Hyper-opinion Evidential Deep Learning for Out-of-Distribution Detection

Jingen Qu
School of Computer Science and Technology
Tongji University, Shanghai, China
newcity@tongji.edu.cn

Yufei Chen*
School of Computer Science and Technology
Tongji University, Shanghai, China
yufeichen@tongji.edu.cn

Xiaodong Yue
Artificial Intelligence Institute
Shanghai University, Shanghai, China.
yswantfly@shu.edu.cn

Wei Fu
School of Computer Science and Technology
Tongji University, Shanghai, China
cs_fuwei@outlook.com

Qiguang Huang
School of Computer Science and Technology
Tongji University, Shanghai, China
1753543@tongji.edu.cn

NeurIPS 2024

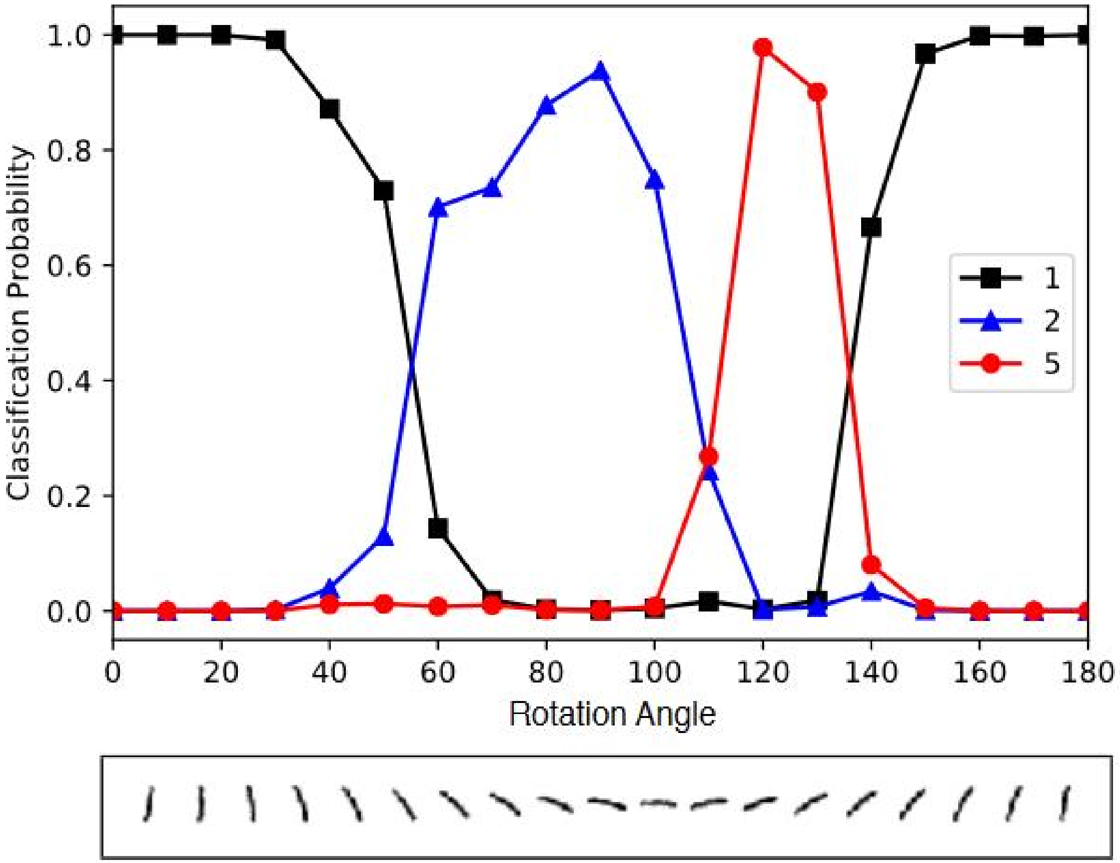
Motivation

- Deep learning models are trained under the implicit assumption that the training and test data are drawn from the same distribution.

- When a deep learning model encounters an input that differs from its training data, it may be overconfident with wrong prediction.

Out-of-distribution (OOD) detection

- Far OOD data
- Near OOD data



Motivation

- The post-hoc methods
  - ✓ MSP, ODIN, GradNorm.
  - ✓ Only take effect at inference phase and are easy to use, but rely on the performance of the pretrained model.
- The training methods
  - ✓ MC-dropout, G-ODIN, LogitNorm.
  - ✓ Require more computational resources.
- Evidential deep learning
  - ✓ Without additional computation.

Evidential Deep Learning to Quantify Classification Uncertainty

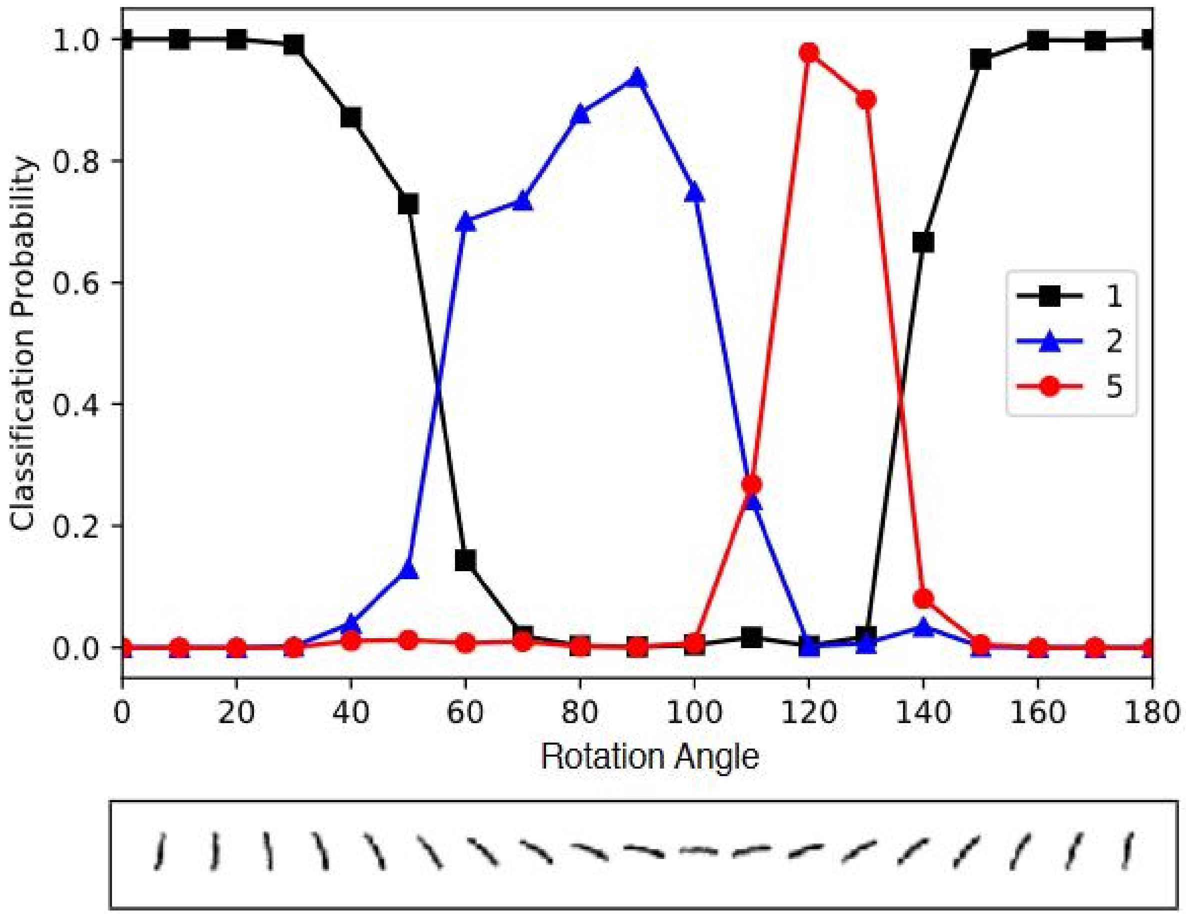
Murat Sensoy
Department of Computer Science
Ozyegin University, Turkey
murat.sensoy@ozyegin.edu.tr

Lance Kaplan
US Army Research Lab
Adelphi, MD 20783, USA
lkaplan@ieee.org

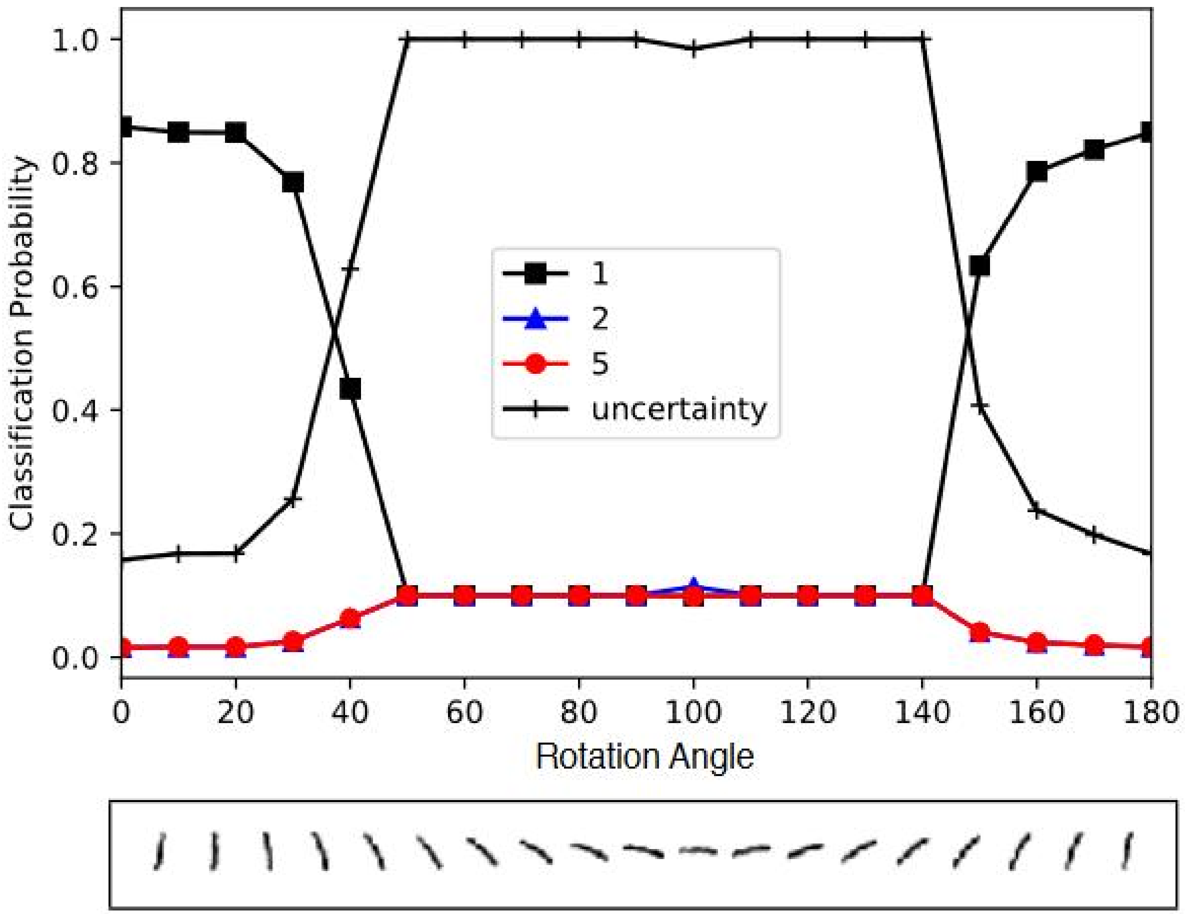
Melih Kandemir
Bosch Center for Artificial Intelligence
Robert-Bosch-Campus 1, 71272 Renningen, Germany
melih.kandemir@bosch.com

NeurIPS 2018

Evidential Deep Learning Review



Softmax-based neural network



Evidential neural network

Evidential Deep Learning Review

- Consider a simple K-class classification network $f(x|\theta)$.

Belief mass $\boldsymbol{b} = (b_1, b_2, \cdots, b_K)$.

Uncertainty mass u.

$$u + \sum_{k=1}^{K} b_k = 1 \qquad u \geq 0 \quad b_k \geq 0$$

$$b_k = \frac{e_k}{S} \qquad u = \frac{K}{S} \qquad S = \sum_{i=1}^{K}(e_i + 1)$$

Evidence is a measure of the amount of support collected from data in favor of a sample to be classified into a certain class.

$$\boldsymbol{e} = f(x|\theta)$$

The softmax layer is replaced with an activation layer, e.g., ReLU, to ascertain non-negative output.

Evidential Deep Learning Review

- Dirichlet distribution

$$D(\mathbf{p}|\boldsymbol{\alpha}) = \begin{cases} \frac{1}{B(\boldsymbol{\alpha})}\prod_{i=1}^{K} p_i^{\alpha_i - 1} & \text{for } \mathbf{p} \in \mathcal{S}_K, \\ 0 & \text{otherwise,} \end{cases}$$

$$\mathcal{S}_K = \left\{\mathbf{p} \middle| \sum_{i=1}^{K} p_i = 1 \text{ and } 0 \leq p_1, \ldots, p_K \leq 1\right\} \qquad \boldsymbol{\alpha} = [\alpha_1, \cdots, \alpha_K]$$

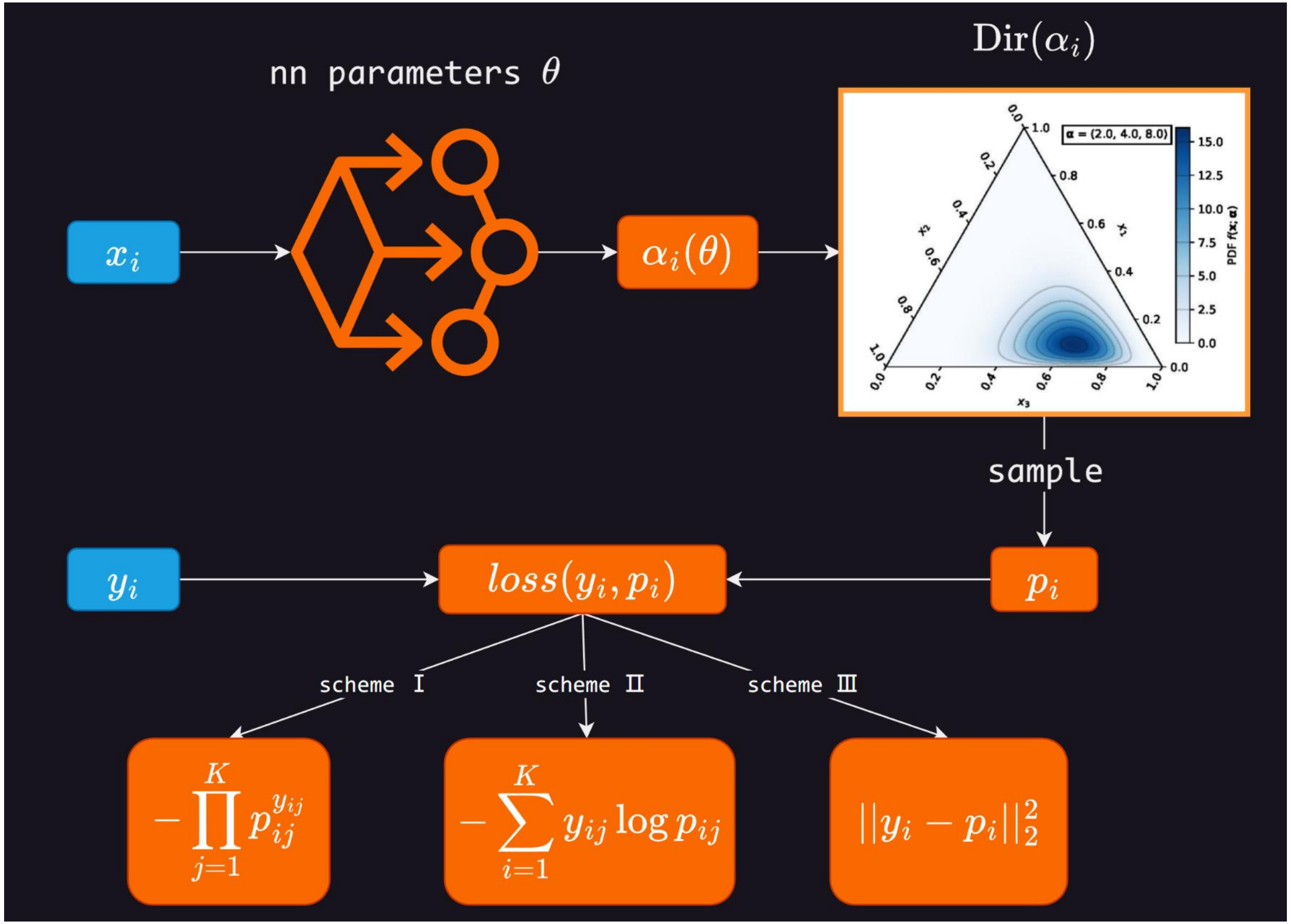
- The Dirichlet strength:

$$S = \sum_{i=1}^{K} \alpha_i$$

$$\updownarrow \quad \alpha_i = e_i + 1$$

$$b_k = \frac{e_k}{S} \qquad u = \frac{K}{S} \qquad S = \sum_{i=1}^{K}(e_i + 1)$$

Evidential Deep Learning Review



- Sampling operation is non-differentiable.
- Sampling involves randomness, making it difficult to accurately measure the obtained $\boldsymbol{\alpha_i}(\boldsymbol{\theta})$.
- The expected probability for the k-th singleton is computed as:

$$\hat{p}_k = \frac{\alpha_k}{S}$$

https://www.zhihu.com/search?type=content&q=evidential%20deep%20learning

Evidential Deep Learning Review

$$-\prod_{j=1}^{K} p_{ij}^{y_{ij}} \quad \rightarrow \quad -\mathop{\mathbb{E}}_{p_i \sim \mathrm{Dir}(\alpha_i)} \left[\prod_{i=1}^{K} p_{ij}^{y_{ij}} \right]$$

$$\begin{aligned}
\mathcal{L}_i(\theta) &= -\log\left(\mathop{\mathbb{E}}_{p_i \sim \mathrm{Dir}(\alpha_i)} \left[\prod_{i=1}^{K} p_{ij}^{y_{ij}} \right] \right) \\
&= -\log\left(\int \prod_{i=1}^{K} p_{ij}^{y_{ij}} \frac{1}{B(\alpha_i)} \prod_{i=1}^{K} p_{ij}^{\alpha_{ij}-1} dp_i \right) \\
&= -\log\left(\frac{B(\alpha_i + y_i)}{B(\alpha_i)} \int \underbrace{\frac{1}{B(\alpha_i + y_i)} \prod_{i=1}^{K} p_{ij}^{\alpha_{ij}+y_{ij}-1} dp_i}_{D(p_i|\alpha_i + y_i)} \right) \\
&= -\log \frac{B(\alpha_i + y_i)}{B(\alpha_i)}
\end{aligned}$$

$$\begin{aligned}
&= -\log \frac{\prod_{j=1}^{K} \Gamma(\alpha_{ij} + y_{ij})}{\Gamma\left(\sum_{j=1}^{K} (\alpha_{ij} + y_{ij})\right)} \frac{\Gamma\left(\sum_{j=1}^{K} \alpha_{ij}\right)}{\prod_{j=1}^{K} \Gamma(\alpha_{ij})} \\
&= -\log\left(\frac{\Gamma(\alpha_{ik} + 1)}{\Gamma(\alpha_{ik})} \frac{\Gamma(\sum_{j=1}^{K} \alpha_{ij})}{\Gamma(1 + \sum_{j=1}^{K} \alpha_{ij})} \right)\Bigg|_{k:y_{ik}=1} = -\log\left(\frac{\alpha_{ik}}{\underbrace{\sum_{j=1}^{K} \alpha_{ij}}_{S_i}} \right)\Bigg|_{k:y_{ik}=1} \\
&= y_{ik} \left(\log S_i - \log \alpha_{ik} \right)\big|_{k:y_{ik}=1} \\
&= \sum_{j=1}^{K} y_{ij} \left(\log S_i - \log \alpha_{ij} \right)
\end{aligned}$$

$\Gamma(\cdot)$ is the *gamma* function

Evidential Deep Learning Review

- Hope the evidence generation process to collapse under specific conditions, directly yielding a result with all evidence values set to zero.

$$u = \frac{K}{\sum_{i=1}^{K}(e_i + 1)} = \frac{K}{\sum_{i=1}^{K}(0 + 1)} = 1$$

$$\alpha_i = e_i + 1 = 1 \longrightarrow \sum_{i=1}^{N} \mathrm{KL}\left(D(p_i|\alpha_i)||D(p_i|\mathbf{1})\right)$$

$$\downarrow$$

$$D(p_i|\tilde{\alpha}_i), \tilde{\alpha}_i = y_i + (1 - y_i) \odot \alpha_i$$

$$\downarrow$$

$$\mathcal{L}(\theta) = \sum_{i=1}^{N} \mathcal{L}_i(\theta) + \lambda_t \sum_{i=1}^{N} \mathrm{KL}\left(D(p_i|\tilde{\alpha}_i)||D(p_i|\mathbf{1})\right) \qquad \lambda_t = \min(1, t/10)$$

Evidential Deep Learning Review

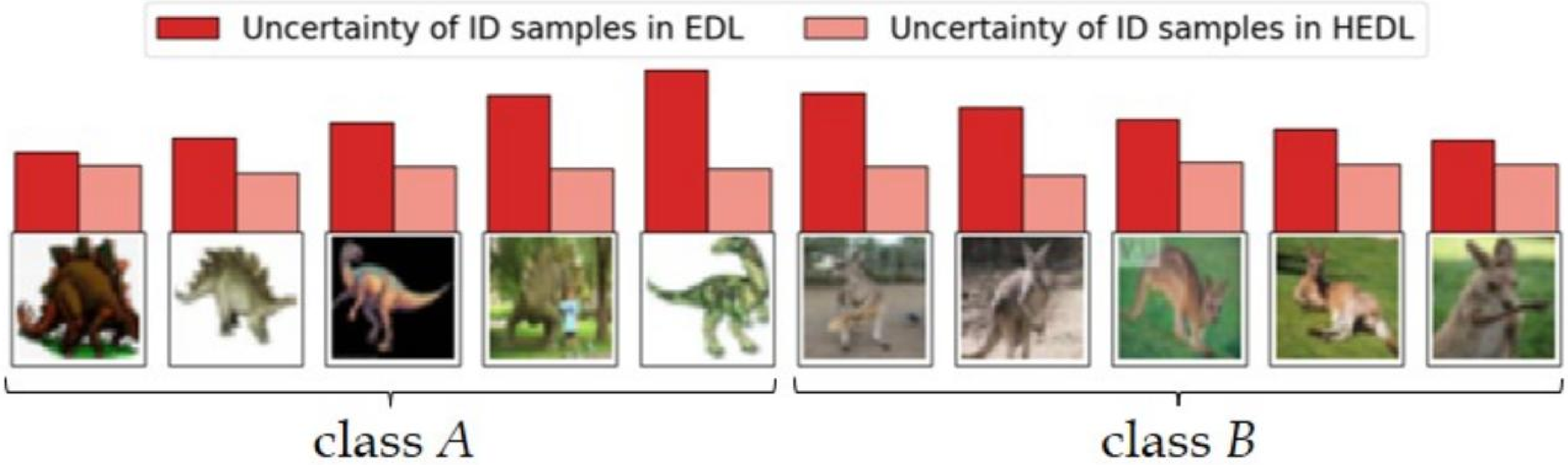
$$KL[D(\mathbf{p}_i|\tilde{\boldsymbol{\alpha}}_i) \mid\mid D(\mathbf{p}_i|\mathbf{1})] = \log\left(\frac{\Gamma(\sum_{k=1}^{K}\tilde{\alpha}_{ik})}{\Gamma(K)\prod_{k=1}^{K}\Gamma(\tilde{\alpha}_{ik})}\right) + \sum_{k=1}^{K}(\tilde{\alpha}_{ik}-1)\left[\psi(\tilde{\alpha}_{ik}) - \psi\Big(\sum_{j=1}^{K}\tilde{\alpha}_{ij}\Big)\right]$$

$\Gamma(\cdot)$ is the *gamma* function

$\psi(\cdot)$ is the *digamma* function

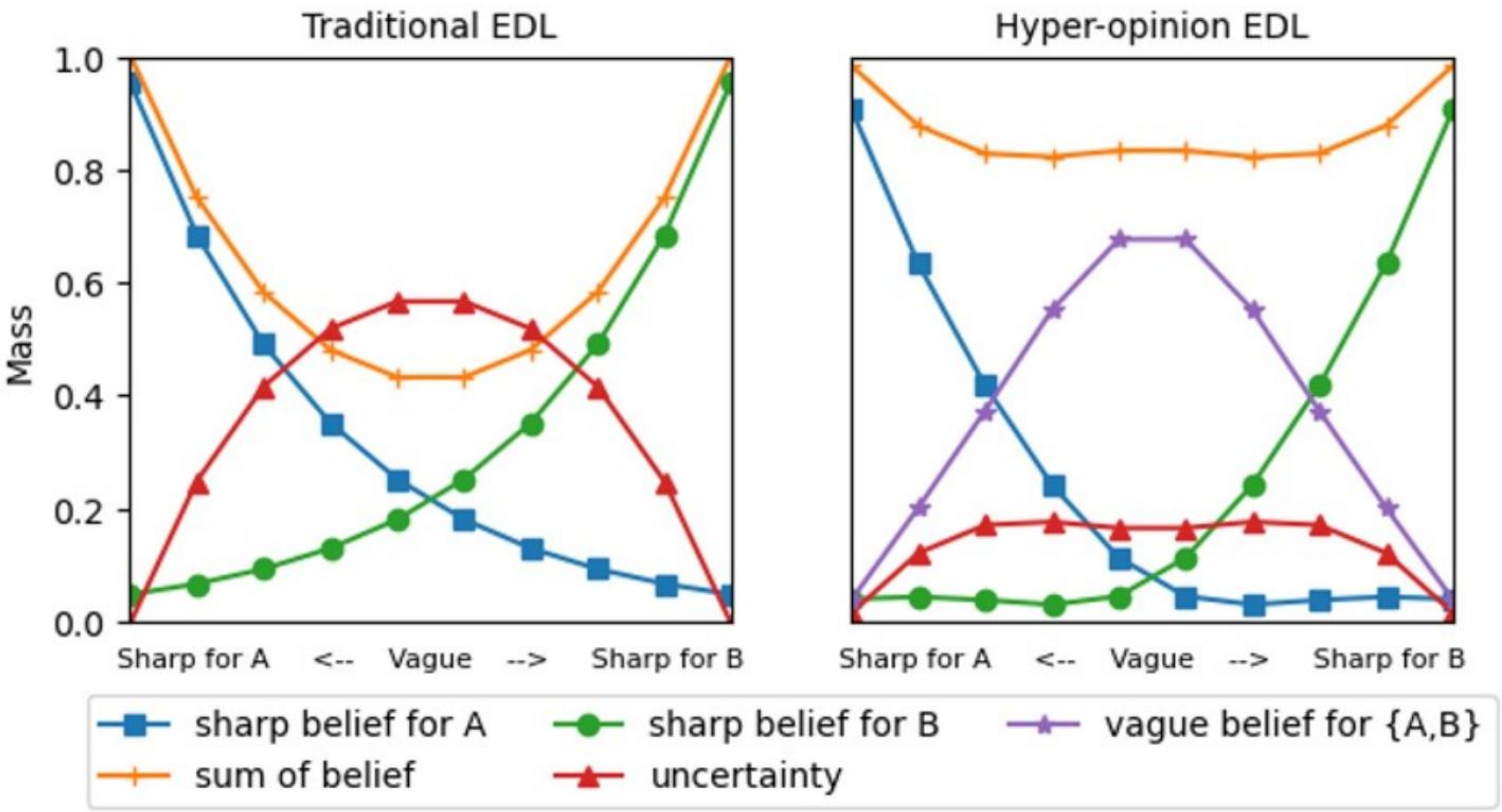
Motivation

- EDL only captures the evidence which supports single category and rejects others. As a result, EDL is unable to effectively leverage vague evidence, such as features supporting a composite set containing multiple categories.



- The parameters of fully-connected layer in EDL models are facing vanishing gradient problem when number of category in datasets rises.

Hyper-opinion Evidential Deep Learning



Hyper-opinion Evidential Deep Learning

- Opinion or multinomial-opinion

Domain $\mathbb{X} = \{1,\ 2,\ \cdots,\ K\}$ → Opinion
- Belief masses $\langle b_1, b_2, \cdots, b_K \rangle$
- Uncertainty mass u

- Hyper-opinion

Powerset ↓

Hyper-domain $R(\mathbb{X}) = P(\mathbb{X})\ /\ \{\mathbb{X},\ \emptyset\}$

↓

Hyper-opinion
$$\begin{cases} b^H : \mathcal{R}(\mathbb{X}) \rightarrow [0,1] \\ u + \sum_{x \in \mathcal{R}(\mathbb{X})} b^H_x = 1. \end{cases}$$

$$\mathbb{X} = \{1,\ 2,\ 3\}$$

$$P(\mathbb{X}) = \begin{Bmatrix} \emptyset, \{1\}, \{2\}, \{3\}, \\ \{1,\ 2\}, \{1,\ 3\}, \{2,\ 3\}, \\ \mathbb{X} = \{1,\ 2,\ 3\} \end{Bmatrix}$$

$$R(\mathbb{X}) = \begin{Bmatrix} \{1\}, \{2\}, \{3\}, \\ \{1,\ 2\}, \{1,\ 3\}, \{2,\ 3\} \end{Bmatrix}$$

Sharp belief masses

Vague belief masses

Hyper-opinion Evidential Deep Learning

- For each singleton $k \in \mathbb{X}$, sharp belief mass is

$$b_k^{\mathrm{S}} = b_k^H \ , \forall k \in \mathbb{X}.$$

- Allocate vague belief masses to each singleton $k \in \mathbb{X}$

$$b_k^{\mathrm{V}} = \sum_{x \in \mathcal{C}(\mathbb{X})} a(k|x) b_x^H, \quad a(k|x) = \frac{a_k}{\sum_{i \in x} a_i}, \quad \forall k \in \mathbb{X}, \forall x \in \mathcal{C}(\mathbb{X}) = \mathcal{R}(\mathbb{X})/\mathbb{X}$$

where $a(k|x)$ is relative base rate. When no prior information is available, $a(k|x)$ can be simplified to

$$a(k|x) = \frac{1}{|x|}, \quad \forall k \in \mathbb{X}, \forall x \in \mathcal{C}(\mathbb{X}),$$

where $|x|$ is the cardinality of x.

$\mathcal{C}(\mathbb{X}) = \{\{1, 2\}, \{1, 3\}, \{2, 3\}\}$ $\quad a(1|\{1, 2\}) = a(2|\{1, 2\}) = \cdots = a(3|\{2, 3\}) = \frac{1}{2}$

$b_{\{1,2\}}^H \quad b_{\{1,3\}}^H \quad b_{\{2,3\}}^H$

$b_1^V = \frac{1}{2} b_{\{1,2\}}^H + \frac{1}{2} b_{\{1,3\}}^H$

Hyper-opinion Evidential Deep Learning

- The belief b_x^H and the uncertainty u are computed as

$$b_x^H = \frac{e_x^H}{S} \quad \text{and} \quad u = \frac{KW_{prior}}{S}, \quad S = \sum_{x \in \mathcal{R}(\mathbb{X})} e_x^H + KW_{prior}$$

$$u = \frac{|R(\mathbb{X})|}{S} \qquad S = \sum_{x \in R(\mathbb{X})} (e_x^H + 1)$$

- Opinion projection

$$b_k = b_k^{\mathrm{V}} + b_k^{\mathrm{S}}, \forall k \in \mathbb{X}.$$

$$\alpha_k = b_k S + 1$$

Hyper-opinion Evidential Deep Learning

Proof 1. Consider the neural network forward propagation in EDL

$$o_k = W\boldsymbol{z} + bias, \tag{18}$$

$$e_k = ReLU(o_k), \tag{19}$$

$$\alpha_k = e_k + \frac{W_{prior}}{K}, \tag{20}$$

$$\mathcal{L}_i(\Theta) = \sum_{j=1}^{K} y_{ij}\Big(\psi(S_i) - \psi(\alpha_{ij})\Big), \tag{21}$$

where $bias$ stands for the bias of the fully-connected layer, $\boldsymbol{z}$ represents the feature extracted by the neural network. We can write expressions for all partial derivatives as follows:

$$\frac{\partial o_k}{\partial W} = \boldsymbol{z}, \quad \frac{\partial \alpha_k}{\partial e_k} = 1, \tag{22}$$

$$\frac{\partial \mathcal{L}}{\partial \alpha_k} = \left(\frac{1}{S^2} + \sum_{i=1}^{\infty} \frac{1}{(i+S)^2} - \frac{y_k}{\alpha_{gt}^2} - \sum_{i=1}^{\infty} \frac{y_k}{(i+\alpha_{gt})^2}\right), \tag{23}$$

$$\frac{\partial e_k}{\partial o_k} = \begin{cases} 0 & \text{if} \quad o_k \le 0 \\ 1 & \text{otherwise.} \end{cases} \tag{24}$$

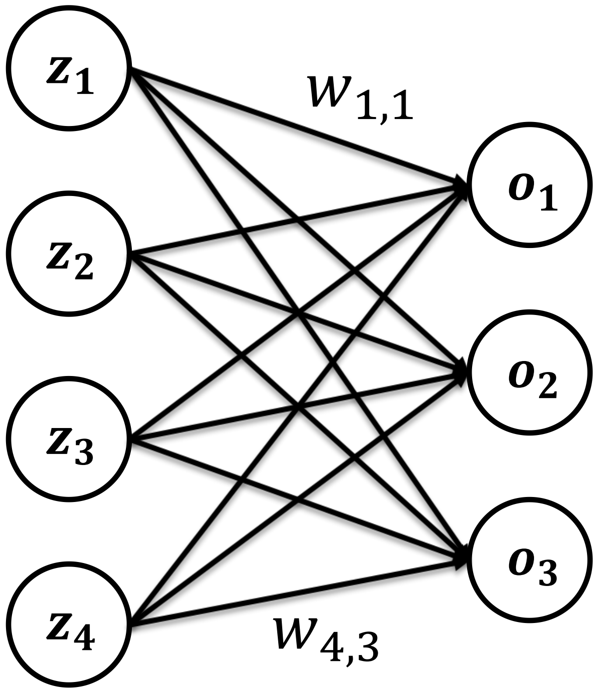
Therefore by the chain rule, we can calculatethe the gradient *w.r.t.* W as:

$$\frac{\partial \mathcal{L}}{\partial W} = \frac{\partial \mathcal{L}}{\partial \alpha_k}\frac{\partial \alpha_k}{\partial e_k}\frac{\partial e_k}{\partial o_k}\frac{\partial o_k}{\partial W} = \frac{\partial \mathcal{L}}{\partial \alpha_k}\frac{\partial e_k}{\partial o_k}\boldsymbol{z}, \tag{25}$$

Hyper-opinion Evidential Deep Learning

Evidence is a measure of the amount of support collected from data in favor of a sample to be classified into a certain $x \in R(\mathbb{X})$.

- Activate the features extracted by neural network for ascertaining non-negative evidence within the hyper-domain.



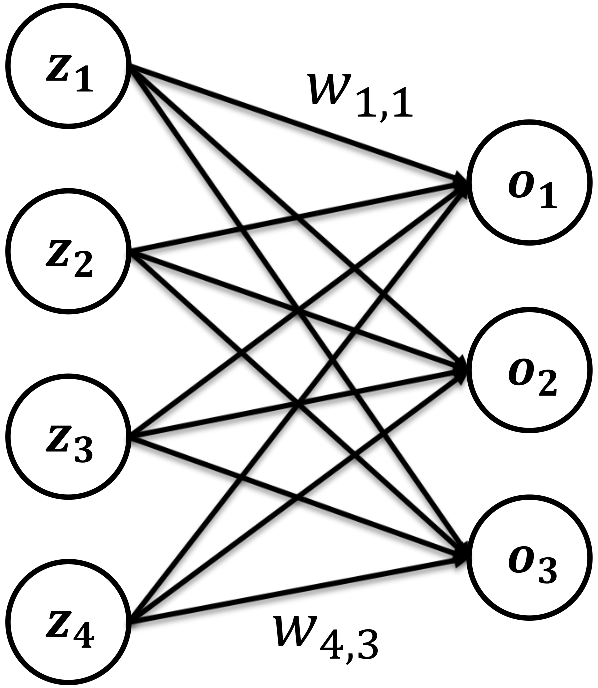
- Apply a unit step activation function to the parameters of the fully connected layer, eg., Heaviside function

$$H(x) = \begin{cases} 1, x > 0 \\ 0, else. \end{cases}$$

$$W \longrightarrow W^S = H(W)$$

W^S represents the information of set each belief mass supports.

Hyper-opinion Evidential Deep Learning



W^S	o_1	o_2	o_3
z_1	1	0	0
z_2	0	1	0
z_3	0	0	1
z_4	0	1	1

W_x^S

$W^S_{N,\ K}$	1	2	3
$b^H_{\{1\}}$	1	0	0
$b^H_{\{2\}}$	0	1	0
$b^H_{\{3\}}$	0	0	1
$b^H_{\{1,2\}}$	1	1	0
$b^H_{\{1,3\}}$	1	0	1
$b^H_{\{2,3\}}$	0	1	1

$$b_k = \sum_{x \in \mathcal{R}(\mathbb{X}} (b_x^H W^p_{x,k}),$$

$$W^p_{x,k} = \frac{a_k H(W_{x,k})}{\sum_{i=1}^{K}(a_i H(W_{x,i}))} = \frac{a_k W^S_{x,k}}{\sum_{i=1}^{K}(a_i W^S_{x,i})}, \quad k \in \mathbb{X}, x \in \mathcal{R}(\mathbb{X})$$

Hyper-opinion Evidential Deep Learning

- In practical terms, this projection is executed by applying a linear transformation to the output of the fully connected layer.

$$\boldsymbol{b} = \boldsymbol{o} \cdot G(W, \boldsymbol{b}^H), \quad G(W, \boldsymbol{b}^H) = \frac{W^p \boldsymbol{b}^H}{W \boldsymbol{b}^H}$$

To ensure that proposed HEDL is not associate with similar problem, considering the forward propagation of HEDL:

$$o_k = W\boldsymbol{z} + bias, \tag{26}$$

$$e_k = o_k G(W, \boldsymbol{b}^H), \tag{27}$$

$$\alpha_k = e_k + \frac{W_{prior}}{K}, \tag{28}$$

the gradient *w.r.t.* W is calculated by chain rule:

$$\frac{\partial \mathcal{L}}{\partial W} = \frac{\partial \mathcal{L}}{\partial \alpha_k} \frac{\partial \alpha_k}{\partial e_k} \frac{\partial e_k}{\partial o_k} \frac{\partial o_k}{\partial W},$$

Hyper-opinion Evidential Deep Learning

$$\frac{\partial o_k}{\partial W} = \boldsymbol{z}, \quad \frac{\partial \alpha_k}{\partial e_k} = 1, \tag{22}$$

$$\frac{\partial \mathcal{L}}{\partial \alpha_k} = \left(\frac{1}{S^2} + \sum_{i=1}^{\infty} \frac{1}{(i+S)^2} - \frac{y_k}{\alpha_{gt}^2} - \sum_{i=1}^{\infty} \frac{y_k}{(i+\alpha_{gt})^2} \right), \tag{23}$$

---

where $\frac{\partial \mathcal{L}}{\partial \alpha_k}, \frac{\partial \alpha_k}{\partial e_k}, \frac{\partial o_k}{\partial W}$ are known items that won't cause vanishing gradient problem. Consider

$$\frac{\partial e_k}{\partial o_k} = \frac{\partial o_k G(W, \boldsymbol{b}^H)}{\partial o_k} = G(W, \boldsymbol{b}^H), \tag{30}$$

where $W, \boldsymbol{b}^H$ are all detached variables that are irrelevant variables in this partial derivative item, implying that $G(W, \boldsymbol{b}^H)$ remains constant during the backward process.

$$\frac{\partial \mathcal{L}}{\partial W} = \frac{\partial \mathcal{L}}{\partial \alpha_k} G(W, \boldsymbol{b}^H) \boldsymbol{z}. \tag{31}$$

Hyper-opinion Evidential Deep Learning

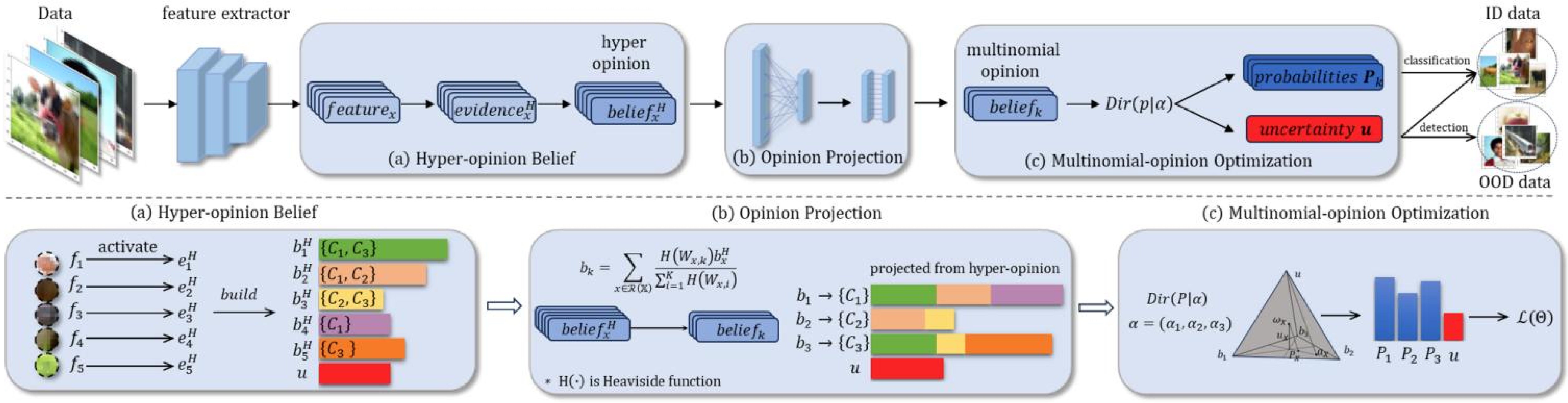


Figure 2: Framework of HEDL. HEDL framework is composed of three integral components. The first part transfers the extracted features to evidence and models them with in hyper-opinion framework. Subsequently, the second component projects the hyper-opinion to multinomial-opinion. Ultimately, the framework optimizes the opinion to attain precise classification and to furnish robust uncertainty estimations for OOD detection.

Experiments

Method	OOD Datasets												ID data
	SVHN			Textures			Place365			Average			
	FPR95↓	AUPR↑	AUROC↑	FPR95↓	AUPR↑	AUROC↑	FPR95↓	AUPR↑	AUROC↑	FPR95↓	AUPR↑	AUROC↑	Acc.↑
CIFAR-10													
MSP[16]	51.87	78.19	90.88	59.89	91.28	88.72	57.64	70.24	89.03	56.47	79.90	89.54	95.06
ODIN[31]	67.92	42.13	73.32	51.10	82.25	80.70	50.51	50.27	82.55	56.51	58.22	78.86	95.06
openGAN[26]	99.39	33.90	53.56	98.24	61.48	42.22	99.44	19.55	36.58	99.02	38.31	44.12	95.06
GradNorm[21]	91.65	78.89	53.91	98.09	48.05	52.07	92.46	86.63	60.50	94.07	71.19	55.49	95.06
VIM[66]	14.41	93.76	97.22	20.78	97.36	96.06	47.52	72.83	90.08	27.57	87.98	94.46	95.06
KNN[61]	33.32	92.31	95.13	46.01	95.93	92.77	43.78	80.15	91.82	41.04	89.47	93.23	95.06
DICE[59]	67.78	73.19	86.43	67.48	85.38	80.14	56.06	57.52	84.43	63.78	72.03	83.66	95.06
RankFeat[58]	64.49	80.33	68.15	59.71	55.39	73.46	43.70	94.66	85.99	55.97	76.79	75.87	95.06
ASH[8]	83.64	89.06	73.46	84.59	72.85	77.45	77.89	94.04	79.89	82.04	85.32	76.93	95.06
SHE[71]	62.74	94.46	86.38	84.60	77.28	81.57	76.36	94.88	82.89	74.57	88.87	83.61	95.06
GEN[38]	28.14	96.37	91.97	40.74	84.71	90.14	47.03	96.67	89.46	38.64	92.58	90.52	95.06
MCDropout[12]	44.58	85.03	92.67	56.60	91.74	88.83	56.20	67.20	88.43	52.47	81.32	89.98	94.95
G-ODIN[19]	8.42	96.63	98.41	23.32	96.03	94.51	39.80	75.49	91.10	23.84	89.39	94.67	94.70
CSI[62]	17.56	**97.75**	95.18	28.95	82.99	90.71	34.76	**96.38**	89.56	27.09	92.37	91.82	91.16
MOS[20]	90.85	70.55	51.09	85.56	90.89	52.91	71.74	78.67	74.15	82.71	80.03	59.38	94.83
VOS[9]	29.92	83.73	93.82	37.38	92.72	91.26	45.37	63.93	88.73	37.55	80.13	91.27	**95.82**
LogitNorm[67]	**5.30**	97.70	**98.86**	30.94	96.32	94.30	31.17	88.11	94.76	22.47	94.04	95.97	94.30
EDL[54]	11.56	88.60	93.92	19.95	99.07	95.70	19.36	93.15	**96.54**	16.96	93.61	95.39	95.72
RED[49]	65.75	29.85	61.30	86.49	71.56	28.06	72.37	19.83	51.16	74.87	40.41	46.84	95.80
HEDL(Ours)	8.43	94.09	96.86	**19.15**	**99.19**	**96.23**	**19.08**	90.14	95.71	**15.55**	**94.47**	**96.27**	95.66

Experiments

Method	OOD Datasets												ID data
	SVHN			Textures			Place365			Average			
	FPR95↓	AUPR↑	AUROC↑	FPR95↓	AUPR↑	AUROC↑	FPR95↓	AUPR↑	AUROC↑	FPR95↓	AUPR↑	AUROC↑	Acc.↑
CIFAR-100													
MSP[16]	83.69	60.76	76.04	83.83	85.24	76.93	81.24	62.39	79.44	82.91	69.46	77.47	77.25
ODIN[31]	89.76	52.36	71.08	78.37	86.67	79.39	81.27	60.85	79.83	83.13	66.62	76.77	77.25
openGAN[26]	83.96	60.85	78.68	86.31	80.18	73.53	88.37	38.87	70.15	86.21	59.96	74.12	77.25
GradNorm[21]	69.90	89.45	76.95	92.51	56.77	64.58	95.32	88.78	69.69	85.91	78.33	70.41	77.25
VIM[66]	82.79	72.82	81.20	**55.90**	92.15	87.41	83.85	56.24	75.76	74.18	73.74	81.46	77.25
KNN[61]	74.27	71.46	82.21	66.40	89.44	83.81	78.74	57.47	79.10	73.13	72.79	81.71	77.25
DICE[59]	79.93	65.95	79.97	80.53	85.41	77.70	80.75	62.76	80.18	80.40	71.37	79.28	77.25
RankFeat[58]	58.49	83.40	72.14	66.87	52.42	69.40	77.42	83.74	63.82	67.59	73.19	68.45	77.25
ASH[8]	46.00	**92.97**	85.60	61.27	68.97	80.72	62.95	91.48	78.76	56.74	84.47	81.69	77.25
SHE[71]	59.15	90.85	80.97	73.29	60.87	73.64	65.24	90.31	76.30	65.89	80.68	76.97	77.25
GEN[38]	55.45	90.36	81.41	61.23	64.52	78.74	**56.25**	**91.90**	80.28	57.64	82.26	80.14	77.25
MCDropout[12]	71.63	67.44	81.31	80.16	86.01	77.93	79.52	61.34	79.20	77.11	71.60	79.48	75.83
G-ODIN[19]	71.62	79.80	86.13	58.01	93.01	**88.35**	78.67	55.45	78.15	69.44	76.09	84.21	74.46
CSI[62]	67.21	91.76	80.24	90.51	51.46	62.22	69.41	88.16	70.99	75.71	77.13	71.15	61.60
MOS[20]	90.58	74.48	59.42	96.32	89.60	46.69	92.64	71.87	60.95	93.18	78.64	55.69	76.98
VOS[9]	98.62	56.36	68.99	94.54	76.20	68.33	97.81	43.20	68.21	96.99	58.59	68.51	77.20
LogitNorm[67]	79.16	75.57	83.03	87.06	79.08	71.53	80.20	63.10	79.84	82.14	72.58	78.13	76.34
EDL[54]	93.05	75.48	81.39	95.48	93.80	71.60	99.30	68.57	76.55	95.94	79.28	76.51	71.40
RED[49]	90.09	62.75	76.41	56.01	96.25	85.29	68.11	64.75	84.46	71.40	74.58	82.05	80.36
HEDL(Ours)	**39.56**	89.22	**93.46**	61.97	**96.85**	85.98	63.89	81.14	**89.32**	**55.14**	**89.07**	**89.59**	**80.40**

Experiments

南京航空航天大学
Nanjing University of Aeronautics and Astronautics

Table 3: Comparison of OOD detection performance between HEDL and other baselines with Flower-102 and CUB-200-2011 as ID dataset.

Method	Flower-102				CUB-200-2011			
	Average OOD performance			ID data	Average OOD performance			ID data
	FPR95↓	AUPR↑	AUROC↑	Acc.↑	FPR95↓	AUPR↑	AUROC↑	Acc.↑
MSP[16]	14.86	95.94	97.42	83.75	30.29	91.18	94.35	75.82
ODIN[31]	4.36	97.63	98.22	83.75	21.92	89.92	96.22	75.82
VIM[66]	6.34	96.70	97.94	83.75	6.71	97.27	98.26	75.82
GradNorm[21]	5.38	97.11	98.81	83.75	32.08	97.68	95.22	75.82
KNN[61]	18.45	88.83	95.30	83.75	14.35	88.63	97.40	75.82
DICE[59]	4.64	97.62	98.95	83.75	25.82	88.83	96.00	75.82
RankFeat[58]	96.57	76.62	60.98	83.75	74.68	83.38	71.09	75.82
ASH[8]	5.16	97.54	98.84	83.75	15.82	92.75	97.07	75.82
SHE[71]	11.69	93.96	97.79	83.75	22.94	96.14	96.18	75.82
GEN[38]	5.25	97.55	98.85	83.75	15.88	92.74	97.06	75.82
MCDropout[12]	14.77	96.22	97.41	83.98	42.46	87.08	91.76	75.83
G-ODIN[19]	56.92	69.88	82.12	24.30	29.51	85.13	93.85	66.74
VOS[9]	39.17	84.52	90.11	78.08	35.98	83.93	89.86	75.92
LogitNorm[67]	41.07	80.34	85.65	77.41	22.69	91.69	95.99	74.84
EDL[54]	100.00	66.95	67.23	66.84	98.03	71.80	75.27	59.87
RED[49]	95.87	80.10	76.45	**84.63**	36.01	94.58	94.89	**76.30**
HEDL(Ours)	**3.98**	**98.73**	**99.07**	84.13	**3.82**	**97.80**	**98.91**	74.62

Experiments

Table 2: Ablation experiment results on Flower-102 and CUB-200-2011. Results show that EDL fails to extract evidence fully. HEDL without projection can extract comprehensive evidence to distinguish ID and OOD samples but fails to classify ID categories. HEDL can further assign evidence correctly and obtain accurate classification.

			Flower-102				CUB-200-2011			
			Average OOD performance			ID data	Average OOD performance			ID data
Multinomial-opinion	Hyper-opinion	Opinion-projection	FPR95↓	AUPR↑	AUROC↑	Acc.↑	FPR95↓	AUPR↑	AUROC↑	Acc.↑
-	-	-	14.86	95.94	97.42	83.75	30.29	91.18	94.35	**75.82**
✓	-	-	100.00	66.95	67.23	66.84	98.03	71.80	75.27	59.87
✓	✓	-	11.90	95.83	97.61	81.40	9.32	91.57	97.82	52.30
✓	✓	✓	**3.98**	**98.73**	**99.07**	**84.13**	**3.82**	**97.80**	**98.91**	74.62

Experiments

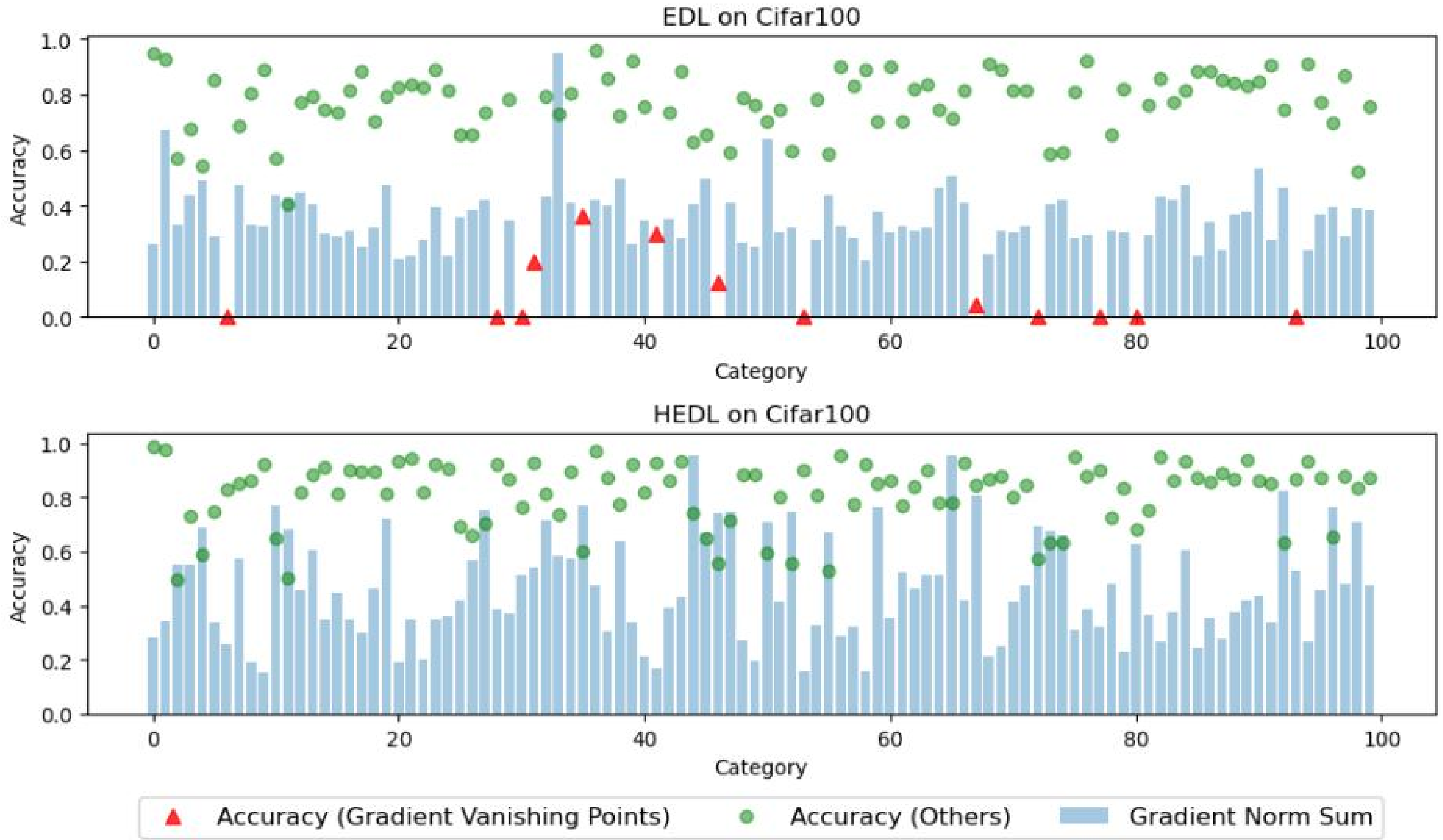
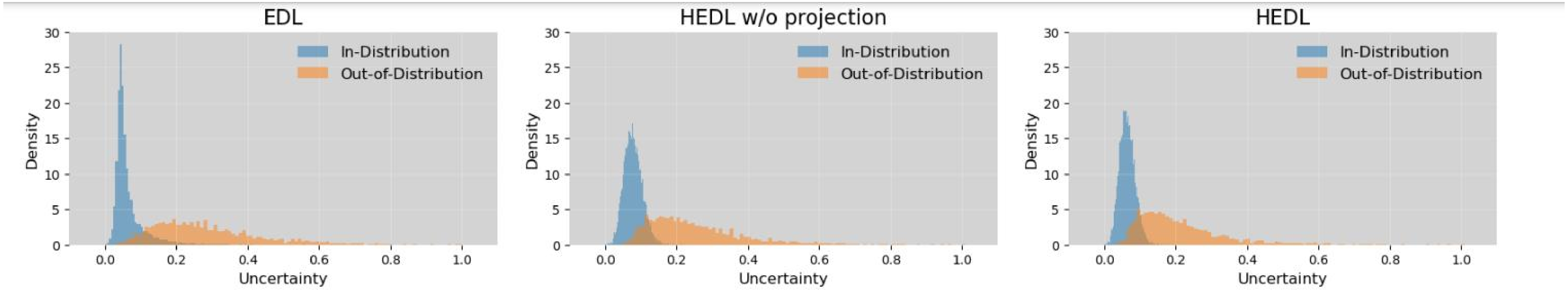
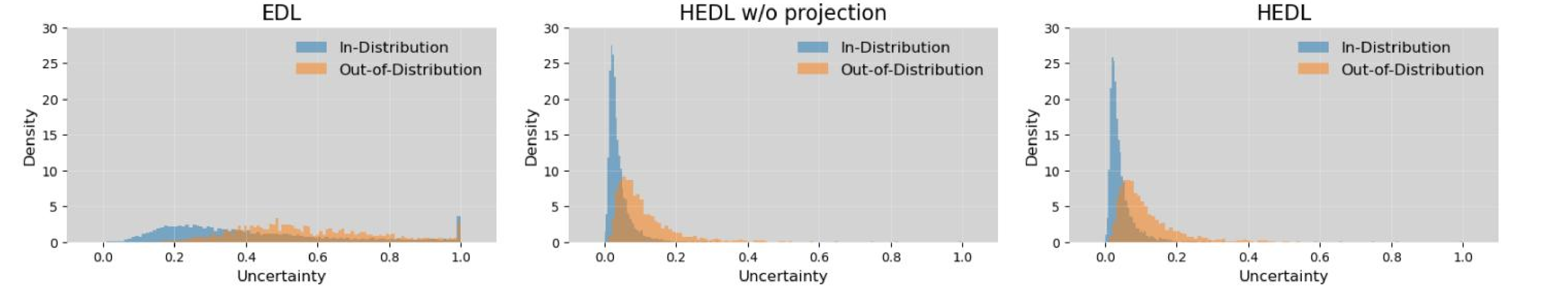


Figure 3: The sum of gradient norms within the fully-connected layer for each category in CIFAR-100 throughout the training process.

Experiments

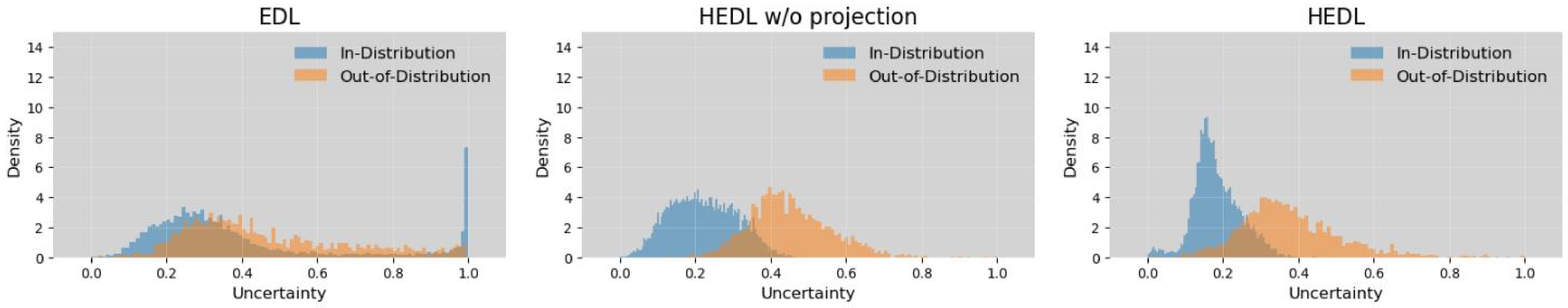


(a) CIFAR-10, the overlap between ID and OOD is 20%, 23%, and 18% for EDL, HEDL w/o projection, and HEDL, respectively.

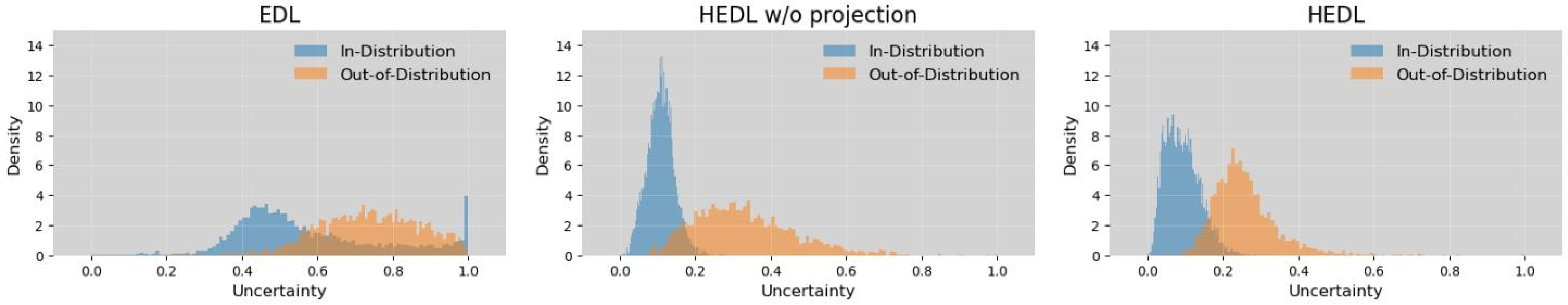


(b) CIFAR-100, the overlap between ID and OOD is 62%, 45%, and 41% for EDL, HEDL w/o projection, and HEDL, respectively.

Experiments



(c) Flower-102, the overlap between ID and OOD is 71%, 26%, and 29% for EDL, HEDL w/o projection, and HEDL, respectively.



(d) CUB-200-2011, the overlap between ID and OOD is 50%, 20%, and 17% for EDL, HEDL w/o projection, and HEDL, respectively.

Experiments

Table 4: Average training time per epoch of EDL and HEDL compared with MSP on different datasets, + indicates more time, and - indicates less time.

Method	Cifar10	Cifar100	Flower-102	CUB-200-2011
EDL	+3.78%	-1.93%	+1.21%	+2.14%
HEDL	+1.62%	+1.02%	-0.74%	+3.57%

THANKS