



Aalto University  
School of Science  
and Technology

# Structured Output Learning with A Random Sample of Spanning Trees

Hongyu Su

Helsinki Institute for Information Technology HIIT  
Department of Information and Computer Science  
Aalto University

November 19, 2014

# Multilabel Classification

- ▶ Multilabel classification is an important research field in machine learning.
  - ▶ For example, a document can be classified as “science”, “genomics”, and “drug discovery”.
  - ▶ Each input variable  $\mathbf{x} \in \mathcal{X}$  is simultaneously associated with multiple output variables  $\mathbf{y} \in \mathcal{Y}, \mathcal{Y} = \mathcal{Y}_1 \times \cdots \times \mathcal{Y}_l$ .
  - ▶ The goal is to find a mapping function that predicts the best values of an output given an input  $f \in \mathcal{H} : \mathcal{X} \rightarrow \mathcal{Y}$ .
- ▶ The central problems of multilabel classification:
  - ▶ The size of the output space  $\mathcal{Y}$  is exponential in the number of microlabels.
  - ▶ The dependency of microlabels needs to be exploited to improve the prediction performance.

# Flat Multilabel Classification

- ▶ Multiple output variables are treated as a “flat” vector.
- ▶ For example, ML-KNN, ADABOOST.MH, MTL, ...
- ▶ It is difficult to take into consideration the correlation of labels.

# Structured Output Learning

- ▶ There is an *output graph* connecting multiple labels.
  - ▶ A set of nodes corresponds to the multiple labels.
  - ▶ A set of edges represents the correlation between labels.
- ▶ Hierarchical classification:
  - ▶ The output graph is a rooted tree or a directed graph defining the different levels of granularities.
  - ▶ For example, SSVM, ...
- ▶ Graph labeling:
  - ▶ The output graph often takes a general form (e.g., a tree, a chain).
  - ▶ For example,  $M^3N$ , CRF, MMCRF, ...
- ▶ The output graph is assumed to be known *apriori*.

# Research Question

- ▶ The output graph is hidden in many applications.
  - ▶ For example, a surveillance photo can be tagged with “building”, “road”, “pedestrian”, and “vehicle”.
- ▶ We focus on the problem in structured output learning when the output graph is not observed.
- ▶ Our approach:
  - ▶ Assume the dependency can be modeled by a complete set of pairwise correlations.
  - ▶ Build a structured output learning model with a complete graph as the output graph.
  - ▶ Solve the optimization problem and the inference problem.

# Contributions

- ▶ A structured output learning model which performs max-margin learning on a random sample of spanning tree.
- ▶ The model is not constrained to the availability of the output graph.
- ▶ The  $\mathcal{NP}$ -hard inference problem can be solved by a polynomial time algorithm with a condition guaranteeing the exact solution.
- ▶ The theoretical analysis and the empirical results verify the performance of the proposed model.

# Model

- ▶ The training examples are given in pair  $S = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^m$ .
- ▶ A complete graph  $G = (E, V)$  is used as the output graph.
- ▶  $\Gamma_G(\mathbf{y}_i)$  is the output feature map on a complete graph  $G$ .
- ▶ Each example is mapped to a joint feature space by a joint feature map

$$\phi_G(\mathbf{x}_i, \mathbf{y}_i) = \varphi(\mathbf{x}_i) \otimes \Gamma_G(\mathbf{y}_i).$$

- ▶ A compatibility score is defined as

$$F(\mathbf{x}, \mathbf{y}; \mathbf{w}) = \langle \mathbf{w}, \phi_G(\mathbf{x}, \mathbf{y}) \rangle = \sum_{e \in E} \langle \mathbf{w}_e, \phi_G(\mathbf{x}, \mathbf{y}_e) \rangle$$

- ▶  $\mathbf{w}$  ensures an input  $\mathbf{x}_i$  with a correct multilabel  $\mathbf{y}_i$  achieves a higher score than with any incorrect multilabel  $\mathbf{y} \in \mathcal{Y}$ .

# Model (cont.)

- The predicted output  $\mathbf{y}(\mathbf{x})$  for a given input  $\mathbf{x}$  is computed by

$$\mathbf{y}(\mathbf{x}) = \operatorname{argmax}_{\mathbf{y} \in \mathcal{Y}} F(\mathbf{x}, \mathbf{y}; \mathbf{w}) = \operatorname{argmax}_{\mathbf{y} \in \mathcal{Y}} \langle \mathbf{w}, \phi_G(\mathbf{x}, \mathbf{y}) \rangle,$$

which is called *inference problem*.

- The inference problem is  $\mathcal{NP}$ -hard for most joint feature maps on the complete graph.



# Learning with A Complete Graph

- ▶ The *margin* of an example  $\mathbf{x}_i$  is

$$\gamma_G(\mathbf{x}_i; \mathbf{w}) = \min_{\mathbf{y} \in \mathcal{Y}} [F(\mathbf{x}_i, \mathbf{y}_i; \mathbf{w}) - F(\mathbf{x}_i, \mathbf{y}; \mathbf{w})].$$

- ▶  $\mathbf{w}$  is solved by *maximum-margin principle* which aims to maximize  $\gamma(\mathbf{x}_i; \mathbf{w})$  over all training example.
- ▶ The problem includes:
  - ▶ The  $\mathcal{NP}$ -hardness of the inference problem on a complete graph.
  - ▶ A large parameter space:  $\Theta(k^2)$
- ▶ We aim to use a joint feature map that allows the inference problem be solved in polynomial time.

# Random Spanning Tree Approximation

- ▶ We proved if a large margin structured output predictor exists, then combining a small sample of random trees will, with a high probability, generate a predictor with good generalization.
- ▶  $\mathcal{T} = \{T_1, \dots, T_n\}$  is a set of spanning trees randomly sampled from the complete graph  $G$ .
- ▶ The compatibility score can be re-defined based on  $\mathcal{T}$  as

$$F_{\mathcal{T}}(\mathbf{x}_i, \mathbf{y}_i; \mathbf{w}_{\mathcal{T}}) = \sum_{t=1}^n \langle \mathbf{w}_{T_t}, \phi_{T_t}(\mathbf{x}_i, \mathbf{y}_i) \rangle.$$

- ▶ The inference problem of predicting the output  $\mathbf{y}_{\mathcal{T}}(\mathbf{x})$  for a given input  $\mathbf{w}$  is

$$\mathbf{y}_{\mathcal{T}}(\mathbf{x}) = \underset{\mathbf{y} \in \mathcal{Y}}{\operatorname{argmax}} F_{\mathcal{T}}(\mathbf{x}, \mathbf{y}; \mathbf{w}_{\mathcal{T}}) = \underset{\mathbf{y} \in \mathcal{Y}}{\operatorname{argmax}} \sum_{t=1}^n \langle \mathbf{w}_{T_t}, \phi_{T_t}(\mathbf{x}, \mathbf{y}) \rangle.$$

# Optimization Problem

- ▶ The margin of an example  $\mathbf{x}_i$  achieved by  $\mathcal{T}$  is

$$\gamma_{\mathcal{T}}(\mathbf{x}_i; \mathbf{w}_{\mathcal{T}}) = \min_{\mathbf{y} \in \mathcal{Y}} \left[ \sum_{t=1}^n \langle \mathbf{w}_{T_t}, \phi_{T_t}(\mathbf{x}_i, \mathbf{y}_i) \rangle - \sum_{t=1}^n \langle \mathbf{w}_{T_t}, \phi_{T_t}(\mathbf{x}_i, \mathbf{y}) \rangle \right].$$

- ▶ To learn  $\mathbf{w}_{T_t}, \forall T_t \in \mathcal{T}$  we solve the optimization problem

$$\begin{aligned} \min_{\mathbf{w}_{T_t}, \xi_i} \quad & \frac{1}{2} \sum_{t=1}^n \|\mathbf{w}_{T_t}\|^2 + C \sum_{i=1}^m \xi_i \\ \text{s.t.} \quad & \sum_{t=1}^n \langle \mathbf{w}_{T_t}, \phi_{T_t}(\mathbf{x}_i, \mathbf{y}_i) \rangle - \max_{\mathbf{y} \neq \mathbf{y}_i} \sum_{t=1}^n \langle \mathbf{w}_{T_t}, \phi_{T_t}(\mathbf{x}_i, \mathbf{y}) \rangle \geq 1 - \xi_i, \\ & \xi_i \geq 0, \forall i \in \{1, \dots, m\}, \end{aligned}$$

# Inference Problem

- ▶ The inference problem of RTA is defined as finding the multilabel  $\mathbf{y}_{\mathcal{T}}(\mathbf{x})$  that maximizes the sum of scores over a collection of trees

$$\mathbf{y}_{\mathcal{T}}(\mathbf{x}) = \operatorname{argmax}_{\mathbf{y} \in \mathcal{Y}} F_{\mathcal{T}}(\mathbf{x}, \mathbf{y}; \mathbf{w}_{\mathcal{T}}) = \operatorname{argmax}_{\mathbf{y} \in \mathcal{Y}} \sum_{t=1}^n \langle \mathbf{w}_{T_t}, \phi_{T_t}(\mathbf{x}, \mathbf{y}) \rangle.$$

- ▶ The inference problem on each individual spanning tree can be solve efficiently in  $\Theta(I)$  by *dynamic programming*

$$\mathbf{y}_{T_t}(\mathbf{x}) = \operatorname{argmax}_{\mathbf{y} \in \mathcal{Y}} F_{T_t}(\mathbf{x}, \mathbf{y}; \mathbf{w}_{T_t}) = \operatorname{argmax}_{\mathbf{y} \in \mathcal{Y}} \langle \mathbf{w}_{T_t}, \phi_{T_t}(\mathbf{x}, \mathbf{y}) \rangle.$$

- ▶ There is no guarantee that there exists a tree  $T_t \in \mathcal{T}$  for which the maximizer of  $F_{T_t}$  is the maximizer of  $F_{\mathcal{T}}$ .

# Fast Inference Over a Collection of Trees

- ▶ For each tree  $T_t$ , instead of computing the best multilabel  $\mathbf{y}_{T_t}$ , we compute  $K$ -best multilabels in  $\Theta(KI)$  time

$$\mathcal{Y}_{T_t,K} = \{\mathbf{y}_{T_t,1}, \dots, \mathbf{y}_{T_t,K}\}.$$

- ▶ Performing the same computation on all trees gives a candidate list of  $n \times K$  multilabels in  $\Theta(nKI)$  time

$$\mathcal{Y}_{\mathcal{T},K} = \mathcal{Y}_{T_1,K} \cup \dots \mathcal{Y}_{T_n,K}.$$

- ▶ For now, we assume the best scoring multilabel of a collection of trees exists in the list  $\mathcal{Y}_{\mathcal{T},K}$ .

# Fast Inference Over a Collection of Trees (cont.)

- Assume

$$\mathbf{y}_K^* = \operatorname{argmax}_{\mathbf{y} \in \mathcal{Y}_{\mathcal{T}, K}} F_{\mathcal{T}}(\mathbf{x}, \mathbf{y}; \mathbf{w}_{\mathcal{T}}).$$

If

$$F_{\mathcal{T}}(\mathbf{x}, \mathbf{y}_K^*; \mathbf{w}_{\mathcal{T}}) \geq \frac{1}{n} \sum_{t=1}^n F_{T_t}(\mathbf{x}, \mathbf{y}_{T_t, K}, \mathbf{w}_{T_t}) = \theta_{\mathbf{x}}(K),$$

then

$$F_{\mathcal{T}}(\mathbf{x}, \mathbf{y}_K^*; \mathbf{w}_{\mathcal{T}}) = \max_{\mathbf{y} \in \mathcal{Y}} F_{\mathcal{T}}(\mathbf{x}, \mathbf{y}; \mathbf{w}_{\mathcal{T}}).$$

# Fast Inference Over a Collection of Trees (cont.)

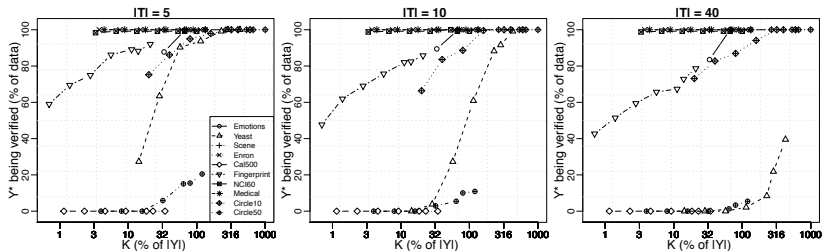
- For example,  $\mathcal{T} = \{T_1, T_2\}$ ,  $\mathcal{Y} = \mathcal{Y}_1 \times \mathcal{Y}_2$ ,  $\mathcal{Y}_i = \{+, -\}$

	$T_1$		$T_2$		$\theta_{\mathbf{x}}(K)$
	$\mathbf{y}_{T_1, K}$	$F_{T_1}(\mathbf{x}, \mathbf{y}_{T_1, K})$	$\mathbf{y}_{T_2, K}$	$F_{T_2}(\mathbf{x}, \mathbf{y}_{T_2, K})$	
$K = 1$	$+-$	5	$--$	4	9
$K = 2$	$++$	4	$-+$	3	7
$K = 3$	$-+$	3	$++$	3	6
$K = 4$	$--$	3	$+-$	2	5

- We proved that with a high probability  $\mathbf{y}_{\mathcal{T}}$  will appear in  $\mathcal{Y}_{\mathcal{T}, K}$ .

# Performance of the Inference Algorithm

- ▶ 10 datasets,  $|\mathcal{T}| = \{5, 10, 40\}$ ,  $K = \{2, 4, 8, 16, 32, 40, 60\}$
- ▶ X-axis is the percentage of examples with exact inference.
- ▶ Y-axis is the value of  $K$  as the percentage of the number of microlabels.
- ▶  $K = 100\%|Y|$  corresponds to a complexity of  $\Theta(nI^2)$ .





# Prediction Performance

DATASET	MICROLABEL LOSS (%)					0/1 LOSS (%)				
	SVM	MTL	MMCRF	MAM	RTA	SVM	MTL	MMCRF	MAM	RTA
EMOTIONS	22.4	20.2	20.1	<i>19.5</i>	<b>18.8</b>	77.8	74.5	71.3	<i>69.6</i>	<b>66.3</b>
YEAST	<i>20.0</i>	20.7	21.7	20.1	<b>19.8</b>	85.9	88.7	93.0	86.0	<b>77.7</b>
SCENE	9.8	11.6	18.4	17.0	<b>8.8</b>	47.2	55.2	72.2	94.6	<b>30.2</b>
ENRON	6.4	6.5	6.2	<b>5.0</b>	<i>5.3</i>	99.6	99.6	92.7	<i>87.9</i>	<b>87.7</b>
CAL500	<b>13.7</b>	<i>13.8</i>	<b>13.7</b>	<b>13.7</b>	<i>13.8</i>	100.0	100.0	100.0	100.0	100.0
FINGERPRINT	<b>10.3</b>	17.3	<i>10.5</i>	<i>10.5</i>	10.7	99.0	100.0	99.6	<i>99.6</i>	<b>96.7</b>
NCI60	15.3	16.0	<i>14.6</i>	<b>14.3</b>	14.9	56.9	<i>53.0</i>	63.1	60.0	<b>52.9</b>
MEDICAL	2.6	2.6	<b>2.1</b>	<b>2.1</b>	<b>2.1</b>	91.8	91.8	63.8	<i>63.1</i>	<b>58.8</b>
CIRCLE10	4.7	6.3	2.6	2.5	<b>0.6</b>	28.9	33.2	20.3	<i>17.7</i>	<b>4.0</b>
CIRCLE50	5.7	6.2	<b>1.5</b>	<i>2.1</i>	3.8	69.8	72.3	<b>38.8</b>	<i>46.2</i>	52.8

**Figure :** Prediction performance of each algorithm in terms of microlabel loss and 0/1 loss. The best performing algorithm is highlighted with boldface, the second best is in italic

# Conclusions

- ▶ Theoretical study shows if a large margin structured output learner exists, then the combination of a random sample of spanning trees will achieve a similar margin with a high probability.
- ▶ The  $K$ -best inference algorithm is tractable which is proved theoretically and empirically.
- ▶ RTA is not constrained by the availability of the output graph, it can therefore be applied to a wider range of multilabel classification problem where the output graph is believed to play an important role during learning.