Multiple label prediction for image annotation with multiple kernel correlation models

Oksana Yakhnenko and Vasant Honavar
Computer Science Department
Iowa State University
{oksayakh, honavar}@cs.iastate.edu

Abstract

Image annotation is a challenging task that allows to correlate text keywords with an image. In this paper we address the problem of image annotation using Kernel Multiple Linear Regression model. Multiple Linear Regression (MLR) model reconstructs image caption from an image by performing a linear transformation of an image into some semantic space, and then recovers the caption by performing another linear transformation from the semantic space into the label space. The model is trained so that model parameters minimize the error of reconstruction directly. This model is related to Canonical Correlation Analysis (CCA) which maps both images and caption into the semantic space to minimize the distance of mapping in the semantic space. Kernel trick is then used for the MLR resulting in Kernel Multiple Linear Regression model. The solution to KMLR is a solution to the generalized eigen-value problem, related to KCCA (Kernel Canonical Correlation Analysis). We then extend Kernel Multiple Linear Regression and Kernel Canonical Correlation analysis models to multiple kernel setting, to allow various representations of images and captions. We present results for image annotation using Multiple Kernel Learning CCA and MLR on Oliva and Torralba [21] scene recognition that show kernel selection behaviour.

1. Introduction

In image annotation problem, the task of assigning a caption (keywords that describe the contents of an image) is significantly more challenging than in a traditional image classification problem for which standard supervised learning methods can be applied. This is due to the fact that the training set in image annotation is a dataset of images with their associated captions, i.e., words that describe the image content without specifically labeling the individual objects, events, or other interesting aspects of the image. The increasing amount of multi-media data presents us with a challenging task of information retrieval and organization. With the increasing availability of image data, there arises the need for image annotation and associating the images with keywords that correspond to the objects, events, or scenes present in the images. Examples of such tasks can be found in organization of electronic medical records and on social network communities such as Flickr and Facebook. Image annotation also allows to add semantical meaning to the image.

The work in image annotation (see discussion in Section 2) relies heavily on the assumption that image features b and image words w are independent given some hidden variable z so that $p(w, b|z) = p(w|z)p(b|z)$ (the variable z means some semantic space in the models described in [10, 4] and images in the models described in [17, 23, 12]). Such probabilistic model is directly related to Canonical Correlation Analysis as shown by Bach as Jordan [2]. Canonical Correlation Analysis and it's kernel variant (Kernel CCA) [14] has been used in the past for image annotation. KCCA finds a projection of images and their captions to a semantic space to maximize the correlation of the projections. Application of KCCA in image annotation and the closely related problem of cross-language retrieval problem i.e., the task of retrieving text written in a language different from the language of the user's query [24], rely on mate-based retrieval to obtain their best reported results. In mate-based retrieval, the query image (or a document in one language) is used to compute the correlation with all the captions associated with images in the training set (or documents in other language) and the caption with the highest correlation is returned in response to the query. Using the learned KCCA model directly to generate a response to the query has been shown to yield worse results than those obtained using mate-based retrieval [24, 15].

Against this background, this paper explores a more direct approach to the image annotation problem. We use the Kernel Multiple Linear Regression (KMLR), model which can be viewed as a discriminative counterpart of KCCA.

978-1-4244-3993-5/09/$25.00 ©2009 IEEE

KMLR generalizes the Multiple Linear Regression (MLR) model [6] by incorporating kernel functions. Hence it can model and learn non-linear relationships between the inputs and outputs. MLR itself is an extension of linear regression to a setting with a multivariate output variable (an output variable with two or more dimensions). This model is similar to a maxtrix factorization model used by Loeff and Farhadi [18] and it uses ℓ_2-loss instead of hinge-loss.

Image representation is a challenging problem. Images can be represented in a variety of ways: from color histograms and to texture descriptors [5], to graphs over image segments [13], to bags-of-visual-words using keypoints detectors or concentric circles from grid sampling [11, 7].

We therefore propose a general framework for KCCA and KMLR to allow for multiple kernel learning. Multiple Kernel Learning finds optimal linear combination of the kernels with $\ell - 1$ norm constraints on the weights [3]. We propose a simple iterative algorithm, similar in spirit to SimpleMKL for SVM [22] and our results demonstrate that KMLR with Multiple Kernel Learning picks the best kernel when using bag-of-visual-words kernels built from keypoints obtained at various keypoint size ($r = 4, 8, 12$ and 16 pixels) during grid sampling.

We evaluate the CCA, MLR and their kernel extensions on the Natural Scene data [21] with approximately 2700 images objects drawn from around 300 possible object classes using various image representations. We then apply Mutliple Kernel learning framework to address the problem of kernel selection. Evaluation of the models on more image annotation datasets and comparison with previous work is currently work in progress.

This paper is organized as follows: we describe the problem of image annotation and related work in Section 2. We describe kernel correlation methods, namely KCCA and KMLR is Section 3. We then introduce the general framework that extends KCCA and KMLR to allow for Multiple Kernel Learning in Section 4. In Section 5 we present results of the models using kernels constructed from bag-of-visual words derived from grid-sampling using circular support of various radii, and show results for kernel selection using MKL-MLR and MKL-CCA. We conclude with discussion in Section 6.

2. Related Work

Duygulu et al. [10] suggested an expectation-maximization model for image annotation. The model is based on that used for machine translation. The model assumes segmentation of the images and the space of segments is treated as one "language" and the space of words is teated as the second "language". The model assumes that the segmentation of the image produces regions that correspond to the actual objects, however this is not always the case for the segmentation algorithms as in some cases the resulting segments span over multiple objects, or the segments partition a given object into parts. Barnard et al. [4] explored a wide range of models, including a Multi-Modal Latent Dirichlet Allocation model, which attempts to correlate image blobs and words in a semantic space. Jeon et al. [17] suggested that the model [10] is analogous to a cross-media relevance model unlike and they showed that their model had better results on the same dataset. The model was then extended to allow for continuous features in [23] and to allow words be sampled from an underlying Bernoulli model [12]. One common feature over these all these models is the assumption that keywords w and image blobs b are independent given some hidden variable z so that $p(w, b|z) = p(w|z)p(b|z)$ (the variable z means some semantic space in the models described in [10, 4] and images in the models described in [17, 23, 12]). Carneiro et al. [8] introduced a Supervised Multiclass Labelling framework which bypasses the modelling of the hidden variable z and models $p(b|w)$ directly, and their results showed the best performance across the previous work. Makadia et al. [20] suggested a very simple approach to image annotation in which they used k-NN model to obtain the most similar images to an unknown image, and to this image they assigned a set of keywords from the nearest neighbors. This approach was shown to outperform all the other approaches on the datasets considered.

3. Kernel models for data with multiple views

We begin by describing the general idea for modeling data with multiple views, and then describe CCA and its discriminative counterpart MLR. We then proceed to describe their extensions to kernel space, KCCA and KMLR, respectively.

Let $\mathbf{x} \in R^n$ and $\mathbf{y} \in R^m$ be the two views of the same instance s. For example, s can be some event, $\mathbf{x}$ can be a photograph of s and $\mathbf{y}$ can be a textual description of s. In order to model the relationship between x and y the goal is to find linear transformation $f : R^n \rightarrow W$ of $\mathbf{x}$ and $h : R^m \rightarrow W$ $\mathbf{y}$ into some lower-dimensional space $W \subset R^k$ where $k < n$ and $k < m$. Canonical Correlation Analysis seeks this projection as to minimize the distance between the projection of $\mathbf{x}$ and projection of $\mathbf{y}$ in W: $\min_{h,f} ||f(\mathbf{x}) - h(\mathbf{y})||^2$.

Multiple Linear Regression, on the other hand attempts to reconstruct y directly: it first projects $\mathbf{x}$ into the lower dimensional space W using linear transformation $f : R^n \rightarrow W$, and then reconstructs the projection into the space R^m using some linear transformation $g : W \rightarrow R^m$ to minimize the error of the reconstruction of $\mathbf{y}$: $\min_{f,g} ||y - g(f(x))||$. Both methods can be reformulated in the dual form, which allows the use of the "kernel trick".

We now proceed to describe the methods in detail.

3.1. Preliminaries

We begin with establishing notation that we will use in the rest of the paper. Let $D = \begin{bmatrix} (\mathbf{x}_1, \mathbf{y}_1) \\ ... \\ (\mathbf{x}_l, \mathbf{y}_l) \end{bmatrix}$ be a sample of size l such that $\mathbf{x}_i \in R^n$ and $\mathbf{y}_i \in R^m$. Let $D_x = \begin{bmatrix} \mathbf{x}_1 \\ ... \\ \mathbf{x}_l \end{bmatrix}$ and $D_y = \begin{bmatrix} \mathbf{y}_1 \\ ... \\ \mathbf{y}_l \end{bmatrix}$ (so that D_x is $l \times n$ matrix and D_y is $l \times m$ matrix). Let $C_{xx} = D_x^\top D_x$ be the correlation matrix for x-view of the data, $C_{yy} = D_y^\top D_y$ be the correlation for the y-view of the data and $C_{xy} = D_x^\top D_y$ be the cross-correlation matrix for x- and y-views.

Let ϕ be the feature space mapping of $\mathbf{x}$: $\mathbf{x} = (x_1 ... x_n) \rightarrow \phi(\mathbf{x}) = (\phi_1(\mathbf{x}), ... \phi_N(\mathbf{x}))$ [9] and φ be the feature space mapping of $\mathbf{y}$. Let $\phi(D_x) = \{\phi(\mathbf{x}_1), ..., \phi(\mathbf{x}_l)\}$ be the feature-space mapping applied to the x-view of the sample and $\varphi(D_y) = \{\varphi(\mathbf{y}_1), ..., \varphi(\mathbf{y}_l)\}$ be the feature space mapping for the y-view of the data. We can now define a kernel function $K(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle$ where $\langle \cdot, \cdot \rangle$ is the dot product (similarly for $\mathbf{y}$). We use $K_x = \phi(D_x)^\top \phi(D_x)$ and $K_y = \varphi(D_y)^\top \varphi(D_y)$ to denote the kernel matrices (symmetric, positive, definite) for x and y views.

3.2. Projection in the direction of Maximum Correlation

3.2.1 Canonical Correlation Analysis

The goal of CCA [16] is to find the basis $\mathbf{w_x}$ for $\mathbf{x}$ and $\mathbf{w_y}$ for $\mathbf{y}$ such that the linear transformation $\mathbf{x} \rightarrow \langle \mathbf{w_x}, \mathbf{x} \rangle$ and $y \rightarrow \langle \mathbf{w_y}, \mathbf{y} \rangle$ applied to the dataset D results in the maximum correlation. The correlation coefficient is defined as

$$\rho = \max_{w_x, w_y} \frac{\mathbf{w_x}^\top C_{xy} \mathbf{w_y}}{\sqrt{\mathbf{w_x}^\top C_{xx} \mathbf{w_x} \mathbf{w_y}^\top C_{yy} \mathbf{w_y}}}$$
$$\text{subject to: } \begin{matrix} \mathbf{w_x}^\top C_{xx} \mathbf{w_x} = 1 \\ \mathbf{w_y}^\top C_{yy} \mathbf{w_y} = 1 \end{matrix}$$

The problem is equivalent to solving a generalized eigen-value problem $A\mathbf{w_x} = \lambda^2 \mathbf{w_x}$ where $A = C_{xx}^{-1} C_{xy} C_{yy}^{-1} C_{yx}$ and the solution to $\mathbf{w_x}$ are the eigen-vectors of A. Then $\mathbf{w_y}$ can be solved for by using $\mathbf{w_y} = \frac{C_{yy}^{-1} C_{yx} \mathbf{w_x}}{\lambda}$.

3.2.2 Regularized Kernel Canonical Correlation Analysis

We briefly summarize the main ideas behind KCCA here. We refer the reader to [1, 24, 15] for details.

First, the weights $\mathbf{w_x}$ and $\mathbf{w_y}$ can be expressed in the dual form as follows: $\mathbf{w_x} = \sum_{i=1}^{l} \alpha_i \mathbf{x}_i = \alpha^\top D_x^\top$ and $\mathbf{w_y} = \sum_{i=1}^{l} \beta_i \mathbf{y}_i = \beta^\top D_y^\top$. Substituting $\mathbf{w_x}$ and $\mathbf{w_y}$ into definition of the objective function ρ , the numerator becomes $\alpha^\top \left(D_x^\top D_x\right) \left(D_y^\top D_y\right) \beta$. The product of the data matrices can be replaced by kernel matrices K_x and K_y and so the original CCA problem can be solved by maximizing $\rho = \max_{\alpha,\beta} \frac{\alpha^\top K_x K_y \beta}{\sqrt{\alpha^\top K_x^2 \alpha \beta^\top K_y^2 \beta}}$. Using regularization to force a non-trivial solution to this problem yields the modified objective function given by:

$$\rho = \max_{\alpha,\beta} \frac{\alpha^\top K_x K_y \beta}{\sqrt{\left(\alpha^\top K_x^2 \alpha + \kappa \alpha^\top K_x \alpha\right) \left(\beta^\top K_y^2 \beta + \kappa \beta^\top K_y \beta\right)}}$$
$$\text{subject to: } \begin{matrix} \alpha^\top K_x^2 \alpha + \kappa \alpha^\top K_x \alpha = 1 \\ \beta^\top K_y^2 \beta + \kappa \beta^\top K_y \beta = 1 \end{matrix}$$

After setting up the Lagrangian, it can be shown that the solution to the maximization problem for α can be obtained by solving the general eigen-value problem $\left(K_x + \kappa I\right)^{-1} K_y \left(K_y + \kappa I\right)^{-1} K_x \alpha = \lambda^2 \alpha$ and β can be solved for using $\beta = \frac{\left(K_y + \kappa I\right)^{-1} K_x \alpha}{\lambda}$. The constraints are then satisfied by normalization.

The eigen-vectors associated with the top k eigen-values then form the basis α and β for the KCCA model.

3.3. Projection in the direction of Least Error

We now review a discriminative counterpart to Canonical Correlation Analysis, and present a principled solution to minimizing the error of reconstruction of multivariate output.

3.3.1 Multiple Linear Regression

The goal of Multiple Linear Regression is to find basis $\mathbf{w_x}$ and $\mathbf{w_y}$ and direction d to minimize square error [6] of reconstruction of $\mathbf{y}$ for a given $\mathbf{x}$.

$$\begin{aligned} \epsilon^2 &= E\left\{ \left\| \mathbf{y} - \sum_{i=1}^{k} d_i \mathbf{w_{yi}} \mathbf{w_{xi}}^\top \mathbf{x} \right\|^2 \right\} \\ &= E\left\{ \left\| \mathbf{y} - d \mathbf{w_y} \mathbf{w_x}^\top \mathbf{x} \right\|^2 \right\} \\ &= E\left\{ \mathbf{y}^\top \mathbf{y} \right\} - 2d \mathbf{w_y}^\top C_{yx} \mathbf{w_x} + d^2 \mathbf{w_x}^\top C_{xx} \mathbf{w_x} \end{aligned}$$

By taking a derivative with respect to the direction d and setting it to 0, $\frac{\partial \epsilon^2}{\partial d} = 0$ yields a closed-form solution to the direction d, namely $d = \frac{\mathbf{w_x}^\top C_{xy} \mathbf{w_y}}{\mathbf{w_x}^\top C_{xx} \mathbf{w_x}}$. Substituting d back in the main equation and re-arranging the terms, the objective function can be reformulated as an alternative maximization

problem to maximize

$$\rho = \max_{w_x w_y} \frac{\mathbf{w_x}^\top C_{xy} \mathbf{w_y}}{\sqrt{\mathbf{w_x}^\top C_{xx} \mathbf{w_x} \mathbf{w_y}^\top \mathbf{w_y}}}$$
$$\text{subject to: } \begin{matrix} \mathbf{w_x}^\top C_{xx} \mathbf{w_x} = 1 \\ \mathbf{w_y}^\top \mathbf{w_y} = 1 \end{matrix}$$

It can be shown that the problem is equivalent to the generalized eigen-value of the form $A\mathbf{w_x} = \lambda^2 \mathbf{w_x}$ where $A = C_{xx}^{-1} C_{xy} C_{yx}$ and $\mathbf{w_x}$ are the eigen-vectors of A. Then $\mathbf{w_y}$ can be solved for using $\mathbf{w_y} = C_{yx}\mathbf{w_x}$.

3.3.2 Kernel Multiple Linear Regression

The generalization of MLR into KMLR by incorporating kernel functions is fairly straightforward and follows ideas similar to those used in the generalization of CCA into KCCA: We apply a similar linear transformation of the weights, and the the kernel trick, to obtain $d = \frac{\alpha^T K_x K_y \beta}{\alpha^T K_x^2 \alpha}$. The objective function can be re-written in the dual form as $\rho = \frac{\alpha^T K_x K_y \beta}{\sqrt{\alpha^T K_x^2 \alpha \beta^T K_y \beta}}$. Using regularization, we obtain the modified objective function given by:

$$\rho = \max_{\alpha,\beta} \alpha^T K_x K_y \beta$$
$$\text{subject to: } \begin{matrix} \alpha^T K_x^2 \alpha + \kappa \alpha^T K_x \alpha = 1 \\ \beta^T K_y \beta + \kappa \beta^T K_y \beta = 1 \end{matrix}$$

By setting up the Lagrangian

$$L(\lambda, \alpha, \beta) = \alpha^T K_x K_y \beta - \frac{\lambda}{2}\left(\alpha^T K_x^2 \alpha + \kappa \alpha^T K_x \alpha\right) - \frac{\lambda}{2}\left(\beta^T K_y \beta + \kappa \beta^T K_y \beta\right)$$

and setting $\frac{\partial L}{\partial \alpha} = 0$ and $\frac{\partial L}{\partial \beta} = 0$ it is straightforward to show that $\beta = \frac{K_x \alpha}{\lambda}$ and that $\frac{1}{(1+\kappa)}\left(K_x + \kappa I\right)^{-1} K_y K_x \alpha = \lambda^2 \alpha$ which yields the corresponding generalized eigen-value problem. The eigen-vectors associated with the top k eigen-values are then used to compute the basis α, β and the direction b.

3.4. Keyword reconstruction

Given the weights from CCA, MLR, and their kernel extensions KCCA and KMLR, we can use two approaches to produce the annotation to be output for a given input. The first approach provides the direct reconstruction of the keywords by using the weights obtained from CCA and MLR to rank the keywords to be included in the annotation for a given input image. The second, the so-called mate-based reconstruction [24], uses the model to generate, for the given input image, a ranking of the captions in the training set and then assigns the highest scoring caption to the input image.. The two annotation strategies are described below.

3.4.1 Direct reconstruction

In the case of CCA, let $\mathbf{q}$ be the column vector representation of an image for which we want to retrieve the keywords. Let $\mathbf{w_x} = \{\mathbf{w_{x1}}, ..., \mathbf{w_{xk}}\}$ and $\mathbf{w_y} = \{\mathbf{w_{y1}}, ..., \mathbf{w_{yk}}\}$ be the matrices obtained by stacking the eigen-vectors (as columns) corresponding to the the top k eigen-values obtained by solving the generalized eigen-value problem for CCA. We use the value of correlation to assign a score to each keyword: $s = \mathbf{w_y} \mathbf{w_x}^\top \mathbf{q}$, and use the keywords associated with the highest t scores from vector s to produce the annotation.

Similarly, in the case of MLR the solution is given by $\mathbf{w_x}$, $\mathbf{w_y}$ and $\mathbf{d} = [d_1 ... d_k]$. We use the reconstruction score $s = \sum_{i=1}^{k} d_i \mathbf{w_{yi}} \mathbf{w_{xi}}^\top \mathbf{q}$ and select the keywords associated with the highest t scores to include in the annotation.

The assignment of the scores in the case of KCCA and KMLR are similar. Let $\alpha = \{\alpha_1 ... \alpha_k\}$ and $\beta = \{\beta_1 ... \beta_k\}$ be the solutions to KCCA. . Let $K_T(q)$ be the column vector of kernel similarity functions between q and the training images. We first compute the weights for the keywords $\mathbf{w_y} = \alpha^\top D_y^\top$ where D_y is the training matrix of the y view. . The score for KCCA is then computed using $s = \mathbf{w_y} \beta^\top K_T(q)$. The score for KMLR is computed similarly using the solution of KMLR as follows: $s = \sum_{i=1}^{k} d_i \mathbf{w_{iy}} \beta_i^\top K_T(q)$.

3.4.2 Mate-based reconstruction

In mate-based reconstruction given a query image, each caption from the training test is assigned a score, and the highest scoring caption is provided as the output. Let s be the score of a query computed as in the case of direct reconstruction describe above (for CCA, KCCA, MLR or KMLR). Then the captions associated with the images in the training set are ranked using $r = s^\top D_y^\top$ where D_y is the training matrix of the y-view.

4. Multiple Kernel Learning framework for KCCA and KMLR

Given $K_1 ... K_m$ kernels, the goal of Multiple Kernel Learning is to find the optimal combination of the kernels. Consider a convex combination $K = \sum_{i=1}^{k} \eta_i K_i$ such that $\sum_{i=1}^{k} \eta_i = 1$, $\eta_i \geq 0, \forall i$ [3]. The $\ell 1$ norm on the multipliers may drive some ηs to 0 and therefore kernel selection is also performed. We consider the Multiple Kernel Learning set-up similar to that in SimpleMKL for SVM [22] to avoid the need of reformulating the maximization problems for KCCA and KMLR and take advantage of the available solutions.

4.1. Generalized optimization problem

$$
\begin{aligned}
&\text{maximize: } \operatorname{trace}(\alpha^{'} K_x K_y \beta) \\
&\text{subject to: } \alpha^{'} \left(K_x + \kappa I\right) K_x \alpha = 1 \\
&\quad \text{KCCA: } \left\{ \; \beta^{'} \left(K_y + \kappa I\right) K_y \beta = 1 \right. \\
&\quad \text{KMLR: } \left\{ \; \left(1+\kappa\right) \beta^{'} K_y \beta = 1 \right. \\
&\qquad\qquad K_x = \textstyle\sum_{i=1}^{M_x} \eta_i^x K_{xi} \quad \sum_{i=1}^{M_x} \eta_i^x = 1 \\
&\qquad\qquad K_y = \textstyle\sum_{i=1}^{M_y} \eta_i^y K_{yi} \quad \sum_{i=1}^{M_y} \eta_i^y = 1 \\
&\qquad\qquad \eta_i^x, \eta_i^y \geq 0
\end{aligned}
$$

As in SimpleMKL [22], we consider alternative optimization problem:

$$
\begin{aligned}
&\max\nolimits_{\eta^x, \eta^y} J(\eta^x, \eta^y) \\
&\text{subject to: } \eta_i^x, \eta_i^y \geq 0, \sum_{i=1}^{M_x} \eta_i^x = 1, \sum_{i=1}^{M_y} \eta_i^y = 1
\end{aligned}
$$

$$
J(\eta^x, \eta^y) = \begin{cases}
\max_{\alpha,\beta} & \alpha^{'} K_x K_y \beta \\
\text{s. t.} & \alpha^{'} \left(K_x + \kappa I\right) K_x \alpha = 1 \\
\text{KMLR:} & \beta^{'} \left(K_y + \kappa I\right) K_y \beta = 1 \\
\text{KCCA :} & \left(1+\kappa\right) \beta^{'} K_y \beta = 1 \\
& K_x = \sum_{i=1}^{M_x} \eta_i^x K_{xi} \\
& K_y = \sum_{i=1}^{M_y} \eta_i^y K_{yi}
\end{cases}
$$

Let $J^*(\eta)$ be the objective function where where α^* and β^* are the optimal solutions (K eigenvectors for the highest eigenvalues) to $J(\eta^x, \eta^y)$. We then can apply an iterative maximization procedure to solve for ηs: 1) given ηs find solutions α^* and β^* that maximize $J^*(\eta)$ and 2) given α^* and β^* find solutions η^*that maximize $J(\eta)$. For the first step, we use the same eigen-value solution as described in Section 3. For the second step, we used reduced gradient algorithm in order to satisfy the ℓ_1 constraints on ηs.

4.2. Reduced gradient algorithm

Following SimpleMKL, we propose to use reduced gradient algorithm. Let η_m^x be the non-zero entry of η^x

$$
\begin{aligned}
[\nabla_{red} J]_m^x &= \tfrac{\partial J}{\partial \eta_m^x} - \tfrac{\partial J}{\partial \eta_\mu^x}, \forall m \neq \mu \\
[\nabla_{red} J]_\mu^x &= \textstyle\sum_{m \neq \mu} \left[\tfrac{\partial J}{\partial \eta_m^x} - \tfrac{\partial J}{\partial \eta_\mu^x} \right]
\end{aligned}
$$

where μ as the index of the largest component of η^x (similar construction is done for η^y).

Then the direction for the x component of the gradient direction is

$$
D_x = \begin{cases}
0 & \eta_m = 0, \tfrac{\partial J}{\partial \eta_m^x} - \tfrac{\partial J}{\partial \eta_\mu^x} < 0 \\
\tfrac{\partial J}{\partial \eta_m^x} - \tfrac{\partial J}{\partial \eta_\mu^x} & \eta_m > 0, m \neq \mu \\
\sum_{m \neq \mu} \left[\tfrac{\partial J}{\partial \eta_\mu^x} - \tfrac{\partial J}{\partial \eta_m^x} \right] & m = \mu
\end{cases}
$$

and similarly for the y component for the direction.

We use a similar algorithm as in SimpleMKL modified to take into account several ℓ_1 constraints.

Algorithm 1 Generalized MKL for KCCA/KMLR

Initialize $\eta_i^x = \frac{1}{m_x}$ and $\eta_i^y = \frac{1}{m_y}$
Set $\eta = [\eta^x, \eta^y]$
while Convergence criteria not met **do**
  Compute $\alpha^*$and β^* for $J(\eta^x, \eta^y)$ using $K_x(\eta^y)$ and $K_y(\eta^y)$
  set $\mu_x = \arg\max_m \eta_{x_m}$, $\mu_y = \arg\max_m \eta_{y_m}$
  Compute $\frac{\partial J}{\partial \eta}$ and the ascent direction $D = [D_x, D_y]$
  $J^\dagger = \infty$, $\eta^\dagger = \eta$, $D^\dagger = D$
  while $J^\dagger > J(\eta)$ **do**
    $\eta = \eta^\dagger$, $D = D^\dagger$
    Set $\nu_x = \arg\min_{\{m | D_{xm} < 0\}} -\frac{\eta_{xm}}{D_{xm}}$
    Set $\nu_y = \arg\min_{\{m | D_{ym} < 0\}} -\frac{\eta_{ym}}{D_{ym}}$
    Set $\nu = \arg\min_{\nu_x, \nu_y} -\frac{\eta}{D}$, $\gamma_{\max} = -\frac{d_\nu}{D_\nu}$,
    $\eta^\dagger = \eta + \gamma D$
    $D_{\mu_x}^\dagger = D_{\mu_x} - D_{\nu_x}$, $D_{\nu_x}^\dagger = 0$
    $D_{\mu_y}^\dagger = D_{\mu_y} - D_{\nu_y}$, $D_{\nu_y}^\dagger = 0$
    compute $J^\dagger$ using $K_x(\eta^{y\dagger})$ and $K_y(\eta^{y\dagger})$
  end while
  Line search along D for $\gamma \in [0, \gamma_{max}]$
  $\eta^\dagger = \eta + \gamma D$
end while

The following conditions are used as stopping criteria: the number of iterations, change in the value of J, or change in value of η.

4.2.1 Gradient

The computation of the derivatives is fairly straight-forward and the derivatives for KCCA and KMLR are identical:

$$
\frac{\partial J(d)}{\partial \eta_i^x} = \sum_{i=1}^{K} \alpha_i^{*'} K_{xi} K_y \beta_i^*
$$

$$
\frac{\partial J(d)}{\partial \eta_i^y} = \sum_{i=1}^{K} \alpha_i^{*'} K_x K_{yi} \beta_i^*
$$

where J and α^*, β^* is the objective function for KCCA or KMLR with its respective solutions.

5. Experiments

We now describe datasets, experimental set-up and results.

5.1. Data

We use natural scene data [21] derived from LabelMe. It consists of approximately 2700 images in 8 natural scene categories. In addition to the categories, each image is annotated with a number of objects with the total number of possible object classes is 305. On average, each caption consists of 15 keywords. For each image we use the set of object keywords (that the images are annotated with) as the caption. Following similar evaluation procedures as in previous work that used this dataset [21, 7], we randomly split the data into the training and testing sets using 100 images in each category for training, and the rest for testing.

Given the training and the test sets, the images are processed as follows. We rescale each image to size of 256 pixels and extract 128-dimensional SIFT features [19] with circular support of radius $r = 4, 8, 12$ and 16 pixels from a grid evenly spaced 10 pixels apart [11] from the training data and use k-means to construct a codebook of size 500. We then use the histogram of the codewords to represent each image (bag-of-visual-words representation). This gives us 4 kernels for each of the keypoint radius.

5.2. Performance Measure

To access the performance of the different predictive models, following Hardoon et al [14], we use precision, recall, and f-score. Let C be a set of predicted keywords (candidate), and let R be a set of actual keywords (reference). Precision is defined as the probability of correctly predicted words in the candidate caption: $Precision(R|C) = \frac{|R \cap C|}{|C|}$. Recall is defined as the probability of correctly predicted words given the reference $Recall(C|R) = \frac{|R \cap C|}{|R|}$. We define f-score as the harmonic mean of precision and recall: $F - score = 2\frac{Precision \cdot Recall}{Precision + Recall}$ Since it is possible to achieve perfect recall (by assigning all possible keywords to the image), we use all three measures to assess the performance. For each test image we compute the average of precision, recall, and the f-measure of reconstructed keywords, and report the average over all the images.

5.3. Results

Annotation based on the choice of kernel We first use CCA, KCCA, MLR and KMLR on data represented using each of the four kernels. We use $K = 10$ eigen-vectors for all 4 algorithms, and regularization parameter $\kappa = 100$ for KMLR and KCCA. For direct reconstruction we use top ranked keywords to reconstruct the caption. The results are shown in Figure 1. Since KMLR is trained to minimize the error, it has the highest performance according to all measures in direct reconstruction, and mate-based reconstruction.

We also observe slight variation in the performance based on the choice of kernels: in case of MLR and KMLR

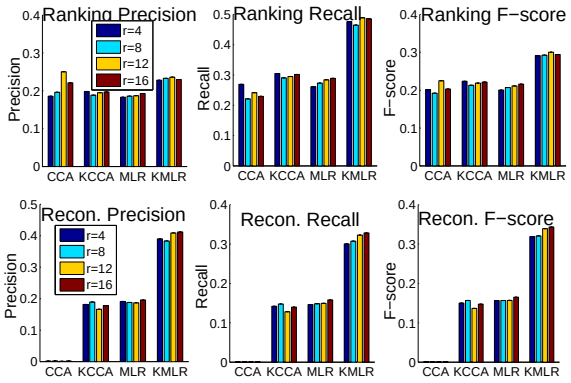

Figure 1.

the kernel derived from support of radius $r = 16$ has the highest precision/recall/f-score measure for KMLR. For KCCA, using mate-based reconstruction, the best performance is achieved using kernels $r = 4$ and $r = 16$, and for reconstruction the best performance is achieved with using kernel $r = 8$.

Multiple Kernel Learning We then use MKL learning framework for KMLR and KCCA and solve the optimization problem. For both KCCA and KMLR, ds for kernels derived from keypoints of $r = 4, 8, 12$ were learned to be zero except for kernel for keypoints of radius $r = 16$, which was set to 1, which means that kernel $r = 16$ was selected as the optimal. For KMLR, this is not surprising since the best performance was achieved for this choice of kernel. It first seems surprising for KCCA, however as we have noticed earlier, it is one of the best kernels for mate-based reconstruction and KCCA is not optimized for direct reconstruction.

6. Conclusion and discussion

Much remains to be done in image annotation, and multiple kernel learning for KCCA and KMLR. We discuss some ongoing work, future work, and open problems in image annotation.

We are currently conducting more experiments with other datasets and benchmarks in image annotation and several datasets in cross-language retrieval.

We presented mutliple kernel framework for KCCA and KMLR to account for kernels in both views of the data, however in our evaluation we used linear kernels. We are currently also extending the work to allow for graph kernels for images to allow for neighborhood dependencies between the image features, and n-gram kernels for the labels, to account for dependency between the words.

We only used several kernels (based on the idea of concentric circles) using grid-sampled keypoints over different

size of keypoint support. We are also working on evaluation of the models using a variety of kernels, including color histograms, textures, graph kernels to find out whether incorporating all this information in a single model is beneficial.

We presented a general framework to find the best kernel combination in the $x-$ space (image input space) and $y-$space (keyword output space), however our experiments are limited to consider the combination in the input space only. It will be of interest to consider various output space kernels as well.

In order to perform reconstruction we used keyword ranking obtained by (K)CCA or (K)MLR, and used the top k scoring keywords. One way of allowing different number of keywords to be assigned to the test image is to use a threshold on the keyword scores. However, the questions of finding a flexible model to assign an optimal number of keywords and determining the optimal threshold remain open.

References

[1] F. Bach and M. Jordan. Kernel independent component analysis. *Journal of Machine Leaning Research*, 3:1–48, 2002. 3

[2] F. Bach and M. Jordan. A probabilistic interpretation of canonical correlation analysis. Technical report, Department of Statistics, University of California,, 2005. 1

[3] F. R. Bach, G. R. G. Lanckriet, and M. I. Jordan. Multiple kernel learning, conic duality, and the smo algorithm. In C. E. Brodley and C. E. Brodley, editors, *ICML*, volume 69 of *ACM International Conference Proceeding Series*. ACM, 2004. 2, 4

[4] K. Barnard, P. Duygulu, N. de Freitas, D. Forsyth, D. Blei, and M. Jordan. Matching words and pictures. *Journal of Machine Learning Research*, 3:11071135, 2003. 1, 2

[5] K. Barnard and D. A. Forsyth. Learning the semantics of words and pictures. In *ICCV*, 2001. 2

[6] M. Borga, T. Landelius, and H. Knutsson. A unified approach to PCA, PLS, MLR and CCA. Technical report, Computer Vision Laboratory, Linkõping University, 1997. 2, 3

[7] A. Bosch, A. Zisserman, and X. Muñoz. Scene classification via pLSA. In *ECCV*, 2006. 2, 6

[8] G. Carneiro, A. B. Chan, P. Moreno, and N. Vasconcelos. Supervised learning of semantic classes for image annotation and retrieval. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, pages 394–410., 2007. 2

[9] N. Christianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*. Cambridge University Press, 2000. 3

[10] P. Duygulu, K. Barnard, J. F. G. de Freitas, and D. A. Forsyth. Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In *ECCV*, 2002. 1, 2

[11] L. Fei-Fei and P. Perona. A bayesian hierarchical model for learning natural scene categories. In *CVPR*, 2005. 2, 6

[12] S. Feng, R. Manmatha, and V. Lavrenko. Multiple bernoulli relevance models for image and video annotation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2004. 1, 2

[13] Z. Harchaoui and F. Bach. Image classification with segmentation graph kernels. In *Proceedings of Computer Vision and Pattern Recognition*, 2007. 2

[14] D. Hardoon, C. Saunders, S. Szedmak, and J. Shawe-Taylor. A correlation approach for automatic image annotation. In X. Li, O. Zaiane, and Z. Li, editors, *Second International Conference on Advanced Data Mining and Applications*, volume 4093, pages 681–692. Springer, 2006. 1, 6

[15] D. Hardoon, S. Szedmak, and J. Shawe-Taylor. Canonical correlation analysis: an overview with application to learning methods. *Neural Computation*, 16:2639–64, 2004. 1, 3

[16] H. Hottelling. Relations between two sets of variates. *Biometrika*, 8:321–377, 1936. 3

[17] J. Jeon, V. Lavrenko, and R. Manmatha. Automatic image annotation and retrieval using cross-media relevance models". In *Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval*, 2003. 1, 2

[18] N. Loeff and A. Farhadi. Scene discovery by matrix factorization. In *European conference on Computer Vision (ECCV)*, 2008. 2

[19] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60:91–110, 2004. 6

[20] A. Makadia, V. Pavlovic, and S. Kumar. A new baseline for image annotation. In *In European Conference on Computer Vision (ECCV)*, 2008. 2

[21] A. Oliva and A. Torralba. Modeling the shape of the scene: a holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42:145–175, 2001. 1, 2, 6

[22] A. Rakotomamonjy, F. R. Bach, S. Canu, and Y. Grandvalet. SimpleMKL. *Journal of Machine Learning Research*, 9:2491–2521, November 2008. 2, 4, 5

[23] J. J. V. Lavrenko, R. Manmatha. A model for learning the semantics of pictures. In *NIPS*, 2003. 1, 2

[24] A. Vinokourov, J. Shawe-Taylor, and N. Cristianini. Inferring a semantic representation of text via cross-language correlation analysis. In *NIPS*, 2002. 1, 3, 4