Soft Margin Consistency Based Scalable Multi-View Maximum Entropy Discrimination

Liang Mao, Shiliang Sun
Shanghai Key Laboratory of Multidimensional Information Processing,
Department of Computer Science and Technology, East China Normal University,
500 Dongchuan Road, Shanghai 200241, China
lmao14@outlook.com, slsun@cs.ecnu.edu.cn

Abstract

Multi-view learning receives increasing interest in recent years to analyze complex data. Lately, multi-view maximum entropy discrimination (MVMED) and alternative MVMED (AMVMED) were proposed as extensions of maximum entropy discrimination (MED) to the multi-view learning setting, which use the hard margin consistency principle that enforces two view margins to be the same. In this paper, we propose soft margin consistency based multi-view MED (SMVMED) achieving margin consistency in a less strict way, which minimizes the relative entropy between the posteriors of two view margins. With a trade-off parameter balancing large margin and margin consistency, SMVMED is more flexible. We also propose a sequential minimal optimization (SMO) algorithm to efficiently train SMVMED and make it scalable to large datasets. We evaluate the performance of SMVMED on multiple real-world datasets and get encouraging results.

1 Introduction

Heterogeneous data analysis is a timely and important task for the artificial intelligence community. For many real-world datasets, features can be naturally partitioned into distinct sets, each of which is regarded as a view. Then each datum can be described by multiple views. For instance, a web page can be described by the text on it and words appearing in the hyperlinks to it, each of which forms a single view. As another example, in multimedia content understanding, multimedia segments can be described by both their audio and video signals.

Multi-view learning is the learning scheme that utilizes the heterogeneous property of datasets. Compared with single view learning, multi-view learning learns a function on each view and train them jointly to improve performance. Xu, Tao and Xu [2013] and Sun [2013] have surveyed the development and applications of multi-view learning.

Maximum entropy discrimination (MED) [Jaakkola *et al.*, 2000] is an effective approach to learn a discriminative classifier as well as consider uncertainties over model parameters, which combines generative and discriminative learning. Rather than find a single classifier parameter $\mathbf{\Theta}$ of the discriminant function $L(\mathbf{\Theta})$ (e.g., $L(X_t|\mathbf{\Theta}) = \boldsymbol{\theta}^{\mathrm{T}} X_t + b$, $\mathbf{\Theta} = \{\boldsymbol{\theta},\, b\}$), MED considers to learn a distribution $p(\mathbf{\Theta})$ over classifier parameter $\mathbf{\Theta}$. After obtaining a joint distribution $p(\mathbf{\Theta}, \boldsymbol{\gamma})$ over $\mathbf{\Theta}$ and margin parameters $\boldsymbol{\gamma}$ by minimizing its relative entropy (also known as Kullback-Leiber divergence, or KL divergence) with respect to some prior target distribution $p_0(\mathbf{\Theta}, \boldsymbol{\gamma})$ under certain large margin constraints, MED marginalizes out $\boldsymbol{\gamma}$ to obtain $p(\mathbf{\Theta})$ [Jebara, 2004]. MED can be extended to a wide variety of learning scenarios, such as feature selection [Jebara and Jaakkola, 2000], multitask learning [Jebara, 2011] and structure learning [Zhu and Xing, 2009; Zhu *et al.*, 2008a; 2008b].

Recently, multi-view maximum entropy discrimination (MVMED) [Sun and Chao, 2013] was proposed as an extension of MED to the multi-view learning setting. It considers a joint distribution $p(\mathbf{\Theta_1}, \mathbf{\Theta_2})$ over the view 1 classifier parameter $\mathbf{\Theta}_1$ and view 2 classifier parameter $\mathbf{\Theta}_2$. Using the augmented joint distribution $p(\mathbf{\Theta}_1, \mathbf{\Theta}_2, \boldsymbol{\gamma})$, MVMED was formulated as follows

$$\min_{p(\mathbf{\Theta}_1,\mathbf{\Theta}_2,\boldsymbol{\gamma})} \mathrm{KL}(p(\mathbf{\Theta}_1, \mathbf{\Theta}_2, \boldsymbol{\gamma}) \,||\, p_0(\mathbf{\Theta}_1, \mathbf{\Theta}_2, \boldsymbol{\gamma}))$$
$$\text{s.t.} \begin{cases} \int p(\mathbf{\Theta}_1, \mathbf{\Theta}_2, \boldsymbol{\gamma})[y_t L_1(X_t^1|\mathbf{\Theta}_1) - \gamma_t] d\mathbf{\Theta}_1 d\mathbf{\Theta}_2 d\boldsymbol{\gamma} \geq 0 \\ \int p(\mathbf{\Theta}_1, \mathbf{\Theta}_2, \boldsymbol{\gamma})[y_t L_2(X_t^2|\mathbf{\Theta}_2) - \gamma_t] d\mathbf{\Theta}_1 d\mathbf{\Theta}_2 d\boldsymbol{\gamma} \geq 0 \\ 1 \leq t \leq N, \end{cases} \tag{1}$$

where $L_1(X_t^1|\mathbf{\Theta}_1)$ and $L_2(X_t^2|\mathbf{\Theta}_2)$ are discriminant functions from two views, respectively. Chao and Sun [2015] also proposed a similar MVMED framework called alternative MVMED (AMVMED), which considers two separate distributions $p_1(\mathbf{\Theta}_1)$ over $\mathbf{\Theta}_1$ and $p_2(\mathbf{\Theta}_2)$ over $\mathbf{\Theta}_2$ and balances KL divergences of their augmented distributions with respect to the corresponding prior distributions. AMVMED was formulated as

$$\begin{aligned} \min_{p_1(\mathbf{\Theta}_1,\boldsymbol{\gamma}),\, p_2(\mathbf{\Theta}_2,\boldsymbol{\gamma})} & \rho \mathrm{KL}(p_1(\mathbf{\Theta}_1, \boldsymbol{\gamma}) \,||\, p_0(\mathbf{\Theta}_1, \boldsymbol{\gamma})) \\ & + (1-\rho)\mathrm{KL}(p_2(\mathbf{\Theta}_2, \boldsymbol{\gamma}) \,||\, p_0(\mathbf{\Theta}_2, \boldsymbol{\gamma})) \\ \text{s.t.} & \begin{cases} \int p(\mathbf{\Theta}_1, \boldsymbol{\gamma})\, [y_t L_1(X_t^1|\mathbf{\Theta}_1) - \gamma_t]\, d\mathbf{\Theta}_1 d\boldsymbol{\gamma} \geq 0 \\ \int p(\mathbf{\Theta}_2, \boldsymbol{\gamma})\, [y_t L_2(X_t^2|\mathbf{\Theta}_2) - \gamma_t]\, d\mathbf{\Theta}_2 d\boldsymbol{\gamma} \geq 0 \\ \int p(\mathbf{\Theta}_1, \boldsymbol{\gamma}) d\mathbf{\Theta}_1 = \int p(\mathbf{\Theta}_2, \boldsymbol{\gamma}) d\mathbf{\Theta}_2 \\ 1 \leq t \leq N. \end{cases} \end{aligned} \tag{2}$$

Unlike conventional multi-view learning methods, MVMED and AMVMED exploit the multiple views in a different style called margin consistency, that is, to enforce the margins from two views to be identical. Although they have provided state-of-the-art multi-view learning performance, this margin consistency requirement may be too strong to fulfill in many cases. For example, all positive margins can lead to the same label prediction in binary classifications. It is thus interesting to explore the possibility of relaxing the requirement. Moreover, as far as we know, the current training algorithms for MVMED and AMVMED are inefficient and not scalable, which prevent people from using them to large datasets.

In this paper, we propose a new multi-view MED framework named soft margin consistency based multi-view MED (SMVMED) which is based on the different principle of soft margin consistency. We give an iterative method to approximate the solution. Compared with MVMED and AMVMED with 'hard' margin consistency that enforces the margins from two views to be identical, SMVMED achieves 'soft' margin consistency by utilizing the sum of two KL divergences $\mathrm{KL}(p(\boldsymbol{\gamma})\,||\,q(\boldsymbol{\gamma}))$ and $\mathrm{KL}(q(\boldsymbol{\gamma})\,||\,p(\boldsymbol{\gamma}))$ in the objective function, where $p(\boldsymbol{\gamma})$ and $q(\boldsymbol{\gamma})$ are the posteriors of two view margins, respectively. By balancing all the involved terms in the objective function, SMVMED is more flexible. Moreover, we propose a sequential minimal optimization (SMO) algorithm [Platt, 1999] to effectively train SMVMED for large datasets under some configurations on priors of model and margin parameters.

The rest of the paper is organized as follows. We first introduce our SMVMED and derive its solution. Then a practical realization of SMVMED will be given. Next we describe the SMO algorithm for fast training and scalability. After that, we report experiments on multiple real-world datasets. Finally, conclusions are given.

2 Soft Margin Consistency Based Multi-view MED

In this section, we give the formal framework of our soft margin consistency based multi-view MED. It achieves margin consistency by minimizing the KL-divergence between the posteriors of margin parameters from two views. We also introduce a trade-off parameter balancing large margin and margin consistency to make the model more flexible.

Suppose we are given a multi-view dataset $\{X_t^1, X_t^2, y_t\}$, $t = 1, \ldots, N$, where X_t^1 and X_t^2 indicate the tth input from view 1 and view 2, respectively, and $y_t \in \{\pm 1\}$ is the label. SMVMED aims to learn two discriminant functions $L_1(X_t^1|\boldsymbol{\Theta}_1)$ and $L_2(X_t^2|\boldsymbol{\Theta}_2)$ for two views, respectively, where $\boldsymbol{\Theta}_1$ and $\boldsymbol{\Theta}_2$ are parameters of these two functions. First, we assume that there are two independent distributions $p(\boldsymbol{\Theta}_1)$ and $p(\boldsymbol{\gamma})$, with their joint distribution $p(\boldsymbol{\Theta}_1, \boldsymbol{\gamma}) = p(\boldsymbol{\Theta}_1)p(\boldsymbol{\gamma})$, where $\boldsymbol{\gamma} = \{\gamma_t\}$, $t = 1, \ldots, N$, is the margin parameter. Here $p(\boldsymbol{\Theta}_1)$ is the posterior of $\boldsymbol{\Theta}_1$ and $p(\boldsymbol{\gamma})$ is the posterior of margins from view 1. Then the same settings are applied to view 2, that is, $q(\boldsymbol{\Theta}_2, \boldsymbol{\gamma}) = q(\boldsymbol{\Theta}_2)q(\boldsymbol{\gamma})$, where $q(\boldsymbol{\gamma})$ is the posterior of margins from view 2. Formally, SMVMED can be formulated as follows:

$$\begin{aligned}
\min_{p(\boldsymbol{\Theta}_1,\boldsymbol{\gamma}),\, q(\boldsymbol{\Theta}_2,\boldsymbol{\gamma})} & \mathrm{KL}(p(\boldsymbol{\Theta}_1)\,||\,p_0(\boldsymbol{\Theta}_1)) \\
& + \mathrm{KL}(q(\boldsymbol{\Theta}_2)||q_0(\boldsymbol{\Theta}_2)) \\
& +(1-\alpha)\mathrm{KL}(p(\boldsymbol{\gamma})\,||\,p_0(\boldsymbol{\gamma})) + (1-\alpha)\mathrm{KL}(q(\boldsymbol{\gamma})||q_0(\boldsymbol{\gamma})) \\
& + \alpha\mathrm{KL}(p(\boldsymbol{\gamma})\,||\,q(\boldsymbol{\gamma})) + \alpha\mathrm{KL}(q(\boldsymbol{\gamma})\,||\,p(\boldsymbol{\gamma})) \\
\text{s.t.} & \begin{cases} \int p(\boldsymbol{\Theta}_1, \boldsymbol{\gamma})\,[y_t L_1(X_t^1|\boldsymbol{\Theta}_1) - \gamma_t]\, d\boldsymbol{\Theta}_1 d\boldsymbol{\gamma} \geq 0 \\ \int q(\boldsymbol{\Theta}_2, \boldsymbol{\gamma})\,[y_t L_2(X_t^2|\boldsymbol{\Theta}_2) - \gamma_t]\, d\boldsymbol{\Theta}_2 d\boldsymbol{\gamma} \geq 0 \\ 1 \leq t \leq N. \end{cases}
\end{aligned} \tag{3}$$

Since we will choose the margin priors that favor large margins, the parameter α above plays the trade-off role of balancing large margin and soft margin consistency.

Since it is tricky to find the solutions making the partial derivatives of the Lagrangian of (3) with respect to $p(\boldsymbol{\Theta}_1, \boldsymbol{\gamma})$ and $p(\boldsymbol{\Theta}_1, \boldsymbol{\gamma})$ be zero, we propose an iterative scheme for finding a solution to (3). In the mth iteration, we successively update $p^{(m)}(\boldsymbol{\Theta}_1, \boldsymbol{\gamma})$ and $q^{(m)}(\boldsymbol{\Theta}_2, \boldsymbol{\gamma})$ by solving the following two problems:

$$\begin{aligned}
& p^{(m)}(\boldsymbol{\Theta}_1, \boldsymbol{\gamma}) \\
= & \operatorname{argmin}_{p^{(m)}(\boldsymbol{\Theta}_1,\boldsymbol{\gamma})} \mathrm{KL}(p^{(m)}(\boldsymbol{\Theta}_1)\,||\,p_0(\boldsymbol{\Theta}_1)) \\
& + (1-\alpha)\mathrm{KL}(p^{(m)}(\boldsymbol{\gamma})\,||\,p_0(\boldsymbol{\gamma})) \\
& + \alpha\mathrm{KL}(p^{(m)}(\boldsymbol{\gamma})\,||\,q^{(m-1)}(\boldsymbol{\gamma})) \\
& \text{s.t.} \begin{cases} \int p^{(m)}(\boldsymbol{\Theta}_1, \boldsymbol{\gamma})\,[y_t L_1(X_t^1|\boldsymbol{\Theta}_1) - \gamma_t]\, d\boldsymbol{\Theta}_1 d\boldsymbol{\gamma} \geq 0 \\ 1 \leq t \leq N, \end{cases}
\end{aligned} \tag{4}$$

and

$$\begin{aligned}
& q^{(m)}(\boldsymbol{\Theta}_2, \boldsymbol{\gamma}) \\
= & \operatorname{argmin}_{q^{(m)}(\boldsymbol{\Theta}_2,\boldsymbol{\gamma})} \mathrm{KL}(q^{(m)}(\boldsymbol{\Theta}_2)\,||\,q_0(\boldsymbol{\Theta}_2)) \\
& + (1-\alpha)\mathrm{KL}(q^{(m)}(\boldsymbol{\gamma})\,||\,q_0(\boldsymbol{\gamma})) \\
& + \alpha\mathrm{KL}(q^{(m)}(\boldsymbol{\gamma})\,||\,p^{(m)}(\boldsymbol{\gamma})) \\
& \text{s.t.} \begin{cases} \int q^{(m)}(\boldsymbol{\Theta}_2, \boldsymbol{\gamma})\,[y_t L_2(X_t^2|\boldsymbol{\Theta}_2) - \gamma_t]\, d\boldsymbol{\Theta}_2 d\boldsymbol{\gamma} \geq 0 \\ 1 \leq t \leq N. \end{cases}
\end{aligned} \tag{5}$$

Notice that we omit $\alpha\mathrm{KL}(q^{(m-1)}(\boldsymbol{\gamma})\,||\,p^{(m)}(\boldsymbol{\gamma}))$ in (4) for the purpose of simplifying the subsequent calculation of derivatives. We think $\alpha\mathrm{KL}(p^{(m)}(\boldsymbol{\gamma})\,||\,q^{(m-1)}(\boldsymbol{\gamma}))$ is a good approximation to $\alpha\mathrm{KL}(p^{(m)}(\boldsymbol{\gamma})\,||\,q^{(m-1)}(\boldsymbol{\gamma})) + \alpha\mathrm{KL}(q^{(m-1)}(\boldsymbol{\gamma})\,||\,p^{(m)}(\boldsymbol{\gamma}))$ since the latter is just the symmetrization version of the former. We omit $\alpha\mathrm{KL}(p^{(m)}(\boldsymbol{\gamma})\,||\,q^{(m)}(\boldsymbol{\gamma}))$ in (5) for the same reason.

Before employing this iterative scheme, we first choose some initial value for $q^{(0)}(\boldsymbol{\Theta}_2, \boldsymbol{\gamma})$. It is a proper choice to initialize $q^{(0)}(\boldsymbol{\Theta}_2, \boldsymbol{\gamma})$ with $q_0(\boldsymbol{\Theta}_2, \boldsymbol{\gamma})$, which makes (4) a standard MED problem. We will use this scheme in this paper.

The Lagrangian of (4) can be written as

$$\begin{aligned}
L = &\int p^{(m)}(\mathbf{\Theta}_1) \log \frac{p^{(m)}(\mathbf{\Theta}_1)}{p_0(\mathbf{\Theta}_1)} d\mathbf{\Theta}_1 \\
&+(1-\alpha) \int p^{(m)}(\boldsymbol{\gamma}) \log \frac{p^{(m)}(\boldsymbol{\gamma})}{p_0(\boldsymbol{\gamma})} d\boldsymbol{\gamma} \\
&+\alpha \int p^{(m)}(\boldsymbol{\gamma}) \log \frac{p^{(m)}(\boldsymbol{\gamma})}{q^{(m-1)}(\boldsymbol{\gamma})} d\boldsymbol{\gamma} \\
&-\sum_{t=1}^{N} \int p^{(m)}(\mathbf{\Theta}_1, \boldsymbol{\gamma}) \lambda_{1,t}^{(m)} [y_t L_1(X_t^1|\mathbf{\Theta}_1) - \gamma_t] \, d\mathbf{\Theta}_1 d\boldsymbol{\gamma} \\
= &\int p^{(m)}(\mathbf{\Theta}_1, \boldsymbol{\gamma}) \log \frac{p^{(m)}(\mathbf{\Theta}_1, \boldsymbol{\gamma})}{p_0(\mathbf{\Theta}_1)[p_0(\boldsymbol{\gamma})]^{1-\alpha}[q^{(m-1)}(\boldsymbol{\gamma})]^{\alpha}} \\
&-\sum_{t=1}^{N} \int p^{(m)}(\mathbf{\Theta}_1, \boldsymbol{\gamma}) \lambda_{1,t}^{(m)} [y_t L_1(X_t^1|\mathbf{\Theta}_1) - \gamma_t] \, d\mathbf{\Theta}_1 d\boldsymbol{\gamma},
\end{aligned} \tag{6}$$

where $\boldsymbol{\lambda}_1^{(m)} = \{\lambda_{1,t}^{(m)}\}$ is a set of nonnegative Lagrange multipliers, one for each classification constraint. After taking the partial derivative of (6) with respect to $p^{(m)}(\mathbf{\Theta}_1, \boldsymbol{\gamma})$ and setting it to zero, we will obtain the solution to (4) which has the following form

$$\begin{aligned}
p^{(m)}(\mathbf{\Theta}_1, \boldsymbol{\gamma}) = &\frac{1}{Z_1^{(m)}(\boldsymbol{\lambda}_1^{(m)})} p_0(\mathbf{\Theta}_1)[p_0(\boldsymbol{\gamma})]^{1-\alpha}[q^{(m-1)}(\boldsymbol{\gamma})]^{\alpha} \\
&\exp\left\{\sum_{t=1}^{N} \lambda_{1,t}^{(m)} [y_t L_1(X_t^1|\mathbf{\Theta}_1) - \gamma_t]\right\},
\end{aligned} \tag{7}$$

where $Z_1^{(m)}(\boldsymbol{\lambda}^{(m)})$ is the normalization constant. $\boldsymbol{\lambda}_1^{(m)}$ is set by finding the unique maximum of the following concave objective function:

$$J_1^{(m)}(\boldsymbol{\lambda}_1^{(m)}) = -\log Z_1^{(m)}(\boldsymbol{\lambda}_1^{(m)}). \tag{8}$$

Apply the same analysis to (5) and we will obtain the its solution, which has the following form

$$\begin{aligned}
q^{(m)}(\mathbf{\Theta}_2, \boldsymbol{\gamma}) = &\frac{1}{Z_2^{(m)}(\boldsymbol{\lambda}_2^{(m)})} q_0(\mathbf{\Theta}_2)[q_0(\boldsymbol{\gamma})]^{1-\alpha}[p^{(m)}(\boldsymbol{\gamma})]^{\alpha} \\
&\exp\left\{\sum_{t=1}^{N} \lambda_{2,t}^{(m)} [y_t L_2(X_t^2|\mathbf{\Theta}_2) - \gamma_t]\right\},
\end{aligned} \tag{9}$$

where $\boldsymbol{\lambda}_2^{(m)} = \{\lambda_{2,t}^{(m)}\}$ is another set of Lagrange multipliers. $\boldsymbol{\lambda}_2^{(m)}$ is set by finding the maximum of the following objective function:

$$J_2^{(m)}(\boldsymbol{\lambda}_2^{(m)}) = -\log Z_2^{(m)}(\boldsymbol{\lambda}_2^{(m)}). \tag{10}$$

After each iteration, we calculate the relative error between values of (8) from two successively iterations and that of (10), respectively, and utilize them for determining convergence. When the relative errors

$$\frac{J_1^{(m)}(\boldsymbol{\lambda}_1^{(m)}) - J_1^{(m-1)}(\boldsymbol{\lambda}_1^{(m-1)})}{J_1^{(m-1)}(\boldsymbol{\lambda}_1^{(m-1)})} \tag{11}$$

and

$$\frac{J_2^{(m)}(\boldsymbol{\lambda}_2^{(m)}) - J_2^{(m-1)}(\boldsymbol{\lambda}_2^{(m-1)})}{J_2^{(m-1)}(\boldsymbol{\lambda}_2^{(m-1)})} \tag{12}$$

are both less than some tolerance ϵ, the iteration ends. Then we obtain $p(\mathbf{\Theta}_1)$ and $q(\mathbf{\Theta}_2)$ and use the following formulas as decision rules for a new example (X^1, X^2) from view 1 and view 2, respectively

$$\hat{y}_1 = \text{sign}\left(\int p(\mathbf{\Theta}_1) \, L_1(X_t^1|\mathbf{\Theta}_1) \, d\mathbf{\Theta}_1\right), \tag{13}$$

$$\hat{y}_2 = \text{sign}\left(\int p(\mathbf{\Theta}_2) \, L_2(X_t^2|\mathbf{\Theta}_2) \, d\mathbf{\Theta}_2\right), \tag{14}$$

or use two views together

$$\begin{aligned}
\hat{y} = \text{sign}\big(&\tfrac{1}{2}\int p(\mathbf{\Theta}_1) \, L_1(X_t^1|\mathbf{\Theta}_1) \, d\mathbf{\Theta}_1 \\
&+\tfrac{1}{2}\int p(\mathbf{\Theta}_2) \, L_2(X_t^2|\mathbf{\Theta}_2) \, d\mathbf{\Theta}_2\big) \, .
\end{aligned} \tag{15}$$

3 Practical Realization

In this section, we discuss the practical realization of SMVMED. Since the priors $p_0(\mathbf{\Theta}_1, \boldsymbol{\gamma})$ and $q_0(\mathbf{\Theta}_2, \boldsymbol{\gamma})$ play an important role in the solution, we shall give their concrete formulations. Also, we use the linear classifier assumptions, that is,

$$L_1(X_t^1|\mathbf{\Theta}_1) = \boldsymbol{\theta}_1^{\mathrm{T}} X_t^1 + b_1, \tag{16}$$

$$L_2(X_t^2|\mathbf{\Theta}_2) = \boldsymbol{\theta}_2^{\mathrm{T}} X_t^2 + b_2. \tag{17}$$

We will show that the particular configuration leads to the implementation of an SMO algorithm for fast training.

Suppose

$$p_0(\mathbf{\Theta}_1, \boldsymbol{\gamma}) = p_0(\mathbf{\Theta}_1) p_0(\boldsymbol{\gamma}) = p_0(\boldsymbol{\theta}_1) p_0(b_1) p_0(\boldsymbol{\gamma}), \tag{18}$$

$$q_0(\mathbf{\Theta}_2, \boldsymbol{\gamma}) = q_0(\mathbf{\Theta}_2) q_0(\boldsymbol{\gamma}) = q_0(\boldsymbol{\theta}_2) q_0(b_2) q_0(\boldsymbol{\gamma}), \tag{19}$$

where $p_0(\boldsymbol{\theta}_1)$ and $q_0(\boldsymbol{\theta}_2)$ are Gaussian distributions with mean 0 and standard deviation I, $p_0(b_1)$ and $q_0(b_2)$ are set to non-informative Gaussian distributions, and $p_0(\boldsymbol{\gamma})$ and $q_0(\boldsymbol{\gamma})$ are assumed to be fully factorized, namely,

$$p_0(\boldsymbol{\gamma}) = \prod_{t=1}^{N} p_0(\gamma_t), \tag{20}$$

$$q_0(\boldsymbol{\gamma}) = \prod_{t=1}^{N} q_0(\gamma_t), \tag{21}$$

with $p_0(\gamma_t) = q_0(\gamma_t) = \frac{c}{\sqrt{2\pi}} e^{-\frac{c^2}{2}(1-\gamma_t)^2}$, a Gaussian prior with mean 1 that encourages large margins.

Under the above configuration, we can derivate a concrete form of (8) and (10). Before doing that, we first introduce some notations to simplify the derivation. We use k to indicate the times of optimization problems that we have solved and set $l = (k \bmod 2) + 1$. Let $Z^{(k)}(\boldsymbol{\lambda}^{(k)})$ represent $Z_1^{(m)}(\boldsymbol{\lambda}_1^{(m)})$ when $k = 2m - 1$, and represent $Z_2^{(m)}(\boldsymbol{\lambda}_2^{(m)})$ when $k = 2m$. $J^{(k)}(\boldsymbol{\lambda}^{(k)})$ is used in the same manner. After that, $Z_1^{(m)}(\boldsymbol{\lambda}_1^{(m)})$ in (7) and $Z_2^{(m)}(\boldsymbol{\lambda}_2^{(m)})$ in (9) become

$$\begin{aligned}
&Z^{(k)}(\boldsymbol{\lambda}^{(k)})\\
&=\int N(\boldsymbol{\theta}_l|\mathbf{0},\mathrm{I})\exp\left\{\boldsymbol{\theta}_l^{\mathbf{T}}\left(\sum_{t=1}^{N}\lambda_t^{(k)}y_tX_t^l\right)\right\}d\boldsymbol{\theta}_l\\
&\quad N(b_l|0,\sigma_l^2)\exp\left\{b_l\left(\sum_{t=1}^{N}\lambda_t^{(k)}y_t\right)\right\}db_l\\
&\quad \exp\left\{-\sum_{t=1}^{N}\left(\sum_{i=1}^{k}\alpha^{k-i}\lambda_t^{(i)}\right)\right\}\\
&\quad \prod_{t=1}^{N}\frac{c}{\sqrt{2\pi}}e^{-\frac{c^2}{2}(1-\gamma_t)^2}d\boldsymbol{\gamma}\\
&\quad \exp\left\{\alpha\sum_{t=1}^{N}\left[\sum_{i=1}^{k-1}\left(\alpha^{k-1-i}\lambda_t^{(i)}\right)^2-\frac{\left(\sum_{i=1}^{k-1}\alpha^{(k-1-i)}\lambda_t^{(i)}\right)^2}{2c^2}\right]\right\}\\
&=\exp\left\{\frac{1}{2}\left(\sum_{t,\tau=1}^{N}\lambda_t^{(k)}\lambda_\tau^{(k)}y_ty_\tau X_t^{l\,\mathrm{T}}X_\tau^l\right)\right.\\
&\quad \left.+\frac{\sigma^2}{2}\left(\sum_{t=1}^{N}\lambda_t^{(k)}y_t\right)^2\right\}\\
&\quad \exp\left\{-\sum_{t=1}^{N}\left(1-\frac{\sum_{i=1}^{k-1}\alpha^{k-i}\lambda_t^{(i)}}{c^2}\right)\lambda_t^{(k)}+\sum_{t=1}^{N}\frac{\left(\lambda_t^{(k)}\right)^2}{2c^2}\right\}\\
&\quad \exp\left\{-\left(1-\frac{1}{\alpha}\right)\frac{\sum_{t=1}^{N}\left(\sum_{i=1}^{k-1}\alpha^{k-i}\lambda_t^{(i)}\right)^2}{2c^2}\right\}.
\end{aligned}\tag{22}$$

Substituting it into (8) and (10), we will get

$$\begin{aligned}
&J^{(k)}(\boldsymbol{\lambda}^{(k)})\\
&=\sum_{t=1}^{N}\left(1-\frac{\sum_{i=1}^{k-1}\alpha^{k-i}\lambda_t^{(i)}}{c^2}\right)\lambda_t^{(k)}-\sum_{t=1}^{N}\frac{\left(\lambda_t^{(k)}\right)^2}{2c^2}\\
&\quad -\frac{1}{2}\left(\sum_{t,\tau=1}^{N}\lambda_t^{(k)}\lambda_\tau^{(k)}y_ty_\tau X_t^{l\,\mathrm{T}}X_\tau^l\right)\\
&\quad -\frac{\sigma^2}{2}\left(\sum_{t=1}^{N}\lambda_t^{(k)}y_t\right)^2+\left(1-\frac{1}{\alpha}\right)\frac{\sum_{t=1}^{N}\left(\sum_{i=1}^{k-1}\alpha^{k-i}\lambda_t^{(i)}\right)^2}{2c^2}.
\end{aligned}\tag{23}$$

According to the non-informative prior assumption on b_1 and b_2, we will have $\sigma_l^2 \to \infty$, which requires that $\sum_{t=1}^{N}\lambda_t^{(k)}y_t = 0$. Thus we have the following dual optimization problem

$$\begin{aligned}
\max_{\boldsymbol{\lambda^{(k)}}}\ &\sum_{t=1}^{N}\left(1-\frac{\sum_{i=1}^{k-1}\alpha^{k-i}\lambda_t^{(i)}}{c^2}\right)\lambda_t^{(k)}-\sum_{t=1}^{N}\frac{\left(\lambda_t^{(k)}\right)^2}{2c^2}\\
&-\frac{1}{2}\left(\sum_{t,\tau=1}^{N}\lambda_t^{(k)}\lambda_\tau^{(k)}y_ty_\tau X_t^{l\,\mathrm{T}}X_\tau^l\right)\\
&+\left(1-\frac{1}{\alpha}\right)\frac{\sum_{t=1}^{N}\left(\sum_{i=1}^{k-1}\alpha^{k-i}\lambda_t^{(i)}\right)^2}{2c^2}\\
\text{s.t.}\ &\begin{cases}\boldsymbol{\lambda^{(k)}}\geq\mathbf{0}\\ \sum_{t=1}^{N}\lambda_t^{(k)}y_t=0.\end{cases}
\end{aligned}\tag{24}$$

The Lagrange multipliers $\boldsymbol{\lambda}^{(k)}$ are set by solving the convex optimization problem (24). After each iteration, we obtain two sets of Lagrange multipliers and substitute them into (8) and (10). Then we can use (11) and (12) to judge convergence. After the iteration converges, we compute (22) using $\boldsymbol{\lambda}_1^{(m)}$ from the last iteration and substitute (16), (18), (20) and (22) into (4) to obtain the solution $p(\boldsymbol{\Theta}_1,\boldsymbol{\gamma})$. Similarly, using $\boldsymbol{\lambda}_2^{(m)}$ from the last iteration to compute (22) and substituting (17), (19), (21) and (22) into (5) will give $q(\boldsymbol{\Theta}_2,\boldsymbol{\gamma})$. The decision rule using view 1 can be given as

$$\hat{y}_1=\operatorname{sign}\left(\hat{\boldsymbol{\theta}}_1^{\mathrm{T}}X^1+\hat{b}_1\right),\tag{25}$$

where $\hat{\boldsymbol{\theta}}_1$ and $\hat{b}_1$ are the expected values of discriminant function parameters. $\hat{\boldsymbol{\theta}}_1$ is obtained as follows

$$\begin{aligned}
\hat{\boldsymbol{\theta}}_1&=\int p(\boldsymbol{\Theta}_1,\boldsymbol{\gamma})\boldsymbol{\theta}_1d\boldsymbol{\Theta}_1d\boldsymbol{\gamma}=\int p(\boldsymbol{\theta}_1)\boldsymbol{\theta}_1d\boldsymbol{\theta}_1\\
&=\textstyle\sum_{t=1}^{N}\lambda_{1t}^{(m)}y_tX_t^1.
\end{aligned}\tag{26}$$

With (26) substituted into (25), the prediction rule becomes

$$\hat{y}_1=\operatorname{sign}\left(\sum_{t=1}^{N}\lambda_{1t}^{(m)}y_tX_t^{1\,\mathrm{T}}X^1+\hat{b}_1\right),\tag{27}$$

Similarly, the prediction rules using view 2 and using two views together are given as

$$\hat{y}_2=\operatorname{sign}\left(\sum_{t=1}^{N}\lambda_{2t}^{(m)}y_tX_t^{2\,\mathrm{T}}X^2+\hat{b}_2\right),\tag{28}$$

$$\begin{aligned}
\hat{y}=\operatorname{sign}\bigg(&\frac{1}{2}\left(\sum_{t=1}^{N}\lambda_{1t}^{(m)}y_tX_t^{1\,\mathrm{T}}X^1+\hat{b}_1\right)\\
&+\frac{1}{2}\left(\sum_{t=1}^{N}\lambda_{2t}^{(m)}y_tX_t^{2\,\mathrm{T}}X^2+\hat{b}_2\right)\bigg).
\end{aligned}\tag{29}$$

We will give the solution to $\hat{b}_1$ and $\hat{b}_2$ in the next section.

3.1 Remarks on Margin Priors

Jaakkola et al. [3] discussed different margin priors and their corresponding potential terms in the objective function of the dual optimization problem. Both MVMED and AMVMED choose the exponential distribution $p_0(\gamma_t) = ce^{-c(1-\gamma_t)}$ as the margin prior which leads to a logarithmic potential term. In this paper, we use the Gaussian distribution $p_0(\gamma_t) = q_0(\gamma_t) = \frac{c}{\sqrt{2\pi}}e^{-\frac{c^2}{2}(1-\gamma_t)^2}$ as margin priors since the corresponding potential term makes the key optimization problem of SMVMED a quadratic programming problem, which can be solved efficiently.

4 Sequential Minimal Optimization

In this section, we propose an SMO algorithm to solve the convex optimization problem (24) which plays a central role in SMVMED. We will also give the solution to $\hat{b}_1$ and $\hat{b}_2$ in (27), (28) and (29). SMO solves (24) by successively performing direction search on a small subset of dataset called working set [Bottou and Lin, 2007]. With respect to the working set selection scheme, we use the second order working set selection scheme [Fan *et al.*, 2005].

First, we rewrite the objective function in (24) as

$$D(\boldsymbol{\lambda}^{(\boldsymbol{k})}) = \sum_{t=1}^{N} \left(1 - \frac{\sum_{i=1}^{k-1} \alpha^{k-i} \lambda_t^{(i)}}{c^2} \right) \lambda_t^{(k)} - \sum_{t=1}^{N} \frac{\left(\lambda_t^{(k)}\right)^2}{2c^2} \\ -\tfrac{1}{2} \left(\sum_{t,\tau=1}^{N} \lambda_t^{(k)} \lambda_\tau^{(k)} y_t y_\tau K_{t\tau}^l \right) \tag{30}$$

where we omit the term irrelevant to $\boldsymbol{\lambda}^{(\boldsymbol{k})}$ and replace ${X_t^l}^{\mathrm{T}} X_\tau^l$ with $K_{t\tau}^l$ so that nonlinear classifiers can be taken into account. Then we write the constraints on $\lambda_t^{(k)}$ as constraints on $y_t \lambda_t^{(k)}$, that is,

$$y_t \lambda_t^{(k)} \in [\mathrm{A_t},\, \mathrm{B_t}] = \begin{cases} [0,\, +\infty]\,, y_t = 1 \\ [-\infty,\, 0]\,, y_t = -1. \end{cases} \tag{31}$$

We give the optimality criterion next. Let $\boldsymbol{\lambda}^* = \{\lambda_t^*\}$, $t = 1, \ldots, N$, be the solution to (31). $\boldsymbol{\lambda}^*$ should satisfy the optimization constraints. Let $\boldsymbol{g}^* = \{g_t^*\}$, $t = 1, \ldots, N$, be the derivatives of (30) with respect to $\boldsymbol{\lambda}^*$.

$$g_t^* = \frac{\partial D(\boldsymbol{\lambda}^*)}{\partial(\boldsymbol{\lambda}_t^*)} = 1 - \frac{\sum_{i=1}^{k-1} \alpha^{k-i} \lambda_t^{(i)}}{c^2} - y_t \sum_{\tau=1}^{N} y_\tau \lambda_\tau^* G_{t\tau}^l, \tag{32}$$

where

$$G_{t\tau}^l = \begin{cases} K_{t\tau}^l, & t \neq \tau \\ K_{t\tau}^l + \frac{1}{c^2}, & t = \tau. \end{cases} \tag{33}$$

Define $\boldsymbol{\lambda}^\epsilon = \{\lambda_t^\epsilon\}$, $t = 1, \ldots, N$, as

$$\lambda_u^\epsilon = \lambda_u^* + \begin{cases} +\epsilon y_u, u = t \\ -\epsilon y_u, u = \tau \\ 0, \quad \text{otherwise}, \end{cases} \tag{34}$$

where (t, τ) is a pair of subscripts such that $y_t \lambda_t^* < \mathrm{B_t}$ and $y_\tau \lambda_\tau^* > \mathrm{A}_\tau$, and ϵ is a small positive quantity. Since $\boldsymbol{\lambda}^*$ is the solution to (31), the first order expansion

$$D(\boldsymbol{\lambda}^\epsilon) - D(\boldsymbol{\lambda}^*) = \epsilon\,(y_t g_t^* - y_\tau g_\tau^*) + o(\epsilon) \leq 0 \tag{35}$$

suggests that $y_t g_t^* - y_\tau g_\tau^*$ is necessarily negative. This hold for all pairs (t, τ) such that $y_t \lambda_t^* < \mathrm{B_t}$ and $y_\tau \lambda_\tau^* > \mathrm{A}_\tau$. Therefore we can write the following optimality criterion

$$\exists \rho \in \mathbb{R}, \quad \max_{t \in \{t | y_t \lambda_t^* < \mathrm{B_t}\}} y_t g_t^* \leq \rho \leq \min_{\tau \in \{\tau | y_\tau \lambda_\tau^* > \mathrm{A}_\tau\}} y_\tau g_\tau^*. \tag{36}$$

To start with, we initialize $\lambda_t = 0$ and $g_t = 1 - \frac{\sum_{i=1}^{k-1} \alpha^{k-i} \lambda_t^{(i)}}{c^2}$, $t = 1, \ldots, N$. Then we successively choose a pair of subscripts (t, τ) as the working set and perform direction search on it until the optimality criterion (36) is satisfied. The second order working set selection scheme can be formulated as

$$\begin{aligned} t = & \operatorname*{argmax}_{u \in \{u | y_u \lambda_u < \mathrm{B_u}\}} y_u g_u, \\ \tau = & \operatorname*{argmax}_{u \in \{u | y_u \lambda_u > \mathrm{A_u}\}} \frac{(y_t g_t - y_u g_u)^2}{2\left(G_{tt}^l + G_{uu}^l - 2G_{tu}^l\right)} \\ & \text{s.t.} \quad y_t g_t > y_u g_u. \end{aligned} \tag{37}$$

For each working set, we maximize (30) by performing direction search along a direction $\boldsymbol{v}$ containing only two non-zero coefficients: $v_t = y_t, v_\tau = y_\tau$. This direction search is expressed by the following optimization problem

$$\eta = \operatorname*{argmax}_{\eta \in \{\eta | \eta \geq 0\}} D\left(\boldsymbol{\lambda} + \eta \boldsymbol{v}\right), \tag{38}$$

where $\boldsymbol{\lambda}$ is the starting point. The solution to (38) is

$$\eta = \min \left\{ \mathrm{B_t} - y_t g_t, y_\tau g_\tau - \mathrm{A}_\tau, \frac{(y_t g_t - y_\tau g_\tau)}{2\left(G_{tt}^l + G_{\tau\tau}^l - 2G_{t\tau}^l\right)} \right\}. \tag{39}$$

Gradients and coefficients are updated as follows

$$\forall u \in \{1, \ldots, N\},\; g_u = g_u - \eta y_u G_{tu}^l + \eta y_u G_{\tau u}^l, \tag{40}$$

$$\lambda_t = \lambda_t + y_t \eta, \tag{41}$$

$$\lambda_\tau = \lambda_\tau + y_\tau \eta. \tag{42}$$

Repeat the above procedure until optimality criterion (36) is satisfied, and we will obtain solution to (24). With respect to $\hat{b}_l$ in the prediction rules, we set $\hat{b}_l = \rho$.

5 Experiment

In this section, we evaluate SMVMED on real-world datasets: Course, Ads and Indoor. We compare soft margin consistency and hard margin consistency by performing SMVMED, MVMED and AMVMED on Course and Ads. Since Indoor is a large dataset that can not be handled by MVMED and AMVMED, we compare SMVMED with SVM-2K [Farquhar *et al.*, 2005] on it to show the scalability. For prediction functions, besides using two views $\mathrm{sign}(f_1)$ and $\mathrm{sign}(f_2)$ separately, the hybrid prediction function is also taken into consideration. That is, we also consider $\mathrm{sign}(\frac{1}{2}(f_1 + f_2))$ for SMVMED, MVMED and SVM-2K and $\mathrm{sign}(\rho f_1 + (1 - \rho) f_2))$ for AMVMED, where ρ is chosen from $\{0, 0.1, \ldots, 1.0\}$. Among all the three prediction functions , the one having the highest validation accuracy will be selected. The linear kernel is used in all the experiments. All of the experiments are executed on an Intel(R) Core(TM) i7-3667U 2.00GHz CPU with 8GB of RAM using Matlab R2014a.

5.1 Soft Margin Consistency vs. Hard Margin Consistency

We evaluate the performance of our soft margin based multiview MED by comparing it with MVMED and AMVMED on Course and Ads. Note that the original implementations of MVMED and AMVMED use the exponential prior. Here we also adapt them with the Gaussian prior to facilitate comparisons with SMVMED. We give a description of each used dataset below.

- Course: The dataset is the web-page dataset used in the co-training experiment [Blum and Mitchell, 1998]. It is a subset of the WebKB dataset which contains web pages collected by World Wide Knowledge Base (Web -> Kb) project of the CMU text learning group from computer science departments of four universities. It contains 230

Data	MVMED (exp)	MVMED (Gaussian)	AMVMED (exp)	AMVMED (Gaussian)	SMVMED
Course	93.61 ± 0.72	94.26 ± 0.77	93.80 ± 1.13	93.95 ± 1.13	$\mathbf{94.36} \pm 1.08$
Ads	94.73 ± 1.94	95.07 ± 1.67	95.33 ± 1.75	95.27 ± 1.92	$\mathbf{96.40} \pm 1.55$

Table 1: Average accuracies and standard deviations for Course and Ads

Data	MVMED (exp)	MVMED (Gaussian)	AMVMED (exp)	AMVMED (Gaussian)	SMVMED
Course	$262.20s$	$9.80s$	$203.16s$	$9.79s$	$\mathbf{1.41}s$
Ads	$78.93s$	$2.65s$	$62.92s$	$1.38s$	$\mathbf{0.88}s$

Table 2: Average training times for Course and Ads

course pages and 821 non-course pages, with view 1 being words in a web page and view 2 being words in the hyperlinks to the page. The dimensions of the two views are 500 and 82, respectively.

- Ads: The dataset consists of 459 ads images and 2820 non-ads images [Kushmerick, 1999]. We randomly select 600 examples as the used dataset. View 1 contains 587 features describing the image itself. The other 967 features form view 2.

We randomly select half of the dataset as the training set, and the rest is divided into the validation set and the test set equally. Parameter c in SMVMED, MVMED and AMVMED is independently chosen from $\{2^1, 2^2, \ldots, 2^5\}$ for Course, and from $\{2^1, 2^2, \ldots, 2^{15}\}$ for Ads. Parameter α in SMVMED is chosen from $\{0, 0.1, \ldots, 1.0\}$. All the experiments are performed ten times.

The average accuracies and standard deviations in percentage for Course and Ads are shown in Table 1, and the corresponding average training times with validated parameters are shown in Table 2. From Table 1, we can see that SMVMED performs better than MVMED and AMVMED. Table 2 shows that SMVMED and the Gaussian prior MVMED and AMVMED train much faster than the exponential prior MVMED and AMVMED, and that SMVMED is the fastest.

5.2 Scalability

We examine the scalability of SMVMED on a large dataset Indoor and compare it with SVM-2K. The dataset is the UJI-IndoorLoc indoor localization database [Joaquın *et al.*, 2014]. The training set consists of 19936 examples including 5249 Wifi fingerprints collected from building 0, 5195 from building 1 and 10444 form building 2. The test set consists of 536 Wifi fingerprints collected from building 0, 306 from building 1 and 268 from building 2. We perform experiments on both the whole dataset and a subset containing only examples from building 0 and 1. For the whole dataset experiment, we use examples from building 0 and 1 as the positive class and the others as the negative class. Each Wifi fingerprint is characterized by the detected wireless access points and the corresponding received signal strength intensity. We divide 520 intensity values into two views, the dimensions of which are 300 and 220, respectively. Each intensity value is randomly assigned to one of the two views. Since the performance of SMVMED is not sensitive to parameter c on the dataset, we fix it to 1. Parameter α is chosen from $\{0, 0.1, \ldots, 1.0\}$.

	Accuracy		Time	
	SVM-2K	SMVMED	SVM-2K	SMVMED
Subset	99.88	**100**	$876.51s$	$\mathbf{203.13}s$
Indoor	99.55	**100**	$3113.19s$	$\mathbf{448.69}s$

Table 3: Accuracies and training times for Indoor and its subset

Table 3 shows the accuracies for Indoor and its subset, and the corresponding training times. From Table 3, we can find that SMVMED performs better than SVM-2K and is much faster. Also, with the increase of the dataset size, the trainng time of SMVMED grows more slowly than that of SVM-2K.

The dataset used here includes 19936 training examples. However, SMVMED is scalable for larger datasets. From (37), (39) and (40), we can see that only the diagonal and two rows of G^l are required at the same time in SMO. Therefore, once these elements of G^l of a dataset can be cached, it can be handled by SMVMED.

5.3 Summary

The above experimental results show that SMVMED performs better than MVMED and AMVMED, and is much faster. The choice of the Gaussian margin prior greatly improves the training speed. On the large dataset Indoor, SMVMED performs better than SVM-2K and trains much faster.

6 Conclusion

We have presented soft margin consistency based scalable MED, with an SMO algorithm for efficient training. Different from the hard margin consistency used in MVMED and AMVMED, a different principle of less strict soft margin consistency is employed. By balancing large margin and margin consistency, SMVMED is more flexible. Furthermore, the use of the Gaussian prior and SMO makes SMVMED very efficient. Experimental results on multiple real-world datasets have shown its good performance and scalability.

Acknowledgments

The corresponding author Shiliang Sun would like to thank supports from the National Natural Science Foundation of China under Project 61370175.

References

[Blum and Mitchell, 1998] Avrim Blum and Tom M. Mitchell. Combining labeled and unlabeled data with co-training. In *Proceedings of the 11th Annual Conference on Computational Learning Theory*, pages 92–100, 1998.

[Bottou and Lin, 2007] Léon Bottou and Chih-Jen Lin. Support vector machine solvers. In *Large Scale Kernel Machines*, pages 1–28. Cambridge, MA., 2007.

[Chang *et al.*, 2013] Xu Chang, Tao Dacheng, and Xu Chao. A survey on multi-view learning. *arXiv preprint:1304.5634*, 2013.

[Chao and Sun, 2015] Guoqing Chao and Shiling Sun. Alternative multiview maximum entropy discrimination. *IEEE Transactions on Neural Networks and Learning Systems*, 99:1–12, 2015.

[Fan *et al.*, 2005] Rong-En Fan, Pai-Hsuen Chen, and Chih-Jen Lin. Working set selection using second order information for training support vector machines. *Journal of Machine Learning Research*, 6:1889–1918, 2005.

[Farquhar *et al.*, 2005] Jason D. R. Farquhar, David R. Hardoon, Hongying Meng, John Shawe-Taylor, and Sándor Szedmák. Two view learning: SVM-2K, theory and practice. *Advances in Neural Information Processing Systems*, 18:355–362, 2005.

[Jaakkola *et al.*, 2000] Tommi Jaakkola, Marina Meila, and Tony Jebara. Maximum entropy discrimination. *Advances in Neural Information Processing Systems*, 12:470–476, 2000.

[Jebara and Jaakkola, 2000] Tony Jebara and Tommi Jaakkola. Feature selection and dualities in maximum entropy discrimination. In *Proceedings of the 16th Conference in Uncertainty in Artificial Intelligence*, pages 291–300, 2000.

[Jebara, 2004] Tony Jebara. *Machine Learning: Discriminative and Generative*. Kluwer Academic, 2004.

[Jebara, 2011] Tony Jebara. Multitask sparsity via maximum entropy discrimination. *Journal of Machine Learning Research*, 12:75–110, 2011.

[Joaquın *et al.*, 2014] Torres-Sospedra Joaquın, Montoliu Raúl, Martınez-Usó Adolfo, Avariento Joan P, Arnau Tomás J, Mauri Benedito-Bordonau, and Huerta Joaquın. Ujiindoorloc: A new multi-building and multi-floor database for wlan fingerprint-based indoor localization problems. In *Proceedings of the 5th International Conference on Indoor Positioning and Indoor Navigation*, pages 261–270, 2014.

[Kushmerick, 1999] Nicholas Kushmerick. Learning to remove internet advertisements. In *Proceedings of the 3rd Annual Conference on Autonomous Agents*, pages 175–181. ACM, 1999.

[Platt, 1999] John Platt. Fast training of support vector machines using sequential minimal optimization. *Advances in Kernel Methods – Support Vector Learning*, 3, 1999.

[Sun and Chao, 2013] Shiliang Sun and Guoqing Chao. Multi-view maximum entropy discrimination. In *Proceedings of the 23rd International Joint Conference on Artificial Intelligence*, pages 1706–1712, 2013.

[Sun, 2013] Shiliang Sun. A survey of multi-view machine learning. *Neural Computing and Applications*, 23(7-8):2031–2038, 2013.

[Zhu and Xing, 2009] Jun Zhu and Eric P. Xing. Maximum entropy discrimination markov networks. *Journal of Machine Learning Research*, 10:2531–2569, 2009.

[Zhu *et al.*, 2008a] Jun Zhu, Eric P. Xing, and Bo Zhang. Laplace maximum margin markov networks. In *Proceedings of the 25th International Conference on Machine Learning*, pages 1256–1263, 2008.

[Zhu *et al.*, 2008b] Jun Zhu, Eric P. Xing, and Bo Zhang. Partially observed maximum entropy discrimination markov networks. *Advances in Neural Information Processing Systems*, 21:1977–1984, 2008.