

Refining Image Annotation Using Contextual Relations Between Words

Yong Wang

Department of Computer Science
Queen Mary, University of London
Mile End Road, London, UK, E1 4NS
ywang@dcs.qmul.ac.uk

Shaogang Gong

Department of Computer Science
Queen Mary, University of London
Mile End Road, London, UK, E1 4NS
sgg@dcs.qmul.ac.uk

ABSTRACT

In this paper, we present a probabilistic approach to refine image annotations by incorporating semantic relations between annotation words. Our approach firstly predicts a candidate set of annotation words with confidence scores. This is achieved by the relevance vector machine (RVM), which is a kernel based probabilistic classifier in order to cope with nonlinear classification. Given the candidate annotations, we model semantic relationships between words using a conditional random field (CRF) model where each vertex indicates the final decision (true / false) on a candidate annotation word. The refined annotation is given by inferring the most likely states of these vertices. In the CRF model, we consider the confidence scores given by the RVM classifiers as local evidences. In addition, we utilise Normalized Google distances (NGD's) between two words as their contextual potential. NGD is a distance function between two words obtained by searching a pair of words using the Google search engine. It has a simple mathematical formulation with a foundation in Kolmogorov theory. We also propose a learning algorithm to tune the weight parameters in the CRF model. These weight parameters control the balance between the local evidence of a single word and the contextual relation between words. Our experiments on the Corel images demonstrate the effect of our approach.

Categories and Subject Descriptors

H.3.3 [Information Systems]: INFORMATION STORAGE AND RETRIEVAL

General Terms

Algorithms, Experimentation

Keywords

Image Annotation, Semantic Relation, Normalized Google Distance.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

CIVR'07, July 9–11, 2007, Amsterdam, The Netherlands.
Copyright 2007 ACM 978-1-59593-733-9/07/0007 ...\$5.00.

1. INTRODUCTION

With the prevalence of digital imaging devices such as webcams, phone cameras and digital cameras, image data accessible to computer users are now exponentially increased. To that end, an urgent issue is how to browse and retrieve this large volume of images. A possible solution is by content based image retrieval (CBIR) where two images are considered to be similar in semantics if they are close in the space of visual features. This is useful in the case where only image data are available, but limited in any semantic interpretation of images. Another approach is by annotating images with text keywords. With ideal annotations, image retrieval can be solved by well developed techniques in text retrieval. Unfortunately, most of the image data are unannotated or only partially annotated. Manual annotation is impractical for a large number of images. Although in the special case of web images, rough annotation can be extracted from the associated text such as the surrounding web text and titles, these annotations are very noisy because the associated text are not strongly correlated with the images and they completely ignore the visual content of images. Confronted with these challenges, automatic image annotation (AIA) aims to improve the performance of current image retrieval systems. The basic idea is that given a small dataset of images with annotations, we aim to train an image annotation system that is capable of annotating any new images automatically.

Current techniques for AIA can be summarized into two categories. First, image annotation is formulated as an image classification problem [2, 11, 5]. Specifically, each concept (annotation word) is viewed as a unique class label. For example, in a binary classification setting, we train a single classifier for each concept individually. To annotate new images, these classifiers are applied to a new image one by one and the final annotation results are obtained by ranking their posterior class probabilities [2]. Alternatively, we can train a multi-class classifier altogether. This can reduce the overall complexity of the classifier since there are many common features shared by different concepts [5]. The advantage of the classification approach is that we have various mature machine learning techniques available such as Naive Bayes, SVM, Bayes Point Machines and the 2-D multi-resolution Hidden Markov Model [11] etc. The second approach represents annotation words and images as features in different modalities. Image annotation is then realized by modeling the joint distribution of the visual features and the textual features together on the training data and predicting the missing textual features for a given new image. A common point of these approaches is to decompose

an image into a number of sub units such as image regions to generate the similar form of discrete features as that of text. For example, Barnard *et al.* [10] proposed a translation model for image annotation by viewing the images (a set of blobs) and textual words as two different languages. Other works along this line include the cross-media relevance model (CMRM) [7], the multiple Bernoulli relevance model (MBRM) [14], the latent Dirichlet allocation model (LDA) [1] and the continuous relevance model (CRM) [17].

However, there is a problem remained unsolved, *i.e.*, the *semantic gap*. All the above approaches try to annotate images based solely on their visual features. On the other hand, our human cognitive understanding of images is far removed from comparing low level visual features. Whilst effort is made to improve image annotation based on visual features alone, a number of works have begun to incorporate semantic knowledge from different sources into image annotation, complementary to the existing visual feature based approaches. To incorporate semantic knowledge into image annotation, there are three major issues: (i) How to represent semantic knowledge? There are mainly two types of representation considered by existing work, we call them *hierarchical* relationship [6, 15, 19] and *flat* relationship [9, 18] respectively. In a *hierarchical* relationship, concepts are related by parent-and-child relation. More general concepts are usually placed in the top level of the hierarchy. For example, the concept scene can be divided into indoor and outdoor. An outdoor scene can be further divided into several categories such as sunset, beach, city *etc.* The second form, *flat* relationship, considers pairwise relationships between two concepts. The focus here is the relatedness between two concepts. Two concepts can be related for a number of reasons. For example, *sunset* is more related with *outdoor* than with *indoor* and *desk* is more related with *chair* than with *lake*. (ii) How to build a knowledge database? The most popular method is to extract semantic knowledge from existing ontologies such as *WordNet* [13, 6, 15, 9], which is a semantic lexical database providing a number of possible relations between English words. Manual construction [19] or automatic learning [18] of semantic knowledge from external datasets has also been demonstrated. (iii) How to integrate the knowledge database into the annotation process. This is largely dependent on the stages of an annotation process where the knowledge is incorporated. For example, Srikanth *et al.* [15] introduced ontologies the preprocessing of visual features, *i.e.*, the visual vocabulary to improve the translation model. Wu *et al.* [19] modelled the causal strength between concepts by a directed acyclic graph (DAG) and the annotation is inferred bases on this graph structure. Jin *et al.* [9] and Wang *et al.* [18] integrate ontologies in the post processing stage to refine the initial annotations.

In this paper, we present a novel ontology based image annotation framework. Specifically, we propose a Conditional Random Field (CRF) [8] framework to incorporate the contextual relations between words as a kind of *flat* relationship extracted automatically from Google search. For a given image, our framework firstly predicts a candidate set of annotation words and then refine them by integrating contextual relationships between concepts. Our work is mostly related to that of Jin *et al.* [9] and Wang *et al.* [18], but differs in a number of ways as follows:

- (1) In the approach of Jin *et al.* [9], the confidence scores of the initial annotation word are discarded in the fol-

lowing refining stages, while we take the same strategy as that in Wang *et al.* [18] to keep these confidence scores and integrate them with the contextual knowledge on the annotation words in the refining stage.

- (2) The ontologies used in Jin *et al.* [9] is provided by a specific knowledge database (WordNet) constructed by human experts, while ours is extracted automatically from the world wide web (WWW). The WWW is probably the largest digital text database on the earth, and the latent semantic context information entered by millions of independent users average out to provide a meaningful sense of the contextual relations between words. Wang *et al.* [18] also attempted to extract the ontologies from the Internet, but our measure of contextual strength is motivated by the normalized Google distance (NGD) proposed by Cilibrasi and Vitanyi [4]. NGD has shown to be able to discover the meaning of words by the extensive experiments [4].
- (3) The approach of integrating ontologies into the annotation in [9] is based on a rule based method which removes the noisy words from the initial annotation, while in [18] they model the annotation refining process as a random walk with restarts. Different from [18] and [9], we model the assignment decision (true/false) of a word from the initial annotation as a conditional binary random field. The advantage of the CRF approach is that it can provide a coherent probabilistic fusion approach taking into account the individual probabilistic label assignment and the contextual relations between annotation words simultaneously. Moreover, we provide a learning process to tune the balance between the local evidence of each words and the contextual relation between words.

The rest of this paper is organized as follows: section 2 presents an overview of the key components of our automatic image annotation refining framework. Section 3 discusses the probabilistic binary image classification approach based on the bag of visual words model. In section 4 we introduce the normalized Google distance and compare it with the ontologies in WordNet. Section 5 describes the conditional random field which is used to fuse ontologies into image annotation. We present our experimental results in section 6 and conclude in section 7.

2. OVERVIEW OF THE FRAMEWORK

Fig.1 illustrates the flowchart of our framework of refining image annotation. There are three key components as described in the following,

- (1) Binary image classifiers
The first key component is the binary classifiers for each concept (here we use *a concept* and *an annotation word* interchangeably). They provide an initial set of annotation words for a given image. Each annotation word is assigned with a confidence score independently from other annotated words. This can be achieved by many of the existing automatic image annotation algorithms. The only requirement is that it can annotate images with confidence scores. In this work, we have proposed a bag of visual words approach for this purpose.

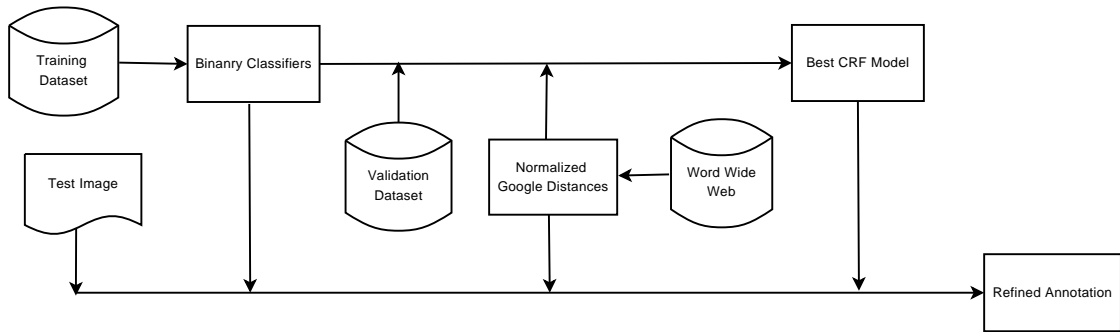


Figure 1: A framework for automatic image annotation. We refine candidate annotations generated using binary classifiers by further taking into account contextual relations between words modeled by a CRF model.

(2) NGD’s from WWW

The normalized Google distances (NGD’s) is the second key component of our framework. A NGD is a distance value between two words extracted from the WWW. A NGD indicates the contextual relation between two words, *e.g.*, the word *sky* is more likely to appear together with *clouds* than with *indoor*. Since they are some sort of general knowledge independent from the training images with annotation, we view it as a sort of ontologies.

(3) CRF Model

The CRF model is the third component in refining the image annotation. CRF is employed to integrate together the ontologies about textual data from WWW and the independent mappings from visual features to a single textual word into a coherent refined annotation. In this model, annotation words with strong positive contextual relation (*e.g.*, *sky* and *clouds*) can support each other while words with negative contextual relation (*e.g.*, *sky* and *indoor*) can contradict each other.

3. BINARY IMAGE CLASSIFICATION

Image annotation can be treated as an image classification problem as follows. For each concept, we take all the training images which have this concept as one of its ground truth labels as the positive samples and those which do not have as the negative samples. After feature extraction, a binary classifiers can be trained for each concept independently. Thus, the critical issues here is how to extract visual features from an image and which classification approach to take. In our work, we take an image as a bag of visual words and extract the histogram of visual words as the visual feature [3]. To address the second issue in image annotation, we adopt relevance vector machine (RVM) [16] as our probabilistic classifier. RVM has several advantages: (i) it is a probabilistic classifier. (ii) it works in the kernel trick as that in Support Vector Machines (SVM), so that it can map the original feature space to an implicit high dimensional feature space. (iii) it is an implementation of sparse Bayesian learning theory so that the relevance vectors involved in the final classifiers are sparser than the support vectors involved in a SVM classifier.

3.1 Image as a Bag of Words

We take an image as a bag of visual words. To this end, we firstly partition an image into a number of small patches by a regular grid. Some previous works have used the image blob representation. An image blob is obtained by an unsupervised image segmentation such as normalized cut segmentation algorithm. We do not take this approach mainly because, image segmentation is still an unsolved issue and the unsupervised segmentation bring about extra computation. A regular grid is much simpler to implement. Among the various local feature descriptors, SIFT [12] has demonstrated its out-performance in some benchmark evaluation. SIFT is a 128-D histogram descriptor of the gradient orientation over a patch. The gradients are center-weighted by a Gaussian weighting function. The weighted gradients are then accumulated over 4×4 sub-regions in each of which a histogram of gradient orientation in 8 orientation bins are computed. The final 128-D descriptor is the concatenated vector of the 16 8-D sub descriptors. SIFT descriptor has been shown to be a feature which is distinctive enough but robust to affine transformation, additive noise and change of illumination. The original SIFT descriptor is proposed for grayscale images. We believe color is an informative feature for image annotation. To combine SIFT with color features, we concatenate SIFT with a 6-D color descriptor for each patch, where the color descriptor include the means and variances of the R, G and B components.

With each image patch described by a color SIFT feature descriptor, we construct a visual vocabulary by running *k*-means clustering algorithm on the local feature descriptors extracted from a subset of the training images. The size of the vocabulary has some influence on the classification performance. Our initial experiments show that the classification performance is better with a larger vocabulary size. We call the resulted clustering centers as visual words, as an analogy to the text words. With this visual vocabulary each image patch can be assigned to one of the visual words. At this stage each image has been transformed into a unordered set of visual words. The histogram of the visual words in an image is then normalized. This is the final global feature descriptor of an image.

3.2 Probabilistic Classification

Relevance vector machine (RVM) is a supervised learning algorithm based on the Bayesian estimation theory. It is reported to yield nearly identical performance to, if not better than, that of SVM in several benchmark studies. A key feature of RVM is that it can yield a solution function

that depends on a much smaller number of kernel functions, called relevance vectors than the number of support vectors in SVM. This sparsity offers RVM good generalization ability and a simple structure in the classifier. What is more, RVM is a probabilistic classifier that it predicts the class label with a probability score.

Since RVM has a close relationship with SVM, we begin with the classification function of SVM. Given training samples $\mathbf{x}_i, i = 1, 2, \dots, N$, the classification function $f(\mathbf{x})$ of SVM is given by

$$f(\mathbf{x}) = \sum_{i=1}^N w_i K(\mathbf{x}, \mathbf{x}_i) + w_0$$

where $\mathbf{w} = (w_0, w_1, w_2, \dots, w_N)$ is the model weights and $K(\cdot, \cdot)$ is a kernel function. In RVM, this linear model is generalized by applying the logistic sigmoid function $\sigma(f) = 1/(1 + e^{-f})$ to $f(\mathbf{x})$ and the probability of the class label y_i given the training sample x_i is given by

$$P(y_i|\mathbf{x}_i; \mathbf{w}) = \sigma\{f(\mathbf{x}_i)\}^{y_i} [1 - \sigma\{f(\mathbf{x}_i)\}]^{1-y_i} \quad (1)$$

According to Tipping [16], the parameters \mathbf{w} , can be determined using Bayesian estimation. To this end, a sparse prior is introduced on w_i . Specifically, these parameters are assume to be statistically independent and each obeys a zero-mean Gaussian distribution with variance α_i^{-1} , i.e.,

$$P(\mathbf{w}|\boldsymbol{\alpha}) = \prod_{i=0}^N \mathcal{N}(w_i|0, \alpha_i^{-1})$$

where $\boldsymbol{\alpha} = (\alpha_0, \alpha_1, \dots, \alpha_N)$. The posterior likelihood of \mathbf{w} can then be formulated as

$$\mathcal{L}(\mathbf{w}) = P(\mathbf{w}|\boldsymbol{\alpha}) \prod_{i=1}^N P(y_i|\mathbf{x}_i; \mathbf{w})$$

Unlike the maximum a posterior (MAP) estimation, the hyper-parameters $\boldsymbol{\alpha}$ are not manually chosen but estimated automatically through an iterative procedure (refer to [16]). In the real implementation, most of the variance parameters α_i goes to infinity, indicating the weight w_i is peaked around zero.

4. ONTOLOGIES FROM WWW

4.1 Semantic Knowledge from WordNet

An ontology is a data model that represents a domain. It is used to reason about the objects and the relations between them in that domain. Over the years, intensive effort has been made in building ontologies in a computer-digestible form. One of such examples is WordNet, which is trying to establish semantic relations between common objects. WordNet is organized in a hierarchical structure in that each word is provided with at least one synset (set of synonyms). Most synsets are connected to other synsets via a number of semantic relations such as *hypernyms* (Y is a hypernym of X if every X is a (kind of) Y), *hyponyms* (Y is a hyponym of X if every Y is a (kind of) X), *antonym* (opposite meaning of each other) *etc.* Although these semantic knowledge is readily available, how to use them in an application is a challenging issue. As for refining image annotation, the most popular way is to simplify the semantic knowledge into the semantic similarity between concepts.

This simplification has two limitations: Firstly, a lot of detailed semantic knowledge present in WordNet has been lost in this simplification. By the semantic similarity between concepts, we know how much they are related but we do not know the actual reason. Secondly, WordNet does not provide a quantitative measure of the semantic similarity between concepts. So various measures [9] have been proposed to derive the semantic similarity from WordNet, but none of these measures are perfect and sometimes they contradict with human ratings.

4.2 Normalized Google Distance

Realizing the limitations of computing the semantic similarity from WordNet. we take a different approach. Instead of relying on WordNet, we resort to the largest text database, i.e., the WWW, to mine the semantic relation between concepts automatically. The method, called the normalized Google distance (NGD) is firstly proposed by Cilibrasi *et al.* [4]. NGD is a measure of semantic relation between two words obtained by just typing them as the search term in Google's search engine. It has solid foundation in Kolmogorov complexity theory while has a concise mathematical formulation as follows,

$$\text{NGD}(w_1, w_2) = \frac{\max\{\log f(w_1), \log f(w_2)\} - \log f(w_1, w_2)}{\log M - \min\{\log f(w_1), \log f(w_2)\}} \quad (2)$$

where w_1 and w_2 represent the two words in consideration. $f(w_1)$ and $f(w_2)$ are the numbers of the webpages returned by Google search engine when typing w_1 and w_2 as the search term respectively. $f(w_1, w_2)$ is the number of webpages returned when typing w_1 and w_2 together as the search term. M is the index size of Google. To understand how the formulation in Eq.(2) comes out, firstly we define the semantic relation between two words w_1 and w_2 as the following distance function,

$$D_1(w_1, w_2) = \min\{p(w_1|w_2), p(w_2|w_1)\} \quad (3)$$

where

$$p(w_1|w_2) = \frac{p(w_1, w_2)}{p(w_2)} = \frac{f(w_1, w_2)/M}{f(w_2)/M} = \frac{f(w_1, w_2)}{f(w_2)}$$

and $p(w_2|w_1)$ is similar. The same D_1 distance function as that in Eq.(3) has been taken to measure the semantic relation between concepts by Wang *et al.* [18]. However, the direct use of the D_1 distance does not give good results in experiments according to [4]. One reason is that the difference among small probabilities have increasing significance when smaller probabilities are involved. Another reason is that two notions that have very small probabilities each and have D_1 distance ε are much less similar than two notions that have much larger probability and have the same distance. To resolve the first problem, we take the negative logarithm of the items being minimized, resulting in

$$D_2(w_1, w_2) = \max\{\log 1/p(w_1|w_2), \log 1/p(w_2|w_1)\}$$

To resolve the second problem, we normalize $D_2(w_1, w_2)$ with the maximum of $\log 1/p(w_1)$ and $\log 1/p(w_2)$. Altogether the following normalized distance is obtained

$$D_3(w_1, w_2) = \frac{\max\{\log 1/p(w_1|w_2), \log 1/p(w_2|w_1)\}}{\max\{\log 1/p(w_1), \log 1/p(w_2)\}}$$

Finally replace the probabilities $p(w_1), p(w_2), p(w_1|w_2)$ and $p(w_2|w_1)$ with the corresponding frequencies divided by the

index size M , we can get the formulation in Eq.(2). In summary, NGD has several attributes: (a) It is a distance value ranging from 0 to ∞ but most of the NGD's are between 0 and 1; (b) The smaller the NGD is, the stronger the semantic relation is. (c) NGD is scale robust, meaning that it is relatively stable with the change of the index size M . (d) NGD is NOT a metric. It does not obey the triangle inequality rule.

The original NGD is obtained by analyzing the search statistics of generic webpages. For the problem of image annotation, we are more interested in the text description of images. This can be viewed as a special domain NGD. A possible solution is searching the terms on the Google Images instead of the Google search. But Google Images are too noisy and the index size is much smaller than Google search. So we propose an intuitive method by searching a term w only in a subset of webpages which contain at least one word from a list of anchor words (*image, images, photo, photos, picture and pictures*). The underlying intuition is that these webpages are more likely to be related to image data. This method has another advantage: M is now reliable to be estimated by counting the number of webpages containing any anchor word. This is important since the full index size of Google is a miracle and it is changed as time goes on.

4.3 Compare NGD to WordNet

Since both NGD and WordNet provide a semantic similarity between concepts, it is worthwhile to have a comparison. From the definition of NGD, we can find that NGD is actually a measure of the contextual relation, while WordNet focuses on the semantic meaning of words. Nevertheless, in [4], the authors have shown that how the semantic relation derived from NGD can be consistent with that of WordNet. To know more details, readers are referred to [4]. Here we give some examples to have an intuitive understanding of their difference. Firstly, we show that NGD pays more attention to contextual relations rather than semantics. An example is given by three concepts: *mountain, rock* and *lake*. From the perspective of WordNet, the semantic relation between *mountain* and *rock* is stronger than that between *mountain* and *lake* because rocks constitute a mountain, but $NGD(mountain, lake) = 0.3127$, which is less than $NGD(mountain, rock) = 0.4498$. The reason is that mountain and lake appear more frequently together in the text description of images. Secondly, NGD suppresses synonyms but WordNet prefers to synonyms. An example is given by the words *by* and *with*. As these two words are exchangeable in many occasions, either one of them appear in an occasion but not both of them. This results in a NGD value of 3.51, which means they strongly contradict with each other. However, their semantic relation in WordNet is very strong since they are synonyms to each other. Thirdly, NGD is more subjective than WordNet. An example is given by the relation of (*USA, poor*) and (*Kenya, poor*). The two NGD's are 0.5639 and 0.9893 respectively. But there is no reason to believe USA is even poorer than Kenya. The explanation is that many webpages contain both of the words *USA* and *poor* for whatever reasons. In short, both of NGD and WordNet can provide some measure of words relatedness but have different focuses.

5. REFINEMENT BY ONTOLOGY

The probabilistic classifiers described in section 3 can pre-

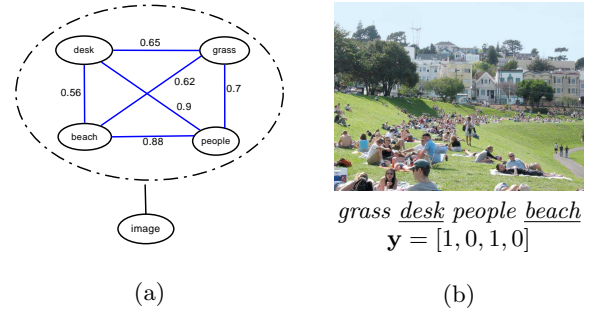


Figure 2: (a) An illustration of the CRF model for annotation refining. Each vertex represents a binary random variable. The numbers besides the edges indicate the contextual strengths. (b) An illustration of a refined annotation represented by an indicator vector. The first line of words is the candidate annotation, the underlined words are removed in the refined annotation.

dict a candidate set of annotation words with confidence scores for a given image. These confidence scores are solely based on the visual features. While if we take an annotation as a concise form of text description of images, they are more than several disjoint words. Particularly, like a normal text paragraph, the context plays an important role in the language usage. Thus, we propose a method to model this context and refine the image annotation by a global decision considering not only a single annotation word but also their pairwise contextual relations. Specifically, we propose a probabilistic framework based on the conditional random field to refine the annotation. In this framework, the final decision of a candidate annotation word is represented by a binary random variable where *true* means it is assigned to the image finally and *false* otherwise. We view the confidence score of each word provided by the binary classifier as a sort of local evidence of its corresponding random decision variable. Furthermore, we view the semantic distance between the annotation words as a sort of potential between these random variables. The CRF modeling of contextual relation between words is illustrated in Fig. 2.

5.1 Conditional Random Field (CRF)

A conditional random field (CRF) is an undirected graphical model in which each vertex represents a random variable whose distribution is to be inferred, and each edge represents a dependency between two random variables. In a CRF, the distribution of each discrete random variable y_i in the graph is conditioned on an input sequence \mathbf{x} . In mathematics, the conditional probability of $\mathbf{y} = (y_1, y_2, \dots, y_n)$ given \mathbf{x} is formulated as,

$$P(\mathbf{y}|\mathbf{x}) = \frac{e^{\psi(\mathbf{y}, \mathbf{x}; \Theta)}}{\sum_{\mathbf{y}'} e^{\psi(\mathbf{y}', \mathbf{x}; \Theta)}}$$

where

$$\psi(\mathbf{y}, \mathbf{x}; \Theta) = \sum_i \sum_k \theta_k^1 f_k^1(y_i, i, \mathbf{x}) + \sum_{i,j} \sum_l \theta_l^2 f_l^2(y_i, y_j, i, j, \mathbf{x}) \quad (4)$$

is the potential function. i, j are used to index the vertices. $f_k^1(y_i, i, \mathbf{x})$ and $f_l^2(y_i, y_j, i, j, \mathbf{x})$ are the node feature function and edge feature function respectively (multiple fea-

ture functions can be used), which are application dependent. $\Theta = \{\theta^1, \theta^2\}$ are the model parameters to learn. For a graphic model with complex structure, the learning process will be expensive. What is more, the training data is usually sparse compared to the high-dimension of the parameter vector. This means that the model learned from the training data does not have good generalization ability in the future new data. Thus, we keep the framework of CRF but reduce the number of parameters to learn by setting most of the parameters as a prior knowledge. Specifically, we change the potential function in Eq. (4) to the following,

$$\psi(\mathbf{y}, \mathbf{x}; \Theta) = \alpha_1 * \sum_i \omega^1(y_i, i, \mathbf{x}) + \alpha_2 * \sum_{i,j} \omega^2(y_i, y_j, i, j)$$

where, ω^1 indicates the local evidence of the state of y_i . It is dependent on the image observation \mathbf{x} . ω^2 is the a prior parameters indicates the contextual potential between the states of two variables y_i and y_j . We take the local evidence as the logarithm of the confidence score given by a binary classifier, i.e.,

$$\omega^1(y_i = 1, i, \mathbf{x}) = \log P_{RVM}(c(y_i) = 1|\mathbf{x}) \quad (5)$$

$$\omega^1(y_i = 0, i, \mathbf{x}) = \log [1 - P_{RVM}(c(y_i) = 1|\mathbf{x})] \quad (6)$$

Here, we use $c(y_i)$ to indicate the concept actually represented by y_i and P_{RVM} is the classification probability according to Eq.(1). We assume the contextual potential is independent from \mathbf{x} and can be obtained as general knowledge provided by the NGD. We convert a NGD into contextual potentials by the following,

$$\omega^2(y_i, y_j, i, j) = \begin{cases} -\log \text{NGD}(c(y_i), c(y_j)) & \text{if } y_i = y_j = 1 \\ 0 & \text{otherwise} \end{cases}$$

The underlying assumption is that only when a concept is present in the image, it has influence to the other concepts. In this simplified CRF model, the parameters to learn, $\Theta = \{\alpha_1, \alpha_2\}$, are the weight parameters to control the balance between local evidence and contextual potential.

5.2 Parameter Estimation

It is hard to manually choose the weight parameters $\Theta = \{\alpha_1, \alpha_2\}$ since the local evidence and contextual potentials come from difference sources. So we propose a learning algorithm which can estimate the two weights from a validation set. Suppose we have a training set and a validation set, the sample of each set is an image with its ground truth annotation words. The algorithm goes as follows: Firstly, we train an image classifier for each concept separately on the training set, as described in section 3. Secondly, for each image in the validation set, these classifier are applied to produce a confidence score for each word from the annotation vocabulary. Thirdly, the annotation words with the top 10 confidence scores or those appear in the ground truth annotation are selected as the candidate set of words. Finally, construct the indicator vector \mathbf{y} for each image and learn Θ by maximizing the log posterior of Θ as

$$\mathcal{L}(\Theta) = \sum_k \log P(\mathbf{y}^k|\mathbf{x}^k) - \frac{\alpha_1^2}{2\sigma^2} - \frac{\alpha_2^2}{2\sigma^2}$$

where k indexes the samples in the validation set. We fit a Gaussian prior with parameter σ on the Θ to prefer to small values of α_1 and α_2 . The indicator vector is constructed in such a way that y_i is *true* if the corresponding concept

appears among the words with top confidence scores and also in the ground truth labels, otherwise it is *false*.

We maximize $\mathcal{L}(\Theta)$ by the deepest gradient descent algorithm. To this end, we need to calculate the first derivative of $\mathcal{L}(\Theta)$ with respect to α_1 and α_2 . Here we focus on the derivative of the log likelihood of a single sample. This is given by the following equation,

$$\begin{aligned} \frac{\partial \log P(\mathbf{y}|\mathbf{x})}{\partial \alpha_1} &= \sum_i \omega^1(y_i, i, \mathbf{x}) - \sum_{\mathbf{y}'} P(\mathbf{y}'|\mathbf{x}) \sum_i \omega^1(y_i', i, \mathbf{x}) \\ &= \sum_i \omega^1(y_i, i, \mathbf{x}) - \sum_i \sum_{\mathbf{y}'} P(\mathbf{y}'|\mathbf{x}) \omega^1(y_i', i, \mathbf{x}) \\ &= \sum_i \omega^1(y_i, i, \mathbf{x}) - \sum_i \sum_{y_i'} P(y_i'|\mathbf{x}) \omega^1(y_i', i, \mathbf{x}) \quad (7) \end{aligned}$$

Similarly, the first derivative with respect to α_2 can be derived as,

$$\begin{aligned} \frac{\partial \log P(\mathbf{y}|\mathbf{x})}{\partial \alpha_2} &= \\ \sum_{i,j} \omega^2(y_i, y_j, i, j) - \sum_{i,j} \sum_{y_i', y_j'} P(y_i', y_j'|\mathbf{x}) \omega^2(y_i', y_j', i, j) \quad (8) \end{aligned}$$

The two marginalized probabilities, $P(y_i|\mathbf{x})$ and $P(y_i, y_j|\mathbf{x})$, can be calculated by belief propagation [8]. Given Eq. (7) and (8), the computation of the first derivative of $\mathcal{L}(\Theta)$ is straightforward.

5.3 Automatically Refine Image Annotation

Given the learned best parameters $\hat{\Theta}$ and a candidate annotation for an image, we can refine this annotation by inferring the most likely state of each indicator variable by the marginalized probability

$$y_i^* = \arg \max_{y_i} P(y_i|\mathbf{x}; \hat{\Theta}) \quad y_i \in \{0, 1\} \quad (9)$$

Alternatively, we can infer the most likely state of the indicator vector all together by

$$\mathbf{y}^* = \arg \max_{\mathbf{y}} P(\mathbf{y}|\mathbf{x}; \hat{\Theta})$$

Both of these two approaches have its own advantages. By the first approach, we can rank the annotation words by their marginalized probabilities. By the second, we can decide the length of the annotation automatically. In this work we have taken the first method.

6. EXPERIMENT

The dataset used in our experiment is the same subset of Corel images as that used in [10]. It contains 5000 color images, each of which has 1 ~ 5 caption words. There are totally 374 caption words. We partition the whole dataset into three subsets: 4000 images as the training set, 500 images as the validation set and 500 images as the testing set.

The first step is training a RVM classifier for each concept. We begin from extracting the color-SIFT descriptors from all the images. We then collect 10,0000 color-SIFT feature descriptors from a subset of the training images and clustering them into 1000 clusters by running k -means clustering algorithm. This is our visual vocabulary. After this, we extract a normalized histogram of the visual words for each image. At this stage, we are ready to construct a training set for each concept and train a RVM classifier. These binary

classifiers are then applied to each image in the validation set and testing set. The result of this is a candidate set of annotation words with confidence score for each image. The second step is training the weight parameters of the CRF model on the validation set. To this end, we construct the training data (the indicator vectors) for the CRF model as discussed in section 5. We use Google search engine to compute the pairwise NGD of all the 374 words. These NGD values are stored and fitted to the CRF model together with the confidence scores provided by the RVM classifiers. We initialized the weight parameters as $\alpha_1 = \alpha_2 = 0.1$ and run the iterative optimization algorithm. To obtain the refined annotation on the test set, the best weight parameters, the pairwise NGD values, the confidence scores of the candidate annotation are fitted together to the CRF model. The top 10 words from the candidate annotation of each image is chosen as the basis of refinement. We rerank these 10 words by their marginalization probability as that in Eq. (9).

To evaluate the performance of the refined annotation, we select the top ranked words from the candidate annotation and refined annotation words respectively and compared the precision and recall values on the top 50 words with the best performance by RVM. The recall value of a specific vocabulary word w is calculated as the number of images correctly annotated with w divided by the number of images with w annotated in the ground truth annotation. The precision value is calculated as the number of images correctly annotated with w divided by the total number of images annotated with w . We compare our method with the Random Walk with Restarts (RWR) approach of Wang et al. [18]. Both of the refining methods take the probabilistic annotations by RVM as the initial annotations. Fig. 3 shows averaged the precision and recall values with different number of top words selected. From the figure we can find that Random Walk with Restarts (RVM_RWR) can improve the annotation performance of RVM but our method outperforms RWR consistently. The less top words selected as annotation, the better the improvement. This is because the contextual relation between correctly annotated words can enhance their rankings. Fig. 4 shows some sample images and their automatic annotations. From these samples, we can find that words which have not been selected by the RVM approach have chance to be selected after refinement because of the support from other annotation words (e.g., *lake* in the first example). On the other hand, words have been selected by the RVM approach are possible to be excluded in the refined annotation because of its negative contextual relation with other words (e.g., *cat* in the forth sample).

7. CONCLUSIONS

In this paper, we presented a probabilistic framework to refine image annotation by incorporation semantic relation between annotation words. Our framework firstly predicts a candidate set of annotation words with confidence scores. This is achieved by the relevance vector machine, which is a probabilistic classifier working in the kernel trick. Given the candidate annotation, we model the context relation between words by a conditional random field model. In the CRF model, each vertex indicate the final decision of a candidate annotation word. Its local evidence is given the the confidence score produced by the RVM classifiers. Its con-

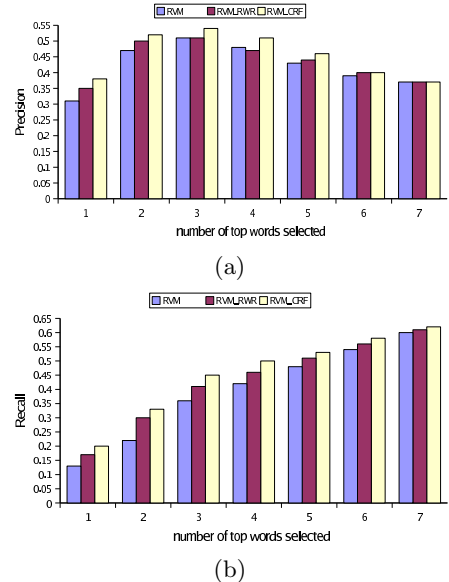


Figure 3: Compare the performances of the annotations generated by RVM and the refined annotations generated by Random Walk with Restarts (RVM_RWR) and CRF (RVM_CRF). (a) the average precision by selecting different number of top ranked words. (b) the average recall by selecting different number of top ranked words.

text relation is given by the normalized Google distance. The NGD is a distance function between two words obtained by just typing these two words as the search terms in the Google search engine. It has a simple mathematical formulation but has a solid theory foundation. We also propose a learning algorithm to learn the weight parameters in the CRF model. These weight parameters control the balance between the local evidence of a single word and the context relation between words. Our experiment results on the Corel images have shown the effect of our approach. We believe that the semantic knowledge from different sources are complementary to each other. In the future, we will research how to combine together the semantic knowledge from NGD and other sources such as WordNet to achieve a better annotation performance.

8. REFERENCES

- [1] D. M. Blei and M. I. Jordan. Modeling annotated data. In *Proceedings of ACM SIGIR*, pages 127–134, 2003.
- [2] E. Chang, K. Goh, G. Sychay, and G. Wu. CBSA: content-based soft annotation for multimodal image retrieval using Bayes point machines. *CSVT, IEEE Transactions on*, pages 26–38, January 2003.
- [3] Chris Dance et al. Visual categorization with bags of keypoints. In *ECCV International Workshop on Statistical Learning in Computer Vision*, 2004.
- [4] R. Cilibrasi and P. Vitanyi. The google similarity distance. *Knowledge and Data Engineering, IEEE/ACM Transactions on*. To appear, also available at <http://cilibras.com/>.

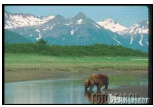








		
RVM: building mountain garden water people	RVM: grass landscape tiger people	RVM: water birds building ocean landscape
RVM_CRF: moun- tain lake building ocean	RVM_CRF: grass plant tiger landscape	RVM_CRF: water ocean boat bird building
		
RVM: water flower rock cat tree	RVM: grass plant people flower animal	RVM: building water mountain ocean clouds
RVM_CRF: water rock tree beach flower	RVM_CRF: grass plant flower garden people	RVM_CRF: building mountain water clouds sky
		
RVM: snow building water sand	RVM: building landscape water beach ocean	RVM: animals rock tree building people
RVM_CRF: snow mountain water lake ice	RVM_CRF: building landscape ocean sky city	RVM_CRF: animals tree rock grass building

Figure 4: Some sample images with their automatic annotations. Annotations in the first row (RVM) are the candidate annotation generated by RVM's. Annotations in the second row (RVM_CRF) are the refined annotation by the CRF model.

- [5] G. Carneiro, and N. Vasconcelos. Formulating semantic image annotation as a supervised learning problem. In *Proceedings of IEEE CVPR*, June 2005.
- [6] Y. Gao and J. Fan. Incorporating concept ontology to enable probabilistic concept reasoning for multi-level image annotation. In *Proceedings of ACM Workshop on MIR*, pages 79–88, 2006.
- [7] J. Jeon et al. Automatic image annotation and retrieval using cross-media relevance models. In *Proceedings of ACM SIGIR*, pages 119–126, 2003.
- [8] J. Lafferty, A. McCallum and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of ICML*, 2001.
- [9] Y. Jin, L. Khan, L. Wang, and M. Awad. Image annotations by combining multiple evidence & wordNet. In *Proceedings of ACM MULTIMEDIA*, pages 706–715, 2005.
- [10] Kobus Barnard et al. Matching words and pictures. *J. Mach. Learn. Res.*, 3:1107–1135, 2003.
- [11] J. Li and J. Z. Wang. Automatic linguistic indexing of pictures by a statistical modeling approach. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25(9):1075–1088, 2003.
- [12] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.
- [13] G. A. Miller. Wordnet: a lexical database for english. *Commun. ACM*, 38(11):39–41, 1995.
- [14] S. L. Feng et al. Multiple bernoulli relevance models for image and video annotation. In *Proceedings of IEEE CVPR*, June 2004.
- [15] M. Srikanth, J. Varner, M. Bowden, and D. Moldovan. Exploiting ontologies for automatic image annotation. In *Proceedings of SIGIR*, pages 552–558, 2005.
- [16] M. E. Tipping. Sparse bayesian learning and the relevance vector machine. *J. Mach. Learn. Res.*, 1:211–244, 2001.
- [17] V. Lavrenko, R. Manmatha and J. Jeon. A model for learning the semantics of pictures. In *Proceedings of NIPS*, 2003.
- [18] C. Wang, F. Jing, L. Zhang, and H.-J. Zhang. Image annotation refinement using random walk with restarts. In *Proceedings of ACM MULTIMEDIA*, pages 647–650, 2006.
- [19] Y. Wu, E. Y. Chang, and B. L. Tseng. Multimodal metadata fusion using causal strength. In *Proceedings of ACM MULTIMEDIA*, pages 872–881, 2005.