A Scalable Bootstrapping Framework for Auto-Annotation of Large Image Collections

Tat-Seng Chua and Huamin Feng

School of Computing, National University of Singapore,
Singapore 117543

Abstract. Image annotation aims to assign semantic concepts to images based on their visual contents. It has received much attention recently as huge dynamic collections of images/videos become available on the Web. Most recent approaches employ supervised learning techniques, which have the limitation that a large set of labeled training samples is required for effective learning. This is both tedious and time consuming to obtain. This chapter explores the use of a bootstrapping framework to tackle this problem by employing three complementary strategies. First, we train two "view independent" classifiers based on probabilistic SVM using two orthogonal sets of content features and incorporate the classifiers in the co-training framework to annotate regions. Second, at the image level, we employ two different segmentation methods to segment the image into different sets of possibly overlapping regions and devise a contextual model to disambiguate the concepts learned from different regions. Third, we incorporate active learning in order to ensure that the framework is scalable to large image collections. Our experiments on a mid-sized image collection demonstrate that our bootstrapping cum active learning framework is effective. As compared to the traditional supervised learning approach, it is able to improve the accuracy of annotation by over 4% in F_1 without active learning, and by over 18% when active learning is incorporated. Most importantly, the bootstrapping framework has the added benefit that it requires only a small set of training samples to kick start the learning process, making it suitable to practical applications.

Keywords: Bootstrapping, co-training, image annotation, active learning

1. Introduction

For many years, we have been using automated content-based techniques to model and retrieve large image[1] collections using low-level content features such as color, texture

[1] All references to images also imply videos.

and shape [3,4,19,20]. However, such techniques are of insufficient accuracy for many practical applications. An alternative is to invest large amount of pre-processing efforts in automatically annotating images using keywords or concepts[2] so that users may issue concept-based queries to search for images. Here we define annotation as the process of associating one or more pre-defined concepts with new images based on their visual contents. Although many useful image collections come with keyword annotations, the annotations are normally incomplete, and there are many more image collections that do not have such annotations. Thus there is a need to develop (semi) automated techniques to annotate images with semantic concepts accurately and completely.

We view image annotation as a classification process aiming to determine if the content of the whole or part of an image can be classified into one of N pre-defined categories or concepts. We therefore call the automated systems that perform the annotation as **classifiers**. In general, the task of annotating images can be divided into three sub-stages of: (a) segmenting images into meaningful units; (b) extracting appropriate features for each unit; and, (c) associating these features with text. Here different techniques may be used to divide the images into sub-units in Stage (a), which could simply be the whole image, a fixed size block, or a region. Stage (b) is the feature extraction problem at the sub-unit level, while Stage (c) associates content features of sub-unit with concepts, typically using a learning-based method.

One popular approach to achieve automated annotation is to employ a supervised learning approach to perform Stage (c), such as the use of 2D HMM [23] or SVM [7] methods. The main practical problem with such approaches is that a large set of labeled training samples is needed, and it is very tedious and time-consuming to provide such training samples. Moreover, such learning approach is “passive” and is unable to learn incrementally and adapt to changing domain.

To overcome the problems of supervised learning approaches, we propose a bootstrapping cum active learning approach to perform auto image annotation using three strategies. The first strategy is to perform bootstrapping [1] at the region level to derive the concepts. Bootstrapping aims to use a small set of labeled samples to kick-start the learning process from a large unlabeled corpus. To perform bootstrapping, we need a way for the system to evaluate the quality of new annotated samples. This can be achieved by using the co-training technique [4] in which two “view independent” methods independently confirm the quality of newly annotated samples, and learn from each other’s results. This can be accomplished by devising two independent sets of features in Stage (b), and use that to train two independent classifiers in Stage (c). As the aim of this chapter is not to investigate better techniques to model image contents, we consider only the use of two common orthogonal sets of content features

[2] Throughout this chapter, we liberally use the term concept and keyword interchangeably.

– one based on color histogram, and the other on texture and shape features. The use of more advanced features to model image contents is currently being investigated [21].

The second strategy is to employ a contextual model at the image level to disambiguate the set of concepts learned at the image level using the overlapping set of regions generated from two separate image segmentation methods. We employ two separate image segmentation methods as the current segmentation methods are unreliable and unstable. The use of multiple methods helps to minimize the possibility that we may miss a correct segment by using only one method.

The third strategy is to incorporate active learning [24] into the bootstrapping framework to ensure that the framework is scalable to larger problems. This is because mistakes will be made during co-training and the degradation in quality of the "automatically labeled sample set" might be too large for the bootstrapping process to proceed effectively. Thus, in addition to selecting those unlabeled training samples that the co-training approach is most confident of, active learning selects those that the co-training approach is least confident with and repeatedly asks the human users to label and includes them into the "expanded" labeled sample set.

We test our bootstrapping framework using a mid-sized image collection (comprising about 6,000 images from photoCD, CorelCD and Web) and demonstrate that our co-training approach without active learning could improve the performance of annotation by about 4% in terms of F_1 measure as compared to the best traditional supervised learning approach. Of course, the co-training approach has the added benefit of requiring much fewer labeled samples during training. To address the concern that the co-training framework is not scalable, we evaluate the accuracy of the resulting labeled set and found that it is only 78%. This suggests that further bootstrapping using this erroneous labeled set might be a problem since a reasonably accurate labeled set is assumed in most learning scenario. By applying active learning that requires users to label a small set of additional samples, we found that the accuracy of the resulting labeled set is improved dramatically to over 85%. Moreover, it leads to significant improvement in performance of annotation with an F_1 measure of over 58%. The results confirm that co-training with active leading is effective and is scalable to large image collections.

The main contribution of this research is two-fold. First, we devise a co-training and active learning framework to bootstrap the process of annotating large image collections starting from a small number of labeled samples. Second, we demonstrate that the framework is effective and scalable.

The rest of the chapter is organized as follows: Section 2 reviews related research, and Section 3 presents our bootstrapping framework. Section 4 presents the initial experimental results and discussion. Section 5 concludes the chapter with discussion for future work.

2. Related Work

Our work is related to research in three areas: auto image annotation, co-training and active learning. Several recent works deal with the automated or semi-automated attachment of keywords [2,7,13] to image databases. Mori et al. [13] were among the earliest to perform "image-to-word" transformation. They divided the images into fixed-size blocks and trained the clusters of blocks to predict keywords for new images. Barnard and Forsyth [2] segmented the images into regions using Blobworld segmentation [6] and associated keywords to regions in the training set. Chang et al. [7] employed image level content analysis and associated keywords with each image through the application of BPM (Bayes Point Machine). Wang and Li [23] performed image analysis on a fixed size blocks using a 2-D multi-resolution HMM to capture the cross blocks and cross resolution dependencies between blocks for the entire image collection. Jeon et al. [11] also used Blobworld to segment images into regions, and learned the joint distribution of blob regions and concepts.

The above approaches are based on the traditional supervised learning scheme. The training sample set is fixed and much manual effort is needed to come up with a reasonable sized labeled training set. To overcome this problem, Blum and Mitchell [4] proposed a co-training algorithm based on the conditional (view) independence assumption. The algorithm repeatedly trains two classifiers from the labeled data, labels some unlabelled data with the two classifiers and exchanges the newly labeled data between the two classifiers. In this algorithm, one classifier always asks the other classifier to label the most certain data for the collaborator. Since the assumption of view independence cannot always be met in practice, Collins and Singer [8] proposed a co-training algorithm based on "agreement" between the classifiers. Muslea et al. [14] introduced the idea of co-testing which is designed for problems with redundant views or with multiple disjoint sets of features that can be used to learn the target concepts. Nigam and Ghani [15] empirically demonstrated that even bootstrapping (co-training) that violates the view independent assumption can still work better than the traditional learning approach. Cao et al. [5] proposed the use of uncertainty reduction in co-training, in which one classifier selects the most uncertain unlabelled data and asks the other classifier to label. They showed that the natural split of features in co-training algorithm produces the best results. Finally Pierce and Cardie [16] proposed a moderately supervised variant of co-training in which human corrects mistakes made during automatic labeling.

The main issue in active learning is how to choose the most critical instances for users to label manually. The use of uncertainty measurement is one of the popular strategies. Lewis and Gale [12] performed uncertainty sampling to compute an uncertain score to each sample and chose the next sample that the classifier has the least confidence. Zhang and Chen [24] proposed an active learning framework that selects samples automatically based on the criterion that annotating these samples will lead to an overall decrease in the uncertainty of the system.

3. The Co-Training Framework for Image Annotation

Given a scenario that we have a (small) set of labeled regions $\underline{R_L}$ and a (large) set of unlabeled regions $\underline{R_U}$, we discuss our proposed framework based on co-training and active learning to annotate large image collections. To accomplish this, we break the task of annotating images into three sub-stages of: (a) segmenting images into meaningful units, (b) extracting appropriate features for the units, and (c) associating the units in images with concepts. Thus, the problem of auto image annotation can be expressed as:

$$S^p (I_i) \approx \sum R^p{}_{ij} \approx \sum F^q (R^p{}_{ij}) \qquad (1)$$

$$G^a (I_i) \approx G^a (S^p (I_i)) \approx \sum G^a (F^q (R^p{}_{ij})) \rightarrow \underline{L}_c \qquad (2)$$

In (1), function $S^p(I_i)$ performs the transformation of the contents of image I_i. An example of such transformation is the segmentation of the image by converting its contents into meaningful units (or regions/ blocks), i.e $S^p(I_i) \rightarrow \sum R^p{}_{ij}$. The function $F^q(R_{ij})$ selects a set of features to model each unit/region, R_{ij}. In (2), function $G^a(I_i)$ performs the auto-annotation that maps an image to a set of concepts in $\underline{L}_c$. Here $\underline{L}_c$ is the set of lexicon or concepts used to annotate the images. As expressed in (2), if we adopt an approach to segment the image contents into sub-units R_{ij}, then $G^a(I_i)$ can be approximated by an equivalent function to annotate each sub-unit separately and integrating the results of annotations for the overall image.

Equations (1-2) allow us to substitute different models to accomplish each stage of the annotation process independently. For example, we may choose to perform $S^p(I_i)$ by either segmenting the image I_i into regions [6,9] or dividing it equally into fixed blocks [13] in Stage (a). We may use different function $F^q(R_{ij})$ to map the content of each sub-unit R_{ij} into different set of features in Stage (b). Finally, we may choose different machine learning methods, including the bootstrapping technique, to perform the auto-annotation of image I_i in stage (c).

Here, we detail our bootstrapping cum active learning approach to perform auto image annotation using three strategies. The first strategy performs bootstrapping [1] at the region level by using two different classifiers derived using two orthogonal sets of content features. The second strategy employs a contextual model at the image level to disambiguate the set of concepts learned at the image level using the overlapping set of regions generated from two separate image segmentation methods. The third strategy incorporates active learning [24] into the bootstrapping framework to ensure that the framework is scalable to larger problems.

3.1 Region Classifiers for Bootstrapping Process

Given a set of regions for each image, we first discuss how to employ the co-training framework to derive the concepts for each region independently. To initiate the co-training process, we need to develop two (weakly) view independent classifiers. To this end, we adopt different function $F^q(R_{ij})$, $q \in [1,2]$, in Stage (b), to select different set of features to represent the contents of each unit R_{ij} in the image. Here, we simply split the feature set into two disjoint sets as: (a) Set 1: color histogram, and, (b) Set 2: texture and shape features. We denote the feature sets as $F^1(R^p_{ij})$ and $F^2(R^p_{ij})$.

Next we employ a learning function $G^a(R_{ij})$ to perform the annotation by associating the contents of sub-unit R_{ij} with a set of concepts in $\underline{L}_c$. Here we adopt the probabilistic SVM method to train $G^a(R_{ij})$. For different feature sets $F^1(R^p_{ij})$ and $F^2(R^p_{ij})$, we develop two independent classifiers, H^{p1} and H^{p2}, using SVM to map a region into a confident vector of concepts as:

$$H^{p1}\text{:}\ G^a\ (S^p(F^1(R^p_{ij}))\rightarrow \underline{\Phi}^{p1} \qquad (3)$$

$$H^{p2}\text{:}\ G^a\ (S^p(F^2(R^p_{ij}))\rightarrow \underline{\Phi}^{p2}$$

where $\underline{\Phi}^{pq} = \{ v^{pq}_1, v^{pq}_2, .., v^{pq}_N\}$ with $q \in [1,2]$. v^{pq}_j is the confident value for concept $c_j \in \underline{L}_c$, and N is the total number of concepts in $\underline{L}_c$. By combining the outputs from H^{p1} and H^{p2}, we derive the final confidence vector for region R^p_{ij} as:

$$\underline{\phi}^p{}_{ij} = \frac{\sum_{k=1}^{n} v_k^{p1} * v_k^{p2}}{\underline{\phi}_{ij}^{p1} \bullet \underline{\phi}_{ij}^{p2}} \qquad (4)$$

The final confidence vector $\underline{\Phi}^p_{ij}$ can be used to control the assignment of concepts to region. Due to the unreliability of region segmentation method, a single concept may be insufficient to describe the region's contents. We therefore adopt two strategies to assign concepts from $\underline{\Phi}^p_{ij}$ to region R^p_{ij} as: (a) Strategy 1: select only one concept per region; and (b) Strategy 2: select the top k concepts.

We now outline the details of the co-training framework for annotating each region as follows.

- Inputs:

$\underline{R_L}$: an initial collection of (small) labeled regions;

$\underline{R_U}$: a large set of unlabeled regions;

c_z: the concept label of the current classifiers;

β: the number of unlabelled regions to be considered in each iteration of co-training;

M: maximum number of iterations of the co-training process;

m: the iteration number (m=0 initially);
θ: the predefined threshold for selecting the most confident class label;
τ_1, τ_2: the thresholds for selecting one classifier to label over the other.
ε: the tolerance for the least uncertainty regions.

- LOOP:

  While there exist regions without concept labels and m ≤ M:

  - Train classifiers H^{p1} and H^{p2} from the current labeled training set $\underline{R_L}$.
  - Randomly select next set of β unlabelled regions $\underline{R_\beta}$ from $\underline{R_U}$:
    - For each $r_i \in \underline{R_\beta}$, compute the confidence values for all concepts in r_i using classifiers H^{p1} and H^{p2} based on (3).
    - The following conditions are used to determine the assignment of concepts c_z to region r_i:

      Condition 1 (when both Classifiers have high confidence in c_z): When the confidence value for c_z is larger than θ for both H^{p1} and H^{p2}, simply label r_i with concept c_z and add it to the labeled set $\underline{R_L}$.

      Condition 2 (when only one Classifier has high confidence in c_z): If condition 1 is not satisfied, but the confidence value of concept c_z for one classifier is larger than τ_1, while that of the other classifier is less than τ_2, then use the classifier that gives higher confidence value to label the region and add it to the labeled set $\underline{R_L}$.

      Condition 3 (when both Classifiers are uncertain) – to perform the optional active learning step: If the above two conditions are not met, but the confidence values of two classifiers for the class label c_z are around $0.5\pm\varepsilon$, then choose k such instances with the lowest entropy values and optionally asks the user/expert to label the region and add it to the labeled set $\underline{R_L}$.

- Outputs: Two updated Classifiers H^{p1} and H^{p2} and an expanded labeled set $\underline{R_L}$.

The above procedure for co-training the classifiers is performed for each segmentation method $S^p(I_i)$ separately. Optionally, it also incorporates strategy 3 by employing active learning to improve the quality of the automatically annotated sample set.

3.2 Concept Disambiguation at the Image Level

Because of the unreliability and uncertainty in image segmentation, the use of only one segmentation method runs the risk of missing or wrongly segment important regions, thus resulting in the missing of key concepts. In this research, we explore the use of different function $S^p(I_i)$ to segment the image into regions at the same time. In other words, we employ two different segmentation methods based on Blobworld [6] and JSEG [9]. They are denoted as $S^B(I_i)$ and $S^U(I_i)$ respectively. The idea here is to use these two different segmentation methods to segment the image into two separate sets of possibly overlapping regions. We then employ the function $G^a()$ (of Equation 3) to map each region independently into concepts, and develop a contextual model that uses the correlations between the overlapping regions and conflicting concepts to disambiguate the learnt concepts to arrive at the final annotation for each region. We call this process *concept disambiguation*.

A naïve approach to generate the overall annotation for the entire image is simply to aggregate the annotations of all regions. This, however, will not work well because although the concepts for regions are generated independently, there are dependencies between concepts or against concepts and/or regions. For example, certain concepts should not co-occur in adjacent regions or in the same image. To derive such knowledge, we need to make use of the context between regions and concepts to perform concept disambiguation. The contextual relationships between regions can be modeled by computing the overlaps between regions generated from different methods as:

$$Mc_{jk} = U_{\mathrm{R}_{\mathrm{ij}}^{\mathrm{b}}, R_{ik}^{u}} = \frac{\mathrm{R}_{\mathrm{ij}}^{\mathrm{b}} \cap R_{ik}^{u}}{\left|\mathrm{Im}\,age\,\mathrm{I}_{\mathrm{i}}\right|} \qquad (5)$$

where Mc_{jk} contains the overlap between region R^b_{ij} and region R^u_{ik}, normalized by the size of image I_i. The Mc_{jk} values are stored in the overlapping matrix that encodes the overlaps between all regions in the same image.

As we expect the regions generated by different methods to be correlated, we expect the regions R^b_{ij} and R^u_{ik} to have some overlaps (i.e. $Mc_{jk} \neq 0$ for some j and k) and that they share common concepts (i.e. $\underline{\Phi}(R^b_{ij}) \cap \underline{\Phi}(R^u_{ik}) \neq 0$ for some j and k). The disambiguation process will make use of a decision model based on SEE5 to identify the best set of concepts for each region based on this contextual information. The inputs to the decision tree are the region id, its concept vector $\underline{\Phi}^M$, and the list of overlapping regions and their corresponding concept vector $\underline{\Phi}^{S}$‘s. The output of the decision tree is a confidence vector for the main region, $\underline{\Phi}^M$, where the elements of $\underline{\Phi}^M$ are as defined in (3). From $\underline{\Phi}^M$, it is easy to choose the concept(s) for the region. We again employ the same strategies as in Section 3.1 to select one or more concepts to annotate the region. The unions of all resulting concepts are used as the final annotation of the image.

The overall process of our concept disambiguation process is given in Fig. 1.

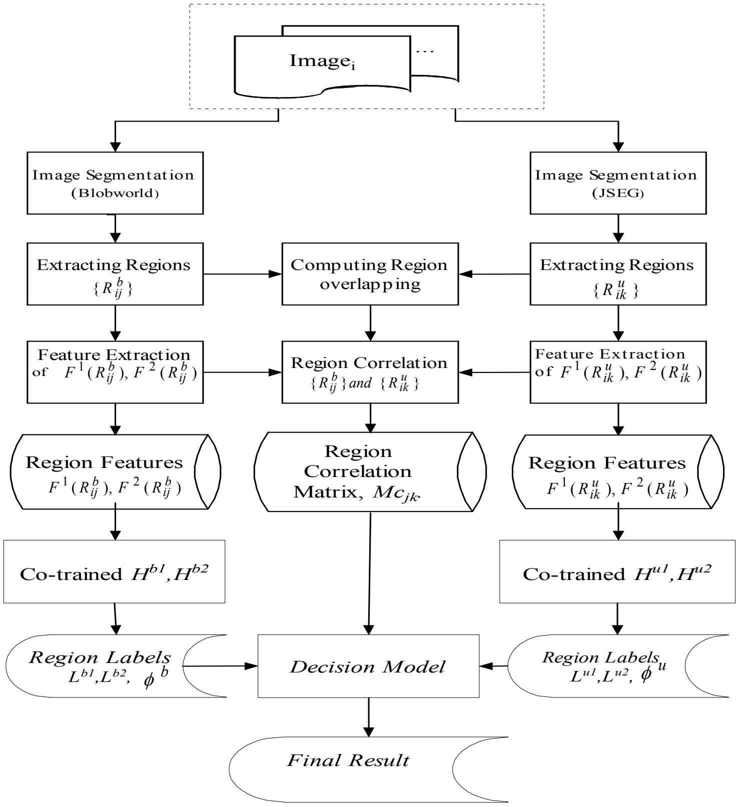

Image$_i$
...
Image Segmentation (Blobworld)
Image Segmentation (JSEG)
Extracting Regions $\{R_{ij}^{b}\}$
Computing Region overlapping
Extracting Regions $\{R_{ik}^{u}\}$
Feature Extraction of $F^1(R_{ij}^{b}), F^2(R_{ij}^{b})$
Region Correlation $\{R_{ij}^{b}\}$ and $\{R_{ik}^{u}\}$
Feature Extraction of $F^1(R_{ik}^{u}), F^2(R_{ik}^{u})$
Region Features $F^1(R_{ij}^{b}), F^2(R_{ij}^{b})$
Region Correlation Matrix, Mc_{jk}
Region Features $F^1(R_{ik}^{u}), F^2(R_{ik}^{u})$
Co-trained H^{b1},H^{b2}
Co-trained H^{u1},H^{u2}
Region Labels L^{b1},L^{b2}, ϕ^{b}
Decision Model
Region Labels L^{u1},L^{u2}, ϕ^{u}
Final Result

Figure 1: The concept disambiguation framework at the image level

4. Experimental Results and Discussions

4.1 Test Data and Methods

To test the effectiveness of our approach, we use an image collection comprising about 6,000 images. The images come from PhotoCD, Web and parts from CorelCD. We randomly selected as sub-set of about 780 images for training, and the rest for testing. We test the annotation of images using 20 concepts derived from Corel CD. Some concepts are general while others are more specific. Examples of general concepts are: animals, computer, food, industry, transport, and indoor etc. Examples of more specific concepts include: people, plant, sports, rock, water, vegetation, snow, sky, beach, road, table, field, aircraft and sunset etc.

For the co-training framework described earlier, we need to select different models at different stages of the process. The models we use are summarized as follows:

a) Feature selection function $F^q(R^p_{ij})$. For each region R^p_{ij}, we use the standard color histogram, texture and shape as the features. For the co-training experiments, we divide the feature set as: F^1 contains the color histogram, and F^2 includes only the texture and shape features.

b) Segmentation methods $S^p(I_i)$. We employ two segmentation methods based on Blobworld [6] ($S^B(I_i)$) and JSEG [9] ($S^U(I_i)$).

c) Image annotation function $G^a(I_i)$. Here we use SVM to train the classifiers, and Decision Tree in the contextual model to disambiguate the concepts learned from different classifiers based on different segmentation methods. Here we experiment with using two types of SVM -- the hard SVM that returns only a single binary decision and the probabilistic SVM (or pSVM) that returns multiple decisions with confidence values. We select SVM with radial basis function (RBF) kernel [22], and use logistic regression to compute the probability of pSVM [17].

In order to test the effectiveness of our co-training method with and without active learning against the traditional machine learning methods, we carry out experiments using the following methods:

a) Traditional machine learning approaches based on probabilistic-SVM. Our earlier work [10] showed that probabilistic-SVM is superior to hard-SVM for this task, thus we consider only the use of probabilistic-pSVM in the experiments here. We combine the feature sets F^1 and F^2 into one set, and choose 400 labeled regions for each concept label to train the classifiers. We experiment with two variants of method as follows:

   pSVM-single: Employ both the Blobworld and Jseg segmentation methods separately and use Decision Tree to perform concept disambiguation. It uses strategy 1 to select only one concept for each region.

   pSVM-multiple: Same as pSVM-single except that it selects multiple concepts for each region.

b) Co-training framework: For the co-training experiment, we choose only 20 labeled seed regions for each concept label to kick-start the co-training process. We again experiment with two variants of co-training methods -- one without active learning and the other with. The resulting methods are denoted as **co-Train(M)** and **co-Train-Active(M)** respectively, where M denotes the maximum of iterations to be performed during co-training.

4.2 Co-Training Experiment

We first test the feasibility of our co-training framework in annotating images. Here we consider only co-training without active learning. Table 1 summarizes our initial results, in which we presents the results of **co-Train** with M =50 and 100. The results are presented in terms of recall, precision and F_1 measures [18]. In addition, we also differentiate between two kinds of results. The first set, which we termed ACR ("automatically checked Result"), compares the learned concepts for each image against the "original annotation" come with the image collection. It does not consider whether the additional concepts learned by the system that are not present in the original annotation are correct. In general, we observe that most images are only assigned with one or few keywords, and they often miss some details of images that are found by the automated methods. As a result, ACR tends to report lower precision for automated methods, as we tend to find more concepts that are correct. In order to fairly evaluate the automated techniques, we present another set of results, which we termed MCR ("manually checked Result"). In MCR, we manually check the learned concepts against the image contents. We consider the learned keywords as correct if it is present in the original annotation or in the image contents. MCR allows us to add more meaningful keywords into the original annotation. For example, the image with only the keywords plane often has sky, cloud, etc. The cloud and sky are likely to be learned in the automated approach, which should be considered as correct.

Table 1: Comparison between Co-training (without active learning) and traditional methods

Note: ACR: Automatically Checked Results; and MCR: Manually Checked Results

Method	**ACR**			**MCR**		
	Re.	**Pr.**	**F_1**	**Re.**	**Pr.**	**F_1**
pSVM-single	45.0	26.5	33.4	50.5	39.0	44.0
pSVM-multiple	47.6	28.5	**35.4**	51.1	44.2	**47.4**
co-Train (50)	48.3	34.3	40.9	51.8	41.0	45.8
co-Train (100)	34.4	57.1	**42.8**	43.1	57.7	**49.3**

From Table 1, we found that the traditional method **pSVM-multiple** performs better than **pSVM-single**, indicating that the strategy of annotating "one region" with "one or more concepts" is more effective. The **pSVM-multiple** could attain an F_1 measure of 35.4% for ACR and 47.4% for MCR.

Table 1 also indicates that by using **co-Train (100)**, we could achieve a superior performance of over 42.8% for ACR and 49.3% for MCR cases. This is about 4% better than the best of traditional method for the MCR case. The results indicate that co-training without active learning is more effective, while requiring much fewer training samples (20 times less than the traditional method) as compared to the traditional supervised learning approach.

Table 1 also shows that by performing more iterations, the performance of **co-Train(M)** improves steadily from M=50 to M=100. This indicates that our method is consistent.

4.3 Scalability Experiment

One major concern with using the co-training framework is that it might not scale up to larger problems. This is because the mistakes made during the co-training process will degrade the quality of the resulting "automatically labeled set". Thus we investigate the incorporation of active learning in the co-training process and evaluate the quality of labeled training set as compared to co-training without active learning. Table 2 lists the accuracy of the resulting labeled sets for both **co-Train(100)** and **co-Train-Active(100)**. We found that the accuracy of resulting labeled set for **co-Train (100)** is about 78.5%, whereas with **co-Train-Active(100)**, the accuracy of the labeled set improves dramatically to 85.7%. As part of the underlying assumption of co-training is that the labeled set should be sufficiently accurate, higher accuracy in the resulting labeled set suggests that further co-training is feasible and that the process is scalable. The active learning process requires users to annotate about 150 additional samples for each class. This is a relatively small amount of efforts for users to achieve superior performance.

Table 2: Accuracy of resulting labeled set after training

Note for pSVM, no new labeled sets are added, and hence the error rate is 0.

Method	# of initial labeled set	# of labeled samples at the end of Co-Training	Accuracy of expanded labeled set
pSVM	25 * 400	10,000	100%
co-Train(100)	25 * 20	4,595	78.5%
co-Train-Active(100)	25 * 20	4,652	85.7%

Table 3: Results of co-training with active learning

Note: ACR: Automatically Checked Results; and MCR: Manually Checked Results

Method	ACR			MCR		
	Re.	**Pr.**	**F_1**	**Re.**	**Pr.**	**F_1**
co-Train-Active (50)	36.9	27.2	31.3	58.6	47.5	52.5
co-Train-Active (100)	34.4	57.1	**42.9**	57.3	59.6	**58.4**

Table 3 shows that the incorporation of active learning leads to significant improvement in the performance of annotation with an F_1 measure of over 58.4%.

4.4 Examples of Image Annotation

Fig. 2 shows some examples of images annotated using our approach. Column 2 of Fig. 2 gives both the original annotation provided by the authors, as well as the annotation learned by our system. The results show that our annotation scheme could give reasonably accurate and complete annotation. Note that as we support only “animal” as the general concept for all types of animals, specific animals such as “dog”, “tiger” etc. are tagged as “animals”, which are considered to be correct.

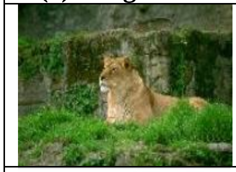
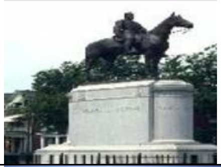
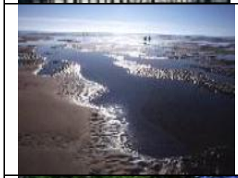
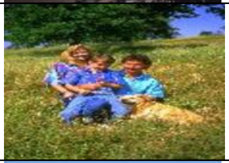
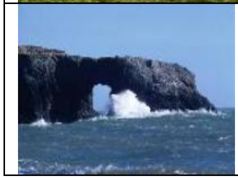
(1) Image	**(2) Keywords**
	Original: tiger, grass, rock Learned: animals, grass
	Original: travel Learned: plant, rock
	Original: water, beach Learned: water, beach, sunset, sky
	Original: people, plant, travel, animals Learned: people, grass, sky
	Original: water Learned: rock, water

Figure 2 Examples of image annotation using our approach

5. Summary

Because of the ambiguity in content-based retrieval techniques, most users prefer to access images using keywords. This brings in a major practical problem of how to (semi-) automatically annotate large image/video archives with text annotations. This research explores the use of bootstrapping approach to perform auto image annotation that requires a small number of labeled training samples to kick start the training process. We carried out experiments using a mid-sized image collection (comprising about 6,000 images). We demonstrated that our co-training framework is able to improve the accuracy of annotation by over 4% in F_1 measure as compared to the traditional supervised learning approach, while requiring only 5% of the labeled samples to kick start the training process. Further we addressed the concern of scalability with co-training by incorporating active learning into the framework. We demonstrated that the use of active learning could further improve the performance of annotation significantly by over 18%, and that the accuracy of the resulting labeled set remains high (>85%), thus ensuring that the co-training framework is scalable.

In evaluating the effectiveness of the bootstrapping techniques, one should also consider the enormous benefits of requiring much fewer training samples (20 times less) as compared to the traditional supervised learning approach to kick start the learning process. This provides a practical approach to deploy the system to handle dynamic environment.

Our results demonstrated that the collaborative bootstrapping approach, initially developed for text processing, could be effectively employed to tackle the challenging problems of multimedia information retrieval. We will carry out further research in the following areas. First, we will further investigate the consistency and scalability of co-training approach by carry out both theoretical study and large-scale empirical experiments. Second, we will explore the use of better content features to model images' contents. Finally, we will research into web image mining based on the images obtained from the web and their surrounding context.

Acknowledgment

The first author would like to thank the National University of Singapore (NUS) for the provision of a scholarship, under which this research is carried out.

References

[1] S. Abney. Bootstrapping. Association for Computational Linguistics (ACL'02).

[2] K. Barnard & D.A. Forsyth. Learning the semantics of words and pictures. IEEE International Conference on Computer Vision II, 408-415 (2001).

[3] K. Barnard, P. Duygulu & D. Forsyth. Clustering Art. IEEE Computer Vision and Pattern Recognition. Pp 434-441. 2001.

[4] A. Blum & T. Mitchell. Combined labeled data and unlabelled data with co-training. Proceeding of the 11th Annual Conference on Computational Learning Theory. 1998.

[5] Y. Cao, H. Li & L. Lian, *Uncertainty reduction in collaborative bootstapping: measure and algorithm*. Association for computational Linguistics (ACL'03). 2003.

[6] C. Carson, M. Thomas, J.M.Hellerstein & J. Malik. BlobWorld: A system for region-based image indexing and retrieval. International Conf Visual Info Sys, 1999.

[7] Edward Chang, Kingshy Goh, Gerard Sychay & Gang Wu. CBSA: content-based soft annotation for multimodal image retrieval using Bayes Point Machines. IEEE Transactions on Circuits and Systems for Video Technology, Special Issue on Conceptual and Dynamical Aspects of Multimedia Content Description 13, 26-38 (2003).

[8] M. Collins & Y. Singer. Unsupervised models for name entity classification. Proceedings of the 1999 Joint SIGDAT Conference on Empirical Methods in Natural language Processing and Very Large Corpora. 1999.

[9] Y. Deng & B.S. Manjunath, *Unsupervised segmentation of color-texture regions in images and video.* IEEE Trans on Pattern Analysis and Machine Intelligence *23,* 800-810. 2001.

[10] Huamin Feng & Tat-Seng Chua. "A bootstrapping approach to annotating large image collection". Workshop on "Multimedia Information Retrieval", organized in part of ACM Multimedia 2003. Berkeley, USA. Nov 2003, 55-62.

[11] J Jeon, V. Lavrenko & R. Manmatha. Automatic image annotation and retrieval using cross-media relevance models. ACM AIGIR '2003, Toronto, Canada. 119-126.

[12] D.D Lewis & W.A. Gale, *A sequential algorithm for training text classifiers*. In proceeding of ACM SIGIR, pp 3-12, 1994.

[13] Y. Mori, H. Takahashi & R. Oka. Image-to-word transformation based on dividing and vector quantizing images with words. First International Workshop on multimedia Intelligent Storage and Retrieval Management (1999).

[14] I. Muslea, S. Minton & C.A. Knoblock, *Selective sampling with co-testing*. CRM Workshop on Combining and Selecting Multiple Models with Machine Learning (2000).

[15] K. Nigam & R. Ghani. Analyzing the effectiveness and applicability of co-training. Proceedings of the 9th International Conference on Information and Knowledge management. 2000.

[16] David Pierce & Claire Cardie. Limitations of co-training for natural language learning from large datasets. Proceeding of the 2001 Conference on Empirical Methods in Natural Language Processing. 2001.

[17] John C. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In ‘Advances in Large Margin Classifiers’, A.J. Smola, P. Bartlett, B. Scholkopf & D. Schuurmans (Eds). MIT Press, 1999.

[18] G. Salton & M.J. McGill. Introduction to modern information retrieval. McGraw Hill. 1983.

[19] John R. Smith and S-F Chang. VisualSeek: A fully automated content-based query system. ACM Multimedia ‘1996. 87-92.

[20] John R. Smith, Milind Naphade & Apostol Natsev. Multimedia Semantic Indexing Using Model Vectors. ICME ’03, 2003.

[21] Rui Shi, Huamin Feng, Tat-Seng Chua & Chin-Hui Lee. An adaptive image content representation and segmentation approach to automatic image annotation. To appear in Conference on Image and Video Retrieval (CIVR’04), Dublin, Jul 2004.

[22] Vladimir Vapnik. The nature of statistical learning theory. Springer, New York, 1995.

[23] James Z. Wang & Jia Li. Learning-based linguistic indexing of pictures with 2-D MHHMs. ACM Multimedia ‘2002, 436-445.

[24] Cha Zhang & Tsuhan Chen. An active learning framework for content-based information retrieval. IEEE transactions on multimedia. 4, 260-268, 2002.