Probabilistic Instance-dependent Label Refinement for Noisy Label Learning

Hao-Yuan He

Joint work with Yu Liu, Ren-Biao Liu, Zheng Xie and Ming Li.

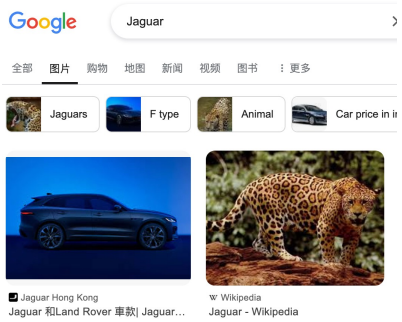
LAMDA Group, School of Artificial Intelligence, Nanjing University

Learning with Noisy Examples

- Label noise is (almost) everywhere.

Crowdsourcing

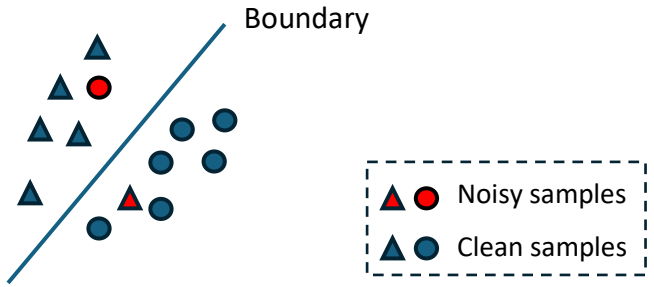
Label from non-experts



Search engine

Label from web-crawler

- Noisy labels hinder learning.



Source from vectorstock and google image

Previous Assumption: Class-conditional Noise (CCN)

- Assume the noise is instance-independent and class-conditional:

$$p(\tilde{y} \mid x, y) = p(\tilde{y} \mid y)$$

- Using the transition matrix, we have: $\boldsymbol{p}_{\tilde{y}|x} = T^\top \boldsymbol{p}_{y|x}$

$$\begin{pmatrix} p(\tilde{y}=1 \mid x) \\ \vdots \\ p(\tilde{y}=c \mid x) \end{pmatrix} = \begin{pmatrix} p(\tilde{y}=1 \mid y=1) & \dots & p(\tilde{y}=c \mid y=1) \\ \vdots & \ddots & \vdots \\ p(\tilde{y}=1 \mid y=c) & \dots & p(\tilde{y}=c \mid y=c) \end{pmatrix} \begin{pmatrix} p(y=1 \mid x) \\ \vdots \\ p(y=c \mid x) \end{pmatrix}$$

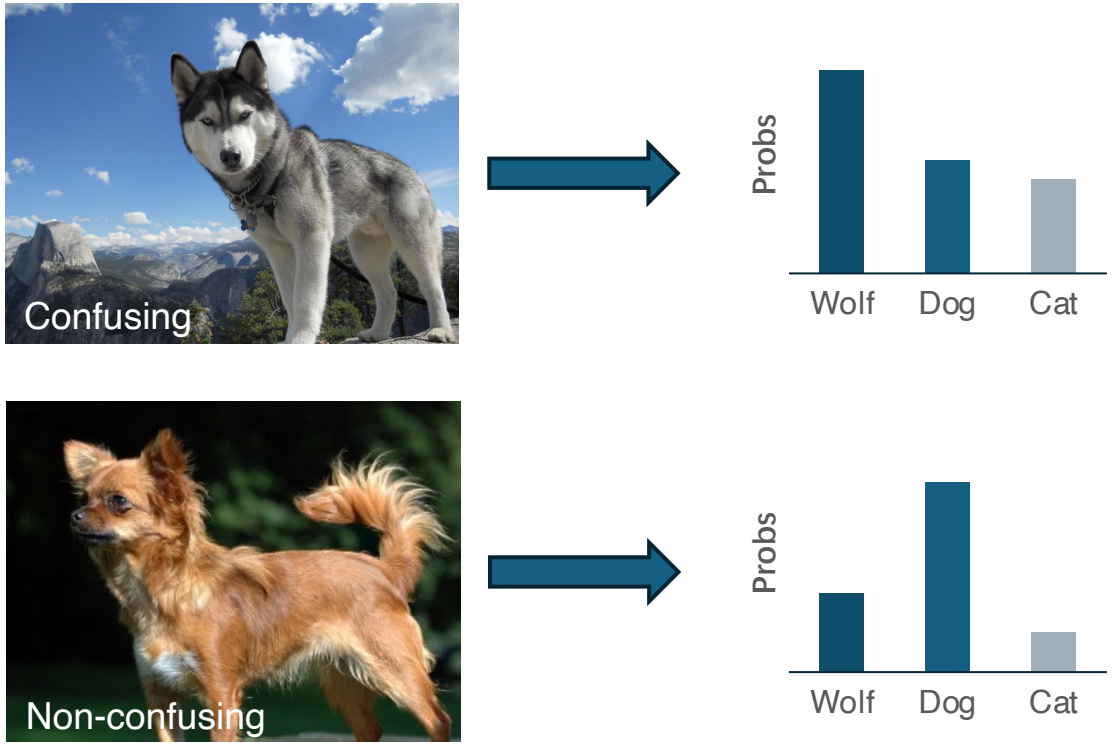
- Loss correction: rewrite the loss to an unbias estimator if given T.

$$\tilde{\ell}_{ir}(f(\boldsymbol{x}), y) = \frac{p(\boldsymbol{x}, y)}{\tilde{p}(\boldsymbol{x}, y)} \ell(f(\boldsymbol{x}), y) = \frac{\boldsymbol{g}_y(\boldsymbol{x})}{(T^\top \boldsymbol{g})_y(\boldsymbol{x})} \ell(f(\boldsymbol{x}), y)$$

$$\mathbb{E}_{(X,\tilde{Y})\sim\tilde{D}} \left[\tilde{\ell}_{ir}(f(X), \tilde{Y}) \right] = \mathbb{E}_{(X,Y)\sim D} [\ell(f(X), Y)]$$

Realistic Noise Type: Instance-dependent Noise (IDN)

- Realistic label noise is always instance-dependent.
  - Confusing instances are more likely to be misclassified.

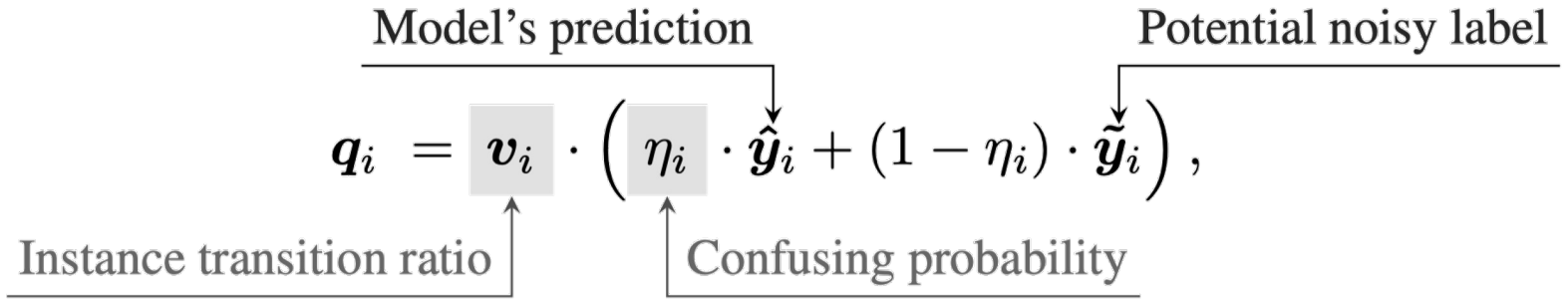


Source from Wikipedia.

π-LR: Probabilistic Instance-dependent Label Refinement

- Main idea:
  - Modeling the true label from the probabilistic perspective.
  - Estimating the confusing probability helps modeling label noise.

- Main result:



Here we slightly change the notation for convenience.

Instance-dependent Noise Modeling

- The estimated true label $\boldsymbol{q}_i = \left[\Pr\left(\boldsymbol{y}_i^1 = 1 \mid \boldsymbol{x}_i\right), \ldots, \Pr\left(\boldsymbol{y}_i^c = 1 \mid \boldsymbol{x}_i\right)\right]^\top$.
  - Consider the jth term:

$$\boldsymbol{q}_i^j = \Pr\left(\boldsymbol{y}_i^j = 1 \mid \boldsymbol{x}_i\right) = \frac{\Pr\left(\tilde{\boldsymbol{y}}_i, \boldsymbol{y}_i^j = 1 \mid \boldsymbol{x}_i\right)}{\Pr\left(\tilde{\boldsymbol{y}}_i \mid \boldsymbol{y}_i^j = 1, \boldsymbol{x}_i\right)} = \underbrace{\Pr\left(\tilde{\boldsymbol{y}}_i \mid \boldsymbol{x}_i\right)}_{\text{denote as } \psi_i} \cdot \frac{\Pr\left(\tilde{\boldsymbol{y}}_i^j = 1 \mid \boldsymbol{y}_i, \boldsymbol{x}_i\right)}{\Pr\left(\tilde{\boldsymbol{y}}_i \mid \boldsymbol{y}_i^j = 1, \boldsymbol{x}_i\right)}$$

  - Using the concept of confusing probability, we can expand the blue term as:

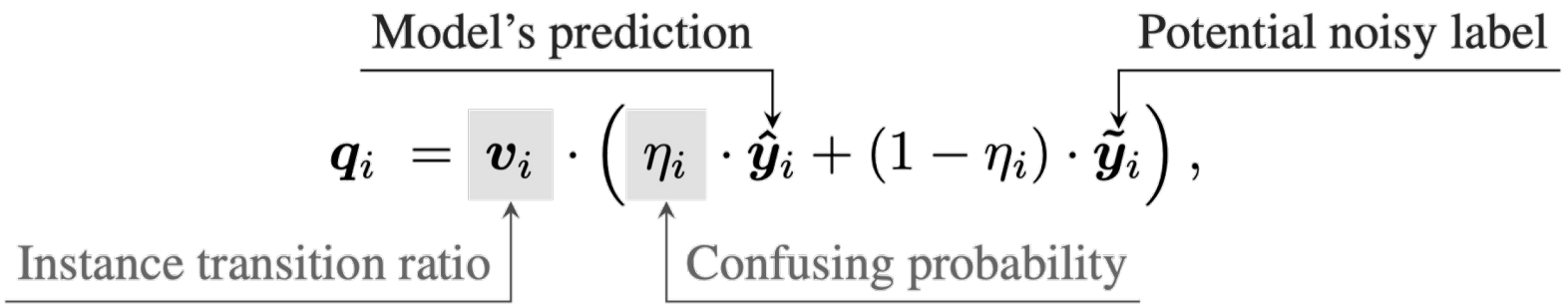
$$\Pr\left(\boldsymbol{y}_i^j = 1 \mid s_i = 0, \tilde{\boldsymbol{y}}_i, \boldsymbol{x}_i\right) \cdot (1 - \eta_i) + \Pr\left(\boldsymbol{y}_i^j = 1 \mid s_i = 1, \tilde{\boldsymbol{y}}_i, \boldsymbol{x}_i\right) \cdot \eta_i$$

    - The first term refer to the non-confusing case, which equals $\mathbb{I}\left(\tilde{\boldsymbol{y}}_i^j = \boldsymbol{y}_i^j\right) = \tilde{\boldsymbol{y}}_i^j$
    - The second term can be estimated by the prediction of a trained model.
    - Denote the resident part as $\boldsymbol{v}_i = \Pr(\tilde{\boldsymbol{y}} \mid \boldsymbol{x}_i) / \Pr(\tilde{\boldsymbol{y}}_i \mid \boldsymbol{y}_i^j = 1, \boldsymbol{x}_i)$
  - Overall, we get

$$\boldsymbol{q}_i = \boldsymbol{v}_i \cdot (\eta_i \cdot \hat{\boldsymbol{y}}_i + (1 - \eta_i) \cdot \tilde{\boldsymbol{y}}_i)$$

What Next?

- Recall the main result:

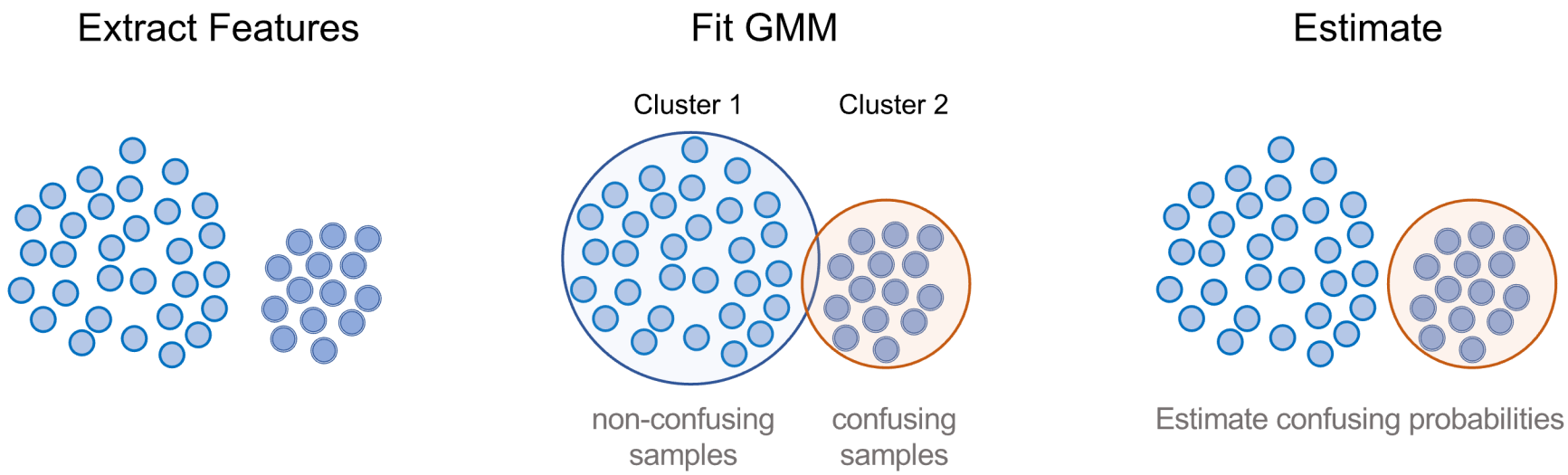


- Two key components:
  - Estimate the confusing probability.
  - Estimate the instance transition ratio.

Estimation of Confusing Probabilities

- Challenge: No direct supervision
  - Assumptions:
    - The distributions of the confused and non-confused samples are different.
    - Specifically, here we adopt the Gaussian mixture model (GMM) assumption.

- Estimation process



Optimization for Instance Transition Ratios

- Challenge: Intractable probabilistic inference.

$$\boldsymbol{v}_i = \Pr(\tilde{\boldsymbol{y}} \mid \boldsymbol{x}_i) / \Pr(\tilde{\boldsymbol{y}}_i \mid \boldsymbol{y}_i^j = 1, \boldsymbol{x}_i)$$

- Set this term as learnable parameters.
  - Derived optimization object via variational inference:

$$\ell(\Theta) = \sum_{i \in [N]} \log \Pr\left(\tilde{\boldsymbol{y}}_i \mid \boldsymbol{x}_i; \Theta\right)$$

$$\geq \mathbb{E}_{i \in [N], j \in [c]} \left[\boldsymbol{q}_i^j \cdot \log \left[\Pr\left(\tilde{\boldsymbol{y}}_i, \boldsymbol{y}_i^j = 1 \mid \boldsymbol{x}_i; \Theta \right) \right] \right] + \text{ const.}$$

$$\mathcal{L}_v = -\frac{1}{N \cdot c} \sum_{i \in [N]} \sum_{j \in [c]} \boldsymbol{q}_i^j \log \left(\psi_i \cdot (\eta_i \cdot \hat{\boldsymbol{y}}_i + (1 - \eta_i) \cdot \tilde{\boldsymbol{y}}_i) \right)$$

Only update $\boldsymbol{v_i}$

Overall Procedure

Input: Training set $\{(\boldsymbol{x}_i, \tilde{\boldsymbol{y}}_i)\}_{i=1}^{N}$; training steps T; estimation step list $\mathcal{T}$.

Output: Optimized parameters θ

Initialize $\eta_i = 0, \forall i \in [N]$

Initialize $\boldsymbol{v}_i = \mathbf{1}, \forall i \in [N]$

For $t = 1$ to T **do**

    Estimate the true label as $\boldsymbol{q}_i = \boldsymbol{v}_i \cdot (\eta_i \cdot \hat{\boldsymbol{y}}_i + (1 - \eta_i) \cdot \tilde{\boldsymbol{y}}_i)$.

    Calculate the loss terms.

    Update θ.

    If $t \in \mathcal{T}$ **then**

        Estimate $\eta_i, \forall i \in [N]$

    End if

End for

Overall Procedure

- Classification loss with refined label:

$$\mathcal{L}_c = \frac{1}{N} \sum_{i \in [N]} \text{CrossEntropy}\left(\boldsymbol{q}_i, \hat{\boldsymbol{y}}_i\right)$$

- Evidence lower bound term for updating $\boldsymbol{v_i}$:

$$\mathcal{L}_v = -\frac{1}{N \cdot c} \sum_{i \in [N]} \sum_{j \in [c]} \boldsymbol{q}_i^j \log\left(\psi_i \cdot \left(\eta_i \cdot \hat{\boldsymbol{y}}_i + (1 - \eta_i) \cdot \tilde{\boldsymbol{y}}_i\right)\right)$$

- Regularization terms.

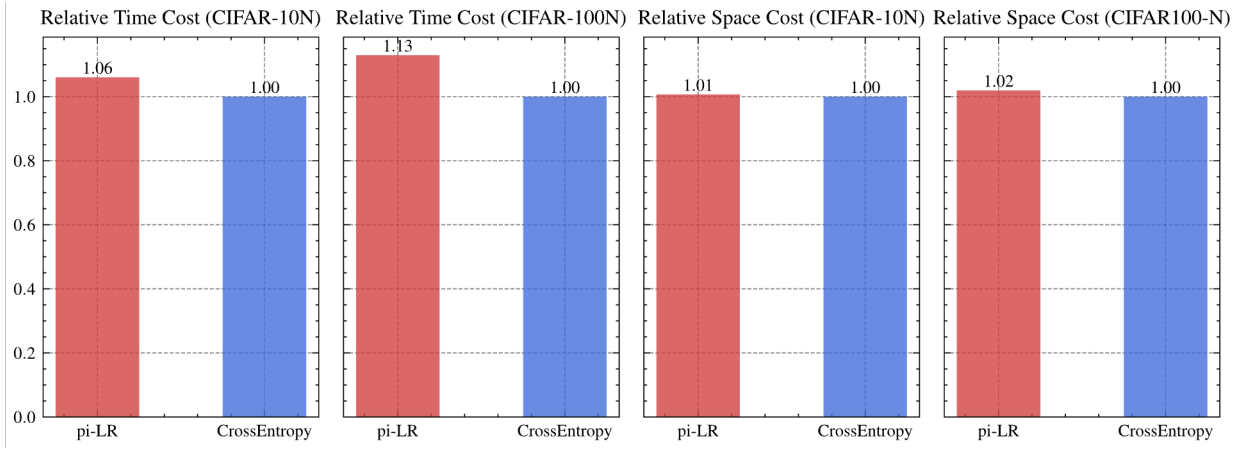
Optimization Objects

Input: Training set $\{(\boldsymbol{x}_i, \tilde{\boldsymbol{y}}_i)\}_{i=1}^N$; training steps T; estimation step list $\mathcal{T}$.
Output: Optimized parameters θ
Initialize $\eta_i = 0, \forall i \in [N]$
Initialize $\boldsymbol{v}_i = \mathbf{1}, \forall i \in [N]$
For $t = 1$ to T **do**
  Estimate the true label as $\boldsymbol{q}_i = \boldsymbol{v}_i \cdot (\eta_i \cdot \hat{\boldsymbol{y}}_i + (1 - \eta_i) \cdot \tilde{\boldsymbol{y}}_i)$.
  Calculate the loss terms.
  Update θ.
  If $t \in \mathcal{T}$ **then**
    Estimate $\eta_i, \forall i \in [N]$
  End if
End for

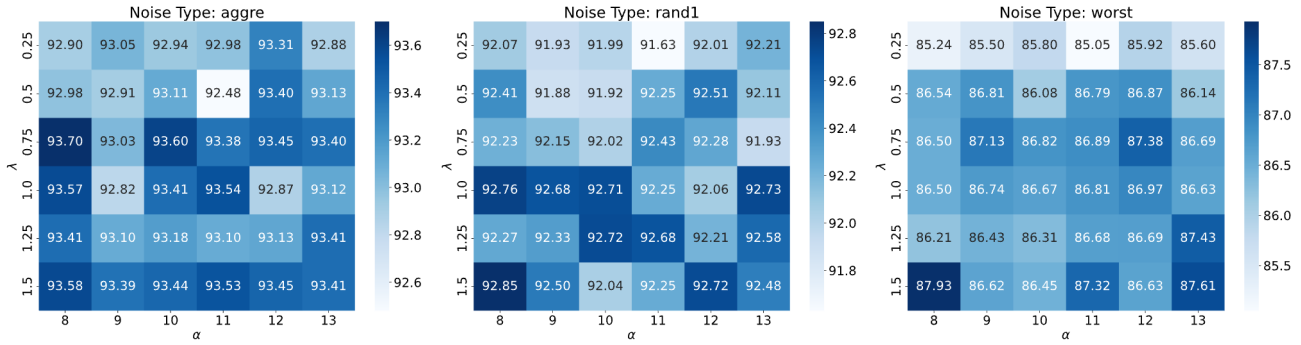
Empirical Studies

Methods	Random 1	Random 2	Random 3	Aggregate	Worst	Noisy
CE	85.02±0.65	86.46±1.79	85.16±0.61	87.77±0.38	77.69±1.55	55.50±0.66
Forward	86.88±0.50	86.14±0.24	87.04±0.35	88.24±0.22	79.79±0.46	57.01±1.03
Backward	87.14±0.34	86.28±0.80	86.86±0.41	88.13±0.29	77.61±1.05	57.14±0.92
GCE	87.61±0.28	87.70±0.56	87.58±0.29	87.85±0.70	80.66±0.35	56.73±0.30
Peer Loss	89.06±0.11	88.76±0.19	88.57±0.09	90.75±0.25	82.53±0.52	57.59±0.61
VolMinNet	88.30±0.12	88.27±0.09	88.19±0.41	89.70±0.21	80.53±0.20	57.80±0.31
F-div	89.70±0.40	89.79±0.12	89.55±0.49	91.64±0.34	82.53±0.52	57.10±0.65
ELR	91.46±0.38	91.61±0.16	91.41±0.44	92.38±0.64	83.58±1.13	58.94±0.92
PM	85.12±2.90	87.55±0.13	84.83±3.18	88.78±0.95	84.83±1.13	29.87±0.08
CAL	90.93±0.31	90.75±0.30	90.74±0.24	91.97±0.32	85.36±0.16	61.73±0.42
CORES	89.66±0.32	89.91±0.45	89.79±0.50	91.23±0.11	83.60±0.53	61.15±0.73
π-LR (Ours)	92.02±0.32	91.96±0.28	92.09±0.12	92.99±0.24	86.76±0.42	62.73±0.46

Robust against realistic label noise



Low time and space overhead



Insensitive to hyper-parameters

π-LR outperforms compared SOTA methods and keeps efficient in both time and space.

Conclusions

- Take Home Message:
  - Estimating the confusing probability helps modeling IDN label noise.
  - π-LR shows robustness against realistic label noise and keeps efficient.

- Future Directions:
  - Better optimization process of the instance transition ratio $\boldsymbol{v_i}$
  - Expand to other weakly-supervised learning scenarios.

Thank you!

Q&A

Contact me: hehy@lamda.nju.edu.cn