdots\cdots, S_k\}$ so as to minimize the within-cluster sum of squares:

$$\arg\min_S \sum_{p=1}^{k} \sum_{V^q \in S_p} \left\| V^q - \mu_p \right\|^2 \quad (6)$$

where μ_p is the mean of S_p.

To obtain the final cluster results, K-means performs an iterative refinement technique. In statistics and machine learning, K-means clustering is a method of cluster analysis which aims to partition *n* observations into *k* clusters in which each observation belongs to the cluster with the nearest mean. K-means clustering is often used on large data sets since its complexity is linear in *n*, the number of elements to be clustered. K-means is a family of partitional clustering algorithms that iteratively assigns each element to one of *k* clusters according to the centroid closest to it and recomputes the centroid of each cluster as the average of the cluster. As the initial centroids are randomly selected, the resulting clusters vary in quality. Some sets of initial centroids lead to poor convergence rates or poor cluster quality.

Given an initial set of *k* means $\{m_1^{(1)}, m_2^{(1)}, \cdots\cdots, m_k^{(1)}\}$, which may be specified randomly or by some heuristic, the algorithm proceeds by alternating between two steps:
Assignment step: Assign each observation to the cluster with the closest mean.

$$S_p^{(l)} = \{V_q : \left\| V_q - m_p^{(l)} \right\| \le \left\| V_q - m_{p^*}^{(l)} \right\|, p^* = 1, 2, \cdots\cdots, k\} \quad (7)$$

Update step: Calculate the new means to be the centroid of the observations in the cluster.

$$m_p^{(l+1)} = \frac{1}{\left| S_p^{(l)} \right|} \sum_{V_q \in S_p^{(l)}} V_q \quad (8)$$

When this process converged, we can obtain the centroid vector *M.* The centroid of a cluster is constructed by averaging the feature vectors of a subset of the cluster members.

$$M = \{m_1^*, m_2^*, \cdots\cdots, m_k^*\} \quad (9)$$

The algorithm is deemed to have converged when the assignments no longer change, and obtain the final centroid set S.

B. Edge Weight Computing

After the candidate annotations are clustered into *k* sets, we use the clustering results to construct a graph. The cluster centroid is used to be node of the graph, and distance between two centroids is denoted as edge weight. In this paper, we use cosine similarity to measure the distance between two centroids.

$$Sim(m_i^*, m_j^*) = \frac{m_i^* \cdot m_j^*}{\left\| m_i^* \right\| \left\| m_j^* \right\|} \quad (10)$$

V. Web Image Annotation Using a Parallel MAX-CUT Algorithm

In this section, we will present the proposed algorithm in detail.

A. Parallel algorithm for MAX-CUT problem

As our work concentrates on Web image annotation, the size of candidate annotation set is usually very large. Therefore, our work is based on a parallel version of Goeman's randomized 0.87856 approximation scheme for finding the maximum-cut in a graph[11].

Let *W* be the matrix whose $(i, j)^{th}$ element is W_{ij} for $i = 1, ..., n$ and $j = 1, ..., n$. Then MAX CUT can be equivalently formulated as:

$$\text{MAX CUT: } \text{Maximize}_{Y,x} \frac{1}{4} \sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij} - W \bullet Y \quad (11)$$

s.t. $x_j \in \{-1, 1\}$, $j = 1, ..., n.$ $Y = xx^T$.

The matrix $Y = xx^T$ is a symmetric rank-1 positive semidefinite matrix. We can interpret IQP as restricting x_i to be a l-dimensional vector of unit norm. Relaxations can be defined by allowing x_i to be a multidimensional vector v_i of unit Euclidean norm. Since the linear space spanned by the vectors v_i has dimension at most n, we can assume that these vectors belong to $\mathbb{R}^n$, or more precisely to the n-dimensional unit sphere SP_n. We replace $(1 - x_i x_j)$ by $(1 - v_i \cdot v_j)$, where $v_i \cdot v_j$ represents the inner product(or dot product) of v_i and v_j. The resulting relaxation is a semidefinite program, which can be denoted as follows:
RELAXED MAX CUT:

$$\text{Maximize } \frac{1}{4} \sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij} (1 - v_i \cdot v_j) \quad (12)$$

s.t. $v_i \in SP_n$ $i = 1, ..., n.$

The first step of their algorithm is to find a vector configuration which maximizes

$$Z_v = \frac{1}{2}\sum_{i<j} w_{ij}(1 - v_i \cdot v_j) \quad (13)$$

Such a configuration can be found in polynomial time using a positive semidefinite programming algorithm and incomplete Cholesky decomposition. The problem has to be reformulated as an unconstrained optimization problem if the methods just stated are to be applicable. The objective function is designed as follows.

$$h(v_1, \cdots\cdots, v_n) = \sum_{i<j} w_{ij} \frac{v_i \cdot v_j}{\|v_i\|\|v_j\|} \quad (14)$$

Let v_i^k denote the *k*-th component of the *i*-th vector and let $d_i^k = \partial h / \partial v_i^k$. Then we have

$$d_i^k = v_i^k (v_i \cdot C_i) - C_{ik} \quad (15)$$

assuming that all v_i have norm one, where C is the matrix formed by the column vectors.

$$C_i = \sum_{j \in adj(i)} w_{ij} v_j \quad (16)$$

where $adj(i) = \{j \in [n], w_{ij} \neq 0\}$ and C_{ik} denotes the *k*-th component of C_i. Thus, the value of ∇h can be computed as follows:

1) Computing C by Eq. 16
2) Computing $T_i = v_i \cdot C_i, \forall i \in [n]$
3) Computing $d_i^k = v_i^k T_i - C_{ik}$

Next, graph partitioning could perform in a parallel mode. Every processor q stores two vectors T and $T^{(q)}$. T_i and $T_i^{(q)}$ denote the i-th components of these vectors, and we assign each processor q a set SP_q of components such that $|SP_{qi}| \geq |SP_{qi}| - 1$ for all processors q_i, q_j.

The objective function is as follows.

1) for all q in parallel:

$\forall i \in [n]$, $k \in S_q$: $C_{ik} = \sum_{j \in adj(i)} w_{ij} v_j^k$

$\forall i \in [n]$: $T_i^{(q)} = \sum_{k \in S_q} v_i^k C_{ik}$

2) compute $\sum_q T^{(q)}$ using global communication

3) return $\sum_i T_i$

B. Algorithm description

The graph-based Web image annotation algorithm is illustrated as follows.

Graph-based Web image annotation algorithm

Input: An image I with t candidate annotations which is denoted as $\{A_1, A_2, \cdots\cdots, A_t\}$

Candidate annotations clustering:

Using K-means to cluster the candidate annotations into k set $S = \{S_1, S_2, \cdots\cdots, S_k\}$, the centroid set are $M = \{m_1^*, m_2^*, \cdots\cdots, m_k^*\}$

Graph establishing:

1) $W_{ij} = Sim(m_i^*, m_j^*)$

2) if $W_{ij} < \delta$ then let $W_{ij} = 0$

3) Building a graph with cluster centroids as the nodes and using W_{ij} as edge weight.

Graph cutting by paralleled algorithm

4) Solve MAX CUT problem, and obtain an optimal set of vectors v_i

5) Let r be a vector uniformly distributed on the unit sphere S_n.

6) Partition nodes of the graph into two parts, where $S = \{i | v_i \bullet r \geq 0\}$ and $\overline{S} = \{j | v_j \bullet r < 0\}$

Final annotations obtaining

7) if $\frac{\sum_{g,h \in S, g>h} W_{gh}}{|S|} \geq \frac{\sum_{\overline{g},\overline{h} \in \overline{S}, \overline{g}>\overline{h}} W_{\overline{g}\overline{h}}}{|\overline{S}|}$

Choose annotations in the clusters of which the centroid belonged to $\overline{S}$ as final annotations

8) else

Choose the remainder annotations as final annotations

Output: The final annotations of image I decided by the above process.

Our graph-based Web image annotation algorithm consists of four steps. In step 1, candidate annotations are clustered by K-means algorithm. Step 2 constructs the graph according to candidate annotations clustering results. Cluster centroids serve as graph nodes, and the distance between two centroids are used as edge weight. To reduce the computation cost, the edge of which the weight is less than a predefined threshold (denoted as δ) is deleted from the graph. In step 3, we perform a parallel graph bipartition algorithm to divide candidate annotations into two parts. In the final step, final annotations are chosen from the dividing results of step 3.

VI. An Example

In this section, we show an example to demonstrate our proposed algorithm.

Nowadays, people around the world use photos to visually communicate with others and present their feelings about a vacation, a party, a news event, or about virtually any topic related to their daily lives. Photo forums provide an energetic environment for people to share and discuss photography. To attract more attention, most photographers are very enthusiastic about providing metadata such as a title, category, location, camera setting, and description to the uploaded photos. And numerous volunteer users provide ratings and critiques to photos that they particularly like and enjoy[6].

For example, a photo entitled "*the rock*" at "http://www.photosig.com/go/photos/view?id=416147" (shown in Fig.2) has the following metadata:

Figure 2. A photo crawled from PhotoSIG.

Title: *the rock*

Photographer's Description: *One of the smaller landmasses in the Galapagos chain, popularly called "Kicker Rock" just after sunrise.*

Photographer's Category: *Landscape, Sea and Sand, Tourist.*

One of the critiques: *The rock truly catches the focus in this pic and the colors are on point as welll... Not really feeling the clouds that much they seem to look photoshopped.*

For this image, the annotations in Table 1 and the words in title and photographer's category field are obtained as the candidate annotations. The candidate annotation set is {*landscape, sea, sand, tourist, rock, sky, cloud, bird, clap, reflection, water, black, real, way, blue, depth, photograph, eye, place, center, capture, horizon, angle, sharpness, island, people, frame, crop, subject, border, nature, flawless, life* }.

TABLE I. ANNOTATIONS EXTRACTED FROM WEB PAGE WITH HIGH OCCURRENCE NUMBER

Annotation	Occurrence number	Annotation	Occurrence number
rock	*186*	*center*	*15*
sky	*151*	*capture*	*15*
cloud	*117*	*horizon*	*15*
bird	*108*	*angle*	*14*
clap	*60*	*sharpness*	*14*
reflection	*40*	*island*	*14*
water	*39*	*people*	*13*
black	*30*	*frame*	*12*
real	*29*	*sea*	*12*
way	*28*	*crop*	*12*
blue	*24*	*subject*	*12*
depth	*23*	*border*	*12*
photograph	*22*	*nature*	*12*
eye	*17*	*flawless*	*11*
place	*16*	*life*	*11*

After running our method, all candidate annotations are divided into two parts, and the final annotation set is {*landscape, sea, sand, tourist, rock, sky, cloud, bird, reflection, water, way, blue, island, people, nature* }. The conclusion can be drawn that our algorithm performs well in this example.

VII. EXPERIMENTS

We design three experiment scenarios to evaluate the performance of our approach. The image datasets we used are crawled from photographers' forums. Photos from various photo forums are of higher quality than personal photos, and are also more appealing to public users than personal photos. In addition, photos uploaded to photo forums require rich metadata about the title, camera setting, category, and description provided by photographers. These metadata are the most precise descriptions for photos and undoubtedly can be indexed to solve the relevance problem in a search engine. More importantly, there are volunteer users in each web community actively providing valuable ratings for these photos. The rating information is of great value in solving the photo quality ranking problem[6].

In experiment 1, we test the performance of our approach under different datasets, we select three community website for photographers that allows members to critique one another's work, which are Photosig[7], Flickr[8] and Photo[9].The proposed algorithm was evaluated on Web images which are randomly selected 200 images from the above photo forum sites and candidate annotations are also extracted from Web pages.

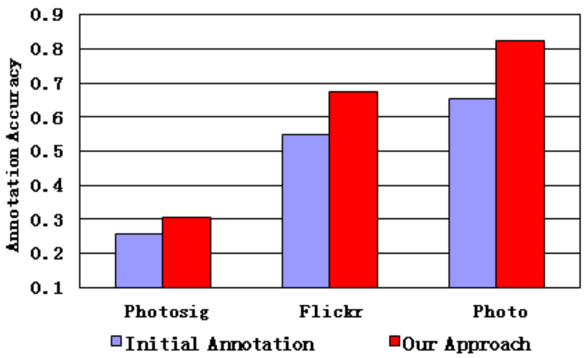

Figure 3. Accuracy enhancement results for three datasets.

We show the annotation results for three different dataset. In Fig.3, the Y-axis represents annotation accuracy and the X-axis represents three different noisy annotation sets. For the dataset extracting from Photosig, before annotating process, we see 25.74% accuracy and we have observed that our approach increases the accuracy to 30.56% through refinement. For other noisy dataset (Flickrs, Photo), the candidate annotation accuracy are 54.89% and 65.47%, and annotation results accuracy enhance to 67.41% and 82.55%. For the three datasets Photosig, Flickr and Photo, after annotating process, annotation accuracy are increased by 18.73%, 22.81%, 26.09% respectively. From the results of experiment 1, we find that the performance of our approach depends highly on dataset quality. In a word, the higher the dataset quality, the higher annotation accuracy enhancement.

In experiment 2 and experiment 3, Web images crawled from PhotoSIG[7] act as the test image set which has been used in experiment 1. We manually annotate these 200 Web images as the ground truth annotations. The initial annotations are extracted from the hosting web page firstly. The words with high occurrence number and the words in title and category field are obtained after stemming and stop words removing.

To evaluate the proposed algorithm, three annotation methods are compared with our approach as follows: 1) the WordNet-based method(WNM)[2], 2) an image annotation refinement algorithm using random walk with restarts(RWRM)[3], 3) a framework for knowledge-based image annotation refinement through randomized approximation of weighted maximum cut problem(KBIAR-MC)[5]. We use the same candidate annotation set for our approach, WNM, RWRM and KBIAR-MC, therefore, annotation performance can be evaluated fairly.

For all the candidate annotations, a fixed number of annotations are chosen as final annotations. The number of final annotations is denoted as *N*. For the performance metric, we adopt top *N* precision and coverage rate to measure the performance of final annotations. Top *N* precision (denoted as *P(N)*) measures the precision of top *N* ranked annotations for an image. Top *N* coverage rate (denoted as *C(N)*) is defined as the percentage of images which are correctly annotated by at least one word among the first *N* ranked annotations. *precise(i,N)* is the number of correct annotations in top *N* ranked annotations of image *i*, and *T* is the test image set. If at least one correct annotation of image *i* is belonged to the top *N* ranked annotations, *coverage(i,N)* is set 1, otherwise 0. To evaluate the performance of final annotations, the precision and coverage rate are adopted together in our experiment.

$$P(N)=\frac{\sum_{i\in T} precise(i,N)}{|T|\cdot N} \qquad (17)$$

$$C(N)=\frac{\sum_{i\in T} coverage(i,N)}{|T|} \qquad (18)$$

In experiment 2, we test the annotation precision when the number of initial annotations changing (shown in Fig.4). Experiment 3 shows the results in a different view, in which we show the annotation coverage rate when the number of initial annotations varying (shown in Fig.5). Fig.6 shows the annotation results of four selected images from PhotoSIG. From the experimental results, the conclusion can be drawn that our method works more effectively than other methods. The reasons lie in that the word clustering scheme and graph bipartition algorithm used in our approach work effectively.

VIII. Conclusions and Future Work

In this paper, we have presented a novel approach to annotate Web images by candidate annotations clustering and parallel graph partition algorithm. Our method is mainly made up of three steps. Firstly, the candidate annotations are clustered to construct a weighted graph. Next, the graph are divided into two parts in parallel mode. Finally, one of the two parts is selected to generate final annotations.

In the future, we would like to extend our work in the following directions. First, we will use more effective schemes to cluster candidate annotations. Next, we will test the performance of our approach on other large scale datasets .

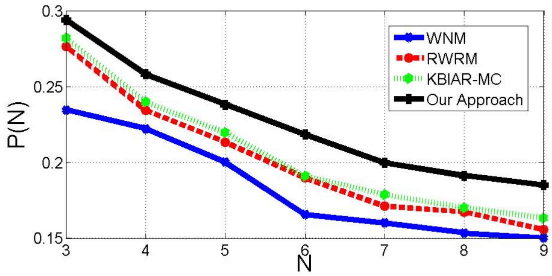

Figure 4. Precision comparison with number of final annotations.

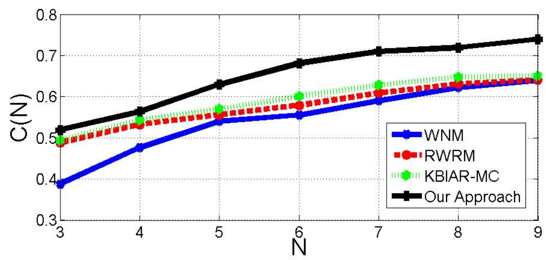

Figure 5. Coverage rate comparison with number of final annotations.

Acknowledgment

This work is supported by the National Natural Science Foundation of China (Grant No.60970048), Research Foundation of Shandong Economic University ("Research on Key Problems of Automatic Image Annotation", 2008) and Project of Shandong Province to Support Excellent Young Teacher as Domestic Visiting Scholar, 2008.

References

[1] Mei Wang, Xiangdong Zhou, and Hongtao Xu, Web Image Annotation Based on Automatically Obtained Noisy Training Set, APWeb 2008, LNCS 4976, 2008, pp.637-648.
[2] Jin, Y., Khan, L., Wang, L., and Awad M. Image Annotations By Combining Multiple Evidence & Wordnet. Proc. of ACM Multimedia, Singapore, 2005, pp.706-715.
[3] Changhu Wang, Feng Jing, Lei Zhang, Hong-Jiang Zhang. Image annotation refinement using random walk with restarts. Proc. of the 14th annual ACM international conference on Multimedia, 2006, pp. 647-650.
[4] Changhu Wang, Feng Jing, Lei Zhang, Hong-Jiang Zhang. Content-Based Image Annotation Refinement. Proc. of

IEEE Computer Vision and Pattern Recognition, 2007, pp. 1-8.
[5] Yohan Jin, Latifur Khan, B.Prabhakaran. To be Annotated or not?: the Randomized Approximation Graph Algorithm for Image Annotation Refinement Problem. ICDE2008 Workshop, 2008.
[6] Zhang, L., Chen, L., Jing, F., Deng, K.F.,Ma,W.Y.: EnjoyPhoto-a vertical image search engine for enjoying high-quality photos. In: Proceedings of ACM Multimedia 2006, pp.367-376.
[7] PhotoSIG: http://www.photosig.com.
[8] Flickr: http://www.flickr.com.
[9] Photo: http://www.photo.net.
[10] R.M. Karp. Reducibility among combinatorial problems. Complexity of Computer Computations, Plenum Press, 1972, pp. 85-103.
[11] Homer, S., Peinado, M., A highly parallel algorithm to approximate MaxCut on distributed memory architectures. 9th International Parallel Processing Symposium Proceedings, 1995, pp.113-117.
[12] X. Rui, M. Li, Z. Li, W.-Y. Ma, and N. Yu. Bipartite Graph Reinforcement Model for Web Image Annotation. In: Proceedings of the 15th international conference on ACM Multimedia 2007, pp.585-594.
[13] Wang, X.J., Zhang, L., Jing, F., Ma, W.Y., AnnoSearch: Image Auto-Annotation by Search. In: Proc. CVPR 2006, pp.1483-1490.
[14] Hua, Z.G., Wang, X.J., Liu, Q.S., Lu, H.Q.: Semantic Knowledge Extraction and annotation for Web Images. In: Proc. ACM Multimedia 2005, pp.467-470.
[15] Y. A. Aslandogan and C. T. Yu. Diogenes: A web search agent for person images. In: Proceedings of the eighth ACM international conference on Multimedia 2000, pp.481-482.
[16] D. Cai, S. Yu, J. Wen, W. Ma. Vips: a vision-based page segmentation algorithm, Microsoft Technical Report (MSR-TR-2003-79), 2003.
[17] Lin, D. 1998. Automatic retrieval and clustering of similar words. Proceedings of COLING/ACL-98. Montreal, Canada, pp.768-774.
[18] Xin-Jing Wang, Lei Zhang, Xirong Li, Wei-Ying Ma. Annotating Images by Mining Image Search Results. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 30, Issue 11, 2008, pp.1919-1932.
[19] Wong, R.C.F, Leung, C.H.C. Automatic Semantic Annotation of Real-World Web Images. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 30, Issue 11, 2008, pp.1933-1944.
[20] Cai, D., He, X.F., Li, Z.W., Ma,W.Y., Wen, J.R., Hierarchical Clustering of WWW Image Search Results Using Visual, Textual and Link Information. In: Proc. ACM Multimedia 2004, pp. 952-959.
[21] J. Jia, N. Yu, and X.-S. Hua. Annotating personal albums via web mining. In: Proceedings of ACM Multimedia 2008, pp.459-468.

Image	Image ID in Photosig	Candidate Annotations	Our Approach
	513957	Skater, ice, sharp, action, pose, reflection, speed, silver, glove, team, position, magazine, sport, background, lens, capture, subject, power, face, body	Skater, ice, sport, speed, pose, capture, body, face, glove, team
	553303	Animal, nature, rural, lighting, color, herd, sheep, sun, mood, scene, effect, view, mist, fog, flock, dream, sky, circle, picture, look	Animal, nature, rural, herd, sheep, sun, fog, mist, flock, sky
	564324	Beach, sky, sand, water, zanzibar, boat, sea, cloud, white, bright, winter, shadow, color, scene, picture, blue, cold, line, edge	Beach, sky, sand, water, boat, sea, cloud, winter, cold, blue
	642804	Lighting, dog, atmosphere, walk, forest, person, woods, darkness, nature, exposure, grass, beauty, landscape, nature, mood, depth, beauty, frame, shadow, picture	Dog, forest, woods, nature, grass, person, landscape, walk, beauty

Figure 6. Annotation results of four Web images.

Zheng Liu, born in 1980, earned a B.S. and M.S. degree in Computer Science & Technology from Shandong University, in 2002 and 2005 respectively. After graduate school, he joined school of Computer Science & Technology, Shandong Economic University in 2005. Now he is pursuing his Ph.D. degree in Shandong University, and works with the Information Retrieval Group.

His current research interests include machine learning, pattern recognition and multimedia data mining, currently he is more interested in image retrieval.