Joint SVM for Accurate and Fast Image Tagging

Hanchen Xiong, Sandor Szedmak and Justus Piater

IIS, University of Innsbruck

ESANN 04/2014

Outline

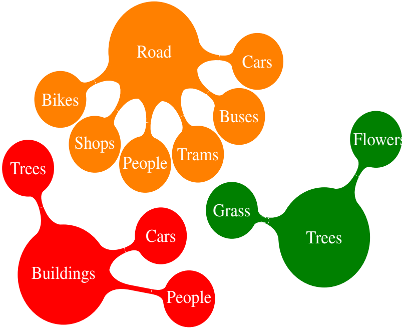
1. Problem statement
2. Learning strategy
3. Joint SVM
4. Optimization model
5. Experiment

Problem statement

- **Image tagging**:
  - Assign **context** relating **labels** to images
- The **labels** are mainly **binary**
  - **True** or **False** values of a set **propositions** applied on the those images:
    - ★ *“Is a tree on the image?”*
    - ★ *“Are there some buildings on the image?”*
    - ★ *“Is a see on the image?”*
    - ★ . . .
  - ⇒ *Yes*, *Yes*, *No*, . . .
- **Task:**
  - Given an **image predict** the **labels**!

How to exploit the interdependency of the labels

- Might labels imply other lables:
- **Graph** of **label nodes**, **implications** are **arcs**.
  - label ⇒ label
    - ★ “Buildings” ⇒ “People”
    - ★ “Trees” ⇒ “Grass”
    - ★ “Cars” ⇒ “Road”
    - ★ “Cars” ⇒ “Road signs”
- Higher order relations, hypergraph
  - label set ⇒ label set
    - ★ “Cars”,“People” ⇒ “Road”,“Buildings”
- Undirected, mutual implications.



Examples of known approaches to interdependency

- Ignore interactions ...
  - Apply plenty of binary classifiers, e.g. SVMs ...
- Max-Margin Markov Networks,
  - Taskar(2003)
  - Tsochantaridis(2005)
- Least-square approaches,
  - Cortes(2005)
- See collection of approaches in
  - Bakir(2007)
- Recurrent neural network,
  - Gomez(2008)

- Main problems in addressing the interaction are
  - **the very high computational complexity**,
  - **not too much satisfactory results...**.

A possible learning strategy

Attacking the problem via properly chosen feature representation.

Embedding where the structures of the input and output objects are represented in properly chosen spaces(e.g. Hilbert, ...).

Optimization has to find the similarity based matching between the input and the output representations.

Inversion(Pre-image problem) has to recover the best fitting output structure of its representation.

Embedding

Embedding

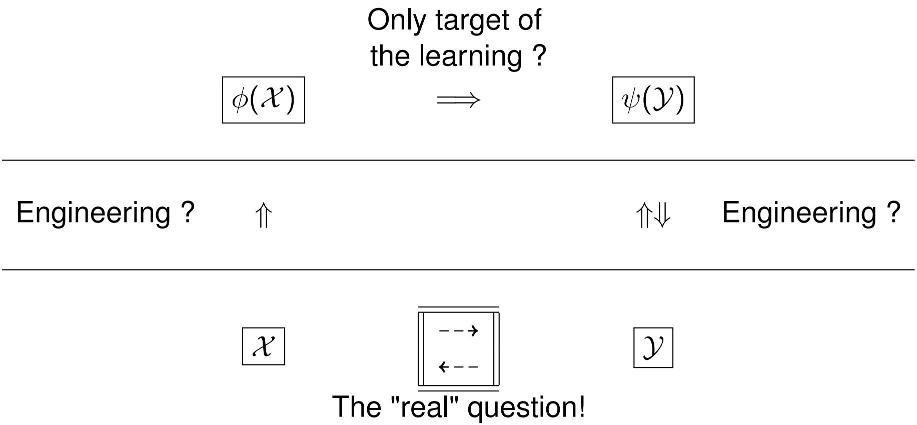
$$\phi : \overbrace{\text{input space}}^{\mathcal{X}} \rightarrow \overbrace{\text{feature space}}^{\mathcal{H}_\phi}$$

$$\psi : \overbrace{\text{output space}}^{\mathcal{Y}} \rightarrow \overbrace{\text{label space}}^{\mathcal{H}_\psi}$$

Similarity transformation $\widetilde{\mathbf{W}} = (\mathbf{W}, \mathbf{b}) \Rightarrow \psi(\mathbf{y}) \sim \widetilde{\mathbf{W}}\phi(\mathbf{x})$

Inversion $\psi^{-1}(\mathbf{Y})$

Learning as task to find “Probably, approximately isomorph” relations



Only target of the learning ?

Engineering ?

Engineering ?

The "real" question!

Joint SVM

$$\begin{array}{ll}
\min & \frac{1}{2}\sum_{t}^{T}\|\mathbf{w}_t\|^2 \\
& +C\sum_{t=1}^{T}\sum_{i=1}^{m}\xi_t^{(i)} \\
\hline
\text{w.r.t.} & \forall\ t \Rightarrow \mathbf{w}_t \in \mathbb{R}^{\mathcal{H}_\phi}, \\
& b_t \in \mathbb{R}, \\
& \forall\ t, i \Rightarrow \xi_t^{(i)} \in \mathbb{R}, \\
\hline
\text{s.t.} & \sum_{t=1}^{T}\left(y^{(i)}\mathbf{w}'_t\phi(\mathbf{x}^{(i)}) + b_t\right) \\
& \qquad \geq T - \sum_{t=1}^{T}\xi_t^{(i)}, \\
& \xi_t^{(i)} \geq 0,\ t = 1, \ldots, T, \\
& i = 1, \ldots, m
\end{array}$$

$$\begin{array}{ll}
\min & \frac{1}{2}\|\mathbf{W}\|_F^2 \\
& +C\sum_{i=1}^{m}\bar{\xi}^{(i)} \\
\hline
\text{w.r.t.} & \mathbf{W} \in \mathbb{R}^{T\times\mathcal{H}_\phi}, \\
& \mathbf{b} \in \mathbb{R}^T, \\
& \hat{\boldsymbol{\xi}} \in \mathbb{R}^m, \\
\hline
\text{s.t.} & \langle\mathbf{y}^{(i)}, \mathbf{W}\phi(\mathbf{x}^{(i)}) + \mathbf{b}\rangle \\
& \qquad \geq 1 - \bar{\xi}^{(i)}, \\
& \bar{\xi}^{(i)} \geq 0, \\
& \qquad i = 1, \ldots, m
\end{array}$$

$$\begin{array}{llll}
\mathbf{y}_i & = \left[y_1^{(i)}, \ldots, y_T^{(i)}\right], & \mathbf{b} & = \left[\frac{b_1}{T}, \ldots, \frac{b_T}{T}\right], \\
\mathbf{W} & = \left[\frac{w'_1}{T}, \ldots, \frac{w'_T}{T}\right], & \bar{\xi}^{(i)} & = \frac{1}{T}\sum_{t=1}^{T}\xi_t^{(i)}.
\end{array}$$

Dual problem

$$\begin{array}{ll}
\text{min} & \sum_{i,j=1}^{m} \alpha_i \alpha_j \overbrace{\langle \psi(\mathbf{y}^{(i)}), \psi(\mathbf{y}^{(j)}) \rangle}^{\kappa_{ij}^{\psi}} \overbrace{\langle \phi(\mathbf{x}^{(i)}), \phi(\mathbf{x}^{(j)}) \rangle}^{\kappa_{ij}^{\phi}} - \sum_{i=1}^{m} \alpha_i, \\
\text{w.r.t.} & \forall i \rightarrow \alpha_i \in \mathbb{R}, \\
\text{s.t.} & \sum_{i=1}^{m} \alpha_i \psi(\mathbf{y}^{(i)}) = \mathbf{0}, \ \text{## only if bias is used} \\
 & 0 \leq \alpha_i \leq C, \ i = 1, \dots, m.
\end{array}$$

- κ_{ij}^{ϕ} **input kernel**,
- κ_{ij}^{ψ} **output kernel**,
- The objective function is a symmetric function of the input and the output.

Reinterpretation of the normal vector **w**

Original
- $y_i \in \{-1, +1\}$ binary outputs
- **w** is the normal vector of the separating hyperplane.

New
- $y_i \in \mathcal{Y}$ arbitrary outputs
  - $\psi(y_i) \in \mathcal{H}_\psi$ embedded labels in a linear vector space
- **w**$'$ is a linear operator projecting the input space into the output space.
  - The aim to find the highest similarity between the output and the projected input.

The output space is a one dimensional subspace in the SVM.

Primal problems

	Binary class learning	Vector label learning
	Support Vector Machine(SVM)	Maximum Margin Robot(MMR)
min	$\frac{1}{2}\underbrace{\boxed{\mathbf{w}'\mathbf{w}}}_{\lVert\mathbf{w}\rVert_2^2}+C\mathbf{1}'\boldsymbol{\xi}$	$\frac{1}{2}\underbrace{\boxed{\mathbf{tr}(\mathbf{W}'\mathbf{W})}}_{\lVert\mathbf{W}\rVert_{Frobenius}^2}+C\mathbf{1}'\boldsymbol{\xi}$
w.r.t.	$\boxed{\mathbf{w}:\mathcal{H}_\phi\to\mathbb{R}}$, normal vec. $\boxed{b\in\mathbb{R}}$, bias $\boldsymbol{\xi}\in\mathbb{R}^m$, error vector	$\boxed{\mathbf{W}:\mathcal{H}_\phi\to\mathcal{H}_\psi}$, linear operator $\boxed{\mathbf{b}\in\mathcal{H}_\psi}$, translation(bias) $\boldsymbol{\xi}\in\mathbb{R}^m$, error vector
s.t.	$\boxed{y_i(\mathbf{w}'\phi(\mathbf{x}_i)+b)}\geq 1-\xi_i$ $\boldsymbol{\xi}\geq\mathbf{0},\ i=1,\dots,m$	$\boxed{\langle\psi(\mathbf{y}_i),\mathbf{W}\phi(\mathbf{x}_i)+\mathbf{b}\rangle_{\mathcal{H}_\psi}}\geq 1-\xi_i$ $\boldsymbol{\xi}\geq\mathbf{0},\ i=1,\dots,m$

Prediction

No bias

The linear operator:

$$\mathbf{W} = \sum_{i=1}^{m} \alpha_i \psi(\mathbf{y}^{(i)}) \phi(\mathbf{x}^{(i)})'$$

Prediction in the label space:

$$\begin{aligned} \psi(\mathbf{y}) &= \mathbf{W}\phi(\mathbf{x}) \\ &= \sum_{i=1}^{m} \alpha_i \psi(\mathbf{y}^{(i)}) \boxed{\underbrace{\langle \phi(\mathbf{x}^{(i)}), \phi(\mathbf{x}) \rangle}_{\kappa^{\phi}(\mathbf{x}^{(i)},\mathbf{x})}} \end{aligned}$$

Prediction when the labels are implicit

An approach

Assume the set of outcomes is known

$$\mathbf{y} \in \widetilde{\mathcal{Y}} \quad \Leftarrow \text{Set of the possible outputs}$$

$$\mathbf{y}^* = \arg\max_{\mathbf{y}\in\widetilde{\mathcal{Y}}} \psi(\mathbf{y})'\mathbf{W}\phi(\mathbf{x})$$

$$= \arg\max_{\mathbf{y}\in\widetilde{\mathcal{Y}}} \sum_{i=1}^{m} \alpha_i \overbrace{\langle\psi(\mathbf{y}),\psi(\mathbf{y}^{(i)})\rangle}^{\kappa^{\psi}(\mathbf{y},\mathbf{y}^{(i)})} \overbrace{\langle\phi(\mathbf{x}^{(i)}),\phi(\mathbf{x})\rangle}^{\kappa^{\phi}(\mathbf{x}^{(i)},\mathbf{x})}$$

an example, Gaussian feature mapping of the labels:

$$= \arg\max_{\mathbf{y}\in\widetilde{\mathcal{Y}}} \sum_{i=1}^{m} \alpha_i \kappa^{\phi}(\mathbf{x}^{(i)},\mathbf{x}) \exp(-\tfrac{1}{2}(\mathbf{y}^{(i)}-\mathbf{y}^{(j)})'\Sigma^{-1}(\mathbf{y}^{(i)}-\mathbf{y}^{(j)}))$$

$$\widetilde{\mathcal{Y}} = \{\mathbf{y}^{(1)},\ldots,\mathbf{y}^{(K)}\} \subset \mathbb{B}^{T}$$

Experiment

Dataset

Dataset:	Corel5K benchmark
Input(image) features:	INRIA features, LEAR project
Output features:	260 binary labels, (average 3.5 per image)
Training items:	$N = 4500$ images
Test items:	$N = 500$ images

Experiment

Results

	Training	Testing	Testing Accuracy		
	Time (sec)		Precision	Recall	F1
Joint SVM (Gaussian)	81	**7**	**0.408**	0.371	**0.389**
Joint SVM (Polynomial)	**76**	9	0.391	0.357	0.373
Independent SVMs	6285	317	0.105	0.123	0.113
Best result of Makadia et al. (2010)	-	-	0.27	0.32	0.29
Recent result of Chen et al. (2013)	-	-	0.32	**0.43**	0.37

Figure : Comparison between joint SVM and independent SVMs.

Epilogue

“Young man,

in mathematics you don’t understand things.
You just get used to them.”

John von Neumann, one of the greatest mathematician of the Twenty Century.

This is the End

Thanks!