# Evidential Knowledge Distillation

Liangyu Xiang[1, 2] Junyu Gao[1, 2 †] Changsheng Xu[1, 2, 3]
[1]State Key Laboratory of Multimodal Artificial Intelligence Systems (MAIS),
Institute of Automation, Chinese Academy of Sciences (CASIA)
[2]School of Artificial Intelligence, University of Chinese Academy of Sciences (UCAS)
[3]Peng Cheng Laboratory, ShenZhen, China
xiangliangyu2023@ia.ac.cn; {junyu.gao, csxu}@nlpr.ia.ac.cn

## 1. Proof and Derivation

### 1.1. Proof of Theorem 1

For a single sample $x_i$, the student's risk yields the following:

$$\begin{aligned} R_i^S &= -\sum_{i=1}^{K} \mathbb{E}_{Dir(\boldsymbol{\alpha}_i^T)}[p_i^T] log \mathbb{E}_{Dir(\boldsymbol{\alpha}_i^S)}[p_i^S] \\ &= \mathbb{E}_{Dir(\boldsymbol{\alpha}_i^T)}[-\sum_{i=1}^{K} p_i^T log \mathbb{E}_{Dir(\boldsymbol{\alpha}_i^S)}[p_i^S]] \\ &= \mathbb{E}_{\boldsymbol{p}^T \sim Dir(\boldsymbol{\alpha}_i^T)}[R_i^S(\boldsymbol{p}^T)] \end{aligned} \tag{1}$$

In other words, we arrive at the following definition:

$$R_i^S(\boldsymbol{p}^T) = -\sum_{i=1}^{K} p_i^T log \mathbb{E}_{Dir(\boldsymbol{\alpha}_i^S)}[p_i^S] \tag{2}$$

The empirical empirical risk $\hat{R}^S$ of $N$ samples and the expected risk $R^S$ on the whole data distribution $D$ are defined as:

$$\hat{R}^S = \frac{1}{N}\sum_{i=1}^{N} R_i^S = \frac{1}{N}\sum_{i=1}^{N} \mathbb{E}_{\boldsymbol{p}^T \sim Dir(\boldsymbol{\alpha}_i^T)}[R_i^S(\boldsymbol{p}^T)] \tag{3}$$

$$R^S = \mathbb{E}_D\{-\sum_{i=1}^{K} \mathbb{E}_{Dir(\boldsymbol{\alpha}^T)}[p_i^T] log \mathbb{E}_{Dir(\boldsymbol{\alpha}^S)}[p_i^S]\} \tag{4}$$

It is noted that the expected risk $R^S$ is constant given a student, since the teacher is fixed and the randomness of the training set is integrated out in expectation.

Since $\sum_{i=1}^{N} \mathbb{E}_{\boldsymbol{p}^T \sim Dir(\boldsymbol{\alpha}_i^S)} e^{\frac{N}{\gamma}(R^S - \hat{R}_i^S(\boldsymbol{p}^T))}$ is a non-negative random variable related to the training set $D_t$, we can derive the following from Markov's inequality:

$$P_{D_t \sim D^N}(\sum_{i=1}^{N} \mathbb{E}_{\boldsymbol{p}^T \sim Dir(\boldsymbol{\alpha}_i^S)} e^{\frac{N}{\gamma}(R^S - \hat{R}_i^S(\boldsymbol{p}^T))} \leq \frac{1}{\delta}\mathbb{E}_{D_t \sim D^N}\sum_{i=1}^{N} \mathbb{E}_{\boldsymbol{p}^T \sim Dir(\boldsymbol{\alpha}_i^S)} e^{\frac{N}{\gamma}(R^S - \hat{R}_i^S(\boldsymbol{p}^T))}) \geq 1 - \delta \tag{5}$$

Taking the logarithm of both sides of the inner inequality and converting the integral variable, we obtain the following:

$$\begin{aligned} &P_{D_t \sim D^N}(log(\sum_{i=1}^{N} \mathbb{E}_{\boldsymbol{p}^T \sim Dir(\boldsymbol{\alpha}_i^T)}[\frac{Dir(\boldsymbol{\alpha}_i^S)}{Dir(\boldsymbol{\alpha}_i^T)} e^{\frac{N}{\gamma}(R^S - \hat{R}_i^S(\boldsymbol{p}^T))}]) \\ &\leq log(\frac{1}{\delta}\mathbb{E}_{D_t \sim D^N}\sum_{i=1}^{N} \mathbb{E}_{\boldsymbol{p}^T \sim Dir(\boldsymbol{\alpha}_i^S)} e^{\frac{N}{\gamma}(R^S - \hat{R}_i^S(\boldsymbol{p}^T))}) \geq 1 - \delta \end{aligned} \tag{6}$$

Since $log(\frac{1}{\delta}\mathbb{E}_{D_t \sim D^N}\sum_{i=1}^{N} \mathbb{E}_{\boldsymbol{p}^T \sim Dir(\boldsymbol{\alpha}_i^S)} e^{\frac{N}{\gamma}(R^S - \hat{R}^S(\boldsymbol{p}^T))})$ is determined by $\delta$, it can be represented as $C(\delta)$. Applying Jensen's inequality to the concave function $log(x)$, we have:

$$\begin{aligned} P_{D_t \sim D^N}(&\sum_{i=1}^{N} \mathbb{E}_{\boldsymbol{p}^T \sim Dir(\boldsymbol{\alpha}_i^T)}[log(\frac{Dir(\boldsymbol{\alpha}_i^S)}{Dir(\boldsymbol{\alpha}_i^T)}) \\ &+ \frac{N}{\gamma}(R^S - \hat{R}_i^S(\boldsymbol{p}^T))] \leq C(\delta)) \geq 1 - \delta \end{aligned} \tag{7}$$

As a result, the PAC bound of EKD is derived:

$$\begin{aligned} &P_{D_t \sim D^N}(R^S \leq \hat{R}^S + \frac{\gamma}{N}\sum_{i=1}^{N} KL(Dir(\boldsymbol{\alpha}_i^T)|Dir(\boldsymbol{\alpha}_i^S)) \\ &+ C(\delta)) \geq 1 - \delta \end{aligned} \tag{8}$$

With $\delta$ fixed, the student's expected risk is bounded above by its empirical risk and the divergence between its second-order distribution and that of the teacher.

### 1.2. Derivation of objectives in EKD

In EKD, three optimization objectives are utilized: the cross-entropy objective, the first-order distillation objective,

†Corresponding author.

and the second-order distillation objective. This section presents a detailed derivation of each of these objectives.

In evidential deep learning, the model's predictions take the form of a second-order Dirichlet distribution. Based on this, the classic cross-entropy loss is integrated over category probabilities, resulting in the following evidential cross-entropy:

$$\begin{aligned}\mathcal{L}_{EDL-ce} &= \int \left[\sum_{i=1}^{K} -y_i \log(p_i)\right] Dir(\boldsymbol{p}|\boldsymbol{\alpha}) d\boldsymbol{p} \\ &= \int \sum_{i=1}^{K} -y_i \log(p_i) \frac{\Gamma(\alpha_0)}{\prod_{i=1}^{K}\Gamma(\alpha_i)} \prod_{i=1}^{K} p_i^{\alpha_i - 1} d\boldsymbol{p} \\ &= \int \sum_{i=1}^{K} -y_i \frac{\Gamma(\alpha_0)}{\prod_{i=1}^{K}\Gamma(\alpha_i)} (\log(p_i) \prod_{i=1}^{K} p_i^{\alpha_i - 1}) d\boldsymbol{p}\end{aligned} \tag{9}$$

According to the form of the Dirichlet distribution, the marginal probability distribution of $p_i$ is $Beta(\alpha_i, \alpha_0 - \alpha_i)$. As a result, the following expression is derived:

$$\begin{aligned}\mathbb{E}[log(p_i)] &= \int log(p_i) p_i^{\alpha_i - 1}(1-p_i)^{\alpha_0 - \alpha_i - 1} dp_i \\ &= \psi(\alpha_i) - \psi(\alpha_0)\end{aligned} \tag{10}$$

Combining the above two equations, we arrive the final expression:

$$\begin{aligned}\mathcal{L}_{EDL-ce} &= \int \sum_{i=1}^{K} -y_i \mathbb{E}[log(p_i)] dp_i \\ &= \sum_{i=1}^{K} y_i \left(\psi(\alpha_0) - \psi(\alpha_i)\right)\end{aligned} \tag{11}$$

Before introducing the first-order distillation objective, we first provide the expression for the expectation of the second-order Dirichlet distribution. Let $\hat{\boldsymbol{p}}^T = (\hat{p}_1^T, \hat{p}_2^T, ..., \hat{p}_K^T)$ denote the expectation of the teacher's Dirichlet distribution. Then $\hat{p}_i^T$ is calculated as: $\hat{p}_i^T = \frac{\alpha_i^T}{\alpha_0^T}$. For the student, a similar result holds: $\hat{p}_i^S = \frac{\alpha_i^S}{\alpha_0^S}$. The first-order distillation objective can be derived by applying the Kullback-Leibler (KL) divergence to $\hat{\boldsymbol{p}}^T$ and $\hat{\boldsymbol{p}}^S$:

$$\begin{aligned}\mathcal{L}_{1st} &= KL(\hat{\boldsymbol{p}}^T || \hat{\boldsymbol{p}}^S) \\ &= \sum_{i=1}^{K} \hat{p}_i^T \log(\frac{\hat{p}_i^T}{\hat{p}_i^S}) \\ &= \sum_{i=1}^{K} \frac{\alpha_i^T}{\alpha_0^T} log(\frac{\alpha_i^T \alpha_0^S}{\alpha_0^T \alpha_i^S})\end{aligned} \tag{12}$$

The second-order distillation objective also uses the KL divergence, but it is applied directly to the Dirichlet distributions of the teacher and student. The detailed derivation is as follows:

$$\begin{aligned}\mathcal{L}_{2nd} &= KL(Dir(\boldsymbol{p}^T|\boldsymbol{\alpha}^T)||Dir(\boldsymbol{p}^S|\boldsymbol{\alpha}^S)) \\ &= \int Dir(\boldsymbol{\alpha}^T) log(\frac{Dir(\boldsymbol{\alpha}^T)}{Dir(\boldsymbol{\alpha}^S)}) d\boldsymbol{p} \\ &= \int \frac{1}{\boldsymbol{B}(\boldsymbol{\alpha}^T)} \prod_{i=1}^{K} p_i^{\alpha_i^T - 1} \log(\frac{\frac{1}{\boldsymbol{B}(\boldsymbol{\alpha}^T)} \prod_{i=1}^{K} p_i^{\alpha_i^T - 1}}{\frac{1}{\boldsymbol{B}(\boldsymbol{\alpha}^S)} \prod_{i=1}^{K} p_i^{\alpha_i^S - 1}}) d\boldsymbol{p} \\ &= \frac{1}{\boldsymbol{B}(\boldsymbol{\alpha}^T)} \int \prod_{i=1}^{K} p_i^{\alpha_i^T - 1} [\log(\frac{\Gamma(\alpha_0^T)}{\Gamma(\alpha_0^S)}) \\ &- \sum_{i=1}^{K} \log(\frac{\Gamma(\alpha_i^T)}{\Gamma(\alpha_i^S)}) + \sum_{i=1}^{K} (\alpha_i^T - \alpha_i^S) log(p_i)] d\boldsymbol{p} \\ &= \log(\frac{\Gamma(\alpha_0^T)}{\Gamma(\alpha_0^S)}) - \sum_{i=1}^{K} log(\frac{\Gamma(\alpha_i^T)}{\Gamma(\alpha_i^S)}) \\ &+ \frac{1}{\boldsymbol{B}(\boldsymbol{\alpha}^T)} \int \sum_{i=1}^{K} (\alpha_i^T - \alpha_i^S) \log(p_i) \prod_{i=1}^{K} p_i^{\alpha_i^T - 1} d\boldsymbol{p} \\ &= log \frac{\Gamma(\alpha_0^T)}{\Gamma(\alpha_0^S)} - \sum_{i=1}^{K} log \frac{\Gamma(\alpha_i^T)}{\Gamma(\alpha_i^S)} \\ &+ \sum_{i=1}^{K} (\alpha_i^T - \alpha_i^S)(\psi(\alpha_i^T) - \psi(\alpha_0^T))\end{aligned} \tag{13}$$

The optimization effect of first-order distillation on the proportions of network outputs is straightforward, as it directly acts on $\boldsymbol{p}^S$. However, some may question how second-order distillation achieves alignment in terms of class magnitudes. To address this, we provide a more detailed explanation from the perspective of derivatives. The partial derivative of $\alpha_i^S$ in the second-order distillation objective can be written as:

$$\begin{aligned}\frac{\partial \mathcal{L}_{2nd}}{\partial \alpha_i^S} &= -\frac{\partial \log(\Gamma(\alpha_0^S))}{\partial \alpha_i^S} + \frac{\partial \log(\alpha_i^S)}{\partial \alpha_i^S} - \psi(\alpha_i^T) + \psi(\alpha_0^T) \\ &= \psi(\alpha_0^T) - \psi(\alpha_0^S) + \psi(\alpha_i^S) - \psi(\alpha_i^T)\end{aligned} \tag{14}$$

From the above formulation, it can be inferred that the gradients of the second-order loss with respect to the outputs of different classes are non-conflicting. This ensures that as the Dirichlet parameter $\alpha_i^S$ for any given class approaches $\alpha_i^T$, the second-order loss decreases monotonically. In contrast, for the first-order loss, which focuses on optimizing inter-class proportions, the adjustment of $\alpha_i^S$ toward $\alpha_i^T$ may cause significant shifts in the proportions of other classes, potentially leading to an increase in the loss. Consequently, the second-order loss is characterized as intra-class magnitude alignment, while the first-order loss is described as inter-class proportion alignment.

Table 1. Results on the CIFAR-100 validation set. The teacher and student networks share the same architecture but differ in either depth or width. "Softmax" indicates that the model uses the vanilla KD probabilistic model, while "EDL" indicates that the model are trained by evidential cross entropy loss. "*" denotes the results obtained by replacing the softmax-based teacher networks with evidential networks.

| Experiment Group | | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| Teacher | Architecture | ResNet32×4 | VGG13 | WRN-40-2 | WRN-40-2 | ResNet56 | ResNet110 | ResNet110 |
| | Softmax | 79.42 | 74.64 | 75.61 | 75.61 | 72.34 | 74.31 | 74.31 |
| | EDL | 79.53 | 74.96 | 75.41 | 75.41 | 73.09 | 74.67 | 74.67 |
| Student | Architecture | ResNet8×4 | VGG8 | WRN-40-1 | WRN-16-2 | ResNet20 | ResNet32 | ResNet20 |
| | Softmax | 72.50 | 70.36 | 71.98 | 73.26 | 69.06 | 71.14 | 69.06 |
| | EDL | 72.77 | 70.67 | 71.84 | 73.36 | 69.22 | 70.9 | 69.22 |
| Logit | KD [5] | 73.33 | 72.98 | 73.54 | 74.92 | 70.66 | 73.08 | 70.67 |
| | KD* | 73.82 | 73.55 | 73.26 | 74.80 | 70.96 | 73.15 | 70.81 |
| | DKD [16] | 76.32 | 74.68 | 74.81 | 76.24 | 71.97 | 74.11 | 71.06 |
| | DKD* | 76.28 | 74.69 | 74.14 | 75.27 | 71.71 | 73.62 | 71.54 |
| | Logit_Stand [10] | 76.62 | 74.36 | 74.37 | 76.11 | 71.43 | 74.17 | 71.48 |
| | Logit_Stand* | 76.41 | 74.47 | 73.99 | 75.62 | 71.00 | 73.50 | 71.00 |
| | EKD(Ours) | **77.21** | **74.71** | **74.43** | **76.15** | **71.48** | **73.68** | **71.51** |

## 2. Relationship between EKD and classical KD

It is observed that both EKD and classical KD methods utilize first-order categorical probabilities for distillation. Consequently, we try to explore the similarities and differences between the first-order probabilities in EKD and classical KD.

Beginning with the network output logits $z = \{z_1, z_2, ..., z_K\}$, classical KD apply a softmax function to derive probabilities:

$$p_i = \frac{exp(z_i)}{\sum_{j=1}^{K} exp(z_j)} \tag{15}$$

where $p_i$ refers to the $i$-th dimension of categorical probability $\boldsymbol{p}$. On the other hand, the first-order probability in evidential theory are obtained by:

$$\hat{p}_i = \frac{\sigma(z_i) + \lambda}{\sum_{j=1}^{K} (\sigma(z_j) + \lambda)} \tag{16}$$

where $\sigma(\cdot)$ denote the evidential activation function.

Upon comparing the two equations above, their forms are remarkably similar. In fact, Eq. 15 can be regarded as a special case of Eq. 16 when $f = exp$ and $\lambda = 0$. Here, $\lambda$ represents a prior Dirichlet distribution, denoted as $Dir(\boldsymbol{\lambda})$. Therefore, the classical first-order probability is equivalent to the expected probability in the absence of any prior distribution. In other words, the first -order alignment in EKD further clarifies the source of the first-order probabilities, which are the expectations of second-order distributions. Moreover, we take into account the prior Dirichlet distribution inherent in the training dataset.

## 3. Implementation Details

We adopt the experimental settings from previous work [2, 10, 16]. In the experiments on CIFAR-100, we utilize the SGD optimizer with 240 epochs. When the student network is ResNets [3], WRNs [14], or VGGs [9], the initial learning rate is set to 0.05. For MobileNets [6, 8] and ShuffleNets [15], the initial learning rate is set to 0.01. The learning rate decayed by a factor of 0.1 at epochs 150, 180, and 210. As the batch size is 64, the momentum and weight decay were set to 0.9 and $5e-4$, respectively. The cross-entropy weight is fixed at 1, while the weights for first-order and second-order distillation are set to 4.5 each. Additionally, in the second-order distillation objective, we adopted the linear warmup strategy from DKD to prevent numerical overflow during the initial training phase.

In the experiments on ImageNet, the number of epochs for SGD optimization is set to 100, and the batch size is increased to 512. The initial learning rate is set to 0.2 and decays by a factor of 0.1 every 30 epochs. The momentum and weight decay are configured to 0.9 and $1e-4$, respectively. The cross-entropy weight is set to 1, while the weights for the first-order and second-order distillation objectives are set to 2 and 0.01, respectively.

As mentioned in Section 3.1 of the main text, the network outputs' logits are transformed through an evidential activation function, for which we adopt the exponential function, and then added to a prior weight $\lambda$. Typically, $\lambda$ is a manually specified hyperparameter [1], but we define it as a trainable parameter that depends solely on the training dataset. Additionally, we distill the prior weights by directly transferring the $\lambda$ values from the trained teacher network to the student network.

## 4. Further Remarks

**The Effect of Evidential Networks.** Since EKD employs second-order predictions supported by evidential theory, we adopt evidential student and teacher networks which outperform softmax-based networks by an average of 0.2% . This raises potential fairness concerns, primarily regard-

Table 2. Top-1 accuracy of various ViT student models on CIFAR100. Teacher model is ResNet56.

| Architecture | DeiT-Ti [11] | PiT-Ti [4] | PVT-Ti [12] | PVTv2 [13] |
|---|---|---|---|---|
| Softmax | 65.08 | 73.58 | 69.22 | 77.44 |
| EDL | 64.77 | 73.48 | 69.02 | 76.68 |
| KD [5] | 73.25 | 75.47 | 73.60 | 78.81 |
| AutoKD [7] | 78.58 | 78.51 | 77.48 | 79.37 |
| Logit_Stand [10] | 78.55 | 78.76 | 78.43 | 78.43 |
| EKD(Ours) | **78.64** | **79.33** | **78.74** | **79.80** |

ing the fixed teacher network, as all methods require the student network to be trained from scratch during distillation. To ensure fairness, We replace the softmax-based teacher networks with evidential networks in several representative methods, including KD [5], DKD [16] and Logit_Stand [10].

As shown in Table 1, evidential networks have varying effects, both positive and negative, on the three distillation methods across different network architectures. On average, vanilla KD [5] achieves a performance improvement of 0.17% on evidential networks, whereas DKD [16] and Logit_Stand [10] experience decreases of 0.13% and 0.37%, respectively. These results indicate that teacher networks trained with evidential theory are both essential and unique to EKD, as other methods are not inherently compatible with evidential networks. Thus, the comparisons presented in this study are fair, as the most suitable teacher network was chosen for each method.

Another fairness consideration is that evidential student networks generally outperform softmax-based networks by a slight margin, with an average improvement of 0.2%. More importantly, under these conditions, students trained with EKD achieve absolute performance gains of 1.80% over KD [5], 0.52% over DKD [16], and 0.23% over Logit_Stand [10]. These results demonstrate that EKD delivers substantial improvements over KD and marginally outperforms state-of-the-art methods.

**Distilling ViTs.** Logit-based distillation methods operate solely from the perspective of network predictions, making them inherently model-agnostic. As a result, these methods can be directly applied to ViT models. The summarized results for ViT are presented in Table 2. EKD achieves state-of-the-art performance on half of the networks and demonstrates comparable performance to other methods on the remaining networks. These results underscore the potential of EKD for deployment in more advanced network architectures.

## References


[1] Mengyuan Chen, Junyu Gao, and Changsheng Xu. R-edl: Relaxing nonessential settings of evidential deep learning. In *The Twelfth International Conference on Learning Representations*, 2024. 3

[2] Pengguang Chen, Shu Liu, Hengshuang Zhao, and Jiaya Jia. Distilling knowledge via knowledge review. In *CVPR*, pages 5008–5017, 2021. 3

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 3

[4] Byeongho Heo, Sangdoo Yun, Dongyoon Han, Sanghyuk Chun, Junsuk Choe, and Seong Joon Oh. Rethinking spatial dimensions of vision transformers. In *CVPR*, pages 11936–11945, 2021. 4

[5] Geoffrey Hinton. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 3, 4

[6] Andrew G Howard. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017. 3

[7] Lujun Li, Peijie Dong, Zimian Wei, and Ya Yang. Automated knowledge distillation via monte carlo tree search. In *CVPR*, pages 17413–17424, 2023. 4

[8] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *CVPR*, pages 4510–4520, 2018. 3

[9] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 3

[10] Shangquan Sun, Wenqi Ren, Jingzhi Li, Rui Wang, and Xiaochun Cao. Logit standardization in knowledge distillation. In *CVPR*, pages 15731–15740, 2024. 3, 4

[11] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *ICML*, pages 10347–10357. PMLR, 2021. 4

[12] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *CVPR*, pages 568–578, 2021. 4

[13] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pvt v2: Improved baselines with pyramid vision transformer. *Computational Visual Media*, 8(3):415–424, 2022. 4

[14] Sergey Zagoruyko. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016. 3

[15] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *CVPR*, pages 6848–6856, 2018. 3

[16] Borui Zhao, Quan Cui, Renjie Song, Yiyu Qiu, and Jiajun Liang. Decoupled knowledge distillation. In *CVPR*, pages 11953–11962, 2022. 3, 4