Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Image annotation via graph learning

Jing Liu[a,*], Mingjing Li[b], Qingshan Liu[a], Hanqing Lu[a], Songde Ma[a]

[a]Institute of Automation Chinese Academy of Sciences, No. 95, Zhongguancun East Road, Beijing 100080, China
[b]Microsoft Research Asia, Beijing 100080, China

ARTICLE INFO

Article history:
Received 15 December 2007
Received in revised form 21 March 2008
Accepted 16 April 2008

Keywords:
Graph learning
Image annotation
Image similarity
Word correlation

ABSTRACT

Image annotation has been an active research topic in recent years due to its potential impact on both image understanding and web image search. In this paper, we propose a graph learning framework for image annotation. First, the image-based graph learning is performed to obtain the candidate annotations for each image. In order to capture the complex distribution of image data, we propose a Nearest Spanning Chain (NSC) method to construct the image-based graph, whose edge-weights are derived from the chain-wise statistical information instead of the traditional pairwise similarities. Second, the word-based graph learning is developed to refine the relationships between images and words to get final annotations for each image. To enrich the representation of the word-based graph, we design two types of word correlations based on web search results besides the word co-occurrence in the training set. The effectiveness of the proposed solution is demonstrated from the experiments on the Corel dataset and a web image dataset.

1. Introduction

With the advent of digital imagery, the number of digital images has been growing rapidly and there is an increasing requirement on indexing and searching these images effectively. Systems using non-textual (image) queries have been proposed but many users found it hard to represent their queries using abstract image features. Most users prefer textual queries, i.e., keyword-based image search, which is typically achieved by manually providing image annotations and searching over these annotations using a textual query. However, manual annotation is an expensive and tedious procedure. Thus, automatic image annotation is necessary for efficient image retrieval.

Generally, image annotation methods aim to learn the semantics of untagged images from annotated images according to image similarities. Its probabilistic interpretation is to find a set of keywords w^* (in a given vocabulary V) that maximize the joint probability of $P(w, I_q)$ as follows[1]:

$$\begin{aligned} w^* &= \arg\max_{w \subset V} P(w, I_q) \\ &= \arg\max_{w \subset V} \sum_{I_i \in T} P(w|I_i)P(I_q|I_i)P(I_i) \end{aligned} \tag{1}$$

where I_q is an untagged image, T is a set of annotated images, $P(I_q|I_i)$ denotes the probability that I_i is relevant (or similar) to I_q, and $P(w|I_i)$ represents the likelihood that I_i can be annotated with w.

From the formulation, a basic image annotation system consists of two relations: image-to-image relation (IIR) and image-to-word relation (IWR). Typically, IIR is built with visual features, which is available given a dataset. IWR indicates the likelihood of a word given an image, which is a word model learned from annotated images. Given IIR and IWR, annotations of the untagged images can be achieved by the similarity propagation. In addition, word-to-word relation (WWR) can be explored to refine the annotations so as to maintain the semantic consistence among them.

Then, we formulate the problem of image annotation into a graph learning framework, which includes the image-based graph learning and the word-based graph learning. The image-based graph learning is first performed to learn the relationships between images and words, i.e., to obtain the candidate annotations for each image, and then the word-based graph learning is used to refine the obtained relationships by exploring word correlation.

How to build a similarity graph is very important in graph learning. A good graph should reflect a deep understanding of the data structure and help to mine potential knowledge as much as possible. In previous studies, the graph is often constructed by k-NN or ε-ball-based pairwise similarities. However, these pairwise relations may not satisfy the requirements of image annotation on the large image database, due to the limited domain knowledge and the complex distribution of image data. In order to better capture the complex distribution of image data, in this paper, we propose a nearest

* Corresponding author. Tel.: +86 1062542971.
E-mail address: liujingmgm@gmail.com (J. Liu).

[1] Here we assume that the events of observing words w and image I_q are mutually independent once we pick the training image I_i.

0031-3203/$ - see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2008.04.012

spanning chain (NSC) method to construct the image-based graph, i.e., the NSC-based image graph. The edge-weights of this graph are calculated with the chain-wise statistical information. This calculation can leverage the structural knowledge existing in the real data distribution and so outperform the traditional representation of pairwise similarities.

Since the image-based graph learning does not take the relationship between words into account, it is inevitable to produce semantically inconsistent annotations. Thus, we develop the word-based graph learning to refine the obtained annotations from the image-based graph learning, which is performed by exploring the word correlation. To enrich the word correlation, we not only use the word co-occurrence in the annotated dataset, but also design two types of word correlations in the web context by using Google image searcher. These three word correlations are combined with a linear model to build the word-based graph.

The main contributions in this paper are:

- A graph learning framework is proposed for image annotation. In this framework, we not only target at learning the relationship between images and words by the image-based graph learning, but also explore the word correlation by the word-based graph learning to further improve the performance of image annotation.
- We introduce and further improve the NSC method for the image-based graph construction, which replaces the pairwise similarities with the chain-wise statistical similarities. It can well adapt to the data distribution.
- We design two new word correlations in the web context in addition to the word co-occurrence in the training set. Specially, the two web-based correlations provide more general and robust estimations and open a new domain for word correlation estimation.

The rest of the paper is organized as follows. Related work is briefly reviewed in Section 2. The proposed annotation framework is presented in Section 3. The implementation of two-phrases graph learning for image annotation are addressed in Sections 4 and 5 separately. The experiments are reported in Section 6. Finally, the conclusion and future work are given.

2. Related work

Machine learning techniques have been used extensively in the field of image analysis [1,2], and there is not an exception for image annotation. We can classify them into three categories, i.e., the classification methods, the probabilistic modeling methods, and the graph-based methods.

The classification methods treat each semantic keyword or concept as an independent class, and assign each keyword or concept one classifier. Work such as linguistic indexing of pictures [3], image annotation using SVM [4], and Bayes point machine [5] falls into this category.

The probabilistic modeling methods aim to learn a relevance model to represent the correlation between images and keywords. The early work [6] applied a translation model (TM) to translate a set of blob tokens (obtained by clustering image regions) to a set of annotation keywords. Jeon et al. [7] assumed that image annotation could be viewed as analogous to the cross-lingual retrieval problem and proposed a cross-media relevance model (CMRM). Lavrenko et al. [8] proposed a continuous-space relevance model (CRM), in which the image is segmented into regions and each region is described with a continuous-valued feature vector. Given a set of training images with annotations, a joint probabilistic model of image features and words is estimated. Then the probability of the image regions belonging to a word can be predicted. Compared with the CMRM, the CRM directly models continuous features, so it does not rely on clustering and avoids the granularity issues. Feng et al. [9] proposed another model based on the multiple Bernoulli distribution to generate words instead of the multinomial one as in the CRM. The Gaussian mixture model, the latent Dirichlet allocator (LDA), and the correspondence LDA incorporate latent variables to link image features with keywords [10]. Carneiro et al. [11] proposed a supervised multiclass labeling (SML) method, in which a two-level mixture probabilistic model, i.e., one mixture density estimated for each image, the other mixture associated with all images annotated with a common semantic label, is built to learn the correspondence between images and their labels. Some studies also integrate the word correlation in the annotation process, such as the coherent language model (CLM) [12], the correlated label propagation (CLP) [13], and the wordnet-based method (WNM) [14].

Recently, the graph-based methods have achieved much success in the field of image and video analysis including image annotation. One of the representative work was developed by Tong et al. [15], which supported keyword propagation for image retrieval. Actually, it is performed by propagating the keywords from the labeled images to the unlabeled images by visual similarities. It is domain independent and the parameters are easy to tune, but this model directly used the pairwise relations over images to construct a k-NN similarity graph and did not consider the correlations between words.

In our previous work [16], we proposed a graph model based on the NSC to annotate images. The model gives a statistical estimation of image similarities, which is gained by exploring the pairwise connections in multiple full-length NSCs. However, the sparsity of those connections in NSCs and the high expense on calculation constrain the performance of the estimation. In other side, word correlations were used to refine annotations for each image, in which only the semantic consistence among top annotations for each image was leveraged, while the annotating probabilities for each annotation obtained from the NSC-based graph learning were not considered. The semantic relatedness from WordNet and word co-occurrence in training set were combined to estimate the word correlation in Ref. [16]. However, the estimation which depends on manually defined lexicons (WordNet and a training dataset), can only handle a limited number of words and relationships among them. To attack all above problems, in this paper, we explicitly present a graph learning framework for image annotation and propose some improvements on the construction of image-based graph and word-based graph. Besides, more comprehensive experiments are performed to demonstrate the effectiveness of the proposed solution.

3. Image annotation via graph learning

In this section, we will first overview the graph learning algorithm [17], and then present the annotation framework via graph learning.

3.1. Graph learning algorithm

Given a set of points $X = x_1, x_2, \ldots, x_N \subset \Re^d$, x_i denotes a d-dimensional feature vector. The labeling matrix $Y^{N\times c}$ is initialized according to prior knowledge, where c is the number of categories. Typically, $y_{ij} = 1$ if x_i is initially labeled as the jth category and $y_{ij} = 0$ otherwise. The element in the matrix $R^{N\times c}$ is defined as the confidence of each point relevant to each category, while its stable state is derived from an iterative procedure as follows:

Step 1: Construct the similarity matrix $W^{N\times N}$ as

$$W_{ij} = \exp\left[-\frac{dis(x_i, x_j)}{\sigma^2}\right] \tag{2}$$

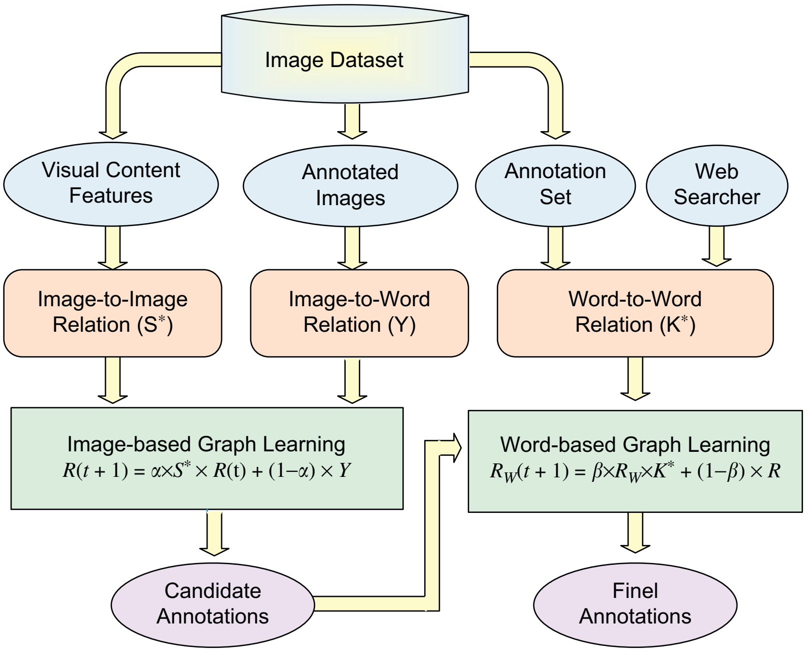


Image Dataset
Visual Content Features
Annotated Images
Annotation Set
Web Searcher
Image-to-Image Relation (S*)
Image-to-Word Relation (Y)
Word-to-Word Relation (K*)
Image-based Graph Learning
$R(t+1) = \alpha \times S^{*} \times R(t) + (1-\alpha) \times Y$
Word-based Graph Learning
$R_W(t+1) = \beta \times R_W \times K^{*} + (1-\beta) \times R$
Candidate Annotations
Finel Annotations

Fig. 1. Illustration of the proposed image annotation framework.

where $dis(\cdot)$ is a certain distance metric (L1 distance is used in our implementation). Note that $W_{ii}=0$ because there are no loops in the graph.

Step 2: Symmetrically normalize W by

$$S = D^{-1/2} W D^{-1/2} \tag{3}$$

where D is a diagonal matrix and $D_{ii} = \sum_{j=1}^{N} W_{ij}$.

Step 3: Do iteration according to Eq. (4), until convergence.

$$R(t+1) = \alpha S \cdot R(t) + (1-\alpha) Y, \quad R(0) = Y \tag{4}$$

where t denotes the number of iterations and α is the propagating parameter.

Step 4: Decide the label for each unlabeled point according to the convergent matrix of R^*.

According to the above description, the graph learning method has two key components: the similarity graph (S) and the initial state representation (Y). The former describes the data distribution over all the labeled and unlabeled data, and the latter provides prior expert information for the learning process. So the graph learning method provides a natural way to solve the automatic annotation by the label propagation.

3.2. *Overview of image annotation framework*

We propose a graph learning framework for image annotation, which is illustrated in Fig. 1. This framework contains two graph models, i.e., the image-based graph model and the word-based graph model.

Firstly, given the annotated training set and the visual features of all the images, the image-based graph learning aims to propagate labels from the annotated images to the un-annotated images by their visual similarities. In this model, the image-based similarity graph (S^*) is built with the NSC technique, which can capture the data distribution efficiently. As for the initial IWR (Y), we utilize the multiple Bernoulli model to indicate the probability of the word given an image, rather than the traditional indicative function. The details are described in Section 4.

The purpose of the word-based graph learning is to refine the annotations obtained from the image-based graph learning. Three kinds of word correlations are designed to build the word-based graph (K^*). One is based on the annotated training set. The other two are based on analyzing the web context with the help of popular image searcher. The details are given in Section 5.

4. Image-based graph learning

Traditionally, the similarity graph is constructed with pairwise similarities directly, while the structural distributing information among data has not been considered suitably. But such information usually provides valuable knowledge to identify data, which can be observed from the toy examples in Fig. 2 intuitively. To better capture the data distribution, we try to explore a kind of chain-wise distributing information to refine the pairwise relationships and propose the following NSC method, and apply it to build the image-based graph for image annotation.

4.1. *Nearest spanning chain (NSC)*

The proposed NSC is similar to the minimum spanning tree (MST) [19], which connects all the points in a dataset. However, the NSC is a sequential chain, which only has one-to-one connections without any branches, while the MST usually has some branches, i.e., one-to-many connections. So the NSC is much simpler than the MST. The proposed NSC mainly has three properties as follows:

Property 1. N points (two end points and $N-2$ mid points) and $N-1$ edges form one sequential chain.

Property 2. Two end points are connected with one edge, respectively, and other mid-points are connected with two edges.

Property 3. The points in one chain are connected according to the rule of the nearest in residue (*NinR*). It means that a point should find its nearest neighbor in the residual un-selected points.

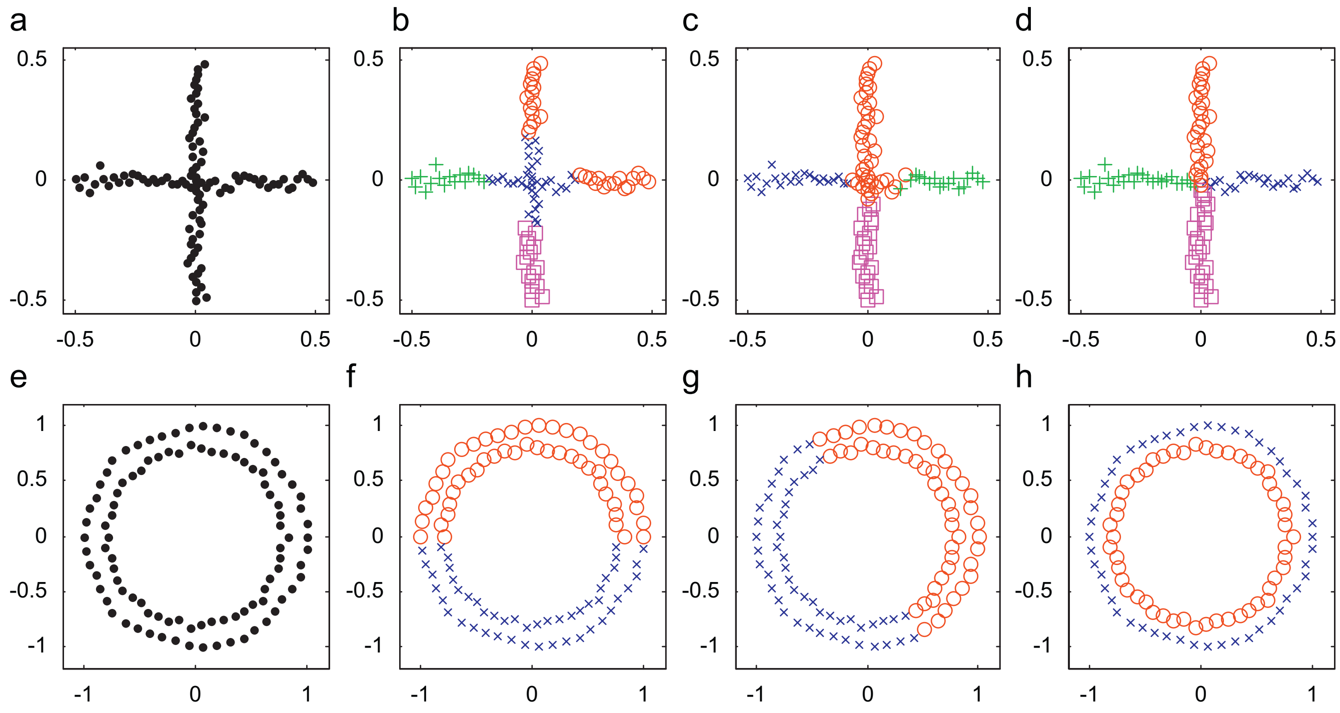


Fig. 2. Toy examples: (a) toy data distribute along two perpendicularly crossing lines, in which the distributing variances are 0.02; (b)–(d) the clustering results based on three types of similarity matrixes, which are the traditional pair-wise one [17], the original NSC-based one [16] and the improved sub-NSC-based one in this paper, respectively, using the spectral clustering algorithm [18]; (e) toy data distribute on two closing circles, whose radius are set as: radius1 = normrnd(1, 0.02) and radius2 = normrnd(0.8, 0.02), respectively; (e)–(h) the clustering results based on the three types of similarity matrixes as used in (b)–(d).

Based on the above three properties, two nearer (more similar) points should have a higher probability to be connected together and two farther (more un-similar) points have a lower probability to be connected. Therefore, if multiple NSCs are given, some useful statistical information can be mined to reflect the data distribution to some extent. For clarity, a toy example is given in Fig. 3, in which both sub-figures on the top present the data distribution along a unit circle and many examples of the NSCs, respectively. For instance, the point 2 is nearer to the point 1 than to the point 9, while in the NSCs, the connection of (1–2 or 2–1) occurs more frequently than the connection of (2–9 or 9–2).

4.2. NSC-based image graph

Intuitively, the statistical information in multiple NSCs can enrich the description of the pairwise relations. How to present a robust expression becomes quite important. According to our previous proposal in Ref. [16], the frequency of every pairwise connection in the NSCs is considered as the statistical expression. Considering that the NSC is generated sequentially according to the rule of the *NinR*, the connection occurring in the front part of the NSC gains more confidence than that of in the rear part. Similarly, the connections between the same point-pair occurring in different positions in multiple NSCs, should have different contributions to the statistical representation. The weight should be inversely related to the occurring sequence. So we define the statistical description (C_{ij}) in the NSCs for points i and j as follows:

$$C_{ij} = \sum_{n=1}^{N} seq\_w_{ij}^{n} \cdot \delta_{ij}^{n}, \quad seq\_w_{ij}^{n} = \exp\left(\frac{\lambda_1}{idx\_i + idx\_j}\right) \tag{5}$$

where $seq\_w_{ij}^{n}$ denotes the sequential weight for the connection between points i and j in the nth chain, idx_i (or idx_j) is the occurring index of the point i (or j) in the nth chain. And $\delta_{ij}^{n} = 1$ if the points i and j are directly connected in the chain, and zero otherwise.

With the statistical description, the NSC-based graph for a N-points dataset is constructed as follows:

Step 1: Build N full-length NSCs starting from every point by repeating the following sub-steps: (i) Select one point as the starting point of one NSC. (ii) According to the rule of the *NinR*, absorb all other points into the NSC sequentially and obtain an N-length chain including all the N points.

Step 2: Learn statistical information from N NSCs according to Eq. (5).

Step 3: Construct the traditional similarity matrix as Eq. (2).

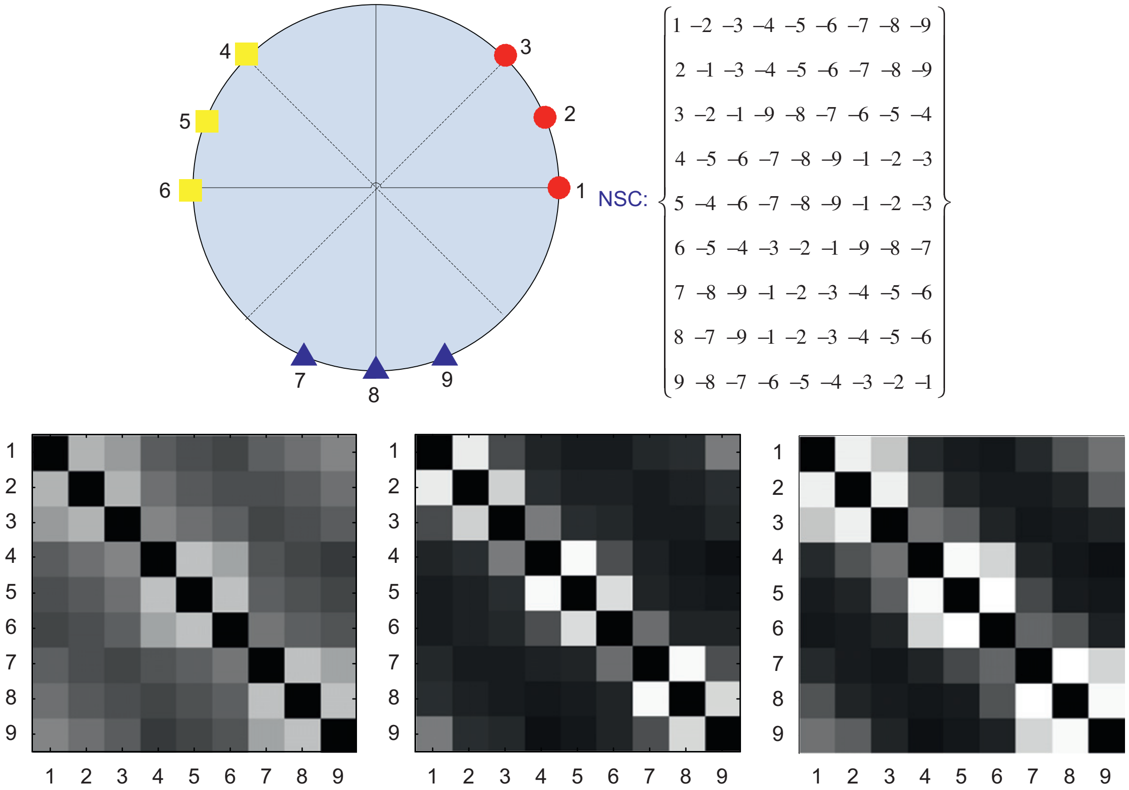
Step 4: Obtain a refined similarity matrix W^* by weighting W with C as

$$W_{ij}^{*} = C_{ij} \cdot W_{ij} \tag{6}$$

Step 5: Normalize W^* as Eq. (3) to obtain the NSC-based image graph S^*.

4.3. Improved sub-NSC-based image graph

The above NSC-based graph is built with the pairwise direct connections in N full-length NSCs (i.e., each NSC contains all the points in the dataset). Actually many point-pairs are never connected directly in the NSCs, which lead to a sparse statistical matrix (C). In addition, the calculation with all the direct connections in the full-length NSCs bring with not only high cost on computation, but also the weakened robustness, since the connections in the rear part of NSC is relatively unreliable. To deal with these issues, we propose two improvements with a reasonable assumption that the local distribution of the image data is dense. As it is known, the points are unsimilar when they are far away from each other, so it is



$$\text{NSC:}\left\{\begin{array}{ccccccccc}
1 & -2 & -3 & -4 & -5 & -6 & -7 & -8 & -9 \\
2 & -1 & -3 & -4 & -5 & -6 & -7 & -8 & -9 \\
3 & -2 & -1 & -9 & -8 & -7 & -6 & -5 & -4 \\
4 & -5 & -6 & -7 & -8 & -9 & -1 & -2 & -3 \\
5 & -4 & -6 & -7 & -8 & -9 & -1 & -2 & -3 \\
6 & -5 & -4 & -3 & -2 & -1 & -9 & -8 & -7 \\
7 & -8 & -9 & -1 & -2 & -3 & -4 & -5 & -6 \\
8 & -7 & -9 & -1 & -2 & -3 & -4 & -5 & -6 \\
9 & -8 & -7 & -6 & -5 & -4 & -3 & -2 & -1
\end{array}\right\}$$

Fig. 3. Toy example: the top-left figure gives the toy data distribution, in which the points with the same marks are assumed to be in the same category and the number besides each point represents its index for clarity. The top-right one gives some examples of NSC denoted by the indexes of data. The three figures on thee bottom illustrate the maps of three types of similarity matrix, in which the brighter block indicates that the corresponding two points are more similar. Specially, the bottom-left figure presents the map based on the pairwise similarity as traditional method [17]. The bottom-middle figure presents the map based on the original NSC method [16]. The bottom-right figure presents the map based on the improved sub-NSC-based method in this paper.

enough and feasible to model the data distribution with locally neighboring data.

Based on the local data distribution, we first relax the NSC to the sub-NSC, i.e., we build the sub-NSC with $M(M<N)$ points instead of all the N points in a dataset. Because we focus on the frequency of pairwise connections among multiple NSCs, we hope the connections are as reliable as possible. As discussed above, the connections occurring in the front part of the full-length NSC should have more confidence than the ones in the rear part. Thus, we take the front part of the NSC as the sub-NSC, and use multiple such sub-NSCs to estimate the data distribution, which are demonstrated to be sufficient by the experiments. Furthermore, the sub-NSC-based graph reduces the computational cost, for it only needs to manage partial sorting indices.

In Section 4.2, only the direct pairwise connections are used to compute the statistical matrix C. Actually some indirect connections in the neighborhood can also provide useful information for constructing C. For the toy example shown in Fig. 3, points 1, 2 and 3 always occur in the NSCs as 1-2-3 (or 3-2-1). Besides direct connections like (1, 2) and (2, 3) are similar pairs, the indirect connection (1, 3) should also be a similar pair because it is always linked by the point 2. We call the point 2 the linker between 1 and 3. We use both direct connections and indirect connections to improve the statistical description. Namely we consider not only the connections like 1-2 (or 2-1) and 2-3 (or 3-2), but also the ones like 1-3 (or 3-1) to calculate the statistical matrix C. Certainly, the pair with the direct connection is more reliable to be similar than the pair with the indirect connection, so we give different weights to the connections according to the length of their linkers as well as their occurring positions. Given two points i and j, the weighted statistical description C'_{ij} should be calculated as

$$C'_{ij} = \sum_{n=1}^{N} seq\_w_{ij}^{n} \cdot f_{L_{ij}}^{n} \cdot \delta_{ij}^{n}, \quad f_{L_{ij}}^{n} = \exp\frac{\lambda_2}{L_{ij}+1} \qquad (7)$$

where $\delta_{ij}^{n}=1$ if i,j are neighbors in nth sub-NSC, and zero otherwise, $seq\_w_{ij}^{n}$ is same as in Eq. (5), $f_{L_{ij}}^{n}$ is the weight inversely relevant to the value of L_{ij}, which is the length of the linker for the connection between the points i and j. Specially, $f_{L_{ij}}^{n}$ obtains the maximum when $L_{ij}=0$, i.e, the direct connection is considered.

To make the connection reliable, the maximum length of the linker, which decides the size of the neighborhood in a sub-NSC to select those connections, cannot be larger than the number of points in each consistent category. For the problem of image annotation, those images similar both on visualness and on semantics should form a consistent category. The number of such images is less than the number of images which are only similar on semantics, let alone the total number of images in dataset. Thus the maximum of the linker is greatly less than the size of dataset, which is also demonstrated in our experiments. Then the extra cost from the improvement is neglectable.

We apply the above two improvements in the image graph construction as mentioned in Section 4.2, i.e., using the sub-NSCs to replace the full length NSCs in Step 1 and using Eq. (7) to replace Eq. (5) in Step 2. Then we get an improved image similarity graph for the graph learning.

Here, we take the toy example shown in Fig. 3 to present the effect of these improvements. For the toy data, nine points are distributed along a circle and can be classified into three classes, i.e., $\{1,2,3\}$, $\{4,5,6\}$, and $\{7,8,9\}$. We present a comparison among three similarity maps, which are obtained from different approaches, respectively, in Fig. 3: the traditional one (bottom-left) as in Ref. [17], the original NSC-based one (bottom-middle) as mentioned in Section 4.2 and the improved sub-NSC-based one (bottom-right). The class memberships, either inter- or intra-relationship, are reflected more precisely and clearly in the sub-NSC-based map. The original NSC-based one weakens the partial intra-relationship such as 1–3, 4–6, and 7–9. The traditional one cannot figure out a contrastive map. Thus, the improved graph obtains the best description of data distribution by pulling the inter-relationship farther and drawing the intra-relationship nearer. Similar outperformance of the improved sub-NSC method can be seen from the examples illustrated in Fig. 2.

4.4. *Image-based graph learning for basic annotation*

The labeling matrix is another necessary component during the graph learning. Traditionally, an indicative function is used to present the label information, i.e., the probability of each word given an image is 0 or 1. Obviously, this is quite an absolute representation in that different annotations of an image usually have different reliabilities to represent the semantics of the image. Here, we adopt a probabilistic measure, which assumes a multiple Bernoulli distribution of annotations [9], to initialize the labeling matrix Y as

$$Y_{ij} = P(w_j|I_i) = \frac{\mu\delta_{w_j,I_i} + N_{w_j}}{\mu + N_T} \tag{8}$$

where Y_{ij} indicates the probability of word w_j given image I_i, μ is a smoothing parameter. $\delta_{w_j,I_i} = 1$ if w_j occurs in the annotations of image I_i and zero otherwise. And N_{w_j} is the number of images that contain w_j in their annotations, and N_T is the number of training images.

With the sub-NSC-based graph S^* and the probabilistic labeling matrix Y, we perform the iterative learning process as Eq. (4). When arriving the convergence, the resulting matrix R provides some preliminary annotations. We would further refine them by the following word-based graph learning.

5. Word-based graph learning

The above image-based graph learning only focuses on the visual similarities among images, while the word correlations are not analyzed. As a result, the semantic consistence among the obtained annotations from the image-based graph learning is difficult to be achieved. In this section, we will design a combined estimation of word correlation to build a word-based graph and try to refine those annotations by the word-based graph learning. One aspect of the combination is based on the co-occurrence statistics over the training set. The other two are based on the visual representations of the words, which are generated by web image search.

5.1. *Word co-occurrence in the training set*

Generally speaking, two words with high co-occurrence in the training set will lead to high probability to annotate certain image jointly, such as 'cloud' and 'sky', 'water', and 'fish'. Therefore, the word co-occurrence becomes an informative representation of the word correlation.

Here, we count the word pairs as annotations jointly in the training set to obtain the word co-occurrence. Usually, the more general a word is, the larger chance it will have to associate other words to annotate images. However, such associations usually have low confidence. Thus, we put a lower weight to a frequent (general) word and a higher weight to a rare (specific) word. The weighted co-occurrence (K_{WC}) is calculated as

$$K_{WC}(x,y) = K_C(x,y) \times \log\left(\frac{N_T}{n_x}\right) \tag{9}$$

where $K_C(x,y)$ is the original number of co-occurrence for the words x and y, n_x is the count of x occurring in the annotated images, and N_T is the total number of the training images. Note that $K_{WC}(x,y)$ may be unequal to $K_{WC}(y,x)$. For example, considering two words 'water' and 'fish', in which the former is general but the latter is specific, we can easily infer that there must have 'water' where 'fish' appears, but not vice versa.

5.2. *Visual content-based correlation in the web context*

The above word co-occurrence is a locally statistical word correlation dependent on the training set. To get a more robust analysis, we seek other entrances to enrich the representation of word correlation. Web is considered as the largest public available corpus with aggregate statistical and indexing information. Such huge and valuable resources deserve to our attention. Since visual content is the direct representation of image semantics, it should also contribute to the word correlation. Thus, we design two visual-based word correlations with the help of Google image searcher. They are the visual content-based correlation in the web context and the visual consistence-based correlation in the web context. The former will be presented in the following, and the latter will be discussed in the next subsection.

Given a query, Google image searcher usually can provide some good search results, especially those on the first page. Thus, we treat the top-ranked images as the visual representations of the querying word roughly. And we define the similarity between the visual representations of two query words as their word correlation in the web context, i.e., the content-based word correlation, as follows:

$$K_{CC}(x,y) = K_S(I(x), I(y)) = \sum_{m,n} S_I(I_m(x), I_n(y)) \tag{10}$$

where $K_{CC}(x,y)$ indicates the visual content-based word correlation between words x and y, $I(x)$ and $I(y)$ indicate the resulting image sets by words x and y, respectively, $I_m(x)$ is the mth image in the image set $I(x)$, and $K_S(\cdot)$ is the similarity function between the two image sets, $S_I(\cdot)$ is the similarity function between two images, and $m,n = 1,2,\ldots,T_r$, T_r is the number of top images from the resulting images ($T_r = 10$ in our experiments).

5.3. *Visual consistence-based correlation in the web context*

The visual distribution of the top ranked images also implies much useful information, especially it can reflect the semantic uniqueness in a sense. For example, 'jaguar' as a polysemous word may represent an animal, a car or a plane, so its search results include animal, car, and plane images. These three kinds of images have different colors, shapes, and textures. That is, they are not visually consistent.

When one query is composed of two words with the conjunctive operator (AND), the search images should be indexed with the both words. If the two words are not semantically related to each other, the search results may be very noisy and not visually consistent. Based on this consideration, the visual consistence of the search results is believed to be a good indicator of the word correlation.

We use the variance of visual features to describe the visual consistence. The less variance of visual features corresponds to the more consistent visual content. With multimodal image features extracted, it is preferred that some image features are given more weight than others in calculating the visual variance in order to better imitate actual human visual cognition. Besides, from studies of cognitive psychology, it is learned that human infers overall similarity based on the aspects that are similar among the compared objects, rather than based on the dissimilar ones. Accordingly, we design a new measure called dynamic partial variance (DPV), which focuses on the features with low variances and activates different features for different image sets. Assuming the variances of each dimensional feature among images in the set S are ordered as $var_1(S) \leqslant var_2(S) \leqslant \cdots \leqslant var_d(S)$, the DPV is defined as:

$$DPV(S) = \frac{1}{l} \sum_{i=1}^{l<d} var_i(S) \tag{11}$$

where d is the dimension of the visual feature and l is the number of similar aspects activated in the measure, which is set as $l = 2d/3$ experientially.

To make the DPVs of the resulting images given word-pair queries comparable to each other, we normalize the values according to the semantic uniqueness of each word, i.e., the DPV of the resulting images given a single-word query. Given two words x and y, their visual consistence-based correlation is calculated as follows:

Step 1: Submit 'x', 'y', 'x y' to Google image searcher, respectively, and obtain three sets of images, denoted as S_x, S_y, S_{xy}, which are composed of the top T_r resulting images, respectively.

Step 2: For each image, extract its visual features.

Step 3: For each set, calculate the DPV according to Eq. (11).

Step 4: The correlation between x and y is given as

$$K_{VC}(x, y) = \exp\left(-\frac{\lambda_3 \cdot DPV(S_{xy})}{\min\{DPV(S_x), DPV(S_y)\}}\right) \tag{12}$$

where $\lambda_3 > 0$ is a smoothing parameter.

5.4. *Word-based graph learning for annotation refinement*

Now we obtain the three types of word correlations, and they have different characteristics. The word co-occurrence in the training set provides relatively precise statistical description, but it depends on the training dataset. Though the two word correlations in the web context are universal and independent on any corpus, they may contain some noises due to the diversity of the web data. To make them complement each other, our proposal is to unify them in a linear form as Eq. (13), after they are normalized into [0, 1], respectively. The better performance is expected when more sophisticated combinations are used:

$$K^* = \varepsilon_1 K_{WC} + \varepsilon_2 K_{CC} + \varepsilon_3 K_{VC} \tag{13}$$

where $\varepsilon_1, \varepsilon_2, \varepsilon_3 \in [0, 1]$, $\varepsilon_1 + \varepsilon_2 + \varepsilon_3 = 1$.

As shown in Fig. 1, we take the result matrix R from the image-based graph learning as the initial labeling matrix and take K^* as the word-based graph to perform the word-based graph learning for annotation refinement. Because the words do not have the transitive characteristic strictly, we carry out the propagation for only a few times (2 times in our experiments) as follows:

$$R_W(t+1) = \beta R_W(t) \cdot K^* + (1 - \beta) R \tag{14}$$

where β is the weight to regulate the role of annotation refinement using the word correlation. According to the resulting matrix R_W, the top t keywords are selected as the final annotations for each image ($t = 5$ in our experiments).

6. Experiments

We test the proposed framework on two datasets: the Corel dataset as in Ref. [6] and the web image dataset as in Ref. [20].

6.1. *Corel dataset*

It is publicly available and widely used in evaluating the image annotation methods. The dataset contains 5000 images. Each image is segmented into 1–10 regions. A 36-dimensional feature is extracted from each region, which includes color, texture, and area features as in Ref. [6]. Each image is annotated with 1–5 words. The total number of words is 371. The dataset is divided into two parts: 4500 images are for training and the rest 500 images are for testing. Totally 260 distinct words are found in the testing set. In order to present a fair comparison on the dataset with some related works, all the comparative experiments carried on the dataset use the same visual features of segmented regions.

6.2. *Web dataset*

The 9046 web images are crawled from the web by 48 topics. Each image is annotated with 5–15 keywords, which are extracted from the surrounding text and the tag information by the VIPS algorithm [21] and the standard text processing techniques. There are totally 1153 keywords excluding the rare ones whose occurring frequency is less than 5. A 144-dimensional global feature for each image is extracted, including 36-dimensional color histogram, 64-dimensional color correlogram [22], 20-dimensional Tamura feature [23] and 24-dimensional pyramid wavelet texture feature [24]. There are 8096 images for training and 1000 images for testing. Totally 826 distinct words are found in the testing set.

In the construction of the image-based graph, one image corresponds to one point in the graph, while the edge-weight is decided by the similarity between images. Because the annotations for the Corel dataset are given manually, they have relatively good correspondence to regions. We calculate the similarity between images based on the region-level visual features as in Ref. [8]. For the web dataset, due to the limitation on annotation extraction for the web images, the obtained annotations are usually diverse and inconsistent with region's semantics. Thus, we build the similarity graph model based on the global content features for the web images.

Similar to previous work [6,9,8], the quality of automatic image annotation is measured through the process of retrieving testing images with single keyword. For each keyword, the number of correctly annotated images is denoted as N_c, the number of retrieved images is denoted as N_s, and the number of truly related images in the testing set is denoted as N_r. The precision and recall are computed as follows:

$$precision(w) = \frac{N_c}{N_s}, \quad recall(w) = \frac{N_c}{N_r} \tag{15}$$

We compute the average precision and recall over all the words occurring in the testing images (260 words) to evaluate the performance. In addition, we give another measure to evaluate the coverage of correctly annotated words, i.e., the number of words with non-zero recall, which is denoted as 'NumWords' for short. This metric is important because a biased model can also achieve high precision and recall values by only performing quite well on a small number of common words.

6.3. *Parameter setting*

In this subsection, we will firstly investigate the parameter setting in the construction of the sub-NSC-based graph. For clarity, we

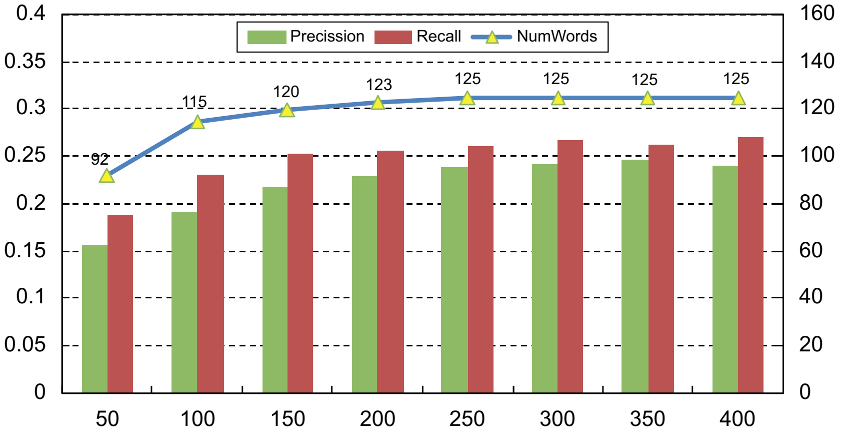


Fig. 4. Performance comparison on Corel dataset with different lengths of sub-NSC (M): the horizontal axis gives the different values of M. The bars present average precision and recall according to the left axis and the line denotes NumWords according to the right axis.

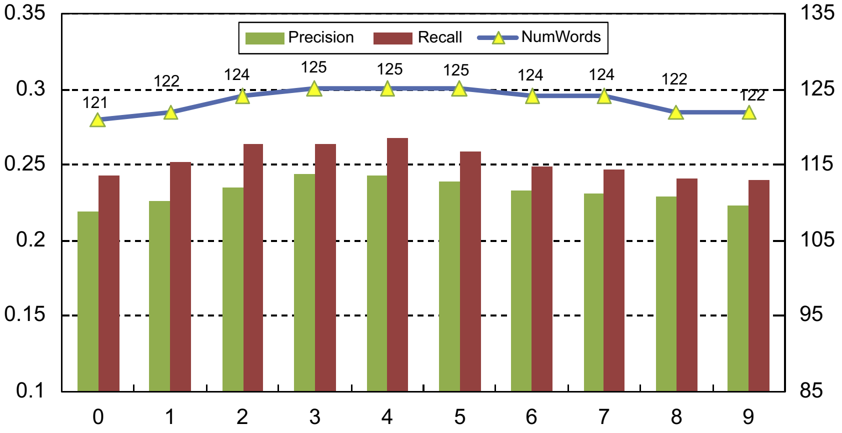


Fig. 5. Performance comparison on Corel dataset with different maximum linker lengths for pair-wise occurrence (L): the horizontal axis gives the different values of L. The bars present average precision and recall according to the left axis and the line denotes NumWords according to the right axis.

only perform the image-based graph learning and do not refine the obtained annotations by the word-based graph learning. The Corel dataset is used to evaluate the performance.

6.3.1. *Length of the sub-NSC*

According to Section 4.3, we need to build N sub-NSCs of M-length ($M < N$) and explore the chain-wise statistical information to model the data distribution. Intuitively, the length should relate to the size of the dataset. However, the longer the sub-NSC, the more expensive the computation. Moreover, too short sub-NSC provides so limited information and makes the similarity matrix sparse to some extent, and so it cannot reflect the data distribution well. As shown in Fig. 4, the longer the length M, the better performance is obtained. When the length arrives at certain value, the performance tends to be stable. Based on this observation, we set M to 300 for the Corel dataset, and set $M = 600$ for the web dataset, because the size of the web dataset is about two times larger than that of the Corel dataset.

6.3.2. *Maximum length of linker for pairwise occurrence*

This is an important parameter in the proposed method, which is denoted as L. When L is zero, it means only the direct pairwise connections are taken into account as in AGAnn [16]. When L is more than zero, it indicates that the indirect connections are also considered besides the direct connections. The performance comparison with different values of L is shown in Fig. 5. We can find something interesting. The increasing value of L can improve the annotation performance. When L is around 4, the performance becomes stable, but when it is long enough ($L > 6$), the performance will be weakened. The reason is that too large or too small neighborhood inside a sub-NSC cannot truly reflect the relationship among the data. Too large L brings with much noise, and too small L does not supply sufficient information. Additionally, since the weights for certain connection, which are inversely related to the length of its linker as in Eq. (7), can leverage the negative effect by a large linker, the relatively stable performance can be obtained over a small range of values. Then, we set L to 4 in following experiments.

In addition, the other parameters are set by cross-validation as follows: $\lambda_1 = \lambda_2 = \lambda_3 = 1.00$, $\varepsilon_1 = 0.40$, $\varepsilon_2 = \varepsilon_3 = 0.30$, and $\alpha = \beta = 0.25$.

6.4. *Comparison among word correlations*

To ensure good semantic relevance of the candidate annotations for each image, we propose to perform the word-based graph learning to refine them. In order to evaluate the proposed word-based graph learning completely, we make a comparison among different word correlations on the Corel dataset, in which the WordNet-based correlation (WNC) [25], the word co-occurrence in the training set (WC), the content-based correlation in the web context (CC) and the visual consistence-based correlation in the web context (VC) are considered. When the candidate annotations are obtained by the proposed sub-NSC-based graph learning method (denoted as GLM + Sub-NSC), the comparison among these different word correlations are listed in Table 1. From the experimental results, we can find some useful observations. First, WC gains obvious improvement on the measure of the NumWords, while it losses a little on the average precision. This demonstrates that the method is capable of connecting more words through the statistical information, but the connections cannot ensure the relatedness on the semantic level. Second, CC and VC achieve more improvements synthetically and the later seems to be better. This is because that the two visual-based correlations by the web searcher provide the word correlations from a more extensive and reasonable level. Additionally, VC is estimated by an adaptive measure, i.e., DPV, so it is much robust to the web noise. Third, WNC shows the worst performance and even a negative role through the annotation refinement. There are 49 out of 371 words that either do not exist in the WordNet lexicon or have no available relations to the other words in the WordNet structure. Thus the sparse relation largely weakens the effect of this measure. Finally, the combination of (WC–CC–VC) achieves the best performance. It shares the advantages from each single correlation and presents a relatively precise and comprehensive measurement.

We also use the MBRM [9] to obtain the candidate annotations, and further test the performance of these word correlations. The results are reported in Table 2. We can see that the combined correlation (WC + CC + VC) still achieves the better performance.

Table 1
Performance comparison on word correlation: based on GLM + Sub-NSC

Model	Precision	Recall	NumWords
GLM + Sub-NSC	0.242	0.267	125
+WNC	0.218	0.249	117
+WC	0.231	0.285	129
+CC	0.243	0.278	127
+VC	0.246	0.283	127
+WC–CC–VC	0.253	0.291	130

Table 2
Performance comparison on word correlation: based on MBRM

Model	Precision	Recall	NumWords
MBRM	0.192	0.231	119
+WNC	0.170	0.219	115
+WC	0.186	0.243	125
+CC	0.195	0.239	124
+VC	0.208	0.238	122
+WC–CC–VC	0.221	0.259	125

6.5. Evaluation on Corel dataset

In the subsection, we use the Corel dataset to evaluate the proposed method based on the two-phrases graph learning. For simplicity, we denote it as TGLM in the following.

Fig. 6 shows the experimental results, where the recall and precision are average over 260 words. In order to further evaluate the performance of the proposed method, we also investigate the performance of some related methods: TM [6], CRM [8], MBRM [9], original graph learning model (GLM) [15], GLM + Sub-NSC (only using the sub-NSC-based graph learning) and GLM + Word (using the combined word correlation of 'WC-CC-VC' to refine the annotation results derived from GLM), AGAnn [16] (using NSC-based graph as in Section 4.2 and another type of combined word correlation-'WNC-WC'), and SML [11].

From Fig. 6, we can draw the following conclusions. First, all the GLM-based methods outperform the others, in which TGLM achieves the best performance. Specifically, compared with MBRM, TGLM achieves a gain of 25.9% in recall and 31.8% in precision. Second, both extensions for GLM with the Sub-NSC method (GLM+Sub-NSC)

Table 3
Retrieval performance comparison on the Corel dataset

Model	260 Words	179 Words	Words (recall > 0)
CRM	0.249	0.265	0.395
MBRM	0.267	0.292	0.443
SML	0.313	–	0.492
AGAnn	0.271	0.327	0.489
TGLM	0.293	0.354	0.521

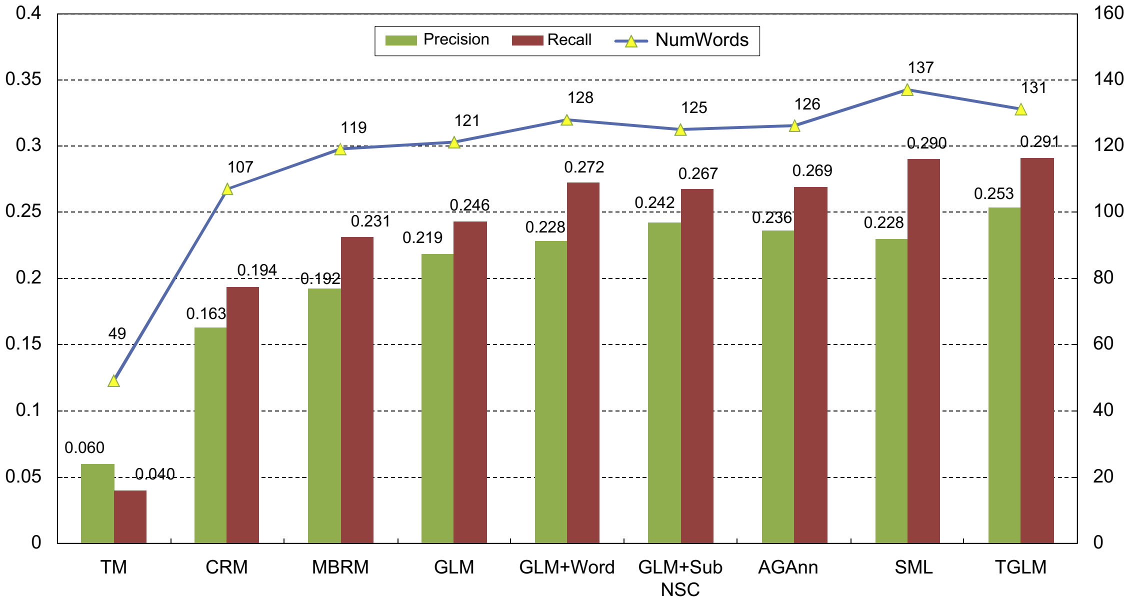


Fig. 6. Performance comparison on Corel dataset: the bars present average precision and recall according to the left axis and the line denotes NumWords according to the right axis.

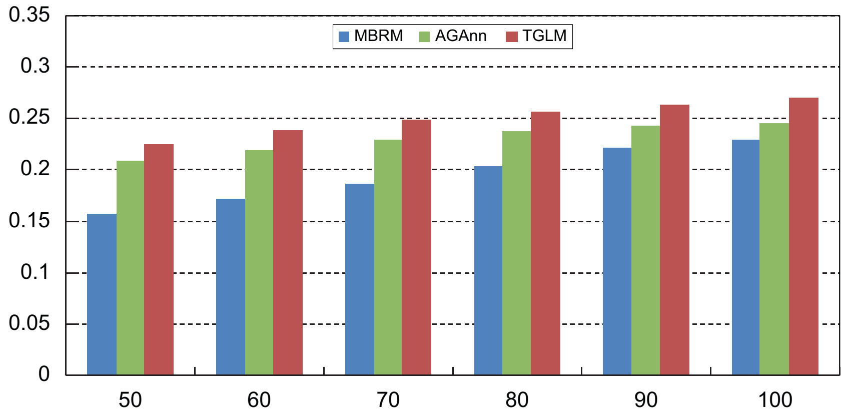


Fig. 7. Average precision comparison among MBRM, AGAnn and TGLM on Web dataset: the horizontal axis denotes the percentage of annotated images in the training set.

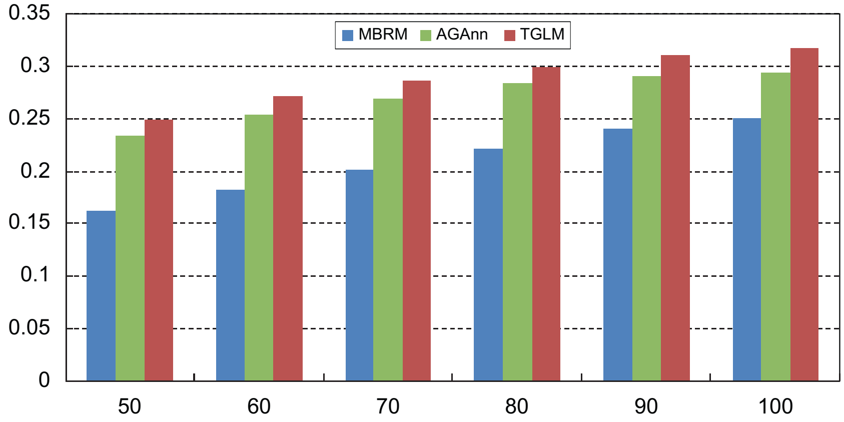


Fig. 8. Average recall comparison among MBRM, AGAnn and TGLM on web dataset: the horizontal axis denotes the percentage of annotated images in the training set.

and the word correlation (GLM+Word) achieve significant improvements. It indicates that the Sub-NSC-based similarity graph can better describe the distribution and the annotation refinement using the combined word correlation also brings some improvement, especially on the NumWords. Third, TGLM outperforms AGAnn in all the evaluation metrics, because TGLM benefits from the improved construction of the image-based graph, the modified initial labeling matrix, and the more comprehensive word correlation. Finally, TGLM and SML achieve the comparable performance. As a state-of-the-art method, SML gains on NumWords, but loses a lot on precision, when compared with our methods.

The ranking order provided with image annotation is very important for image retrieval in that the users would like to find the most related images against their queries as quick as possible. To evaluate the ranking order, we analyze results with single-word queries. That is, given one query word, the system will return all the images that are annotated with the query word, and rank them according to the probabilities of the word given these images. Then a metric called mean average precision (MAP) [8] is used to evaluate the retrieval performance. Average precision is the average of precisions at the ranks where relevant items occur, and MAP is given by averaging over all the queries. Here the relevance means that the ground-truth annotation of the image contains the query word.

In Table 3, we present the values of MAP on all 260 words and on the subset of 179 words that occur at least twice in the testing dataset. The MAPs on the words with non-zero recall are also listed. The results show that TGLM obtains the best performance. It demonstrates that the ranking order obtained from TGLM is preferable.

Table 4
Retrieval performance comparison on the web dataset

Model	826 Words	676 Words	Words (recall > 0)
MBRM	0.199	0.235	0.325
AGAnn	0.251	0.272	0.398
TGLM	0.263	0.321	0.426

6.6. Evaluation on the web image dataset

Currently, there exist almost infinite web images and they usually have extensive semantics and large variation on visual content. Thus, the less requirement on the labeling data and robust modeling for data distribution become important for the web image annotation. Here, we use our collected web dataset to test the performance of the proposed method.

In the experiment, we randomly select the labeled images from the training set with different percentages. Figs. 7 and 8 illustrate the comparisons among TGLM, AGAnn, and MBRM. We can see that TGLM achieves the best performance, while MBRM obtains the worst performance. Especially, MBRM is more sensitive to the varying percentage of the annotated images, while both the graph-based methods can work well even given less labeled images. Compared to AGAnn, TGLM gets the average gain on precisions with the different labeled percentage is 9.7%, and the average gain on recalls is 7.5%. The predominance of TGLM further demonstrates the effect of the proposed improvements in this paper.

We also compare the retrieval performance on the web dataset (8026 training images and 1000 testing images). Table 4 reports the comparisons of MAPs averaged on all the 826 words, the 676 words occurring at least twice, and the words with non-zero recall, respectively. Similarly, the best result is achieved by the proposed TGLM.

7. Conclusions

In this paper, we developed a graph learning framework for image annotation, which contained the image-based graph learning and the word-based graph learning. The image-based graph learning generated the candidate annotations, and the word-based graph learning further refined the candidate annotations to output the final results. To better capture the complex distribution of the image data, the NSC-based technique was proposed to construct the image-based graph. The word-based graph learning was performed by exploring three kinds of word correlations. One is the word co-occurrence in the training set, and the other two are derived from the web context. Extensive experiments on the Corel dataset and the web image dataset demonstrated the effectiveness of the proposed method.

Our goal is to auto-annotate a huge amount of images effectively and efficiently while the required labeled information is as less as possible. Hence in the future, we will work on mining more relevant semantic information from the web pages and building more reliable graph for the web image annotation.

Acknowledgments

The research was supported by National Natural Science Foundation of China (60605004, 60675003, 60723005) and National 863 Project (2006AA01Z315, 2006AA01Z117).

References

[1] D. Tao, X. Li, S. Maybank, Negative samples analysis in relevance feedback, IEEE Trans. Knowledge Data Eng. 19 (4) (2007) 568–580.

[2] D. Tao, X. Tang, X. Li, X. Wu, Asymmetric bagging and random subspace for support vector machine-based relevance feedback in image retrieval, PAMI 28 (7) (2006) 1088–1099.

[3] J. Li, J. Wang, Automatic linguistic indexing of pictures by a statistical modeling approach, PAMI 25 (19) (2003) 1075–1088.

[4] C. Cusano, G. Ciocca, R. Schettini, Image annotation using svm, in: Proceedings of Internet Imaging IV, SPIE 5304, vol. 5304, 2003, pp. 330–338.

[5] E. Chang, G. Kingshy, G. Sychay, G. Wu, Cbsa: content-based soft annotation for multimodal image retrieval using bayes point machines, IEEE Trans. CSVT 13 (1) (2003) 26–38.

[6] P. Duygulu, K. Barnard, J. de Freitas, D.A. Forsyth, Object recognition as machine translation: learning a lexicon for a fixed image vocabulary, in: Proceedings of Seventh European Conference on Computer Vision, UK, 2002, pp. 97–112.

[7] J. Jeon, V. Lavrenko, R. Manmatha, Automatic image annotation and retrieval using cross-media relevance models, in: ACM SIGIR, 2003, pp. 119–126.

[8] V. Lavrenko, R. Manmatha, J. Jeon, A model for learning the semantics of pictures, in: Proceedings of Advance in Neutral Information Processing, 2003.

[9] S.L. Feng, R. Manmatha, V. Lavrenko, Multiple bernoulli relevance models for image and video annotation, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004, pp. 1002–1009.

[10] D. Blei, M. Jordan, Modeling annotated data, in: Proceedings of 26th International Conference on Research and Development in Information Retrieval, 2003.

[11] G. Carneiro, A.B. Chan, P.J. Moreno, N. Vasconcelos, Supervised learning of semantic classes for image annotation and retrieval, PAMI 29 (3) (2007) 394–410.

[12] R. Jin, J. Chai, L. Si, Effective automatic image annotation via a coherent language model and active learning, in: Proceedings of 12th Annual ACM International Conference on Multimedia, 2004, pp. 892–899.

[13] F. Kang, R. Jin, R. Sukthankar, Correlated label propagation with application to multi-label learning, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2006, pp. 1719–1726.

[14] Y. Jin, L. Khan, L. Wang, Image annotations by combining multiple evidence wordnet, in: Proceedings of the 13th Annual ACM International Conference on Multimedia, 2005, pp. 706–715.

[15] H. Tong, J. He, M. Li, W. Ma, H.J. Zhang, C. Zhang, Manifold-ranking based keyword propagation for image retrieval, EURASIP J. Appl. Signal Process. Spec. Issue Inf. Min. Multimedia Database 21 (2006) 1–10.

[16] J. Liu, M.J. Li, W. Ma, Q. Liu, H.Q. Lu, An adaptive graph model for automatic image annotation, in: Eighth ACM International Workshop on Multimedia Information Retrieval, 2006, pp. 61–70.

[17] D. Zhou, O. Bousquet, T. Lal, J. Weston, B. Scholkopf, Ranking on data manifolds, in: Proceedings of 18th Annual Conference on Neural Information Processing System, 2003, pp. 169–176.

[18] M. Meila, J. Shi, A random walks view of spectral segmentation, in: Eighth International Workshop on Artificial Intelligence and Statistic, 2001.

[19] H.C. Thomas, C.E. Leiserson, R.L. Rivest, C. Stein, Minimum spanning trees, Introduction to Algorithms, MIT Press and McGraw-Hill, 2001, pp. 561–579 (Chapter 23).

[20] H. Tong, J. He, M. Li, C. Zhang, M.W.Y., Graph based multi-modality learning, in: ACM MM, 2005, pp. 862–871.

[21] D. Cai, S. Yu, J. Wen, W. Ma, Vips: a vision-based page segmentation algorithm, in: Microsoft Technical Report (MSR-TR-2003-79), 2003.

[22] J. Huang, S. Kumar, M. Mitra, W. Zhu, R. Zabih, Image indexing using color correlogram, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1997, pp. 762–768.

[23] F. Liu, R. Picard, Periodicity, directionality, and randomness: wold features for image modeling and retrieval, PAMI 18 (1996) 722–733.

[24] S.G. Mallat, A theory for multi-resolution signal decomposition: the wavelet representation, PAMI 11 (1989) 674–693.

[25] J. Jiang, D. Conrath, Semantic similarity based on corpus statistics and lexical taxonomy, in: Proceedings on International Conference on Research in Computational Linguistics, 1997.

About the Author–JING LIU is an assistant professor in National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences. She received her B.E. degree in 2001 and M.E. degree in 2004 from Shandong University, and her Ph.D. degree from Institute of Automation, Chinese Academy of Sciences in 2008. Her research interests include machine learning, image content analysis and classification, multimedia information indexing and retrieval, etc.

About the Author–MINGJING LI is a researcher in Microsoft Research China. He received his B.E. in electrical engineering from the University of Science and Technology of China in 1989, and his Ph.D. from the Institute of Automation, Chinese Academy of Sciences in 1995. His research interests include multimedia information indexing and retrieval, machine learning, image content analysis and classification, Chinese handwriting recognition, etc.

About the Author–QINGSHAN LIU is an associate professor in National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences. He received M.E. from Automatic Control from South-East University in 2000, and his Ph.D. degree from Institute of Automation, Chinese Academy of Sciences in 2003. His interests are face analysis, image and video analysis, machine learning, computer vision, mobile multimedia, and web image search, etc.

About the Author–HANQING LU is a professor in National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences. He received his B.E. degree in 1982 and his M.E. degree in 1985 from Harbin Institute of Technology, and Ph.D. degree from Huazhong University of Sciences and Technology in 1992. His research interests include image and video analysis, medical image processing, object recognition, etc.

About the Author–SONGDE MA received his B.S. in Automatic Control from the TsingHua University in 1968, Ph.D. degree in University of Paris in 1983 and "Doctorat d'Etat es Science" in France in 1986 in image processing and computer vision. He was an invited researcher in Computer Vision Laboratory in the University of Maryland in USA in 1983. From 1984 to 1986, he was a researcher in Robot Vision Laboratory in INRIA, France. Prof. MA was a member of the Expert Committee of the National High Technology Program and the chief scientist of the Project "Image, Voice and Natural Language Understanding" of the National Fundamental Research Program. His research interests include computer vision, image understanding and searching, robotics and computer graphics, etc.