Laplacian Adaptive Context-based SVM for Video Concept Detection

Wei Jiang Alexander C. Loui
Corporate Research and Engineering, Eastman Kodak Company, Rochester, NY
{wei.jiang, alexander.loui}@kodak.com

ABSTRACT

Practical semantic concept detection problems usually have the following challenging conditions: the amount of unlabeled test data keeps growing and newly acquired data are incrementally added to the collection; the domain difference between newly acquired data and the original labeled training data is not negligible; and only very limited, or even no, partial annotations are available over newly acquired data. To accommodate these issues, we propose a Laplacian Adaptive Context-based SVM (LAC-SVM) algorithm that jointly uses four techniques to enhance classification: cross-domain learning that adapts previous classifiers learned from a source domain to classify new data in the target domain; semi-supervised learning that leverages information from unlabeled data to help training; multi-concept learning that uses concept relations to enhance individual concept detection; and active learning that improves the efficiency of manual annotation by actively querying users. Specifically, LAC-SVM adaptively applies concept classifiers and concept affinity relations computed from a source domain to classify data in the target domain, and at the same time, incrementally updates the classifiers and concept relations according to the target data. LAC-SVM can be conducted without newly labeled target data or with partially labeled target data, and in the second scenario the two-dimension active learning mechanism of selecting data-concept pairs is adopted. Experiments over three large-scale video sets show that LAC-SVM can achieve better detection accuracy with less computation compared with several state-of-the-art methods.

Categories and Subject Descriptors

H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing; H.3.m [**Information Storage and Retrieval**]: Miscellaneous

General Terms

Algorithms, Experimentation

Keywords

Semantic concept detection, cross-domain, active annotation

WSM'11, November 30, 2011, Scottsdale, Arizona, USA.

1. INTRODUCTION

Rapidly increased amounts of social media data require automatic detection of a broad range of semantic concepts chosen to represent media content, such as objects (*e.g.*, car), scenes (*e.g.*, sunset), events (*e.g.*, birthday), *etc.* In practice, semantic concept detection problems often have the following challenging conditions. First, the amount of unlabeled test data usually grows and newly acquired data are incrementally added to the collection. Second, the domain difference is not negligible, *i.e.*, newly acquired data have different data distribution than the original labeled training data. They come from different users, record different real-world events, have changing characteristics. Finally, due to expensive manual labeling, only very limited (sometimes even no) annotations are available over newly acquired data. We propose a novel solution to accommodate these challenging conditions. The underlying rationale is three-fold.

First, the problem raised by the fixed amount of labeled training data versus the incrementally growing amount of test data that have changing distribution is known in the literature as *cross-domain learning*. The fixed amount of labeled training data are treated as from a *source domain* (also called an auxiliary domain in some previous work), and the incrementally acquired new data are treated as from a *target domain*. By considering the domain difference, several approaches have been developed to adapt data or models from the source domain to classify data in the target domain, such as [5, 6, 8, 9, 18]. However, most previous cross-domain learning methods rely on newly labeled training data in the target domain, while in practice we may not have such data. Therefore, we need a cross-domain learning method that can adapt models from the source domain when there are few or no labeled data in the target domain.

Second, with a very limited number of (or even no) labeled training data in the target domain, it is usually necessary to solicit help from other types of information, such as knowledge about the unlabeled target data and concept relations. *Semi-supervised learning* methods [1, 4] incorporate information of the underlying data structure computed from the unlabeled data, so that a better classifier can be designed for classifying the test data. In addition, concept detection tasks are usually multi-label classification problems, *i.e.*, multiple concepts can be present simultaneously in a single datum. *Multi-concept learning* methods [9, 13] exploit the semantic context, *e.g.*, pairwise concept affinity relations, to enhance classification of individual concepts. We aim to incorporate the semi-supervised and multi-concept learning techniques to improve detection.

Third, we have a partial labeling situation in the target domain. In reality, users generally only annotate a few concepts to a datum, which are present in the datum and are important to describe the datum. Assume we want to detect K concepts. Due to the burden of manual labeling, unless they are required to do so, users normally do not provide full annotations to the whole set of K concepts. Therefore, each datum in our target training set is annotated to a part of the concepts with mostly only positive labels. To cope with this partial labeling issue and to improve the efficiency of manual labeling, we need to actively drive users' annotation. *Active learning* methods have been developed to select the most informative data [16], concepts [7], or data-concept pairs [14] to query users. In our multi-label classification problem, we adopt the data-concept pair selection strategy that can best approximate users' partial labeling situation. We actively select the optimal data-concept pairs to query the user, where each data-concept pair contains a datum and a concept that is most significant to the datum, and the user is asked to provide a binary label to the concept for this datum. Therefore, we aim to develop a cross-domain learning method that can use the partially annotated data-concept pairs to learn concept detectors.

In summary, we propose an algorithm, called *Laplacian Adaptive Context-based SVM* (*LAC-SVM*), to accommodate our needs. LAC-SVM jointly uses cross-domain learning, semi-supervised learning, multi-concept learning, and active learning to enhance classification in practical semantic concept detection problems with the challenging conditions described earlier. LAC-SVM adapts the previous SVM classifiers and concept relations computed from the source domain, while preserving the data affinity relations and concept affinity relations in the target domain. It allows incremental adaptation and can classify new unseen test samples. Also, LAC-SVM can be conducted with or without the presence of new annotated data from the target domain, and can function with partially labeled target training data.

We extensively evaluate LAC-SVM over three video sets: the TRECVID 2007 development set [15], Kodak's consumer benchmark set [11], and the Columbia Consumer Video (*CCV*) set [10]. We evaluate situations of adapting classifiers learned from the TRECVID data to Kodak's consumer data where there is significant domain difference, as well as adapting classifiers within the consumer domain from the CCV data to Kodak's data. We compare LAC-SVM with several state-of-the-art alternatives, such as the cross-domain *Adaptive SVM* (*A-SVM*) [18] and the semi-supervised *Laplacian SVM* (*LapSVM*) [1]. Experiments show that LAC-SVM can achieve better detection accuracy with less computation cost.

2. PROBLEM DEFINITION AND BRIEF REVIEW OF RELATED WORK

A general cross-domain semantic concept detection problem can be described as follows. The goal is to classify K concepts $C_1, \ldots, C_K$ in a target set $\mathcal{X}^{new}$ that is partitioned into a labeled subset $\mathcal{X}^L$ (with size $n^L \geq 0$, where $n^L = 0$ means there are no labeled target data) and an unlabeled subset $\mathcal{X}^U$ (with size $n^U > 0$), *i.e.*, $\mathcal{X}^{new} = \mathcal{X}^L \cup \mathcal{X}^U$. Each data point $\mathbf{x}_i \in \mathcal{X}^L$ is associated with a set of class labels y_{ik}, $k = 1, \ldots, K$, where $y_{ik} = 1$, -1 or 0. $y_{ik} = 1$ or -1 indicates the presence or absence of concept C_k in $\mathbf{x}_i$, and $y_{ik} = 0$ indicates that $\mathbf{x}_i$ is not labeled with respect to C_k. That is, each $\mathbf{x}_i \in \mathcal{X}^L$ is only partially labeled to a part of concepts. In addition to $\mathcal{X}^{new}$, we have a source set $\mathcal{X}^{old}$, whose data characteristics or distribution is different from that of $\mathcal{X}^{new}$, *i.e.*, $\mathcal{X}^{new}$ and $\mathcal{X}^{old}$ are from different domains. A set of classifiers (represented by a set of parameters Θ^{old}) have been learned using $\mathcal{X}^{old}$ to detect $C_1, \ldots, C_K$. Also, a concept affinity matrix $\mathbf{W}^{old}$ has been computed to capture concept affinity relations based on $\mathcal{X}^{old}$. Our task is to adaptively apply previous concept detectors Θ^{old} and concept affinity relations $\mathbf{W}^{old}$ to classify concepts $C_1, \ldots, C_K$ jointly in the target domain, as well as update both concept detectors (into Θ^{new}) and concept affinity relations (into $\mathbf{W}^{new}$) according to $\mathcal{X}^{new}$. The learned Θ^{new} and $\mathbf{W}^{new}$ will better classify $\mathcal{X}^{new}$ and future unseen data from the target domain, compared to the original Θ^{old} and $\mathbf{W}^{old}$.

Combined SVM – Ignoring the domain difference, classifiers such as SVMs can be learned over all available training samples $\tilde{\mathcal{X}}$ from both the source and target domains, $\tilde{\mathcal{X}} = \mathcal{X}^{old} \cup \mathcal{X}^L$. This is the Combined SVM method. However, the influence of new data in $\mathcal{X}^L$ is usually overshadowed by the large amount of data in $\mathcal{X}^{old}$.

Semi-supervised learning – One most popular branch of semi-supervised learning is to use graph regularization [4]. A weighted undirected graph $\mathcal{G}^d = (\mathcal{V}^d, E^d, \mathbf{W}^d)$ can be generated for set $\mathcal{X}^{new}$, where $\mathcal{V}^d$ is the vertices set and each node corresponds to a datum, E^d is the edges set, and $\mathbf{W}^d$ is weights set measuring the pairwise similarities among data points. To detect a concept C_k, under the assumption of label smoothness over $\mathcal{G}^d$, a discriminant function f is estimated to satisfy two conditions: the loss condition – it should be close to given labels y_{ik} for labeled nodes $\mathbf{x}_i \in \mathcal{X}^L$ with $y_{ik} \neq 0$; and the regularization condition – it should be smooth on graph $\mathcal{G}^d$. Among graph-based methods, the LapSVM algorithm [1] is considered one of the states of the art in terms of both classification accuracy and out-of-sample extension ability. Semi-supervised learning methods like LapSVM can be applied directly to cross-domain learning problems by using $\tilde{\mathcal{X}} = \mathcal{X}^{old} \cup \mathcal{X}^L$ as the combined training set. However, such approaches ignore the domain difference, and the classifiers are usually biased by $\mathcal{X}^{old}$.

Cross-domain learning – Cross-domain learning leverages information from a source domain to enhance classification in the target domain [5, 6, 8, 9, 18]. The feature replication method [5] combines samples from $\mathcal{X}^{old}$ and $\mathcal{X}^{new}$, and learns generalities between the two domains by replicating the original features. The cross-domain SVM approach [8] incorporates support vectors from the old domain and weighted combines them with new labeled data $\mathcal{X}^L$ to learn target models. The Domain Transfer SVM method [6] learns a kernel function and an SVM classifier in the target domain, by minimizing the distance of data distribution between the two domains as well as the classification error over combined set $\tilde{\mathcal{X}} = \mathcal{X}^{old} \cup \mathcal{X}^L$. The A-SVM method [18] adapts the old discriminant function trained from $\mathcal{X}^{old}$ into a new discriminant function, by minimizing the deviation between the new decision boundary and the old one, as well as minimizing the classification error over newly labeled target data $\mathcal{X}^L$.

Most previous cross-domain approaches rely on a reasonable sized set of newly labeled training data in the target domain. When applied to our problems where there are no new target training data or only partially labeled target training data, the performance is usually still unsatisfactory. Also, most methods have high computation costs, especially

for large-scale problems, due to the re-training of models using data from both $\mathcal{X}^{old}$ and $\mathcal{X}^{new}$.

Multi-concept learning – Recently, the *Domain Adaptive Semantic Diffusion* (*DASD*) algorithm has been proposed [9], which considers the domain-shift problem when using concept relations. An undirected graph $\mathcal{G}^c=(\mathcal{V}^c, E^c, \mathbf{W}^{c,old})$ is defined to capture semantic concept affinity relations over the source domain. $\mathcal{V}^c$ is the vertices set and each node corresponds to a concept, E^c is the edges set, and $\mathbf{W}^{c,old}$ is the concept affinity matrix. DASD makes an assumption of local smoothness over graph $\mathcal{G}^c$, *i.e.*, if two concepts have high similarity defined in $\mathcal{G}^c$, they frequently co-occur in data samples. Based on this assumption, the discriminant functions over $\mathcal{X}^{new}$ for all concepts can be refined. The major drawback of DASD is the lack of the out-of-sample extension ability, *i.e.*, the target discriminant functions are optimized over the available target data $\mathcal{X}^{new}$, and the learned results can not be easily applied to future unseen test data.

3. THE LAC-SVM ALGORITHM

3.1 No labeled target data

We first develop our algorithm when there is no new labeled target data available. The next subsection will describe the scenario with partially labeled target data.

3.1.1 Discriminative cost function

Similar to previous cross-domain methods such as A-SVM [18], we want the learned new models Θ^{new} to be similar to the previous detectors Θ^{old}, so that we can maintain the discriminant ability carried by Θ^{old}. Therefore, the first part of the joint cost function that our LAC-SVM minimizes is:

$$\min_{\Theta^{new}} \text{Difference}(\Theta^{new}, \Theta^{old}) \ . \quad (1)$$

We use kernel-based SVMs as concept detectors due to their effectiveness in detecting concepts for various data sets [11, 15]. According to the Representer Theorem [17], the discriminant function $f_k(\mathbf{x})$, which is learned from the source domain of a datum $\mathbf{x}$ for a concept C_k, is given as:

$$f_k(\mathbf{x}) = \sum\nolimits_{\mathbf{x}_i \in \mathcal{X}^{old}} \mu_{ki} K(\mathbf{x}_i, \mathbf{x}) = \mathbf{K}(\mathbf{x}; \mathcal{X}^{old})^T \mathbf{u}_k,$$

where $K(\mathbf{x}_1, \mathbf{x}_2)$ is the kernel function of $\mathbf{x}_1$ and $\mathbf{x}_2$, $\mathbf{K}(\mathbf{x}; \mathcal{X}^{old})$ is a vector composed by kernel functions of $\mathbf{x}$ against all data in $\mathcal{X}^{old}$, and $\mathbf{u}_k = [\mu_{k1}, \ldots, \mu_{kn^{old}}]^T$ (n^{old} is the size of $\mathcal{X}^{old}$). Define $\mathbf{U}^{old} = [\mathbf{u}_1, \ldots, \mathbf{u}_K]$. The $n^{old} \times K$ matrix $\mathbf{U}^{old}$ contains all parameters learned from the source domain to generate discriminant functions for classifying K concepts. Our goal is to learn a new $n^{old} \times K$ matrix $\mathbf{U}^{new} = [\tilde{\mathbf{u}}_1, \ldots, \tilde{\mathbf{u}}_K]$ that is similar to $\mathbf{U}^{old}$. That is, we define the following cost function to compute Eqn. (1):

$$\min_{\mathbf{U}^{new}} ||\mathbf{U}^{new} - \mathbf{U}^{old}||_2^2, \quad (2)$$

where $||\cdot||_2$ is the Hilbert-Schmidt norm. The new discriminant function of classifying $\mathbf{x}$ for a concept C_k in the target domain is given by:

$$\tilde{f}_k(\mathbf{x}) = K(\mathbf{x}; \mathcal{X}^{old})^T \tilde{\mathbf{u}}_k. \quad (3)$$

3.1.2 Graph regularization on data points

In order to use the large amount of unlabeled data in the target domain to help classification, we incorporate the assumption of graph smoothness over data points from the semi-supervised learning. Let $\mathcal{G}^d = (\mathcal{V}^d, E^d, \mathbf{W}^d)$ denote the undirected graph over $\mathcal{X}^{new}$ in the new target domain, where each node in the vertices set $\mathcal{V}^d$ corresponds to a data sample. Each entry W_{ij}^d measures the similarity of $\mathbf{x}_i$ and $\mathbf{x}_j$. We have the following cost function:

$$\min_{\tilde{\mathbf{F}}} \frac{1}{2} \sum\nolimits_{\mathbf{x}_i, \mathbf{x}_j \in \mathcal{X}^{new}} W_{ij}^d ||(\tilde{\mathbf{f}}_i^d / \sqrt{d_i^d}) - (\tilde{\mathbf{f}}_j^d / \sqrt{d_j^d})||_2^2, \quad (4)$$

where $\tilde{\mathbf{f}}_i^d = [\tilde{f}_1(\mathbf{x}_i), \ldots, \tilde{f}_K(\mathbf{x}_i)]^T$ comprises discriminant functions of $\mathbf{x}_i$ over all K concepts, and d_i^d is the degree of graph $\mathcal{G}^d$ over node $\mathbf{x}_i$. $\tilde{\mathbf{F}} = [\tilde{\mathbf{f}}_1^d, \ldots, \tilde{\mathbf{f}}_{n^U+n^L}^d]^T$ contains the discriminant functions of the entire target set $\mathcal{X}^{new}$, generated by the parameter matrix $\mathbf{U}^{new}$ over all the K concepts, and

$$\tilde{\mathbf{F}} = \mathbf{K}(\mathcal{X}^{new}; \mathcal{X}^{old}) \mathbf{U}^{new}. \quad (5)$$

$\mathbf{K}(\mathcal{X}^{new}; \mathcal{X}^{old})$ is the kernel matrix of data set $\mathcal{X}^{new}$ against data set $\mathcal{X}^{old}$, and $\mathbf{K}(\mathcal{X}^{new}; \mathcal{X}^{old}) = \mathbf{K}(\mathcal{X}^{old}; \mathcal{X}^{new})^T$. By substituting Eqn. (3) into Eqn. (4), we can get:

$$\min_{\mathbf{U}^{new}} \text{tr}\{\tilde{\mathbf{F}}^T \mathbf{L}^d \tilde{\mathbf{F}}\}/2. \quad (6)$$

$\mathbf{L}^d = \mathbf{I} - \mathbf{D}^{d^{-1/2}} \mathbf{W}^d \mathbf{D}^{d^{-1/2}}$ is the normalized graph Laplacian. $\mathbf{D}^d$ is a diagonal matrix whose entries are row sums of $\mathbf{W}^d$.

3.1.3 Graph regularization on semantic concepts

In order to use the semantic context information, we adopt the assumption of graph smoothness over semantic concepts from the DASD multi-concept learning method, *i.e.*, two concepts having high similarity defined in the concept affinity graph have similar concept detection scores over data samples. Let $\mathcal{G}^{c,new} = (\mathcal{V}^{c,new}, E^{c,new}, \mathbf{W}^{c,new})$ be the undirected graph over concepts in the target domain. Each node in the vertices set $\mathcal{V}^{c,new}$ corresponds to a concept. Each entry $W_{kl}^{c,new}$ gives the edge weight between concepts C_k and C_l. We have the following cost function:

$$\min_{\mathbf{U}^{new}, \mathbf{W}^{c,new}} \text{tr}\{\tilde{\mathbf{F}} \mathbf{L}^{c,new} \tilde{\mathbf{F}}^T\}/2, \quad (7)$$

where $\mathbf{L}^{c,new} = \mathbf{I} - (\mathbf{D}^{c,new})^{-1/2} \mathbf{W}^{c,new} (\mathbf{D}^{c,new})^{-1/2}$ is the normalized graph Laplacian. $\mathbf{D}^{c,new}$ is diagonal whose entries are row sums of $\mathbf{W}^{c,new}$. By introducing a term $\tilde{\mathbf{W}}^{c,new}$:

$$\tilde{\mathbf{W}}^{c,new} = (\mathbf{D}^{c,new})^{-1/2} \mathbf{W}^{c,new} (\mathbf{D}^{c,new})^{-1/2}, \quad (8)$$

Equation (7) can be re-written to:

$$\min_{\mathbf{U}^{new}, \tilde{\mathbf{W}}^{c,new}} \text{tr}\{\tilde{\mathbf{F}} (\mathbf{I} - \tilde{\mathbf{W}}^{c,new}) \tilde{\mathbf{F}}^T\}/2. \quad (9)$$

3.1.4 LAC-SVM

We can combine all three cost functions Eqn. (2), Eqn. (6), and Eqn. (9) into a joint cost function to minimize by our LAC-SVM algorithm:

$$\min_{\mathbf{U}^{new}, \tilde{\mathbf{W}}^{c,new}} Q^{LAC-SVM} = \min_{\mathbf{U}^{new}, \tilde{\mathbf{W}}^{c,new}} \Big[||\mathbf{U}^{new} - \mathbf{U}^{old}||_2^2 + \frac{\lambda^d}{2} \text{tr}\{\tilde{\mathbf{F}}^T \mathbf{L}^d \tilde{\mathbf{F}}\} + \frac{\lambda^c}{2} \text{tr}\{\tilde{\mathbf{F}} (\mathbf{I} - \tilde{\mathbf{W}}^{c,new}) \tilde{\mathbf{F}}^T\} \Big]. \quad (10)$$

By optimizing $Q^{LAC-SVM}$ we can obtain a new parameter matrix $\mathbf{U}^{new}$ that constructs classifiers to classify all K concepts, and the updated normalized concept affinity $\tilde{\mathbf{W}}^{c,new}$. An iterative algorithm can be used to monotonically reduce the cost by coordinate descent towards a local minimum.

Step 1: Optimization with fixed $\tilde{\mathbf{W}}^{c,new}$

When $\tilde{\mathbf{W}}^{c,new}$ is fixed (*i.e.*, $\mathbf{L}^{c,new}$ is fixed), we can learn $\mathbf{U}^{new}$ by gradient descent as:

$$\mathbf{U}^{new}(t) = \mathbf{U}^{new}(t-1) - \alpha_{\mathbf{U}} \frac{\partial Q^{LAC-SVM}}{\partial \mathbf{U}^{new}(t-1)}, \quad (11)$$

$$\partial Q^{LAC-SVM} / \partial \mathbf{U}^{new} = 2\mathbf{U}^{new} - 2\mathbf{U}^{old} + \lambda^d \mathbf{K}(\mathcal{X}^{old}; \mathcal{X}^{new}) \mathbf{L}^d \mathbf{K}(\mathcal{X}^{new}; \mathcal{X}^{old}) \mathbf{U}^{new} + \lambda^c \mathbf{K}(\mathcal{X}^{old}; \mathcal{X}^{new}) \mathbf{K}(\mathcal{X}^{new}; \mathcal{X}^{old}) \mathbf{U}^{new} \mathbf{L}^{c,new}. \quad (12)$$

Step 2: Optimization with fixed $\mathbf{U}^{new}$

When $\mathbf{U}^{new}$ is fixed, Eqn. (10) reduces to:

$$\min_{\tilde{\mathbf{W}}^{c,new}} \tilde{Q} = \min_{\tilde{\mathbf{W}}^{c,new}} \text{tr}\Big\{\tilde{\mathbf{F}} (\mathbf{I} - \tilde{\mathbf{W}}^{c,new}) \tilde{\mathbf{F}}^T\Big\}, \text{ s.t.} \tilde{\mathbf{W}}^{c,new} \geq 0. \quad (13)$$

The constraint $\tilde{\mathbf{W}}^{c,new} \geq 0$ gives a positive matrix $\tilde{\mathbf{W}}^{c,new}$, *i.e.*, we consider positive relations among different concepts here. By introducing a Lagrangian multiplier ζ and taking the derivative of Eqn. (13) with respect to $\tilde{\mathbf{W}}^{c,new}$, we have:

$$\partial \tilde{Q}/\partial \tilde{\mathbf{W}}^{c,new} = 0 \Rightarrow \zeta = -\tilde{\mathbf{F}}^T\tilde{\mathbf{F}}. \quad (14)$$

According to the Karush-Kuhn-Tucker condition [2], for each entry $\tilde{W}^{c,new}_{kl}$ we have:

$$\left[\tilde{\mathbf{F}}^T\tilde{\mathbf{F}}\right]_{kl} \tilde{W}^{c,new}_{kl} = 0. \quad (15)$$

Define $\mathbf{A}$, $\mathbf{A}^+$ and $\mathbf{A}^-$ as follows:

$$\mathbf{A} = \tilde{\mathbf{F}}^T\tilde{\mathbf{F}},\ A^+_{kl} = |A_{kl}| + (A_{kl}/2),\quad A^-_{kl} = |A_{kl}| - (A_{kl}/2). \quad (16)$$

We have the following updating formula to get $\tilde{W}^{c,new}_{kl}$:

$$\tilde{W}^{c,new}_{kl} \leftarrow \tilde{W}^{c,new}_{kl}\sqrt{(A^+_{kl})/(A^-_{kl})}\ . \quad (17)$$

It can be proved that the the above updating formula converges to the global optimal.

The major part of computation lies in matrix multiplications in the second term of Eqn. (12), with $O(\max\{n^{old}, n^{new}\} \times n^{new} \times K)$ complexity if computed straightforwardly (K is usually much smaller than n^{old} and n^{new}). Due to the sparsity of $\mathbf{L}^d$, LAC-SVM can be much faster than other alternatives that retrain SVMs (with the need to solve QP problems) using data from both source and target domains.

It is worth mentioning that if $\lambda^c = 0$, *i.e.*, if we do not consider the concept relations, $\mathbf{U}^{new}$ has closed form solution:

$$\mathbf{U}^{new} = \left[\mathbf{I} + \frac{\lambda^d}{2}\mathbf{K}(\mathcal{X}^{old};\mathcal{X}^{new})\mathbf{L}^d\mathbf{K}(\mathcal{X}^{new};\mathcal{X}^{old})\right]^{-1}\mathbf{U}^{old}. \quad (18)$$

In such a case, the time complexity of obtaining the set of K new target classifiers is about $O({n^{old}}^3)$.

3.2 With partially labeled target data

Here we study the scenario where we have partially labeled data from the target domain (either passively or actively annotated as described in Section 4). For each labeled data $\mathbf{x}_i \in \mathcal{X}^L$, as discussed in Section 2, we have a set of labels $y_{ik},\ k = 1, \ldots, K$. $y_{ik} = 1$ (or -1) indicates that $\mathbf{x}_i$ is labeled as positive (or negative) to C_k, and $y_{ik} = 0$ indicates that $\mathbf{x}_i$ is not labeled for C_k. Intuitively, we can combine $\mathcal{X}^L$ and $\mathcal{X}^{old}$ to retrain classifiers [5, 6, 8], which is usually computationally intensive. Also, users may provide annotations incrementally. Therefore, it is desirable to incrementally adapt $\mathbf{U}^{old}$ according to users' new annotations without retraining classifiers over all of the data.

The LAC-SVM algorithm described in Section 3.1 can be naturally extended to include new labeled data as follows. We add the labeled data $\mathcal{X}^L$ into the set of support vectors by assigning a set of parameters $\mathbf{u}^{new}_i = [\mu^{new}_{1i}, \ldots, \mu^{new}_{Ki}]^T$ to each data sample $\mathbf{x}_i \in \mathcal{X}^L$, where:

$$\mu^{new}_{ki} = \begin{cases} \eta \min_i(\mu_{ki}),\ y_{ik} = -1 \\ y_{ik}\max_i(\mu_{ki}),\ \text{others} \end{cases}. \quad (19)$$

Parameter μ_{ki} is the parameter in the original $\mathbf{U}^{old}$. A weight $\eta \in [0, 1]$ is added to the negative new labeled samples. Due to the unbalancing between positive and negative samples in some real applications, *i.e.*, negative samples significantly outnumber positive ones for some concepts, we may need to treat positive and negative samples unequally. In our experiments, for instance, the negative samples are often 10 to 100 times more than the positive ones, and we empirically set η to be 0.05 or 0.1. A better setting of η can also be obtained through cross-validation.

Let $\mathbf{U}^L = [\mathbf{u}^{new}_1, \ldots, \mathbf{u}^{new}_{n^L}]$. We can get the new amended $(n^{old}+n^L)\times K$ parameter matrix $\hat{\mathbf{U}}^{old} = [\mathbf{U}^{old^T}, \mathbf{U}^{L^T}]^T$. To learn the adapted $(n^{old}+n^L)\times K$ parameter matrix $\mathbf{U}^{new}$ and the updated normalized concept affinity matrix $\tilde{\mathbf{W}}^{c,new}$, we replace $\mathbf{U}^{old}$ with $\hat{\mathbf{U}}^{old}$ and recompute $\tilde{\mathbf{F}}$ in Eqn. (5) as:

$$\tilde{\mathbf{F}} = K(\mathcal{X}^{new}; \mathcal{X}^{old} \cup \mathcal{X}^L)\mathbf{U}^{new},$$

where $K(\mathcal{X}^{new}; \mathcal{X}^{old} \cup \mathcal{X}^L)$ is the kernel matrix of set $\mathcal{X}^{new}$ against the combined set $\mathcal{X}^{old} \cup \mathcal{X}^L$. Then the algorithm described in Section 3.1 can be conducted directly.

Similar to the case without newly labeled target data, the major part of computation comes from the matrix multiplication in Eqn. (12), which is about $O(\max\{(n^{old}+n^L), n^{new}\}\times n^{new}\times K)$. Since the number of newly labeled target data n^L is usually much smaller than n^{old}, the time complexity of LAC-SVM remains almost unchanged. In the case of $\lambda_c = 0$, the closed form solution of $\mathbf{U}^{new}$ in Eqn. (18) turns to:

$$\mathbf{U}^{new} = \left[\mathbf{I} + \frac{\lambda^d}{2}\mathbf{K}(\mathcal{X}^{old}\cup\mathcal{X}^L;\mathcal{X}^{new})\mathbf{L}^d\mathbf{K}(\mathcal{X}^{new};\mathcal{X}^{old}\cup\mathcal{X}^L)\right]^{-1}\hat{\mathbf{U}}^{old}.$$

The time complexity is about $O((n^{old}+n^L)^3)$, which is almost the same with the case without newly labeled target data.

4. ACTIVE DATA-CONCEPT SELECTION

We adopt the active selection mechanism to choose a set of informative data-concept pairs, *i.e.*, data with associated concepts to label, so that the entire data set from the target domain can be better classified to various concepts. Previous two-dimension data-concept selection approaches [14] do not consider domain change issues. They minimize the classification risk over the entire data set, and the selection process is often time consuming, especially for large-scale problems. In our work, the use of data affinity and concept affinity relations enables a simple but effective data-concept pair selection method with small complexity.

The *EigenVector Centrality* (*EVC*) [12] over a graph is widely used to measure the importance of graph nodes. Given a graph $\mathcal{G} = [\mathcal{V}, E, \mathbf{W}]$, the EVC of vertices $\mathcal{V}$ can be described as follows: the eigenvector $\mathbf{s}$ corresponding to the largest eigenvalue of the eigenvalue problem, $\mathbf{Ws} = \lambda\mathbf{s}$, gives the importance of vertices. Based on this idea, we can obtain the importance $\mathbf{s}^d$ of data in $\mathcal{X}$ by eigendecomposition of the data affinity matrix $\mathbf{W}^d$. Also, we can obtain the importance $\mathbf{s}^c$ of concepts by eigendecomposition of $\mathbf{W}^c$.

To determine the importance of data-concept pairs, in addition to $\mathbf{s}^d$ and $\mathbf{s}^c$, we also consider how much a data-concept pair can benefit from the user's annotation. Intuitively, if an automatic concept detector can give accurate prediction and also, this detector is confident about its prediction over a particular datum, we can trust its prediction. From the source domain we can measure the performance of concept classifiers, *e.g.*, through cross-validation. Let p_k denote the accuracy of the classifier from the source domain to detect a concept C_k. Let q_{ki} denote the confidence of a classifier to detect C_k from a datum $\mathbf{x}_i$. q_{ki} can be determined by the distance δ_{ki} between this datum to the decision boundary, *i.e.*, $q_{ki} = 1/(1 + exp(-\delta_{ki}))$. Then we can construct a $K\times n^{new}$ matrix $\mathbf{S}$ where each entry $S_{ki} = (1-p_k)/q_{ki}$ measures how much a data-concept pair $(C_k, \mathbf{x}_i)$ needs help from the user's annotation (n^{new} is the size of $\mathcal{X}^{new}$ in the target domain). Define matrix $\tilde{\mathbf{S}}$ and each entry $\tilde{S}_{ki}$ is:

$$\tilde{S}_{ki} = S_{ki}\cdot s^d_i + \sigma \cdot s^c_k, \quad (20)$$

where s^d_i is the EVC importance of $\mathbf{x}_i$ in $\mathbf{s}^d$, and s^c_k is the EVC importance of C_k in $\mathbf{s}^c$. The first term $S_{ki}\cdot s^d_i$ measures the importance of a data-concept pair $(C_k, \mathbf{x}_i)$ when we treat different concepts equally. $\tilde{S}_{ki}$ gives the final importance of the pair $(C_k, \mathbf{x}_i)$. The value σ is a preset weight parameter

that determines how much we rely on concept relations. For example, σ is empirically set as 0.05 in our experiments, where the concept relations obtained from the source domain is not very strong. We rank entries of matrix $\tilde{\mathbf{S}}$ in descending order and select the top M pairs for the user to label.

5. EXPERIMENTS

We evaluate the LAC-SVM algorithm over three data sets: the TRECVID 2007 development set [15], Kodak's consumer benchmark set [11], and the CCV set [10]. First, we adaptively apply classifiers trained using the TRECVID data to classify Kodak's consumer benchmark videos, where there is significant domain difference. Second, we adaptively apply classifiers trained using the CCV data to classify Kodak's consumer videos. Since both sets are from the consumer domain, we evaluate LAC-SVM when there is moderate domain change. The performance measures are *Average Precision* (*AP*, area under the uninterpolated PR curve) and *Mean AP* (*MAP*, average of APs across various concepts).

5.1 TRECVID 2007 to Kodak's benchmark

The TRECVID 2007 development set contains 50 hours of videos in Dutch (mainly documentary videos). Kodak's consumer benchmark set contains 1358 videos from about 100 actual users. Among the 39 concepts annotated over the TRECVID data, 5 concepts are similar to the concepts annotated over Kodak's benchmark data. They are animal (animal), boat-ship (boat), crowd (crowd), people-marching (parade), and sports (sports), where concepts in parentheses are defined for Kodak's set. We adaptively apply the 5 concept detectors trained over the TRECVID data to Kodak's data. Following experiments in [3], all compared algorithms use the RBF kernel and the global color and texture features.

We evaluate three scenarios where we do not have newly labeled target data, have passively labeled target data, or have actively labeled target data, in Kodak's consumer set. Algorithms in these scenarios are marked by "(n)", "(p)", and "(a)", respectively, *e.g.*, "(n) LAC-SVM", "(p) LAC-SVM", and "(a) LAC-SVM". Figure 1 shows the performance comparison in the first scenario where we compare LAC-SVM with the semi-supervised LapSVM [1] and the original SVM (directly applying TRECVID-based SVMs). For LapSVM, we train classifiers by treating the TRECVID 2007 data as training data and Kodak's consumer data as unlabeled data. This is one intuitive alternative way to learn classifiers using information from both data sets without new annotations. The results show that LAC-SVM can improve the performance of original TRECVID-based SVMs by up to 49% (over parade) in terms of AP on a relative basis. The overall MAP is improved by 8.5%. LapSVM, which treats both data sets as from the same distribution, does not perform well due to the non-negligible domain difference.

Figure 2 shows the MAP comparison in the second scenario with different numbers of passively annotated data from the target domain. A set of randomly selected data are fully annotated into all 5 concepts in Kodak's benchmark set. In this case, the number of annotations N_a is counted as $N_a = N_d * K$, where N_d is the number of randomly selected data and K is the total number of concepts ($K=5$ in this experiment). Results given in Figure 2 are the averaged results over 10 random runs. Figure 3 shows the results in the third scenario with different numbers of actively annotated data from the target domain, *i.e.*, the system actively selects the optimal data-concept pairs for the user to

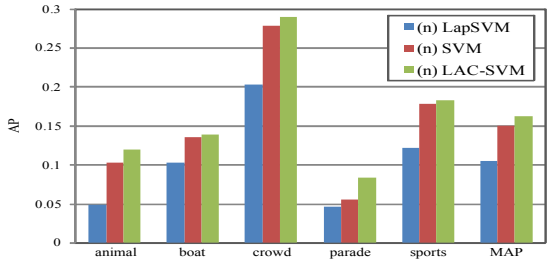

Figure 1: TRECVID to Kodak's benchmark: $n^L = 0$.

annotate in Kodak's benchmark set. In this case, the number of annotations N_a is the number of data-concept pairs labeled by the user. In both scenarios, we compare LAC-SVM with two other alternatives: Combined SVMs using all labeled data from both the TRECVID set and Kodak's benchmark set ("re-SVM"), and the cross-domain A-SVM [18] of adapting TRECVID-based SVMs to Kodak's data. From the figures we can see that for both passive and active annotation, LAC-SVM can effectively improve the classification performance by outperforming the Combined SVM. In comparison, A-SVM can not improve detection because it updates classifiers only based on the few labeled target data that are often biased. The results indicate the advantage of our method by both using information from unlabeled data and adapting classifiers to accommodate the domain change.

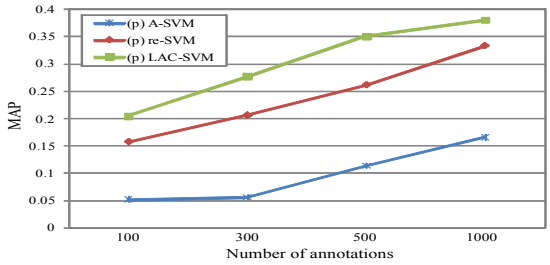

Figure 2: TRECVID to Kodak's benchmark data: with passively annotated new target data.

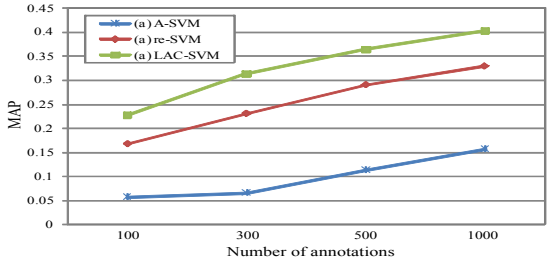

Figure 3: TRECVID to Kodak's benchmark data: with actively annotated new target data.

To evaluate the incremental learning ability of LAC-SVM, we randomly separate Kodak's benchmark set into 3 subsets, and adapt the TRECVID-based SVMs incrementally over these 3 subsets. We conduct 5 random runs and report the average performance. Within each subset, 100 data-concept pairs are actively selected for the user to annotate (*i.e.*, 300 annotations over Kodak's entire set). Table 1 gives the final MAP of the incremental LAC-SVM over Kodak's entire set after 3 incremental adaptations. The table also includes the MAPs of the non-incremental LAC-SVM, Combined SVM and A-SVM, with 300 passive or active annotations over Kodak's entire set. The comparison shows that the incremental LAC-SVM is slightly worse than, yet comparable to, the non-incremental LAC-SVM. Both incremental and non-incremental LAC-SVM algorithms outperform A-SVM and

Table 1: The TRECVID set to Kodak's benchmark data: MAP comparison of various algorithms.

(p) A-SVM	(p) re-SVM	(p) LAC-SVM	(a) A-SVM	(a) re-SVM	(a) LAC-SVM	incremental LAC-SVM
0.0557	0.2058	0.2767	0.0657	0.2306	0.3137	0.3023

Combined SVM. The results demonstrate the effectiveness of LAC-SVM in accommodating the challenging practical scenario where the newly acquired data and users' annotations are accumulated incrementally. With only 300 annotated data-concept pairs (that amount to only 4% annotation rate of Kodak's data in the target domain), the "(a) LAC-SVM" can double the MAP performance of directly applying TRACVID-based SVMs.

5.2 CCV to Kodak's benchmark

Here we evaluate the performance of LAC-SVM within the consumer domain. The CCV data set contains 9317 consumer videos downloaded from the YouTube web sharing site. These videos are annotated to 20 consumer concepts, among which 5 concepts are the same with the concepts annotated over Kodak's benchmark data. They are birthday, beach, parade, playground, and skiing. Therefore, we adaptively apply the 5 detectors trained over the CCV set to Kodak's benchmark data. Following experiments in [10], all compared algorithms use the χ^2 kernel with the *Bag-of-Features* (*BoF*) representation. It is worth mentioning that the actual interconceptual relations over the 5 concepts computed from the CCV set are very weak, *i.e.*, $\mathbf{W}^{c,old} \approx \mathbf{0}$. In such a case, we experiment on the simple version of LAC-SVM where $\lambda_c = 0$ in Eqn. (10). That is, the adapted target classifiers have closed-form solution as described in Eqn. (18). In this situation, the active data-concept selection criterion reduces to a simple version too, *i.e.*, $\sigma = 0$ in Eqn. (20). Figure 4 shows the MAP comparisons with different numbers of passively or actively annotated data from the target domain. From the figure, LAC-SVM can consistently improve the classification performance by outperforming the Combined SVM and A-SVM, which is similar to the conclusions we get in Section 5.1.

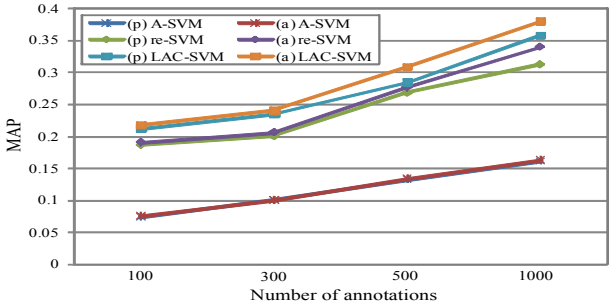

Figure 4: The CCV set to Kodak's benchmark data: with passively and actively annotated new target data.

6. CONCLUSION

We propose an LAC-SVM approach to improve concept detection by jointly using cross-domain, semi-supervised, multi-concept, and active learning. LAC-SVM adaptively applies previous classifiers and concept affinity relations computed from a source domain to detect concepts in the target domain, while incrementally updating both the classifiers and concept relations. Through iteratively conducting active data-concept annotation and model adaptation, LAC-SVM gives an effective framework to accommodate the challenging practical concept detection problem, where there can be large domain changes, few or no partially labeled target data, and incrementally acquired new data and annotations.

7. REFERENCES

[1] M. Belkin, P. Niyogi, and V. Sindhwani, Manifold regularization: a geometric framework for learning from labeled and unlabeled examples. *Journal of Machine Learning Research*, 7(11):2399-2434, 2006.

[2] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, Cambridge, 2004.

[3] S. Chang, D. Ellis, W. Jiang, K. Lee, A. Yanagawa, A. Loui, and J. Luo. Large-scale multimodal semantic concept detection for consumer video. In *Proc. ACM MIR*, pages 255–264, 2007.

[4] O. Chapelle, B. Scholkopf, and A. Ziene. *Semi-supervised Learning*. MIT Press, Cambridge, MA, 2006.

[5] H. Daumé. Frustratingly easy domain adaptation. In *Proc. the 45th Annual Meeting of the Association of Computational Linguistics*, pages 256–263, 2007.

[6] L. Duan, I. Tsang, D. Xu, and S.J. Maybank. Domain transfer svm for video concept detection. In *IEEE CVPR*, pages 1375–1381, 2009.

[7] W. Jiang, S. Chang, and A. Loui. Active context-based concept fusion with partial user labels. In *IEEE ICIP*, pages 2917–2920, 2006.

[8] W. Jiang, E. Zavesky, S. Chang, and A. Loui. Cross-domain learning methods for high-level visual concept classification. In *IEEE ICIP*, 2008.

[9] Y. Jiang, J. Wang, S. Chang, and C.W. Ngo. Domain adaptive semantic diffusion for large scale context-based video annotation. In *IEEE ICCV*, 2009.

[10] Y. Jiang, G. Ye, S. Chang, D. Ellis, and A. Loui. Consumer video understanding: A benchmark database and an evaluation of human and machine performance. In *ACM ICMR*, 2011. Trento, Italy.

[11] A. Loui, J. Luo, S. Chang, D. Ellis, W. Jiang, L. Kennedy, K. Lee, and A. Yanagawa. Kodak's consumer video benchmark data set: Concept definition and annotation. *ACM MIR*, 2007.

[12] M. Newman. *Mathematics of Networks*. The New Palgrave Encyclopedia of Economics, 2nd Edition, L.E. Blume and S.N. Durlauf (eds.), Palgrave Macmillan, Basingstoke, 2008.

[13] G. Qi and *et al*. Correlative multi-label video annotation. In *ACM Multimedia*, pages 17–26, 2007.

[14] G. Qi, X.S. Hua, Y. Rui, J. Tang, and H.J. Zhang. Two-dimensional active learning for image classification. In *IEEE CVPR*, pages 1–8, 2008.

[15] A.F. Smeaton, P. Over, and W. Kraaij. Evaluation campaigns and TRECVid. *ACM MIR*, 2006.

[16] S. Tong and E. Chang. Support vector machine active learning for image retrieval. In *ACM Multimedia*, 2001.

[17] V. Vapnik. *Statistical Learning Theorys*. Wiley-Interscience, New York, 1998.

[18] J. Yang, R. Yan, and A. Hauptmann. Cross-domain video concept detection using adaptive svms. In *ACM Multimedia*, pages 188–197, 2007.