Text-Based Image Retrieval using Progressive Multi-Instance Learning

Wen Li   Lixin Duan   Dong Xu   Ivor Wai-Hung Tsang
School of Computer Engineering
Nanyang Technological University, Singapore
WLI1@e.ntu.edu.sg, {S080003,DongXu,IvorTsang}@ntu.edu.sg

Abstract

*Relevant and irrelevant images collected from the Web (*e.g.*, Flickr.com) have been employed as loosely labeled training data for image categorization and retrieval. In this work, we propose a new approach to learn a robust classifier for text-based image retrieval (TBIR) using relevant and irrelevant training web images, in which we explicitly handle noise in the loose labels of training images. Specifically, we first partition the relevant and irrelevant training web images into clusters. By treating each cluster as a "bag" and the images in each bag as "instances", we formulate this task as a multi-instance learning problem with constrained positive bags, in which each positive bag contains at least a portion of positive instances. We present a new algorithm called MIL-CPB to effectively exploit such constraints on positive bags and predict the labels of test instances (images). Observing that the constraints on positive bags may not always be satisfied in our application, we additionally propose a progressive scheme (referred to as Progressive MIL-CPB, or PMIL-CPB) to further improve the retrieval performance, in which we iteratively partition the top-ranked training web images from the current MIL-CPB classifier to construct more confident positive "bags" and then add these new "bags" as training data to learn the subsequent MIL-CPB classifiers. Comprehensive experiments on two challenging real-world web image data sets demonstrate the effectiveness of our approach.*

1. Introduction

With the rapid adoption of digital cameras, we have witnessed an explosive growth of digital photos. Everyday, a tremendous amount of images together with rich contextual information (*e.g.*, tags, categories and captions) are posted to the Web. There is an increasing interest in developing new systems to help users to retrieve web images. While a large number of content-based image retrieval (CBIR) algorithms (see the recent survey in [6]) have been developed over the past decades, it is more desirable and practical for a user to retrieve photos from the database using textual queries (*i.e.*, tags) as opposed to example images. However, this is a non-trivial task because the performance of text-based image retrieval (TBIR) systems may be significantly degraded when the associated contextual information is noisy and incomplete [3].

Meanwhile, the massive and valuable web data (*i.e.*, web images and the associated rich textual descriptions) have been also exploited for various computer vision tasks. For instance, Torralba *et al.* [20] used a kNN classifier for several object/scene classification tasks by leveraging 80 million loosely labeled tiny images. Wang *et al.* [23] also employed millions of web data for image annotation. Given the massive web data, relevant and irrelevant web images can be readily obtained by using keyword based search. In [8, 9, 10, 19, 21], such relevant and irrelevant web images were used as the loosely labeled training data for image categorization and retrieval (see Section 2 for more details).

In this work, we explicitly handle noise in the loose labels of training images. We first respectively partition the relevant web images and the randomly sampled irrelevant web images into clusters. Following [8, 21], we formulate our task as a multi-instance learning (MIL) problem by treating each cluster as a "bag" and the images in each bag as "instances". Observing that most of the associated textual descriptions are more or less related to the content of web images, we assume that *each positive bag has at least a portion of positive instances*, as suggested in [8]. Note that this new assumption on positive bags is more suitable for our TBIR task when compared with the traditional assumption in most existing MIL work (*e.g.*, [21]) that one positive bag contains at least one positive instance. However, we still use the traditional assumption on negative bags (*i.e.*, each negative bag does not contain any positive instances), as the experiments in [8] show that this assumption generally holds for negative bags. To effectively solve the MIL problem with *constrained positive bags* and predict the labels of instances (images), we present an algorithm called MIL-CPB here, which is a simplified version of the method[1] in [8].

[1] The algorithm in [8] is called generalized multi-instance SVM (GMI-SVM), because it can handle the ambiguities on the instance labels in both

Since the labels of relevant training web images are quite noisy, the constraints on positive bags may not always be satisfied in our application. To cope with this issue, in this work we propose a progressive scheme (referred to as Progressive MIL-CPB, or PMIL-CPB) to refine the quality of the positive bags. Specifically, we construct more confident "bags" from the top-ranked training web images using the current MIL-CPB classifier. After that, these new "bags" are added up to the training data in order to learn the subsequent MIL-CPB classifiers in an iterative fashion. In Section 4, we conduct comprehensive experiments using the challenging NUS-WIDE and Google data sets and the results clearly demonstrate the effectiveness of our approach.

2. Related Work

Loosely labeled training web data has been used for image categorization and retrieval. Fergus *et al*. [9] extended pLSA to handle the large intra-class variability among images returned by the image search engine. Schroff *et al*. [19] developed a multi-modal approach to collect a large image data set, in which different regularization parameters were used for the positive and negative training instances when learning the visual feature based SVM classifier. Meanwhile, researchers also proposed new tag re-ranking methods [3, 14], which are specifically designed for *Flickr* photos that are initially associated with noisy tags. In contrast to [3, 9, 14, 19], we formulate a new MIL problem with constrained positive bags to solve the learning problem with ambiguity in the training samples.

MIL methods [4, 11] have been successfully used in region-based image retrieval. In these applications, images are considered as bags; while regions in the images are considered as instances. In this work, we treat one image cluster as one "bag" and the images inside the bag as "instances". Moreover, the existing MIL approaches (*e.g.*, Diverse Density (DD) [17], EMDD [24], Citation kNN [22]) that can predict the labels of instances are computationally expensive, making them unsuitable for large-scale TBIR applications. In contrast, our method can also efficiently predict the labels of instances (images).

Our work is quite related to [21], in which a variant (referred to as WsMIL here) of Sparse MIL (sMIL) [2] was proposed for image re-ranking and categorization by iteratively updating the weighted mean of instances in the constraint. However, our work is different from [21] in two aspects: 1) the work in [21] used the traditional assumption that one positive bag contains at least one positive instance. We argue that our new assumption that a positive bag contains at least a portion of positive instances is more suitable for this task as most of the associated textual descriptions are more or less related to the content of web images [8]; 2) the work in [21] used pre-defined and fixed "bags" when learning the classifiers and the high-quality positive bags cannot always be obtained due to the noisy web data, while in our work we progressively improve the quality of the positive bags and iteratively add new "bags" to the training data to learn a more robust classifier. Our experiments also demonstrate that PMIL-CPB is better than WsMIL and sMIL for the TBIR task.

Our work is also related to the recent bag-based reranking framework [8], which is specifically designed for improving the web image re-ranking performance. In contrast to [8], this work focuses on how to explicitly handle noise in the loose labels of training images, and our proposed approach PMIL-CPB can effectively handle the challenging case when the constraint on a positive bag is not satisfied in [8].

3. TBIR with Multi-Instance Learning

Given a textual query q, the recent TBIR system in [15] first automatically collects relevant web images whose surrounding texts contain the word q as well as irrelevant web images whose surrounding texts do not contain the word q. Then, these relevant (*resp.*, irrelevant) web images were directly used as positive (*resp.*, negative) training data to learn classifiers (*e.g.*, SVM), and the database images are finally ranked according to their decision values from the learned classifier. We observe that a large portion of relevant web images returned by the image search engine and the method suggested in [15] are accompanied by incorrect labels because the web images are generally associated with noisy textual descriptions (*e.g.*, tags, categories and captions). Therefore, the learned classifiers based on these loosely labeled web images are not robust, which significantly degrades image retrieval performances.

To effectively utilize the noisy web data for image reranking and categorization, Vijayanarasimhan and Grauman [21] proposed a variant (called WsMIL here) of the existing MIL method sMIL [2] to learn classifiers by iteratively updating the weights of the instances in each positive bag. However, in their work the positive bags are assumed to be constructed by using image search engines in multiple languages. Moreover, their approach follows the traditional MIL assumption [1, 2, 4, 7, 13, 17, 22, 24] that a positive bag contains at least one positive instance.

In this work, we first respectively partition the relevant web images and the randomly sampled irrelevant web images into clusters of the same size. Following [8, 21], we also formulate our task as a multi-instance learning (MIL) problem by treating each cluster as a "bag" and the images in each bag as "instances". Based on the observation that negative bags generally do not contain any positive instances and inspired by [8], we formulate our task as a MIL

positive and negative bags. The resultant optimization problem in MIL-CPB is easier in this work, because we only need to handle the ambiguities in positive bags.

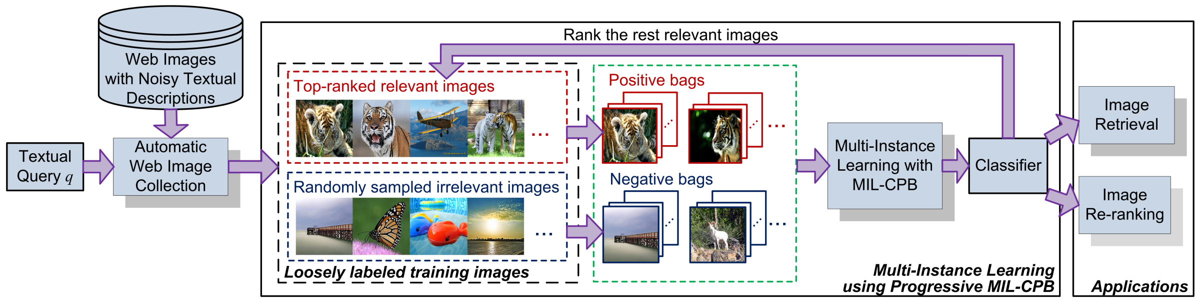
Web Images with Noisy Textual Descriptions
Textual Query q
Automatic Web Image Collection
Rank the rest relevant images
Top-ranked relevant images
Randomly sampled irrelevant images
Loosely labeled training images
Positive bags
Negative bags
Multi-Instance Learning with MIL-CPB
Classifier
Multi-Instance Learning using Progressive MIL-CPB
Image Retrieval
Image Re-ranking
Applications

Figure 1. Flowchart of the proposed approach.

problem with *constrained positive bags*, in which *each positive bag contains at least a portion of positive instances*.

Let us denote each instance as $\mathbf{x}_i$ with its label $y_i \in \{\pm 1\}$, where $i = 1, \ldots, n$. We also represent the label of the bag B_I as $Y_I \in \{\pm 1\}$. The superscript $'$ denotes the transpose of a vector or matrix. Moreover, $\mathbf{I}$ denotes the identity matrix and $\mathbf{0}, \mathbf{1} \in \Re^n$ denote the column vectors of all zeros and ones, respectively. We also define $\mathbf{P} \odot \mathbf{Q}$ as the element-wise product between two matrices $\mathbf{P}$ and $\mathbf{Q}$. The inequality $\mathbf{v} = [v_1, v_2, \ldots, v_j]' \geq \mathbf{0}$ means that $v_i \geq 0$ for $i = 1, \ldots, j$. The new MIL constraints can be interpreted as follows:

$$\begin{aligned} \sum_{i:\, \mathbf{x}_i \in B_I} \frac{y_i + 1}{2} \geq \sigma |B_I| \quad & \text{for} \quad Y_I = 1, \\ \text{and} \quad y_i = -1 \quad & \text{for} \quad Y_I = -1. \end{aligned} \tag{1}$$

Note that our new MIL constraints differ from the traditional MIL assumption in that positive bags are provided with more information, namely, each positive bag contains at least a portion (*i.e.*, σ) of positive instances. Moreover, the traditional MIL assumption is just a special case of our new assumption when setting $\sigma = 1/|B_I|$.

To effectively exploit the constraints on positive bags, we first present an algorithm called MIL-CPB in Section 3.1, which is a simplified version of the method in [8]. We further develop a progressive scheme called Progressive MIL-CPB (PMIL-CPB) in Section 3.2. The overall flowchart of our proposed approach is shown in Fig. 1.

3.1. MIL with constrained positive bags (MIL-CPB)

In our TBIR application, we rank the database images based on their decision values from the decision function (*i.e.*, classifier). And the decision function is to be learned with the instances from the training bags. Let us define the vector of instance labels as $\mathbf{y} = [y_1, \ldots, y_n]'$, and denote the feasible set of $\mathbf{y}$ as $\mathcal{Y} = \{\mathbf{y} | y_i \in \{\pm 1\}, \mathbf{y}$ satisfies the conditions in (1)$\}$. Following [8], we employ the formulation of Lagrangian SVM with the square bias penalty b^2 and the square hinge loss function. We propose to learn the decision function $f(\mathbf{x}) = \mathbf{w}'\phi(\mathbf{x}) + b$ in MIL-CPB by minimizing the following structural risk functional:

$$\begin{aligned} \min_{\mathbf{y} \in \mathcal{Y}, \mathbf{w}, b, \rho, \xi_i} \quad & \frac{1}{2}\left(\|\mathbf{w}\|^2 + b^2 + C \sum_{i=1}^{n} \xi_i^2 \right) - \rho, \\ \text{s.t.} \quad & y_i(\mathbf{w}'\phi(\mathbf{x}_i) + b) \geq \rho - \xi_i, \; i = 1, \ldots, n, \end{aligned} \tag{2}$$

where $\phi : \mathbf{x} \rightarrow \phi(\mathbf{x})$ is a mapping function that maps $\mathbf{x}$ from the original space into a high dimensional space $\phi(\mathbf{x})$, $C > 0$ is a regularization parameter, ξ_i's are slack variables and $2\rho/\|\mathbf{w}\|$ defines the separation between the two opposite classes. It is worth noting that the constraints in (2) are defined for all training instances, which is intrinsically different from the existing work sMIL [2] and WsMIL [21] where the constraint for each positive bag is defined only on the (weighted) mean of instances. One could expect that in our work there would be more support vectors (defined on the instances) to represent a more precise decision boundary, and in return the learned classifier could achieve better prediction performance for instances.

Following [8], we solve (2) in its dual form listed below by introducing a dual variable α_i for each constraint in (2):

$$\min_{\mathbf{y} \in \mathcal{Y}} \max_{\boldsymbol{\alpha} \in \mathcal{A}} -\frac{1}{2} \boldsymbol{\alpha}' \left(\widetilde{\mathbf{K}} \odot \mathbf{y}\mathbf{y}' + \frac{1}{C}\mathbf{I} \right) \boldsymbol{\alpha}, \tag{3}$$

where $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_n]'$, $\mathcal{A} = \{\boldsymbol{\alpha} | \boldsymbol{\alpha} \geq \mathbf{0}, \mathbf{1}'\boldsymbol{\alpha} = 1\}$ is the feasible set of $\boldsymbol{\alpha}$, $\widetilde{\mathbf{K}} = \mathbf{K} + \mathbf{1}\mathbf{1}'$ and $\mathbf{K} = [k(\mathbf{x}_i, \mathbf{x}_j)] \in \Re^{n \times n}$ is the kernel matrix with the kernel function k deduced from the feature mapping $\phi(\cdot)$ (*i.e.*, $k(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)'\phi(\mathbf{x}_j)$). Note that (3) is an integer programming problem with respect to the instance labels y_i's, which is an NP-hard problem and therefore is computationally intractable. To efficiently solve the optimization problem in (3), we present Theorem 1:

Theorem 1. *The lower bound of the objective value of the mixed integer programming problem in (3) is the optimal objective value of the following optimization problem:*

$$\max_{\boldsymbol{\alpha} \in \mathcal{A}} \left\{ \max_{\theta} -\theta : \theta \geq \frac{1}{2} \boldsymbol{\alpha}' \left(\widetilde{\mathbf{K}} \odot \mathbf{y}^t {\mathbf{y}^t}' + \frac{1}{C}\mathbf{I} \right) \boldsymbol{\alpha}, \; \forall \mathbf{y}^t \in \mathcal{Y} \right\}, \tag{4}$$

where $\mathbf{y}^t$ is any feasible solution in $\mathcal{Y}$. And by replacing the inner optimization problem with respect to θ in (4) with its

Algorithm 1 Cutting-plane algorithm for MIL-CPB

```
1: Initialize $y_i = Y_I$ for $\mathbf{x}_i \in B_I$ as $\mathbf{y}^1$, and set $\mathcal{C} = \{\mathbf{y}^1\}$;
2: Use SimpleMKL [18] to solve $\boldsymbol{\alpha}$ and $\mathbf{d}$ in (5) with $\mathcal{C}$;
3: Use $\boldsymbol{\alpha}$ to select a violated $\mathbf{y}^t$ and set $\mathcal{C} = \mathbf{y}^t \cup \mathcal{C}$;
4: Repeat Steps 2 and 3 until convergence.
```

dual form, the optimization problem in (4) is equivalent to the following problem:

$$\min_{\mathbf{d}\in\mathcal{D}} \max_{\boldsymbol{\alpha}\in\mathcal{A}} -\frac{1}{2}\boldsymbol{\alpha}' \left(\sum_{t:\mathbf{y}^t\in\mathcal{Y}} d_t \widetilde{\mathbf{K}} \odot \mathbf{y}^t {\mathbf{y}^t}' + \frac{1}{C}\mathbf{I} \right) \boldsymbol{\alpha}. \quad (5)$$

where the d_t's are dual variables for the inner optimization problem in (4), $\mathbf{d}$ is the vector of d_t's and $\mathcal{D} = \{\mathbf{d}|\mathbf{d} \geq \mathbf{0}, \mathbf{1}'\mathbf{d} = 1\}$ is the feasible set of $\mathbf{d}$.

Proof. The proof is presented in [8]. □

The optimization problem (5) can be deemed as a Multiple Kernel Learning (MKL) problem [18]. Note that the number of base kernels $\widetilde{\mathbf{K}} \odot \mathbf{y}^t {\mathbf{y}^t}'$ in (5) is exponential in size. As a result, the existing state-of-the-art multiple kernel learning methods such as SimpleMKL [18] cannot be used to solve (5), due to the heavy computational cost. To this end, we employ the cutting-plane algorithm [12] to find a subset $\mathcal{C} \in \mathcal{Y}$ of the constraints that can well approximate the original optimization problem. To find the optimal solution, we alternately update the dual variable $\boldsymbol{\alpha}$ and the linear combination coefficient $\mathbf{d}$ during the optimization procedure. Algorithm 1 lists the detailed optimization procedure for MIL-CPB. After obtaining the optimal $\boldsymbol{\alpha}$ and $\mathbf{d}$, we can derive the decision function as follows:

$$f(\mathbf{x}) = \sum_{i:\alpha_i\neq 0} \alpha_i \tilde{y}_i \tilde{k}(\mathbf{x}, \mathbf{x}_i), \quad (6)$$

where $\tilde{y}_i = \sum_{t:\mathbf{y}^t\in\mathcal{C}} d_t y_i^t$ and $\tilde{k}(\mathbf{x}, \mathbf{x}_i) = k(\mathbf{x}, \mathbf{x}_i) + 1$.

Finding the most violated constraint for $\mathbf{y}^t$ in Step 3 of Algorithm 1 is problem-specific, and it is also the most challenging part in the cutting-plane algorithm. According to (4), *the most violated* $\mathbf{y}^t$ is essentially the solution to the optimization problem as $\max_{\mathbf{y}\in\mathcal{Y}} \boldsymbol{\alpha}'(\widetilde{\mathbf{K}} \odot \mathbf{y}\mathbf{y}')\boldsymbol{\alpha}$. We employ a similar process as in [8] to solve this integer optimization problem by enumerating all possible candidates of $\mathbf{y}^t$. Rather than conducting the enumeration for all instances in positive and negative bags as in [8], we only need to enumerate the possible labeling candidates of the instances in positive bags in this work because all instances in the negative bags are assumed to be negative. In the experiments, we set the portion (*i.e.*, σ) of positive instances as 0.6, and we also enforce each bag to contain 15 instances in order to make this problem solvable.

Algorithm 2 Progressive MIL-CPB (PMIL-CPB)

```
1: Initialize $N_{\text{init}}$ positive bags and $N_{\text{init}}$ negative bags for
   the given textual query $q$;
2: Train the MIL-CPB classifier and rank the remaining
   relevant images from the training data set;
3: Construct $N_{\text{inc}}$ positive bags from the top-ranked relevant
   training images and $N_{\text{inc}}$ negative bags from the
   randomly sampled remaining irrelevant training images;
4: Repeat steps 2 and 3 until the maximum number of iterations is reached.
```

3.2. Progressive MIL-CPB (PMIL-CPB)

In our TBIR task, we use the relevant training images to construct positive bags. However, the labels of relevant training web images are usually quite noisy, so the constraints on positive bags in our MIL-CPB algorithm may not always be satisfied. We therefore develop a new scheme called Progressive MIL-CPB to progressively select more confident positive bags.

Specifically, given a textual query q, we first uniformly partition the top-ranked relevant web images and the randomly sampled irrelevant web images into $2N_{\text{init}}$ clusters (*i.e.*, N_{init} positive bags and N_{init} negative bags). The top-ranked relevant web images can be randomly chosen from the images returned by the image search engine. For the *Flickr* images associated with noisy tags, we define the following ranking score [3] for each relevant image $\mathbf{x}$ to collect the top-ranked relevant images:

$$r(\mathbf{x}) = -\tau + \frac{1}{\delta}, \quad (7)$$

where δ is the total number of tags in image $\mathbf{x}$, and τ is the rank position of the textual query q in the tag list of image $\mathbf{x}$.

The $2N_{\text{init}}$ bags are used as training data to learn an initial MIL-CPB classifier (see (6)). Then, we rank the relevant images in the training data set according to their decision values from the initial instance-level decision function, and we choose the $15N_{\text{inc}}$ top-ranked images to construct N_{inc} positive bags with each bag containing 15 instances. We additionally choose N_{inc} negative bags by randomly sampling $15N_{\text{inc}}$ irrelevant images. The $2N_{\text{inc}}$ new bags are added up to the training data to learn an updated MIL-CPB classifier. We iteratively partition the top-ranked training images from the current MIL-CPB classifier to construct more confident "bags" and then add these new "bags" to the training data to learn the subsequent MIL-CPB classifier until the maximum number of iterations is reached. Note that the instances that are already in the positive/negative training bags will not be selected in the subsequent iterations. The overall process is summarized in Algorithm 2.

4. Experiments

In this work, we first test our PMIL-CPB on the NUS-WIDE data set [5] for the large-scale image retrieval task, and then we compare PMIL-CPB with [19, 21] on the Google data set [9] for image re-ranking. We also report the results of MIL-CPB, which is a simplified version of the method in [8] without conducting the progressive bag selection process discussed in Section 3.2.

4.1. Image retrieval on the NUS-WIDE data set

The NUS-WIDE data set [5] consists of 269,648 images collected from *Flickr.com* and their ground-truth annotations for 81 concepts. All images in the NUS-WIDE data set are associated with noisy tags. Therefore, we represent each image by using its visual and textual features. Similar to [5], we extract three types of global visual features: Grid Color Moment (225 dim), Wavelet Texture (128 dim) and Edge Direction Histogram (73 dim). For each image, we further concatenate the three types of visual features to form a 426-dimensional feature vector. PCA is further employed to reduce the dimension of each feature vector by projecting it into a 119-dimensional visual feature space, which preserves 90% of the energy. We also extract the textual feature for each image from its associated tags. Specifically, we have removed the stop words (*e.g.*, "a", "the") and converted the remaining tags into their prototypes. We select the top-200 words with the highest frequency in the training set of NUS-WIDE as the vocabulary. And a 200-dimensional term-frequency feature is then extracted as the textual feature for each image. Above all, for each image, we further concatenate its visual feature $\mathbf{v}$ and its textual feature $\mathbf{t}$ to form the final feature vector representation $\mathbf{x}$, namely, $\mathbf{x} = [\lambda\mathbf{v}', \mathbf{t}']'$, where the parameter λ is empirically fixed as 0.1 in the experiments.

Considering that APR [7], EM-DD [17] and Citation kNN [22] are inefficient for this large-scale TBIR application, we focus on comparisons between our PMIL-CPB and the MIL methods mi-SVM[2] [1], Single Instance Learning SVM (SIL-SVM) [2], MILES [4], sMIL[2] and WsMIL [21]. Note that in SIL-SVM all the instances in positive (*resp.*, negative) bags are treated as positive (*resp.*, negative) samples. For MILES [4], we treat each test image as one test bag in this application, because it can only predict the labels of bags.

Table 1. MAPs (%) of all methods over 81 concepts on the NUS-WIDE data set. Each result in the table is the best among all results obtained by using different numbers (*i.e.*, N) of positive/negative training bags.

	SIL-SVM	mi-SVM	MILES	sMIL	WsMIL	MIL-CPB	PMIL-CPB
MAP	57.54	58.63	53.82	59.71	59.71	61.49	**63.26**

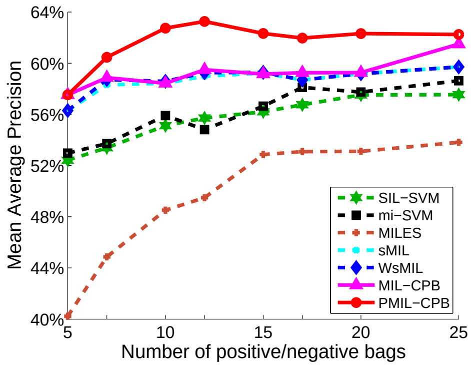

Figure 2. MAPs of all methods over 81 concepts on the NUS-WIDE data set with respect to different numbers (*i.e.*, N) of positive/negative bags.

For mi-SVM, SIL-SVM, MILES, sMIL, WsMIL and MIL-CPB, N positive training bags are constructed by using the top-ranked relevant images decided according to the ranking scores from (7), and N negative training bags are formed by randomly sampling the irrelevant images, where we set $N = 5, 7, 10, 12, 15, 17, 20$ and 25 in the experiments. For our PMIL-CPB, we use the same strategy to construct N_{init} positive training bags as well as N_{init} negative training bags, where $N_{\text{init}} = 5$. With the progressive approach (see Section 3.2), we iteratively add N_{inc} positive and N_{inc} negative training bags, where N_{inc} is set as $2, 3$ or 5 in order to fairly compare PMIL-CPB with other methods using the same number of training bags. Note that we fix the number of instances in each bag to 15 for all the methods. It is worth mentioning that ***the ground-truth labels of the training images are not used*** in the learning process for all methods.

For all methods, we train one-versus-all classifiers using the RBF kernel with the bandwidth parameter set as $1/A$, where A is the mean value of the square distances between training images. The training set consisting of 161,789 images. We evaluate the performances on the test set with 107,859 test images. For performance evaluation, we calculate the non-interpolated Average Precision (AP) [3] based on the 100 top-ranked images. We also define Mean Average Precision (MAP) as the mean of APs over all 81 concepts.

Results: For each method, Table 1 reports its best result among all results obtained by using different N. Fig. 2 shows the MAPs of all methods over 81 concepts using N positive and N negative training bags. From Fig. 2 and Table 1, we have the following observations:

1) MILES achieves the worst MAP in this application possibly because it cannot effectively predict the labels of instances. SIL-SVM generally performs worse than mi-SVM except the case when setting $N = 12$. The explanation is that mi-SVM can cope with the labeling ambiguities on positive bags to some extent while in SIL-SVM all the training instances in positive bags are used as positive samples.

2) sMIL and WsMIL both outperform SIL-SVM and mi-

[2] MI-SVM and mi-SVM achieve similar results in this work, so we only take mi-SVM as an example to report the results.

SVM, which demonstrates that the MIL methods sMIL and WsMIL are more suitable for this application. Moreover, sMIL and WsMIL achieve very similar performances on this data set, which indicates that the iterative update of the weights for the instances in positive bags cannot lead to significant performance improvement for the large-scale TBIR task.

3) The performances of most MIL methods generally increase when using more training bags. The best MAP of MIL-CPB is better than those of sMIL and WsMIL, which demonstrates the effectiveness of MIL-CPB for exploiting the new MIL constraints on the positive training bags.

4) Our progressive approach PMIL-CPB consistently achieves the best results, and it quickly reaches the best MAP by using only 12 positive/negative training bags and then its performance becomes stable. It demonstrates that PMIL-CPB is an effective method for exploiting the new MIL constraints on positive bags, and it also shows that it is beneficial to employ the progressive bag selection scheme to construct high-quality positive training bags.

Purity comparison: We experimentally investigate the *quality* of the training instances from the positive bags in PMIL-CPB by using the ground truth annotations of these images. Note that in (6) the predicted label $\tilde{y}_i$ of each training instance $\mathbf{x}_i$ is obtained by $\tilde{y}_i = \sum_{t:\mathbf{y}^t \in \mathcal{C}} d_t y_i^t$. If $\tilde{y}_i > 0.5$, we predict $\mathbf{x}_i$ as a positive instance; otherwise, $\mathbf{x}_i$ is predicted as a negative instance. For each concept, we define purity $p = m'/m$ as the evaluation metric to analyze the quality of the added relevant images, where m is the number of training instances (referred to as *positively predicted training instances*) that are predicted as positive instances by PMIL-CPB[3] and m' is the number of truly positive instances (according to the ground-truth annotations) among the m positively predicted training instances. For PMIL-CPB, we analyze the purity based on the classifier learned using 20 positive and 20 negative training bags.

For comparison, we also report the purity of the top-ranked m relevant training images from *Init_Ranking* in which the images are ranked according to the initial ranking scores from (7). In the experiment, we set $m = 50, 75, 100, 125, 150, 175$ and 200. The average purities of Init_Ranking and PMIL-CPB over 81 concepts on the NUS-WIDE data set are shown in Fig. 3. From Fig. 3, we observe that the average purity of PMIL-CPB is always much better than that of Init_Ranking, which demonstrates that PMIL-CPB can better cope with the noise in loose labels by effectively exploiting the new MIL constraints as well as using the progressive bag selection scheme. We also observe that

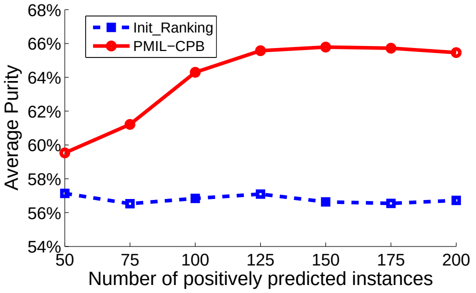

Figure 3. Average purities of Init_Ranking and PMIL-CPB over 81 concepts on the NUS-WIDE data set using different numbers (*i.e.*, m) of positively predicted instances.

the average purity of PMIL-CPB generally increases as the number of positively predicted instances increases. There are two explanations. On one hand, more confident "bags" that contain more pure positive instances are progressively added after each iteration. On the other hand, the learned MIL-CPB classifiers using more confident "bags" also become more robust. We have similar observations with different numbers (*i.e.*, N) of training positive/negative bags.

4.2. Image re-ranking on the Google data set

The Google data set [9] consists of the images returned by Google Image Search. In total, this data set has 7 categories with each having about 600 images on average. According to [9], on average there are 30% "good" images with clear view, 20% "ok" images with extensive occlusion and image noise, and 50% "junk" images that are unrelated to the category. As suggested in [9, 21], we also use four interest point detectors (*i.e.*, Kadir&Brady Saliency operator, Harris-Hessian detector, Difference of Gaussians (DoG) [16], and Edge-Laplace detector), and then we extract the SIFT descriptor [16] from each salient region. For each detector, we construct an independent codebook by clustering the descriptors from the training images with k-means, where we set $k = 200$. The final vocabulary consists of 800 visual words from the combined codebooks. Each image is finally represented as an 800 dimensional token frequency (tf) feature. Since the images in this data set are not associated with any textual descriptions, only the visual feature is used in both the training and testing stages.

In this work, we re-rank the images from the Google data set by using different methods including SIL-SVM, mi-SVM, MILES, MIL-CPB and our proposed method PMIL-CPB. We also report the existing image re-ranking results from [19, 21] on this data set. Following [19, 21], we treat the "ok" images as positive samples in the experiment because we believe the images with extensive occlusion and image noise are the challenging examples for evaluating different methods. For the given category, we construct each positive bag by randomly sampling 15 instances from this category, and each negative bag is obtained by randomly selecting 15 instances from other cate-

[3] In PMIL-CPB, the positively predicted training instances are ranked according to the bag orders of the positive training bags. Recall that we use the progressive approach to increase the number of training bags. Therefore, a bag that is added in an earlier iteration has a higher rank. The initial $N_{\text{init}} = 5$ positive bags are also ranked, because they are constructed by uniformly partitioning the top-ranked relevant images.

Table 2. Mean precisions (%) at 15% recall over 7 categories on the Google data set. Each result in the table is the best among all results obtained by using different numbers (*i.e.*, N) of positive/negative training bags.

	SIL-SVM	mi-SVM	MILES	MIL-CPB	PMIL-CPB
MP	74.45	75.81	74.89	85.61	**86.64**

Table 3. Per-category precisions (%) at 15% recall for 7 categories (*i.e.*, "Airplane", "Cars rear", "Face", "Guitar", "Leopard", "Motorbike" and "Wrist watch") on the Google data set. For better presentation, we denote the name of each category by using its first letter in this table.

	A	C	F	G	L	M	W	Mean
WsMIL [21]	**100**	**81**	57	52	66	**79**	95	75.71
Schroff's [19]	58.5	–	–	70.0	49.6	74.8	**98.1**	70.20
PMIL-CPB	**100**	75.34	**89.91**	**82.74**	**86.15**	76.63	95.72	**86.64**

gories. For all methods, similarly as on the NUS-WIDE data set, we set $N = 5, 7, 10, 12, 15, 17$ and 20 (*i.e.*, $N_{\text{init}} = 5$ and $N_{\text{inc}} = 2, 3$ or 5 for PMIL-CPB). We also adopt the RBF kernel by empirically setting its bandwidth parameter as $0.25/A$, where A is the mean value of the square distances between training images. Following [19, 21], the per-category precision[4] at 15% recall is used for performance evaluation.

In Table 2, we report the mean precision at 15% recall of each method over all 7 categories. We observe that MIL-CPB is better than SIL-SVM, mi-SVM and MILES. Moreover, PMIL-CPB outperforms MIL-CPB, which indicates that our proposed progressive bag selection scheme can automatically construct better positive bags to further improve the performance.

In Table 3, we compare the per-category precision and mean precision at 15% recall of the proposed method PMIL-CPB with the results reported in [19, 21]. Compared with the existing work [19, 21], our method PMIL-CPB enjoys significant performance improvements in 3 out of the 7 categories (*i.e.*, "face", "guitar" and "leopard"), and also achieves the same or similar performances for some categories (*i.e.*, "airplane", "motorbike" and "wrist watch"). The performance of PMIL-CPB is also much better when compared with [19] and [21] in terms of the mean precision at 15% recall.

5. Conclusion

In this paper, we have proposed a new TBIR approach that can effectively exploit loosely labeled web images to learn robust SVM classifiers. First, we partition the relevant and irrelevant web images into clusters, and then we treat each cluster as a "bag" and the images in each bag as "instances". Observing that each positive bag may contain at least a portion of positive instances, we follow [8] and formulate this task as an MIL problem with such constraints on positive bags. To predict the labels of instances (images), we present MIL-CPB which is a simplified version of the method in [8]. Moreover, we propose a novel progressive scheme called PMIL-CPB to automatically and progressively select more confident positive bags, which leads to more robust classifiers. We conduct comprehensive experiments using the challenging NUS-WIDE data set and Google data set, and the results clearly demonstrate the effectiveness of our TBIR approach.

Acknowledgement: This research is supported by the Singapore National Research Foundation & Interactive Digital Media R&D Program Office, MDA under research grant NRF2008IDM-IDM004-018.

[4] In this work, we follow the same setting as in [19, 21], and report the mean of precisions from five rounds of random positive/negarive bag construction processes.

References

[1] S. Andrews, I. Tsochantaridis, and T. Hofmann. Support vector machines for multiple-instance learning. In *NIPS*, 2003.

[2] R. C. Bunescu and R. J. Mooney. Multiple instance learning for sparse positive bags. In *ICML*, 2007.

[3] L. Chen, D. Xu, I. W. Tsang, and J. Luo. Tag-based web photo retrieval improved by batch mode re-tagging. In *CVPR*, 2010.

[4] Y. Chen, J. Bi, and J. Z. Wang. Miles: Multiple-instance learning via embedded instance selection. *T-PAMI*, 28(12):1931–1947, 2006.

[5] T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y. Zheng. Nus-wide: a real-world web image database from national university of singapore. In *CIVR*, 2009.

[6] R. Datta, D. Joshi, J. Li, and J. Z. Wang. Image retrieval: Ideas, influences, and trends of the new age. *ACM Computer Surveys*, 40(2):1–60, 2008.

[7] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Pérez. Solving the multiple instance problem with axis-parallel rectangles. *AI*, 89(1-2):31–71, 1997.

[8] L. Duan, W. Li, I. W. Tsang, and D. Xu. Improving web image search by bag-based re-ranking. *T-IP*, 2011.

[9] R. Fergus, L. Fei-Fei, P. Perona, and A. Zisserman. Learning object categories from google's image search. In *ICCV*, 2005.

[10] M. Fritz and B. Schiele. Decomposition, discovery and detection of visual categories using topic models. In *CVPR*, 2008.

[11] P. V. Gehler and O. Chapelle. Deterministic annealing for multiple-instance learning. In *AISTATS*, 2007.

[12] J. E. Kelley. The cutting plane method for solving convex programs. *SIAM Journal on Applied Mathematics*, 8(4):703–712, 1960.

[13] F. Li and C. Sminchisescu. Convex multiple-instance learning by estimating likelihood ratio. In *NIPS*, 2010.

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

[15] Y. Liu, D. Xu, I. W. Tsang, and J. Luo. Textual query of personal photos facilitated by large-scale web data. *T-PAMI*, 33(5):1022–1036, 2011.

[16] D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60:91–110, 2004.

[17] O. Maron and T. Lozano-Pérez. A framework for multiple-instance learning. In *NIPS*, 1998.

[18] A. Rakotomamonjy, F. R. Bach, and Y. Grandvalet. SimpleMKL. *JMLR*, 9:2491–2521, 2008.

[19] F. Schroff, A. Criminisi, and A. Zisserman. Harvesting image databases from the web. *T-PAMI*, 33(4):754–766, 2011.

[20] A. Torralba, R. Fergus, and W. T. Freeman. 80 million tiny images: A large data set for nonparametric object and scene recognition. *T-PAMI*, 30(11):1958–1970, 2008.

[21] S. Vijayanarasimhan and K. Grauman. Keywords to visual categories: Multiple-instance learning for weakly supervised object categorization. In *CVPR*, 2008.

[22] J. Wang and J.-D. Zucker. Solving the multiple-instance problem: A lazy learning approach. In *ICML*, 2000.

[23] X.-J. Wang, L. Zhang, X. Li, and W.-Y. Ma. Annotating images by mining image search results. *T-PAMI*, 30(11):1919–1932, 2008.

[24] Q. Zhang and S. A. Goldman. EM-DD: An improved multiple-instance learning technique. In *NIPS*, 2001.