Structured Prediction

TECHNISCHE UNIVERSITÄT DARMSTADT

Advanced topics in Machine Learning:
Structured Prediction

TU Darmstadt
June 24, 2015
Oleg Arenz

Overview

TECHNISCHE UNIVERSITÄT DARMSTADT

Today we will look at:

1. Introduction to Structured Prediction
   - Problem Statement
   - Challenges
2. Some Background
   - Conditional Random Fields (CRFs)
   - Maximum Margin Markov Networks (M^3N)
3. A recent approach
   - Efficient Max-Margin Learning using Dual Decompositions

What is Structured Prediction?

TECHNISCHE UNIVERSITÄT DARMSTADT

- A special case of (multivariate) Regression/Classification, i.e. given $\mathcal{D} = \{(x^{(i)}, y^{(i)})\}_i$ we want to learn

$$f : \mathbf{X} \rightarrow \mathbf{Y}$$

- For Structured Prediction each **y** is a *structure*

Examples of Structures

POS-Tagging:

Y:	PRN	VBP	CD	JJ	NNS	PRT	VB	DT	NN
X:	I	know	387420489	different	ways	to	tag	this	sentence.

Semantic Parsing:

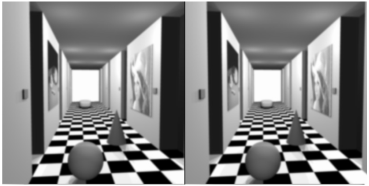
X: How many people live in Darmstadt?

```
Y: SELECT population
     FROM cities
     WHERE STRCMP(name, 'Darmstadt')
```

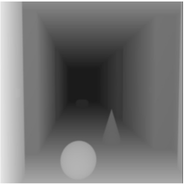
Examples of Structures

Stereo Matching:

X:



Y:



Challenges

TECHNISCHE UNIVERSITÄT DARMSTADT

POS-Tagging:

Y: PRN VBP CD JJ NNS PRT VB DT NN
X: I know 387420489 different ways to tag this sentence.

Two naive ways of applying standard classification:

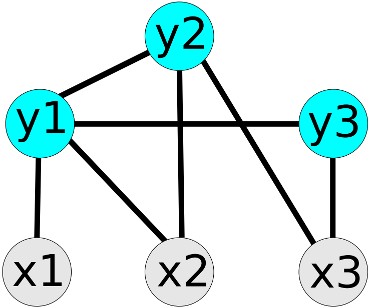
1. learn a separate classifier for each part (word)
    - bad performance
    - fails to exploit the dependences
2. define one label for each possible structure
    - infeasible, too many labels
    - fails to exploit the independences

Structured Prediction is all about *exploiting* the structure!

Conditional Random Fields

TECHNISCHE
UNIVERSITÄT
DARMSTADT

- We need to model the (in)dependences in order to exploit them
- e.g. by using graphical models like CRFs:



- undirected graph
- discriminative method
- models $p(\mathbf{y}|\mathbf{x})$, disregards $p(\mathbf{x})$
- Prediction by Inference

$$p(\mathbf{y}|\mathbf{x}) = \frac{\exp\left(\sum_c \theta_c^\top \phi_c(\mathbf{x}, \mathbf{y})\right)}{\sum_{\mathbf{y}} \exp\left(\sum_c \theta_c^\top \phi_c(\mathbf{x}, \mathbf{y})\right)}$$

Conditional Random Fields
Training

CRFs can be trained by maximizing the conditional log likelihood

$$\mathcal{L} = \sum_N \sum_c \theta_c^\top \phi_c(\mathbf{x_c}, \mathbf{y_c}) - \log Z(\mathbf{x})$$

using the gradient

$$\frac{\partial \mathcal{L}}{\partial \theta_c} = \sum_{i=1}^{N} \phi_c(\mathbf{x_c}^{(i)}, \mathbf{y_c}^{(i)}) - \sum_{i=1}^{N} \sum_{\mathbf{y_c}} p(\mathbf{y_c} | \mathbf{x_c^{(i)}}) \phi_c(\mathbf{x_c}^{(i)}, \mathbf{y_c})$$

"empirical feature counts - expected feature counts"

$\Rightarrow$ Inference needed in the training loop

Conditional Random Fields
Inference

TECHNISCHE UNIVERSITÄT DARMSTADT

- We need $p(\mathbf{y_c}|\mathbf{x})$ for the gradient and $f(\mathbf{x}) = \arg\max_{y} p(\mathbf{y}|\mathbf{x})$ for prediction
- Inference employs general algorithms for graphical models
- Exact solutions (based on DP) only possible for simple models (e.g. trees with small tree width)
  - Belief Propagation
  - Forward Backward
  - Viterbi
  - Junction Tree
- Otherwise approximations are required
  - Loopy Belief Propagation
  - Mean Field Inference
  - Alpha expansion

Maximum Margin Markov Networks
Primal Problem

TECHNISCHE UNIVERSITÄT DARMSTADT

- We are actually not so much interested in getting $p(\mathbf{y}|\mathbf{x})$ right.
- We want to get $f(\mathbf{x}) = \arg\max_{y} p(\mathbf{y}|\mathbf{x})$ right!

$\Rightarrow$ Instead of learning θ_{ML}, M^3N learns θ_{MM} (MM = Maximum Margin)

$$\begin{aligned} \min_{\theta} \quad & \frac{1}{2}||\theta||^2 + C \sum_{\mathbf{x} \in \mathcal{D}} \xi_{\mathbf{x}} \\ \text{s.t.} \quad & \forall_{\mathbf{x} \in \mathcal{D}, \mathbf{y}} \; \theta^\top \Delta \phi_{\mathbf{x}}(\mathbf{y}) \geq \Delta t_{\mathbf{x}}(\mathbf{y}) - \xi_{\mathbf{x}} \\ & \forall_{\mathbf{x} \in \mathcal{D}} \; \xi_{\mathbf{x}} \geq 0 \end{aligned}$$

Here $\Delta t_{\mathbf{x}}(\mathbf{y}) = \sum_i \mathbb{1}_{y_i \neq t(\mathbf{x})_i}$ defines the per-Label loss.

Maximum Margin Markov Networks
Dual Problem

TECHNISCHE UNIVERSITÄT DARMSTADT

$$\max_{\alpha_{\mathbf{x}}(\mathbf{y})} \quad \sum_{\mathbf{x},\mathbf{y}} \alpha_{\mathbf{x}}(\mathbf{y}) \Delta t_{\mathbf{x}}(\mathbf{y}) - \frac{1}{2} || \sum_{\mathbf{x},\mathbf{y}} \alpha_{\mathbf{x}}(\mathbf{y}) \Delta \phi_{\mathbf{x}}(\mathbf{y}) ||^2$$

$$\text{s.t.} \quad \forall_{\mathbf{x}} \sum_{\mathbf{y}} \alpha_{\mathbf{x}}(\mathbf{y}) = C$$

$$\forall_{\mathbf{x},\mathbf{y}} \; \alpha_{\mathbf{x}}(\mathbf{y}) \geq 0$$

- The good news: it's a quadratic program
- The bad news: one constraint for each possible output label

Maximum Margin Markov Networks
Dual Problem

TECHNISCHE UNIVERSITÄT DARMSTADT

$$\max_{\alpha_{\mathbf{x}}(\mathbf{y})} \quad \sum_{\mathbf{x},\mathbf{y}} \alpha_{\mathbf{x}}(\mathbf{y}) \Delta t_{\mathbf{x}}(\mathbf{y}) - \frac{1}{2} || \sum_{\mathbf{x},\mathbf{y}} \alpha_{\mathbf{x}}(\mathbf{y}) \Delta \phi_{\mathbf{x}}(\mathbf{y}) ||^2$$

$$\text{s.t.} \quad \forall_{\mathbf{x}} \sum_{\mathbf{y}} \alpha_{\mathbf{x}}(\mathbf{y}) = C$$

$$\forall_{\mathbf{x},\mathbf{y}} \; \alpha_{\mathbf{x}}(\mathbf{y}) \geq 0$$

- The good news: it's a quadratic program
- The bad news: one constraint for each possible output label

Two key insights:

1. $\alpha_{\mathbf{x}}(\mathbf{y})$ are unnormalized distributions over $\mathbf{y}$ (the constraints tell us)
2. $\alpha_{\mathbf{x}}(\mathbf{y})$ will factorize exactly like the potential functions (the objective tells us)

Maximum Margin Markov Networks
Key Insights

TECHNISCHE UNIVERSITÄT DARMSTADT

Two key insights:

1. $\alpha_{\mathbf{x}}(\mathbf{y})$ are unnormalized distributions over $\mathbf{y}$ (the constraints tell us)
2. $\alpha_{\mathbf{x}}(\mathbf{y})$ will factorize exactly like the potential functions (the objective tells us)

$\Rightarrow$ We don't need to know the α-distributions, but just their marginals:

E.g.: $\sum_{\mathbf{x},\mathbf{y}} \alpha_{\mathbf{x}}(\mathbf{y}) \Delta t_{\mathbf{x}}(\mathbf{y}) = \sum_{\mathbf{x},\mathbf{y}} \alpha_{\mathbf{x}}(\mathbf{y}) \sum_c \Delta t_{\mathbf{c},\mathbf{x_c}}(\mathbf{y_c}) = \sum_{\mathbf{x}} \sum_i \sum_{y_i} \mu_{\mathbf{x}}(y_i) \Delta t_{\mathbf{x}}(y_i)$

using the marginals $\mu_{\mathbf{x}}(\mathbf{y_c}) = \sum_{\mathbf{y} \sim [\mathbf{y_c}]} \alpha_{\mathbf{x}}(\mathbf{y})$

$\Rightarrow$ By solving directly for $\mu_{\mathbf{x_c}}(\mathbf{y_c})$ instead of $\alpha_{\mathbf{x}}(\mathbf{y})$ we need to solve for less variables.

Maximum Margin Markov Networks
Factored Dual

$$\max_{\mu_{\mathbf{x}}(\mathbf{y})} \quad \sum_{\mathbf{x}} \sum_{i,y_i} \mu_{\mathbf{x}}(y_i) \Delta t_{\mathbf{x}}(y_i) - \frac{1}{2} \sum_{\mathbf{x},\mathbf{x}'} \sum_{c,c'} \sum_{\mathbf{y_c}} \sum_{\mathbf{y}'_{\mathbf{c}'}} \mu_{\mathbf{x}}(\mathbf{y_c}) \mu_{\mathbf{x}'}(\mathbf{y}'_{\mathbf{c}'}) \Delta \phi_{\mathbf{x}}(\mathbf{y_c})^\top \Delta \phi_{\mathbf{x}'}(\mathbf{y}'_{\mathbf{c}'})$$

$$\text{s.t.} \quad \forall_{\mathbf{x}} \sum_{i,y_i} \mu_{\mathbf{x}}(y_i) = C$$

$$\forall_{c,\mathbf{x},\mathbf{y_c}} \; \mu_{\mathbf{x}}(\mathbf{y_c}) \geq 0$$

Maximum Margin Markov Networks
Factored Dual

TECHNISCHE UNIVERSITÄT DARMSTADT

$$\max_{\mu_{\mathbf{x}}(\mathbf{y})} \quad \sum_{\mathbf{x}} \sum_{i,y_i} \mu_{\mathbf{x}}(y_i) \Delta t_{\mathbf{x}}(y_i) - \frac{1}{2} \sum_{\mathbf{x},\mathbf{x}'} \sum_{c,c'} \sum_{\mathbf{y_c}} \sum_{\mathbf{y}'_{\mathbf{c}'}} \mu_{\mathbf{x}}(\mathbf{y_c}) \mu_{\mathbf{x}'}(\mathbf{y}'_{\mathbf{c}'}) \Delta\phi_{\mathbf{x}}(\mathbf{y_c})^\top \Delta\phi_{\mathbf{x}'}(\mathbf{y}'_{\mathbf{c}'})$$

$$\text{s.t.} \quad \forall_{\mathbf{x}} \sum_{i,y_i} \mu_{\mathbf{x}}(y_i) = C$$

$$\forall_{c,\mathbf{x},\mathbf{y_c}} \; \mu_{\mathbf{x}}(\mathbf{y_c}) \geq 0$$

Unfortunately it's not that simple. We still need to ensure that the marginals are consistent.

Assuming pairwise potentials, i.e. $\forall_c \mathbf{y}_c = [y_{c,1}, y_{c,2}]^\top$ and a tree, consistency can be enforced by adding constraints

$$\forall c \sum_{y_{c_1}} \mu_{\mathbf{x}}(y_{c_1}, y_{c_2}) = \mu_{\mathbf{x}}(y_{c_2}).$$

Maximum Margin Markov Networks
Learning θ

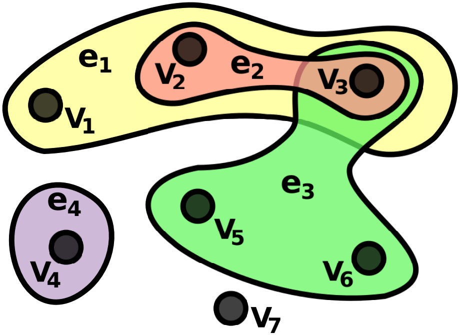
Once, the dual variables have been computed, solving the primal problem is easy:

$$\theta = \sum_{\mathbf{x}} \sum_{(i,j)} \sum_{y_i, y_j} \mu_{\mathbf{x}}(y_i, y_j) \Delta\phi(y_i, y_j)$$

Dual Decompositions for Max Margin Learning

Let's assume we don't want to restrict ourselves to trees with pairwise potentials.

$\Rightarrow$ Let's look at hyperedges instead of edges:



Dual Decompositions for Max Margin Learning
Potential Function

TECHNISCHE UNIVERSITÄT DARMSTADT

Let's define the potentials of our CRF/MRF on the hypergraph as follows:

- a unary potential for each node: $\mathbf{u} = \{u_p\}_{p \in \mathcal{V}}$
- a higher-order potential for each hyperedge: $\mathbf{h} = \{h_c\}_{c \in \mathcal{C}}$

We want the potentials to be parameterized based on their features:

$$u_p^k(y_p|\mathbf{w}) = \mathbf{w}^\top \phi(y_p, \mathbf{x}^k)$$
$$h_c^k(\mathbf{y}_c|\mathbf{w}) = \mathbf{w}^\top \phi_c(\mathbf{y}_c, \mathbf{x}^k)$$

We can then denote the potential of training input k for Hypergraph G as:

$$E_G(\mathbf{u}^k(\mathbf{y}^k|\mathbf{w}), \mathbf{h}^k(\mathbf{y}^k|\mathbf{w})) := \sum_{p \in \mathcal{V}_k} u_p^k(y_p^k) + \sum_{c \in \mathcal{C}_k} h_c^k(\mathbf{y}_c^k)$$

IAS

TECHNISCHE UNIVERSITÄT DARMSTADT

Recall our primal objective

$$\min_{\mathbf{w}} \quad R(\mathbf{w}) + C\sum_{k} \xi_k$$

$$\text{s.t.} \quad \forall_{\mathbf{y}}\ \xi_k \geq E_G(\mathbf{u}^k(\mathbf{y}^k|\mathbf{w}), \mathbf{h}^k(\mathbf{y}^k|\mathbf{w})) - \left(E_G(\mathbf{u}^k(\mathbf{y}|\mathbf{w}), \mathbf{h}^k(\mathbf{y}|\mathbf{w})) - \Delta(\mathbf{y}, \mathbf{y}^k)\right)$$

Since we penalize $\sum_k \xi_k$, the optimal values ξ_k^* satisfy

$$\xi_k^* = E_G(\mathbf{u}^k(\mathbf{y}^k|\mathbf{w}), \mathbf{h}^k(\mathbf{y}^k|\mathbf{w})) - \min\left(E_G(\mathbf{u}^k(\mathbf{y}|\mathbf{w}), \mathbf{h}^k(\mathbf{y}|\mathbf{w})) - \Delta(\mathbf{y}, \mathbf{y}^k)\right)$$

We will assume that the loss decomposes just like our potential.
Let $\bar{\mathbf{u}}^k(\mathbf{y}|\mathbf{w})$ and $\bar{\mathbf{h}}^k(\mathbf{y}|\mathbf{w})$ be the loss-augmented potentials. Then

$$\xi_k^* = E_G(\bar{\mathbf{u}}^k(\mathbf{y}^k|\mathbf{w}), \bar{\mathbf{h}}^k(\mathbf{y}^k|\mathbf{w})) - \min E_G(\bar{\mathbf{u}}^k(\mathbf{y}|\mathbf{w}), \bar{\mathbf{h}}^k(\mathbf{y}|\mathbf{w})) := L_G^k(\mathbf{w})$$

is a hinge loss.

Dual Decompositions for Max Margin Learning
Unconstrainted Problem

TECHNISCHE UNIVERSITÄT DARMSTADT

Substituting ξ_k^* in our primal objective we get an unconstrained optimization problem:

$$\min_{\mathbf{w}} R(\mathbf{w}) + C \sum_k L_G^k(\mathbf{w})$$

$\Rightarrow$ Max Margin Learning is regularized empirical loss minimization based on $L_G^k(\mathbf{w})$.

Unfortunately, evaluating $L_G^k(\mathbf{w})$ is NP-hard.

That's why we need dual decomposition.

Idea:

- Decompose G into smaller sub-Hypergraphs G_i
- Solve the slave problems ($\min_{\mathbf{y}} E_{G_1}, \min_{\mathbf{y}} E_{G_2}, \ldots$)
- Approximate the solution of the master problem ($\min_{\mathbf{y}} E_G$)

G should be decomposed such that

$$\mathcal{V} = \cup_i \mathcal{V}_i \qquad \text{and} \qquad \mathcal{C} = \cup_i \mathcal{C}_i.$$

G_i inherit higher-order potentials but have own unary potentials, i.e.

$$E_{G_i}(\mathbf{u}^i(\mathbf{y}|\mathbf{w}), \bar{\mathbf{h}}(\mathbf{y}|\mathbf{w})) := \sum_{p \in \mathcal{V}} u_p^i(y_p) + \sum_{c \in \mathcal{C}_k} \bar{h}_c(\mathbf{y}_c)$$

The unary potentials should satisfy

$$\sum_{i \in \mathcal{I}_p} u_p^i(y_p) = \bar{u}_p(y_p)$$

We then attain a lower bound on the master problem:

$$\sum_i \min_{\mathbf{y}} E_{G_i}(\mathbf{u}^i(\mathbf{y}), \bar{h}(\mathbf{y})) \leq \min_{\mathbf{y}} E_G(\mathbf{u}(\mathbf{y}), \bar{h}(\mathbf{y}))$$

Let's choose the unary potentials such that the bound is as tight as possible:

$$\text{DUAL}_{\{G_i\}}(\mathbf{u^k}, \mathbf{\bar{h}^k}) = \max_{\mathbf{u}^{k,i}{}_{1\leq i\leq N}} \sum_i \min_{\mathbf{y}} E_{G_i}(\mathbf{u}^{k,i}(\mathbf{y}), \bar{h}^k(\mathbf{y}))$$

$$\text{s.t. } \forall_{p\in\mathcal{V}} \sum_{i\in\mathcal{I}_p} u_p^i = \bar{u}_p$$

Dual Decompositions for Max Margin Learning
The Unconstrained Problem (Again)

TECHNISCHE UNIVERSITÄT DARMSTADT

$$\min_{\mathbf{w}} R(\mathbf{w}) + C \sum_k L_G^k(\mathbf{w})$$

$$L_g^k(\mathbf{w}) = E_G(\bar{\mathbf{u}}^k(\mathbf{y}^k|\mathbf{w}), \bar{\mathbf{h}}^k(\mathbf{y}^k|\mathbf{w})) - \min E_G(\bar{\mathbf{u}}^k(\mathbf{y}|\mathbf{w}), \bar{\mathbf{h}}^k(\mathbf{y}|\mathbf{w}))$$

Dual Decompositions for Max Margin Learning
The Unconstrained Problem (Again)

$$\min_{\mathbf{w}} R(\mathbf{w}) + C \sum_k L_G^k(\mathbf{w})$$

$$L_g^k(\mathbf{w}) = E_G(\bar{\mathbf{u}}^k(\mathbf{y}^k|\mathbf{w}), \bar{\mathbf{h}}^k(\mathbf{y}^k|\mathbf{w})) - \min E_G(\bar{\mathbf{u}}^k(\mathbf{y}|\mathbf{w}), \bar{\mathbf{h}}^k(\mathbf{y}|\mathbf{w}))$$

$$\approx E_G(\bar{\mathbf{u}}^k(\mathbf{y}^k|\mathbf{w}), \bar{\mathbf{h}}^k(\mathbf{y}^k|\mathbf{w})) - \max_{\mathbf{u}^{k,i}{}_{1 \leq i \leq N}} \sum_i \min_{\mathbf{y}} E_{G_i}(\mathbf{u}^{k,i}(\mathbf{y}), \bar{h}^k(\mathbf{y}))$$

Dual Decompositions for Max Margin Learning
The Unconstrained Problem (Again)

TECHNISCHE UNIVERSITÄT DARMSTADT

$$\min_{\mathbf{w}} R(\mathbf{w}) + C \sum_k L_G^k(\mathbf{w})$$

$$\begin{aligned} L_g^k(\mathbf{w}) &= E_G(\bar{\mathbf{u}}^k(\mathbf{y}^k|\mathbf{w}), \bar{\mathbf{h}}^k(\mathbf{y}^k|\mathbf{w})) - \min E_G(\bar{\mathbf{u}}^k(\mathbf{y}|\mathbf{w}), \bar{\mathbf{h}}^k(\mathbf{y}|\mathbf{w})) \\ &\approx E_G(\bar{\mathbf{u}}^k(\mathbf{y}^k|\mathbf{w}), \bar{\mathbf{h}}^k(\mathbf{y}^k|\mathbf{w})) - \max_{\mathbf{u}^{k,i}{}_{1\leq i\leq N}} \sum_i \min_{\mathbf{y}} E_{G_i}(\mathbf{u}^{k,i}(\mathbf{y}), \bar{h}^k(\mathbf{y})) \\ &= \min_{\mathbf{u}^i{}_{1\leq i\leq N}} \left(E_G(\bar{\mathbf{u}}^k(\mathbf{y}^k|\mathbf{w}), \bar{\mathbf{h}}^k(\mathbf{y}^k|\mathbf{w})) - \sum_i \min_{\mathbf{y}} E_{G_i}(\mathbf{u}^{k,i}(\mathbf{y}), \bar{h}^k(\mathbf{y}) \right) \end{aligned}$$

Dual Decompositions for Max Margin Learning
The Unconstrained Problem (Again)

TECHNISCHE UNIVERSITÄT DARMSTADT

$$\min_{\mathbf{w}} R(\mathbf{w}) + C \sum_k L_G^k(\mathbf{w})$$

$$\begin{aligned}
L_g^k(\mathbf{w}) &= E_G(\bar{\mathbf{u}}^k(\mathbf{y}^k|\mathbf{w}), \bar{\mathbf{h}}^k(\mathbf{y}^k|\mathbf{w})) - \min E_G(\bar{\mathbf{u}}^k(\mathbf{y}|\mathbf{w}), \bar{\mathbf{h}}^k(\mathbf{y}|\mathbf{w})) \\
&\approx E_G(\bar{\mathbf{u}}^k(\mathbf{y}^k|\mathbf{w}), \bar{\mathbf{h}}^k(\mathbf{y}^k|\mathbf{w})) - \max_{{\mathbf{u}^{k,i}}_{1\leq i\leq N}} \sum_i \min_{\mathbf{y}} E_{G_i}(\mathbf{u}^{k,i}(\mathbf{y}), \bar{h}^k(\mathbf{y})) \\
&= \min_{{\mathbf{u}^i}_{1\leq i\leq N}} \left(E_G(\bar{\mathbf{u}}^k(\mathbf{y}^k|\mathbf{w}), \bar{\mathbf{h}}^k(\mathbf{y}^k|\mathbf{w})) - \sum_i \min_{\mathbf{y}} E_{G_i}(\mathbf{u}^{k,i}(\mathbf{y}), \bar{h}^k(\mathbf{y}) \right) \\
&= \min_{{\mathbf{u}^i}_{1\leq i\leq N}} \sum_i \left(E_{G_i}(\mathbf{u}^k(\mathbf{y}^k|\mathbf{w}), \bar{\mathbf{h}}^k(\mathbf{y}^k|\mathbf{w})) - \min_{\mathbf{y}} E_{G_i}(\mathbf{u}^{k,i}(\mathbf{y}), \bar{h}^k(\mathbf{y}) \right)
\end{aligned}$$

Dual Decompositions for Max Margin Learning
The Unconstrained Problem (Again)

TECHNISCHE UNIVERSITÄT DARMSTADT

$$\min_{\mathbf{w}} R(\mathbf{w}) + C \sum_k L_G^k(\mathbf{w})$$

$$\begin{aligned}
L_g^k(\mathbf{w}) &= E_G(\bar{\mathbf{u}}^k(\mathbf{y}^k|\mathbf{w}), \bar{\mathbf{h}}^k(\mathbf{y}^k|\mathbf{w})) - \min E_G(\bar{\mathbf{u}}^k(\mathbf{y}|\mathbf{w}), \bar{\mathbf{h}}^k(\mathbf{y}|\mathbf{w})) \\
&\approx E_G(\bar{\mathbf{u}}^k(\mathbf{y}^k|\mathbf{w}), \bar{\mathbf{h}}^k(\mathbf{y}^k|\mathbf{w})) - \max_{\mathbf{u}^{k,i}{}_{1\leq i\leq N}} \sum_i \min_{\mathbf{y}} E_{G_i}(\mathbf{u}^{k,i}(\mathbf{y}), \bar{h}^k(\mathbf{y})) \\
&= \min_{\mathbf{u}^i{}_{1\leq i\leq N}} \left(E_G(\bar{\mathbf{u}}^k(\mathbf{y}^k|\mathbf{w}), \bar{\mathbf{h}}^k(\mathbf{y}^k|\mathbf{w})) - \sum_i \min_{\mathbf{y}} E_{G_i}(\mathbf{u}^{k,i}(\mathbf{y}), \bar{h}^k(\mathbf{y}) \right) \\
&= \min_{\mathbf{u}^i{}_{1\leq i\leq N}} \sum_i \left(E_{G_i}(\mathbf{u}^k(\mathbf{y}^k|\mathbf{w}), \bar{\mathbf{h}}^k(\mathbf{y}^k|\mathbf{w})) - \min_{\mathbf{y}} E_{G_i}(\mathbf{u}^{k,i}(\mathbf{y}), \bar{h}^k(\mathbf{y}) \right) \\
&= \min_{\mathbf{u}^i{}_{1\leq i\leq N}} \sum_i L_{G_i}^k(\mathbf{w}, \mathbf{u}^{k,i})
\end{aligned}$$

Dual Decompositions for Max Margin Learning
Relaxed Problem

TECHNISCHE UNIVERSITÄT DARMSTADT

$$\min_{\mathbf{w},\mathbf{u}^{k,i}} \quad R(\mathbf{w}) + C \sum_k \sum_i L^k_{G_i}(\mathbf{w}, \mathbf{u}^i)$$

$$\text{s.t.} \quad \forall_{p\in\mathcal{V},k} \sum_{i:p\in V_i} u^{k,i}_p = \bar{u}_p$$

The min in L_{G_i} is not differentiable, however the subgradient is easy

$$\nabla_{\text{sub}} - \min_{\mathbf{y}} E_{G_i}(\mathbf{u}^{k,i}(\mathbf{y}), \bar{h}^k(\mathbf{y})) = \nabla - E_{G_i}(\mathbf{u}^{k,i}(\hat{\mathbf{y}}^{k,i}), \bar{h}^k(\hat{\mathbf{y}}^{k,i}))$$

We can now learn $\mathbf{w}$ and $\mathbf{u}^{k,i}$ using their subgradients.
However, instead of updating $\mathbf{u}^{k,i}$ directly, we make use of an auxiliary variable

$$\lambda^{k,i}_p = u^{k,i}_p - \frac{u^k_p}{|\mathcal{I}_p|}$$

Such that the constraint maps to $\forall_p : \sum_{i\in\mathcal{I}_p} \lambda^{k,i}_p = 0$.

Dual Decompositions for Max Margin Learning
Subgradients

TECHNISCHE UNIVERSITÄT DARMSTADT

The subgradients are given by:

$$\frac{\partial - E_{G_i}(\mathbf{u}^{k,i}(\hat{\mathbf{y}}^{k,i}), \bar{h}^k(\hat{\mathbf{y}}^{k,i}))}{\partial \mathbf{w}} = -\sum_{p \in \mathcal{V}_i} \frac{\hat{\phi}_p(\hat{y}_p^{k,i}, \mathbf{x}^k)}{\mathcal{I}_p} - \sum_{c \in \mathcal{C}_i} \hat{\phi}_c(\hat{\mathbf{y}}_c^{k,i}, \mathbf{x}^k)$$

$$\frac{\partial - E_{G_i}(\mathbf{u}^{k,i}(\hat{\mathbf{y}}^{k,i}), \bar{h}^k(\hat{\mathbf{y}}^{k,i}))}{\partial \lambda_{\mathbf{p}}^{\mathbf{k,i}}(l)} = -[\hat{y}_p^{k,i} = l]$$

However we still need to enforce the constraint.

Dual Decompositions for Max Margin Learning
Projected Subgradient Descent

TECHNISCHE UNIVERSITÄT DARMSTADT

After each update, $\lambda_p^{k,i}$ has to be projected back to the convex, feasible set

$$\Lambda = \left\{ \lambda_p^{k,i} \middle| \sum_{i \in \mathcal{I}_p} \lambda_p^{k,i} = 0 \right\}$$

Hence, after each update we have to subtract $\frac{\sum_{i \in \mathcal{I}_p} \lambda_p^{k,i}}{|\mathcal{I}|_p}$. The update then becomes

$$\lambda_p^{k,i}(l) \leftarrow \lambda_p^{k,i}(l) - \alpha_t C \left([y_p^k = l] - [\hat{y}_p^{k,i} = l] \right) - \frac{\sum_{i \in \mathcal{I}_p} \lambda_p^{k,i}(l) + \alpha_t C \left([y_p^k = l] - [\hat{y}_p^{k,i} = l] \right)}{|\mathcal{I}_p|}$$

$$= \lambda_p^{k,i}(l) - \alpha_t C \left([y_p^k = l] - [\hat{y}_p^{k,i} = l] - \frac{\sum_{i \in \mathcal{I}_p} [y_p^k = l] - [\hat{y}_p^{k,i} = l]}{|\mathcal{I}_p|} \right)$$

$$= \lambda_p^{k,i}(l) + \alpha_t C \left([\hat{y}_p^{k,i} = l] - \frac{\sum_{i \in \mathcal{I}_p} [\hat{y}_p^{k,i} = l]}{|\mathcal{I}_p|} \right)$$

Dual Decompositions for Max Margin Learning
Choosing The Decompositions

We have two requirements on the decomposition:

1. The slave problems should be tractable
2. We want a good bound on the loss function

Some notes:

- The dual relaxation with G_{single} (one hyperedge per subgraph) corresponds to the LP relaxation of the IP formulation
- For any decomposition better that G_{single} there will be one sub-hypergraph for which the LP relaxation is not tight
- You can get better than G_{single} by including small loops
- different decompositions that yield the same loss may have different speeds of convergence. E.g. for pairwise MRFs, G_{tree} will correspond to the same relaxation as G_{single} but information can propagate faster.

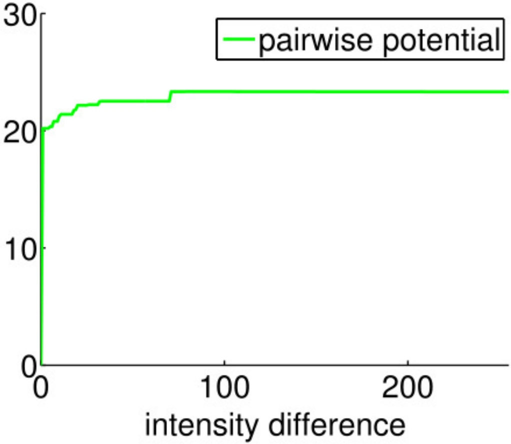
Dual Decompositions for Max Margin Learning
Experiments: Image Denoising

TECHNISCHE UNIVERSITÄT DARMSTADT

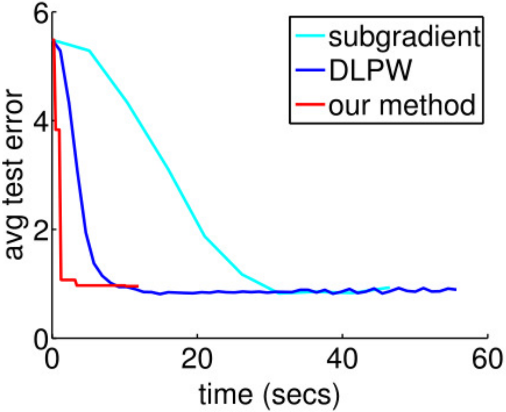
Approach:

- A pairwise model is assumed
- Unary potentials are known: $u_p(\ell) = |\ell - I_p|$
- Pairwise potentials should be learned: $h_{pq}(\ell_p, \ell_q) = V(|\ell_p - \ell_q|)$

Dual Decompositions for Max Margin Learning
Experiments: Image Denoising
Learned V



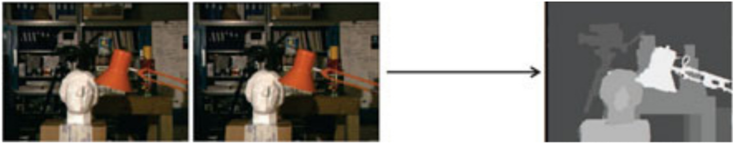
Dual Decompositions for Max Margin Learning
Experiments: Image Denoising
Performance



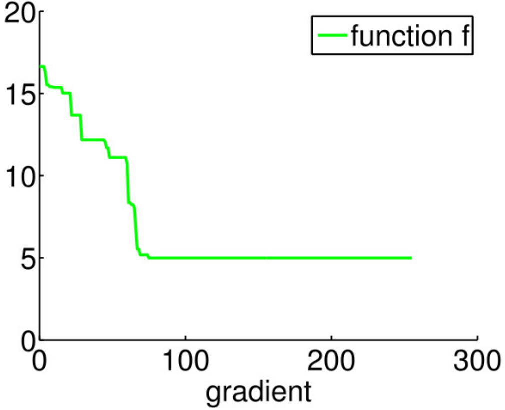
Dual Decompositions for Max Margin Learning
Experiments: Stereo Matching

Approach:

- A pairwise model is assumed
- Unary potentials are known: $u_p(\ell) = |I_p^{\text{left}} - I_{p-\ell}^{\text{right}}|$
- Pairwise potentials should be learned: $h_{pq}(\ell_p, \ell_q) = f(|I_p^{\text{left}} - I_q^{\text{left}}|)[\ell_p \neq \ell_q]$
- A-priori knowledge that f should be decreasing is encoded using an additional Projection on **w**

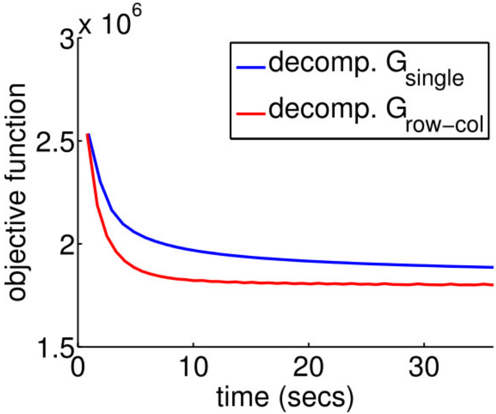


Dual Decompositions for Max Margin Learning
Experiments: Stereo Matching
Learned V



Dual Decompositions for Max Margin Learning
Experiments: Stereo Matching
Performance

- simple model yields larger disparity errors on middlebury dataset than SOTA



Dual Decompositions for Max Margin Learning
Experiments: Knowledge Based Segmentation

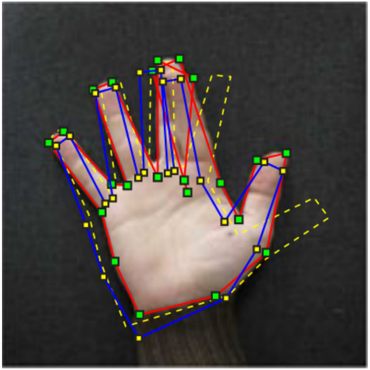
TECHNISCHE UNIVERSITÄT DARMSTADT

Approach:

- generates n control points on the boundary of the known object
- pairwise cliques for the object boundaries
- triplet-cliques to learn sparse, pose-invariant shape priors
- one triplet-clique for each possible combination of three points
- two inner angles $\alpha_c(\mathbf{y}_c)$ and $\beta_c(\mathbf{y}_c)$ as pose-invariant properties
- higher-order potentials are based on a probabilistic model $h_c(\mathbf{y}_c) = -w_c \log p_c(\alpha_c(\mathbf{y}_c), \beta_c(\mathbf{y}_c))$
- L_1 regularization to learn sparse $\mathbf{w}$
- dissimilarity function $\Delta(\mathbf{y}, \mathbf{y}')$ is also zero if y' and y are connected by a similarity transformation

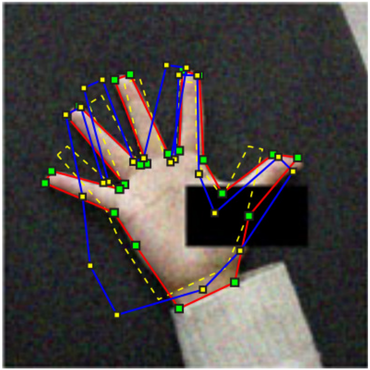
Dual Decompositions for Max Margin Learning
Experiments: Knowledge Based Segmentation

TECHNISCHE UNIVERSITÄT DARMSTADT

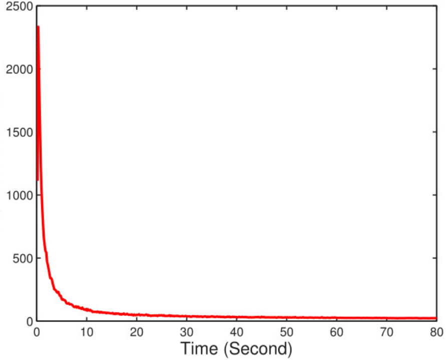


Dual Decompositions for Max Margin Learning
Experiments: Knowledge Based Segmentation

TECHNISCHE UNIVERSITÄT DARMSTADT



Dual Decompositions for Max Margin Learning Experiments: Knowledge Based Segmentation Performance



Learned **w** has only 5.6 percent non-zero elements.

Recap

TECHNISCHE UNIVERSITÄT DARMSTADT

We looked at

1. Conditional Random Fields
   - model conditional distribution $p(\mathbf{Y}|\mathbf{X})$
   - require inference for training/prediction
2. Maximum Margin Markov Networks
   - concentrate on the decision boundary
   - aim at maximizing the loss-augmented margin
   - training can get feasible by replacing the dual variables by marginals
   - specifying the constraints only feasible for simple potential/graph-structures
3. Efficient Max-Margin Learning with Dual Decompositions
   - dual relaxation based on (almost) arbitrary graph decompositions
   - loss function has to decompose over the graph
   - How to decompose the graph?

References

TECHNISCHE UNIVERSITÄT DARMSTADT

- Lafferty, John, Andrew McCallum, and Fernando CN Pereira. "Conditional random fields: Probabilistic models for segmenting and labeling sequence data." 8th International Conf. on Machine Learning (2001), pp. 282-289.
- Taskar, Ben, Carlos Guestrin, and Daphne Koller. "Max-Margin Markov Networks." Advances in Neural Information Processing Systems. 2003.
- Nikos Komodakis, Bo Xiang, and Nikos Paragios. "A Framework for Efficient Structured Max-Margin Learning of High-Order MRF Models" IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 37,no. 7, 2015.