Multi-label Ranking from Positive and Unlabeled Data

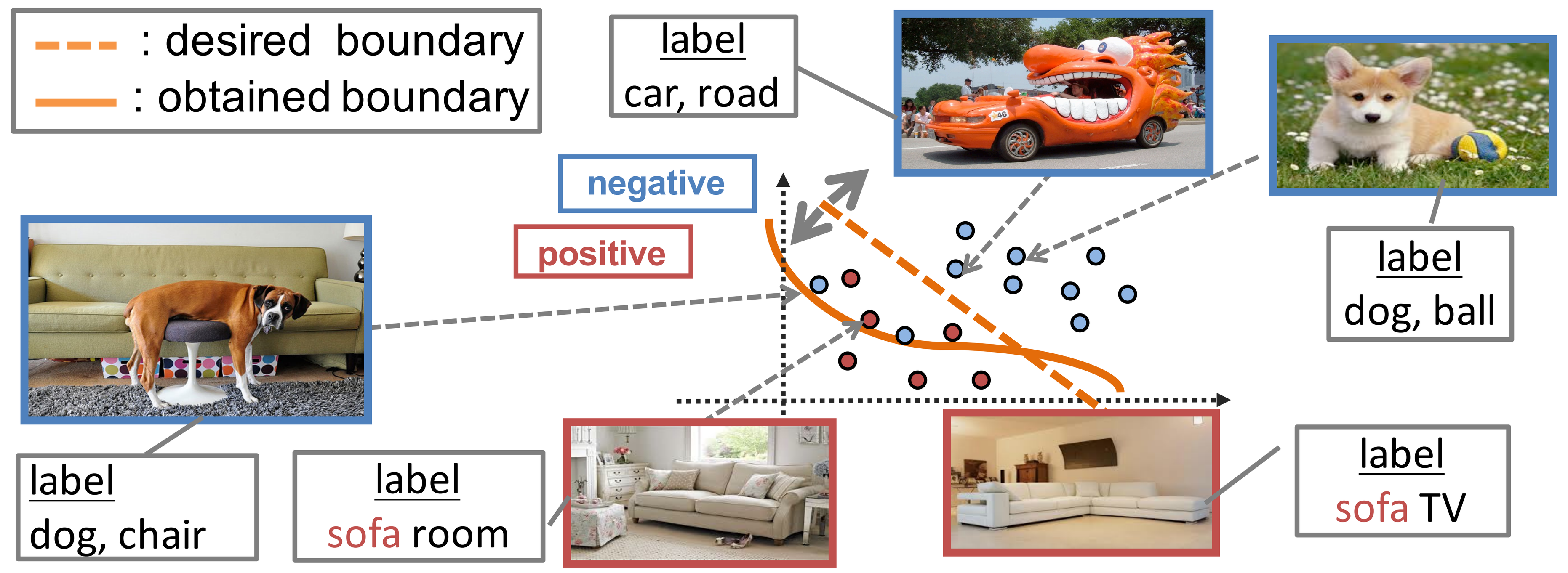
Atsushi Kanehira and Tatsuya Harada, The University of Tokyo

IEEE 2016 Conference on Computer Vision and Pattern Recognition

CVPR 2016

Introduction:

- Multi-label dataset is incomplete.
  - Assigned labels are reliable because they are based on human's judgement.
  - Positive but absent label still exists.
- However, label's incompleteness is usually ignored, and
- it degrades classifier's performance.



❑ Goal:

Training multi-label classifier from incompletely labeled data

We treat incomplete label problem as multi-label PU classification.

What is multi-label PU classification ?

(1) Assigned labels are definitely positive.
(2) Absent labels are not necessarily negative.
(3) Samples are allowed to take multiple labels.

(1), (2) → PU classification
(3) → Multi-label classification

❑ Contributions:

(1) We showed two conditions which should be met in order to train classifier consistently from multi-label PU dataset:

a) Loss function should be weighted properly.
b) Symmetric surrogate loss function should be used.

(2) We demonstrated efficacy by the experiment on several datasets.

Analysis of multi-label PU ranking:

❑ Formulation:

$$\min L_{\text{true}} = \mathbb{E}_{\mathrm{xy}}[R(f(\mathrm{x}),\mathrm{y})]$$

$$R(\mathrm{f}(\mathrm{x}),\mathrm{y}) = p(f_i < f_j | y_i = 1, y_j = 0)$$ (mis-rank rate)

$\mathrm{x} \in \mathbb{R}^d$: sample,

$\mathrm{y} \in \{0,1\}^m$: true label, $s \in \{0,1\}^m$: observed label

where d Is feature dimension and m is the number of classes

minimizing ranking loss, with only observation of s (labeled or not)

❑ Analysis:

a) Loss function should be weighted properly.

We can not estimate mis-rank rate, instead we can observe.

$$R_X(\mathrm{f}(\mathrm{x}),\mathrm{s}) = p(f_i < f_j | s_i = 1, s_j = 0)$$ (pseudo mis-rank rate)

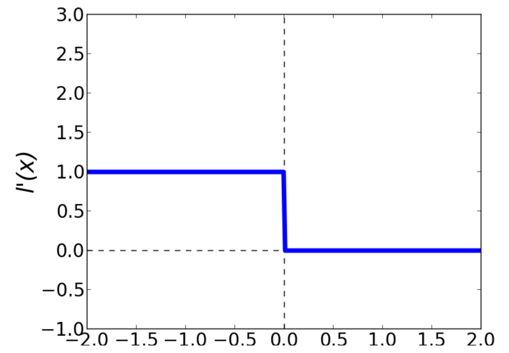
$$L_{\mathrm{PU}} = \mathbb{E}_{\mathrm{xs}}[c_{ij} R_X(f(\mathrm{x}),\mathrm{s})] = L_{\text{true}} - \text{const}$$ (where $c_{ij} = \frac{p(y_i = 1)}{p(s_i = 1, s_j = 0)}$)

- We can minimize loss function only from observed data.

b) Symmetric surrogate loss function should be used.

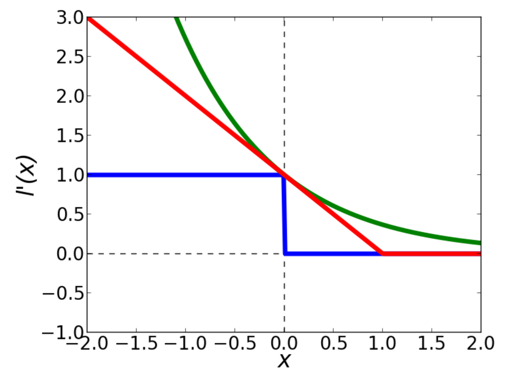
optimization of loss function

$$R(\mathrm{f}(\mathrm{x}),\mathrm{y}) = p(f_i < f_j | y_i = 1, y_j = 0) = \mathbb{E}_{x|y_i=1,y_j=0}[l_{0\text{-}1}(f_i - f_j)]$$



Due to computationally complexity, surrogate loss (e.g. hinge) is usually used.

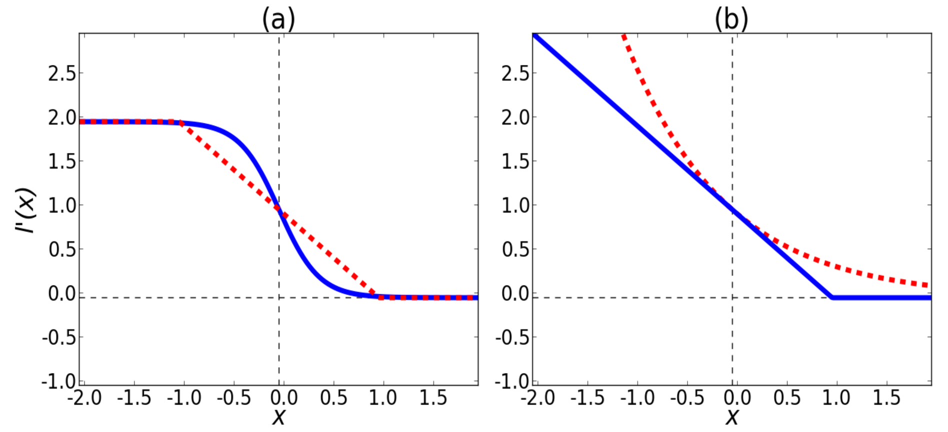
$$= \mathbb{E}_{x|y_i=1,y_j=0}[l'_{\mathrm{sur}}(f_i - f_j)]$$



Using surrogate loss,

$$L'_{\mathrm{PU}} = L'_{\text{true}} + \underbrace{p(y_i = 1, y_j = 1)\mathbb{E}_{x|y_i=1,y_j=1}[l'(f_i - f_j) + l'(f_j - f_i)]}_{\text{Surrogate loss generate bias}}$$

Bias can be cancelled for symmetric surrogate loss.



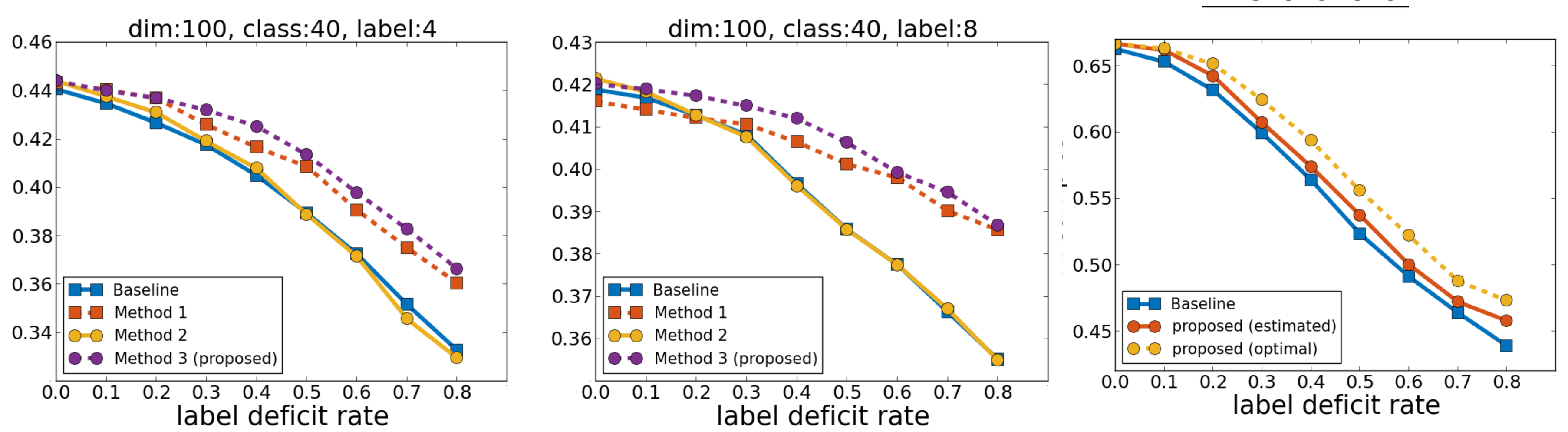
Experiment:

❑ Setting:

- synthetic dataset, image annotation dataset (MSCOCO, NUS-WIDE)
- compared 4 methods, which corresponding to each condition
- trained with data with 0-80% label deficit
- evaluated on Mean Average Precision

	Not symmetric (hinge loss)	Symmetric (ramp loss)
Not weighted	Baseline	Method ②
weighted	Method ①	Method ③(proposed)

❑ Result:



Proposed methods outperform others.

Conclusion:

we derived two conditions to train classifier consistently

a) Loss function should be weighted properly.
b) Symmetric surrogate loss function should be used.