On the Pooling of Positive Examples with Ontology for Visual Concept Learning *

Shiai Zhu†, Chong-Wah Ngo†, Yu-Gang Jiang‡
†Dept of Computer Science, City University of Hong Kong, Kowloon, Hong Kong
‡School of Computer Science, Fudan University, Shanghai, China
shiaizhu2@student.cityu.edu.hk, cwngo@cs.cityu.edu.hk, ygj@fudan.edu.cn

ABSTRACT

A common obstacle in effective learning of visual concept classifiers is the scarcity of positive training examples due to expensive labeling cost. This paper explores the sampling of weakly tagged web images for concept learning without human assistance. In particular, ontology knowledge is incorporated for semantic pooling of positive examples from ontologically neighboring concepts. This effectively widens the coverage of the positive samples with visually more diversified content, which is important for learning a good concept classifier. We experiment with two learning strategies: aggregate and incremental. The former strategy re-trains a new classifier by combining existing and newly collected examples, while the latter updates the existing model using the new samples incrementally. Extensive experiments on NUS-WIDE and VOC 2010 datasets show very encouraging results, even when comparing with classifiers learnt using expert labeled training examples.

Categories and Subject Descriptors

H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing

General Terms

Algorithms, Measurement, Experimentation

Keywords

Visual concepts, training set construction, semantic pooling

1. INTRODUCTION

Acquiring a sufficiently large amount of positive training examples is a key requirement for learning effective concept classifiers. Ideally we should have a training set that maximizes two criteria: coverage and diversity. Coverage specifies the inclusiveness of samples in describing different semantic facets of a concept. Diversity recounts the novelty of samples in presenting the diverse visual appearance of a semantic facet. Constructing a training set with good coverage and diversity is often difficult in practice due to the expensive cost in manual data sampling and labeling.

In recent few years, nevertheless, the proliferation of social media has made the low cost sampling and labeling of training examples a feasible idea [7, 8]. Flickr website, for example, has accumulated more than four billions of images and receives thousands of new uploads per minute. This repository alone already offers a tremendous pool of candidate images for the construction of training sets. However, due to the fact that most images are weakly (noisily) tagged, direct supplying these images for classifier learning without data cleaning will result in poor performance. One straightforward way for dealing with this problem is by bootstrap learning [7, 8]. In [8], an iterative framework that simultaneously learns classifiers and collects positive samples from the Web was proposed. Starting from a classifier learnt from a collection of expert labeled examples, iterative update of the classifier model is performed by including new web images which are predicted as positive by the model itself. Although the approach is expected to collect a clean and larger training set, the newly added images may be homogeneous and lack visual diversity. A similar idea of bootstrap learning was also proposed in [8], but interestingly, for collecting negative training samples from the Web. The sampled images are shown to be more suitable for concept learning than those collected by random sampling, leading to better classification performance. A variant of the bootstrapping strategy is active learning [9], where human intervention is required to iteratively provide manual labels to guide the learning process towards a good concept classifier. In general, bootstrap or active learning could suffer from high computational cost. For instance, as indicated in [8], the overall performance becomes stable after 50 rounds of learning, prediction and sample selection. As a result, the scalability of these learning strategies is in question especially for learning a large set of concept classifiers.

A more efficient way of sampling training examples is to rank the Web images according to their relevancy to a given target concept, and then select the top ranked ones [4, 12]. In [4], visual clustering is carried out to characterize the set of images as a cluster correlation network. Random walk is then performed on the network to rank the image clusters. Heuristics such as larger clusters are more relevant to the target concepts and noisy samples have weak visual correlation with other image clusters are adopted for relevancy reranking. In [12], instead of exploiting visual similarity, semantic field was proposed to predict the relevancy of tag lists

*Area chair: Lei Chen

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

to target concepts by exploiting external knowledge such as Wikipedia. The top-k ranked images are then used as positive examples for concept learning.

In this paper, different from [4, 7, 8, 12] which mainly focused on filtering noisy Web search results for training set construction, we address the issue of training sample enrichment, which is important for learning a good concept classifier with better generalization capability. Specifically, a semantic pooling technique is proposed to enlarge the coverage as well as to diversify the initial collection of sampled images by propagating positive examples among semantically related concepts using ontology. Two kinds of learning: aggregate and incremental learning, are then employed to train better concept classifiers using new samples. Ontology has been frequently adopted for boosting concept annotation performance [1, 5]. The existing works, however, are on the utilization of ontological relationship for concept label propagation or noise removal. Our work is different in the way that the ontology is employed to pool the freely obtained Web images for boosting the concept learning performance with a cleaner and more diverse training set.

2. SEMANTIC POOLING

Our proposed approach for collecting training examples consists two components: 1) collect and filter examples for each node in an ontological structure; and 2) hierarchical pooling of training examples based on the ontology. We elaborate each of the components below.

2.1 Positive Example Sampling

Given a concept node C_x, we first adopt a similar approach as [12] which considers tag-concept relevancy for training example acquisition. The probability of seeing a concept C_x in an image with a tag list $SF =< T_1, T_2, ..., T_n >$ is defined as $P(C_x|SF) = \frac{P(SF|C_x)\times P(C_x)}{P(SF)}$, where $P(C_x)$ is a constant for all the images under consideration when selecting examples for C_x, and thus can be ignored. The computation of $P(C_x|SF)$ is not very stable since the probability of the entire tag list $P(SF)$ is usually extremely small. Thus we approximate $P(SF|Cx)$ as $P(SF) \times (\sum_i P(T_i|C_x)/n)$, which combines the probabilities of observing SF as a whole and seeing each tag of SF in images tagged with concept C_x. With that, $P(SF)$ can be eliminated and $P(C_x|SF)$ can be approximated as:

$$P(C_x|SF) = \frac{\sum_{i=1}^{n} P(T_i|C_x)}{n} \quad (1)$$

where $P(T_i|C_x)$ denotes the likelihood of observing a tag T_i given a concept C_x, and n is the number of tags.

We employ two measures different from [12] for computing Equation (1). The first measure is Flickr Context Similarity (FCS), which was originally proposed in [6], for measuring the visual co-occurrence between two words. For instance, concept "car" is often tagged together with "road" in user labeling. Although both concepts are not closely related by ontological inference, their visual co-occurrence should be high. FCS is defined as $FCS(T_i, C_x) = e^{-NGD(T_i,C_x)/\rho}$, where $NGD(T_i, C_x) = \frac{max\{\log h(T_i), \log h(C_x)\} - \log h(T_i, C_x)}{\log N - \min\{\log h(T_i), \log h(C_x)\}}$, $h(T_i)$ is the number of Flickr images associated with tag T_i, and $h(T_i, C_x)$ is the number of images associated with both T_i and C_x.

The second measure is based on ontological similarity. For this, we adopt WUP [11], defined as $WUP(T_i, C_x) = \frac{2D(S_{T_i,C_x})}{L(T_i,C_x)+2D(S_{T_i,C_x})}$, where S_{T_i,C_x} is the lowest common ancestor of T_i and C_x in WordNet. The function D returns the depth of a concept, while the function L evaluates the path length by traversing from T_i to C_x in WordNet.

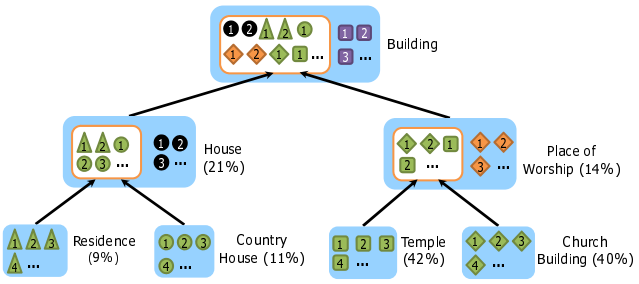

Figure 1: A toy example for illustrating the semantic pooling of training examples for concept "building". Positive samples of child nodes are hierarchically pooled in a bottom-up manner. The percentage in the parentheses indicates the proportion of examples to be propagated to the parent node, computed by Equation (2). The numbered small color boxes with various shapes represent images originally from different nodes, with rankings (indicated by the numbers) computed by Equation (1).

Based on the Bayesian theorem, the conditional probability in Equation (1) can be computed as $P(T_i|C_x) = P(T_i, C_x)/P(C_x)$, where $P(C_x)$ is ignored in our experiments since it is a constant when selecting images for concept C_x. Finally $P(T_i, C_x)$ can be estimated by $FCS(T_i, C_x) \times WUP(T_i, C_x)$. With Equation (1), each image has a $P(C_x|SF)$, based on which the top-k ones are selected as training samples for C_x.

2.2 Ontology-based Pooling

In ontologies like WordNet, child nodes are semantic subsets of parent nodes. Take concept "building" as an example, using the hyponymy relationship in WordNet, nodes under "building" are organized in a sub-tree structure of 6 layers and 268 child nodes. Intuitively, the coverage and diversity of training examples for "building" can be greatly enhanced, by also pooling the samples from the 268 child nodes. With this intuition, we employ WordNet ontology as the ground to propagate positive examples for concept learning.

According to WordNet hyponymy structure, as long as the target concepts are not leaf-nodes, each of them can form a multi-layer tree, with itself as the root node. Figure 1 shows the top three levels of a tree with root node "building". Sample pooling is performed on the tree structure in bottom-up manner. Specifically, taking a tree with only two-layers as an example, positive samples in the child nodes are propagated in proportion to the root node. The proportion is decided based on the popularity of the child node, which is measured based on the total number of images being returned from Flickr. Formally, the percentage of training samples to be propagated from a child node C_i to its parent node can be computed by:

$$P_{C_i} = \frac{f_{C_i}}{\sum_{C_j \in L_i} f_{C_j}} \quad (2)$$

where f_{C_i} is the number of Flickr images tagged with C_i,

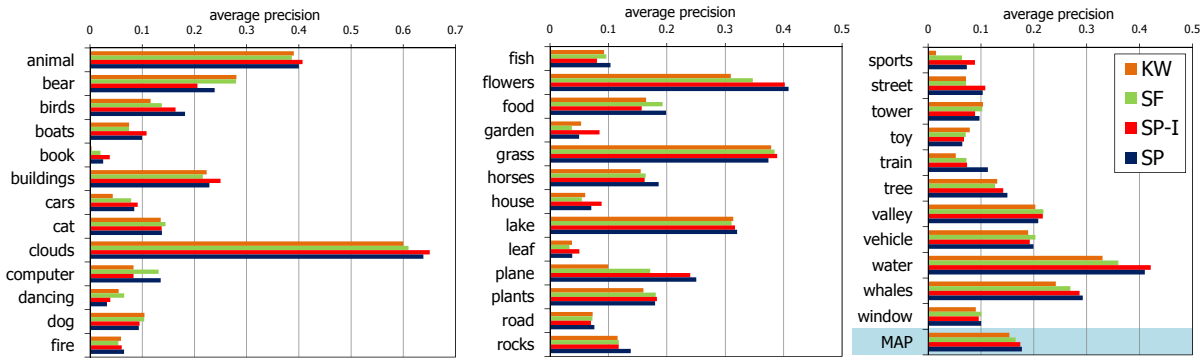

Figure 2: AP comparison on NUS-WIDE testing set

and L_i represents a set of concept nodes on the same layer with C_i in the tree.

The order of selecting samples for pooling is based on the rank list evaluated by Equation (1). In other words, the first image being picked up from a child node is always its top ranked image. Images propagated into a parent node are rescored by the following equation:

$$S(I_k) = (N - k)/N \tag{3}$$

where I_k is the image ranked at the k_{th} position among training examples of the child node C_i, and N is the number of images being propagated from C_i. After the bottom-up propagation process, samples arriving at the root node C^* are then aggregated with the original samples T_{C^*} in C^* as $\widetilde{T}_{C^*} \leftarrow T_{C^*} \cup T_{C_1} \cup T_{C_2} \cup \cdots$, where T_{C_i} denotes the set of positive examples propagated from the child node C_i, and $\widetilde{T}_{C^*}$ is the final set of examples for learning concept C^*. To facilitate successive propagations into nodes on the upper layers, the examples in $\widetilde{T}_{C^*}$ is then ranked by Equation (3), which guarantees that all the top ranked examples of child nodes will be distributed evenly in the ranked list of $\widetilde{T}_{C^*}$ and the orders are retained.

For a tree with more than two levels, similar procedure is carried out recursively from the leaf nodes to the root concept. Figure 1 gives a toy example to illustrate the procedure of semantic pooling for concept "building". The training set $\widetilde{T}_{C^*}$ is enriched with diverse examples from the child nodes.

3. CONCEPT LEARNING

Based on the pool of freely sampled images by ontology, we consider two scenarios for concept learning. Aggregate learning re-trains a classifier using the collection of training examples by semantic pooling. Incremental learning, on the other hand, uses the newly collected samples from the child nodes to update the existing models.

We adopt SVM for aggregate learning, and adaptive SVM (A-SVM) [10] for incremental learning. A-SVM learns a "delta function" $\Delta f(x)$ based on the new examples, and adapts the original SVM model $f^a(x)$ as $f(x) = f^a(x) + \Delta f(x) = f^a(x) + W^T\phi(x)$, where W^T are the parameters to be leant from new samples. A-SVM basically seeks for additional support vectors learnt from newly arrival data to adjust the original decision boundary of a classifier. It optimizes the trade-off that the new boundary should be close to the original one, and meanwhile, can correctly classify the new samples.

4. EXPERIMENT

We primarily use NUS-WIDE [2] which is a large-scale Web image dataset for performance evaluation. The dataset consists of 81 labeled concepts and we conduct experiments on the testing set which contains 107,859 images. Using WordNet, we construct 81 concept trees, with each of the concepts as root node. For each tree, we further remove the leaf nodes that have less than 1,000 images returned by Flickr. Eventually, we consider 37 concepts, each with a tree of depth more than two. Among the 37 concept trees, the average depth is 4 and the average number of child nodes is 55. For instance, concept "building" has 127 nodes of depth 6, and "sports" has 101 nodes of depth 7.

We compare our approach semantic pooling (SP) with two other approaches: 1) semantic field (SF) [12] based on the description in Section 2.1, and 2) simple keyword matching with query expansion (KW). In the experiments, SF and KW respectively sample 2,000 pseudo positive examples from Flickr. To ensure fair comparison such that each approach uses an equal number of training examples, SP also samples 2,000 positive examples. The first 1,000 examples are from the root node, while the remaining 1,000 examples are pooled from the child nodes. All the three approaches use the same set of negative examples, 5,000 per concept, randomly crawled from Flickr. Although there are better ways of collecting negative samples [8], we use random sampling for simplicity as our focus is on the collection of better positive training samples. For concept learning, we adopt the approach in [3] where three SVM models based on bag-of-visual-words, color moment and wavelet textures are trained respectively. Probability predictions from the three SVM models are linearly fused. For SP, in addition to training new SVM models using the newly pooled training sets, we also attempt an incremental learning method based on A-SVM [10]. Results from direct SVM training is indicated by SP, while those from A-SVM is marked by SP-I. Specifically, for SP-I, we use SVM classifiers learnt from SF filtering (without pooling) as original/source models. The models are updated by pooling 1,000 positive samples from child nodes using semantic pooling, and another 1,000 negative examples randomly acquired from Flickr.

We employ Average Precision (AP) to measure the performance of four different approaches. Figure 2 shows the results. SP, with mean AP (MAP) of 0.1774, exhibits better performance than SF (MAP=0.1655) and KW (MAP=0.1536). Among the tested concepts, there are 26 out of 37 concepts

where SP shows higher AP than SF, and only 5 concepts suffer from performance degradation after pooling (around 5%). The performance degradation for these concepts is mostly due to different semantic interpretations between human labels and WordNet. For example, the ground-truth labels in NUS-WIDE regard "panda" as a kind of "bear", while in WordNet, "panda" is not a child node of "bear". Also, "beach ball" and "tree house" are two child nodes of "toy" in WordNet, but are rarely labeled as "toy" in NUS-WIDE. From our observation, the coverage and diversity of training examples for most concepts can be largely enhanced by semantic pooling. Figure 3 shows an example for concept "plane". In 3(a), the sampled examples for root node "plane" are mostly in close view and mixed with false positives. The samples from child nodes, as shown in 3(b), offer a more diverse view in terms of visual and semantic aspects. In addition, as the child node concepts are more specific, the chance of sampling false positives is usually lower than that for the parent concepts. As a result, the AP of concept "plane" is significantly boosted to 0.2402 (by SP) from 0.1715 (by SF). SP-I also improves both SF and KW significantly, with an MAP of 0.1744. Comparing SP-I to SP, there is no clear winner between the two. SP-I is computationally more efficient, but is more sensitive to parameter setting simply because it involves a few more parameters.

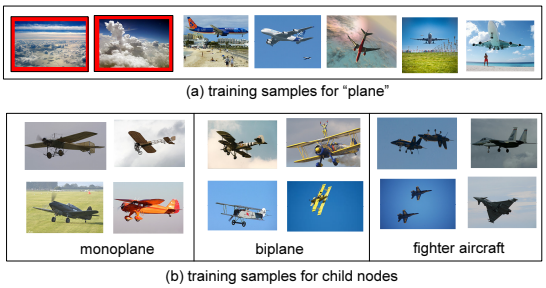

Figure 3: The automatically sampled training examples for "plane" and its child concepts. False positive examples are marked with red boxes.

Comparing our SP result (MAP=0.1774) with the publicly available VIREO-Web81* trained by expert labeled examples (MAP=0.2414) [12], there is still a performance gap. This is not surprising because the NUS-WIDE dataset was constructed two years ago by querying Flickr search engine — images downloaded at the same time may be visually more similar, resulting in the same data distribution (in feature space) between its official training and testing sets. Thus, we believe the performance gap between SP and VIREO-Web81 is partially due to the change of data characteristics of our newly downloaded training images. To verify this, we further conduct another experiment using VOC 2010 dataset. VOC 2010 provides labels for 20 semantic concepts. Among them, 8 concepts overlap with our constructed concept trees and VIREO-Web81. Thus, we evaluate the performance of the 8 concepts on the training set of VOC 2010 which has 10,103 images. Table 1 lists the experimental results. As shown in Table 1, the overall performance of SP (MAP=0.3284) is better than that of VIREO-Web81 (MAP=0.3230). This verifies our suspicion earlier that VIREO-Web81 is good on NUS-WIDE because

*http://vireo.cs.cityu.edu.hk/vireoweb81/

Table 1: Performance comparison of 8 concepts on VOC 2010 dataset.

Concept	SP	SP-I	SF	KW	VIREO-Web81
aeroplane	0.6628	**0.6760**	0.5671	0.4498	0.6519
bird	0.2266	0.2132	0.1924	0.1833	**0.2688**
boat	0.3041	0.3144	0.2461	0.2462	**0.3170**
car	0.3864	**0.4352**	0.3949	0.3274	0.3607
cat	**0.3712**	0.3700	0.3384	0.3253	0.3504
cow	0.1527	0.1344	**0.1630**	0.1556	0.1480
dog	0.3040	**0.3207**	0.3060	0.2706	0.2740
horse	**0.2194**	0.2067	0.1482	0.1513	0.2133
MAP	0.3284	**0.3338**	0.2945	0.2637	0.3230

of data domain over-fitting. In other words, our proposed approach SP is already able to produce training sets comparable to or even better than the expert labeled training sets. In addition, similar to the results on NUS-WIDE, in this experiment we also observe significant performance improvement from SP/SP-I over both SF and KW.

5. CONCLUSIONS

We have presented an ontology-based semantic pooling approach to enrich the coverage and diversity of freely sampled Web images for learning semantic concept detectors. By semantic pooling, consistent performance improvement is observed in our empirical studies on both NUS-WIDE and VOC 2010 datasets. In addition, when discounting the factor due to data distribution shift, the performance of classification models learnt using semantic pooling is comparable to that of detectors trained using expert labeled examples. Currently, we consider only semantic-level ontology for positive sample pooling. Future extension includes the pooling of positive and also negative examples by both semantic and visual co-occurrence relationships.

6. ACKNOWLEDGMENTS

The work described in this paper was fully supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (CityU 119709).

7. REFERENCES

[1] L. Xie et al. Probabilistic visual concept trees. In *ACM Multimedia*, 2010.
[2] T.-S. Chua et al. NUS-WIDE: A real-world web image database from national university of singapore. In *CIVR*, 2009.
[3] Y.-G. Jiang et al. Representations of keypoint-based semantic concept detection: A comprehensive study. *IEEE Trans. on Multimedia*, 12(1):42–53, 2010.
[4] J. Fan, Y. Shen, N. Zhou, and Y. Gao. Harvesting large-scale weakly-tagged image databases from the web. In *CVPR*, 2010.
[5] R. Fergus, H. Bernal, Y. Weiss, and A. Torralba. Semantic label sharing for learning with many categories. In *ECCV*, 2010.
[6] Y.-G. Jiang, C.-W. Ngo, and S.-F. Chang. Semantic context transfer across heterogeneous sources for domain adaptive video search. In *ACM MM*, 2009.
[7] L.-J. Li and L. Fei-Fei. OPTIMOL: automatic object picture collection via incremental model learning. *Int. J. of Computer Vision*, 2009.
[8] X.-R. Li, C. G. M. Snoek, M. Worring, and A. W.M. Smeulders. Social negative bootstrapping for visual categorization. In *ICMR*, 2011.
[9] B. Settles. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2009.
[10] J. Yang, R. Yan, and A. G. Hauptmann. Cross-domain video concept detection using adaptive svms. In *ACM Multimedia*, 2007.
[11] W. Zhibiao and M. Palmer. Verb semantic and lexical selection. In *ACL*, 1994.
[12] S. Zhu, G. Wang, C.-W. Ngo, and Y.-G. Jiang. On the sampling of web images for learning visual concept classifiers. In *CIVR*, 2010.