MoE-KD: Your Teacher Model is Worth Mixture-of-Experts for Knowledge Distillation

Kuo Shi, Wenjie Zhu*, and Bo Peng*

Independent Researcher
kuo.shi@foxmail.com, {wenjie.zhu.cv,bopeng.ml}@gmail.com

Abstract. Knowledge distillation (KD) aims to transfer useful information from a large-scale model (teacher) to a lightweight model (student). Classical KD focuses on leveraging the teacher's predictions as soft labels to regularize student training. However, the exact match of predictions in Kullback-Leibler (KL) divergence could be somewhat in conflict with the classification objective, given that the distribution discrepancies between teacher-generated predictions and ground-truth annotations tend to be fairly severe. In this paper, we rethink the role of teacher predictions from a Mixture-of-Experts (MoE) perspective and transfer knowledge by introducing teacher predictions as latent variables to reformulate the classification objective. This MoE strategy results in breaking down the vanilla classification task into a mixture of easier subtasks with the teacher classifier as a gating function to weigh the importance of subtasks. Each subtask is efficiently conquered by distinct experts that are effectively implemented by resorting to multi-level teacher outputs. We further develop a theoretical framework to formulate our method, termed MoE-KD, as an Expectation-Maximization (EM) algorithm and provide proof of the convergence. Extensive experiments manifest that MoE-KD outperforms advanced knowledge distillers on mainstream benchmarks.

Keywords: Knowledge Distillation · Mixture-of-Experts

1 Introduction

Deep learning has shown its significance by boosting the performance of various real-world tasks such as computer vision [42], natural language processing [19], and reinforcement learning [62]. However, it is worth mentioning that the effectiveness of deep learning generally comes at the expense of huge computational complexity and massive storage requirements. This greatly restricts the deployment of large-scale models (teachers) in real-time applications where lightweight models (students) are preferable due to limited resources. Under this context, with the primary goal of improving the student's performance for the task at hand, knowledge distillation (KD) [26, 72] is introduced as a de facto standard to transfer knowledge from a teacher model to a student model.

* Correspondence to Wenjie Zhu and Bo Peng

The rationale behind KD can be explained from an optimization perspective: there is evidence that high-capacity models can find good local minima due to over-parameterisation [21, 63]. This motivates KD to use such models to facilitate the optimization of lower-capacity models (i.e., the student) during training. Classically, KD is approached by minimizing the Kullback-Leibler (KL) divergence between predictive distributions of the teacher and student [29, 31], the motivation behind which is to leverage the teacher's predictions as soft labels to regularize the student training [**?**, 54]. However, the efficacy of classical KD is challenged by counter-intuitive observations [13, 66]. Specifically, a larger teacher does not necessarily increase a student's accuracy compared to a relatively smaller teacher. This can be attributed to the capacity gap between the two models which makes the discrepancy between their predictions significantly large [33]. On the one hand, some methods [20,53,56,64] develop student-friendly teachers to tackle the poor learning issue of the student model. Unfortunately, such methods suffer from complex distillation procedures and heavy computational costs for re-training the teacher model, therefore not being applicable in practice. On the other hand, ATS [47] separately applies a higher/lower temperature to the correct/wrong class by finding that more complex teachers are more likely to assign a larger score for the correct class or less varied scores for the wrong classes while KD-Zero [46] develops automated searches for distillers without manual architecture modification and KD design.

Despite remarkable progress, it has been largely overlooked the fact that there could be a significantly large discrepancy between ground-truth labels and teacher-generated labels. Our key insight stems from [74], which shows that *pseudo-labels created by machines are naturally imbalanced due to intrinsic data similarity*. This aligns with empirical observations in [55]: *whether the temperature is high or low, the teacher would produce imbalanced predictive distributions even though it is trained on a balanced dataset*. Given the fact that classical KD typically calculates the cross-entropy loss between the ground-truth label and the student's prediction in addition to the KL divergence between the teacher's and student's predictions, this kind of transfer gap makes it ill-prosed to simultaneously align the student's predictive distribution with those mutually exclusive targets, which greatly undermines the power of classical KD.

To get out of this dilemma, this paper rethinks knowledge distillation from a mixture-of-experts (MoE) perspective. The heart of our method, termed `MoE-KD`, lies in leveraging the teacher's predictions as latent variables to rewrite the classification objective. In this way, we arrive at decomposing the student classifier as a convex combination of conditional models. Specifically, each of the conditional models, referred to as an expert, learns to classify a subset of samples, where an input-dependent gating function partitions the dataset into subsets by allocating weights among experts.

To address nontrivial learning problems, we formulate `MoE-KD` as a novel Expectation-Maximization (EM) algorithm [18], where we iteratively estimate the Bayes-optimal posterior distribution of the latent variables given the observed data (in the E-step) and maximize the evidence lower bound (ELBO) of

the reformulated classification objective (in the M-step). We theoretically prove that the ELBO is upper-bounded and our proposed EM algorithm contributes to the convergence of the ELBO (see Section 3.3). Empirically, our proposed `MoE-KD` achieves state-of-the-art performance in various distillation settings regarding teacher-student pairs (homogeneous and heterogeneous) and training datasets (coarse-grained and fine-grained).

2 Preliminary

Notations. We write vectors and matrices as bold-faced lowercase and uppercase characters respectively. All trainable parameters will be subscripted by $\boldsymbol{\theta}$. Let $\mathbf{e}[i]$ be the i-th element of the vector $\mathbf{e} \in \mathbb{R}^K$ and $[K] = \{1, \ldots, K\}$, we then define $\text{softmax}_k(\mathbf{e}) = \exp(\mathbf{e}[k]) / \sum_{i \in [K]} \exp(\mathbf{e}[i])$.
Multi-class Classification. This paper considers K-way classification as a case study, where $\mathcal{X}$ and $\mathcal{Y} = [K]$ denote the input space and label space respectively. Let $\mathbb{P}_{XY}$ be the joint distribution defined over $\mathcal{X} \times \mathcal{Y}$, we are provided with a labelled dataset $\mathcal{D} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\} \sim \mathbb{P}_{XY}^N$, *i.i.d.*, to train a discriminative model by maximizing the following objective over the dataset $\mathcal{D}$:

$$\mathcal{R}_{\text{cls}}(\mathbf{x}_i, y_i; \boldsymbol{\theta}) \triangleq \log P_{\boldsymbol{\theta}}(Y = y_i | \mathbf{x}_i). \tag{1}$$

Knowledge Distillation involves transferring dark knowledge from a teacher model to a student model. Classical KD [29, 31] calculates the cross-entropy between the ground-truth label and student predictions as well as the KL divergence between the predictive distributions of the student and the teacher. Since the teacher is pre-trained and fixed in the context of KD, the overall learning objective of classical KD can be simplified into the following form[1]:

$$\mathcal{R}_{\text{KD}}(\mathbf{x}_i, y_i; \boldsymbol{\theta}) \triangleq \mathcal{R}_{\text{cls}}(\mathbf{x}_i, y_i; \boldsymbol{\theta}) + \alpha \cdot \sum_{k=1}^{K} P^{\mathcal{T}}(Y^{\mathcal{T}} = k | \mathbf{x}_i) \log P_{\boldsymbol{\theta}}^{\mathcal{S}}(Y^{\mathcal{S}} = k | \mathbf{x}_i), \tag{2}$$

where $\alpha > 0$ is a weighting hyper-parameter that balances the importance of the two losses. Note that, in Eq. (2), we have used $\mathcal{T}$ and $\mathcal{S}$ as superscripts to indicate the teacher and student model respectively, which, unless explicitly stated, is considered as a default setting in the rest of this paper.

3 Methodology

3.1 Rethinking KD From a Mixture of Experts Perspective

Motivated by the mixture of experts (MoE) framework [37], we introduce the teacher's class prediction $Y^{\mathcal{T}} \in \{1, 2, \cdots, K\}$ as a latent variable and naturally

[1] For brevity, we omit the constant term $\sum_{k=1}^{K} P^{\mathcal{T}}(Y^{\mathcal{T}} = k | \mathbf{x}_i) \log P^{\mathcal{T}}(Y^{\mathcal{T}} = k | \mathbf{x}_i)$.

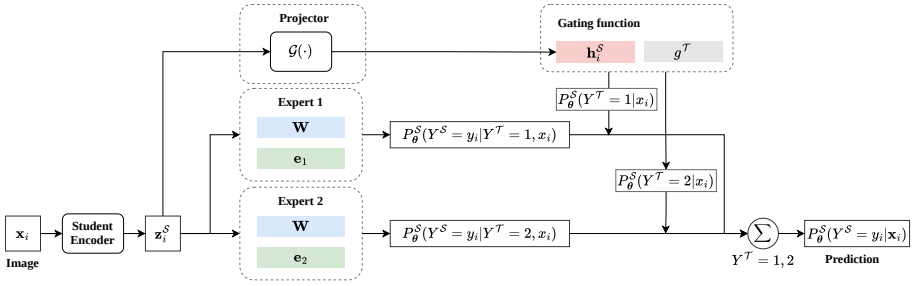

Fig. 1: The graphical illustration of the key idea behind MoE-KD. Here, we demonstrate a 2-way classification example. Best view in color.

extend the vanilla classification objective in Eq. (1) to the following formulation:

$$\log P_{\boldsymbol{\theta}}^{\mathcal{S}}(Y^{\mathcal{S}}=y_i|\mathbf{x}_i)=\log\sum_{k=1}^{K}P_{\boldsymbol{\theta}}^{\mathcal{S}}\left(Y^{\mathcal{S}}=y_i,Y^{\mathcal{T}}=k|\mathbf{x}_i\right) \tag{3}$$

$$=\underbrace{\log\sum_{k=1}^{K}P_{\boldsymbol{\theta}}^{\mathcal{S}}(Y^{\mathcal{S}}=y_i|Y^{\mathcal{T}}=k,\mathbf{x}_i)P_{\boldsymbol{\theta}}^{\mathcal{S}}\left(Y^{\mathcal{T}}=k|\mathbf{x}_i\right)}_{\mathcal{R}_{\texttt{MoE-KD}}(\mathbf{x}_i,y_i;\boldsymbol{\theta})}, \tag{4}$$

where $P_{\boldsymbol{\theta}}^{\mathcal{S}}(Y^{\mathcal{S}}=y_i|Y^{\mathcal{T}}=k,\mathbf{x}_i)$ is one of the experts that classify a subset of samples and $P_{\boldsymbol{\theta}}^{\mathcal{S}}\left(Y^{\mathcal{T}}=k|\mathbf{x}_i\right)$ is a gating function that partitions the dataset into subsets according to the latent semantics by routing each sample to one or a few experts. With this *partition-and-classify* principle, the experts tend to be highly specialized in data points that share similar semantics, which improves training efficiency. While, same as the original MoE, the experts work in the supervised setting, both gating functions and experts are based on neural networks to fit the high-dimensional data. In the following, we will elaborate on how we parameterize each term in Eq. (4) to fit the KD task.

Gating function. The gating function organizes the classification task into K simpler subtasks by weighting the experts based on the semantics of the input sample. Inspired by [22], we formulate the gating function by reusing the pre-trained teacher classifier, namely,

$$P_{\boldsymbol{\theta}}^{\mathcal{S}}\left(Y^{\mathcal{T}}=k|x_i\right)=\mathrm{softmax}_k\left[g^{\mathcal{T}}(\mathbf{h}_i^{\mathcal{S}})\right],\quad \mathbf{h}_i^{\mathcal{S}}=\mathcal{G}(\mathbf{z}_i^{\mathcal{S}}), \tag{5}$$

where $\mathbf{z}_i^{\mathcal{S}}\in\mathbb{R}^d$ denotes the student feature of $\mathbf{x}_i$ and a projector $\mathcal{G}(\cdot)$ transforms from the student feature space $\mathcal{Z}^{\mathcal{S}}$ to the teacher feature space $\mathcal{Z}^{\mathcal{T}}$ for dimension alignment at a relatively small cost.

Experts. Each expert learns to solve a distinct subtask of the classification task arranged by the gating function. Formally, inspired by [10], let $\mathbf{e}_k\in\mathbb{R}^d$ be an expert prototype, we formulate the probability of the sample x_i being

recognized as the y_i-th class by the k-th expert as follows:

$$P_{\boldsymbol{\theta}}^{\mathcal{S}}\left(Y^{\mathcal{S}}=y_i|Y^{\mathcal{T}}=k,x_i\right)=\text{softmax}_{y_i}\left[\mathbf{W}^{\top}(\mathbf{z}_i^{\mathcal{S}}+\mathbf{e}_k)\right]=\text{softmax}_{y_i}\left(\mathbf{W}^{\top}\mathbf{z}_i^{\mathcal{S}}+\mathbf{b}_k\right), \tag{6}$$

where $\mathbf{W}\in\mathbb{R}^{d\times K}$ represents a learnable weight matrix and the expert-specific bias vector $\mathbf{b}_k\in\mathbb{R}^K$ has been re-parameterized by $\mathbf{W}^{\top}\mathbf{e}_k$. To make Eq. (6) benefit from the teacher's logit-level and feature-level knowledge, we implement $\mathbf{e}_k$ based on deep set representations [87]:

$$\mathbf{e}_k=\Psi(\boldsymbol{\mu}_k),\quad \boldsymbol{\mu}_k=\sum_{i=1}^{N}\frac{P^{\mathcal{T}}\left(Y^{\mathcal{T}}=k|\mathbf{x}_i\right)}{\sum_{j=1}^{N}P^{\mathcal{T}}\left(Y^{\mathcal{T}}=k|\mathbf{x}_j\right)}\cdot\mathbf{z}_i^{\mathcal{T}}. \tag{7}$$

where $\Psi(\cdot):\mathcal{Z}^{\mathcal{T}}\rightarrow\mathbb{R}^d$ is another projector that is introduced to match feature dimensions given that Eq. (7) involves a soft aggregation along samples in the teacher embedding space. Although the design of Eq. (7) is similar to pooling by multi-head attention (PMA) in the set transformer [44], we do not rely on a softmax operation to normalize aggregation weights along samples as we never expect any single sample to play a dominant role in representing $\boldsymbol{\mu}_k$. Besides, for a fair comparison, we define $P^{\mathcal{T}}\left(Y^{\mathcal{T}}=k|\mathbf{x}_i\right)$ in accordance with prior works [29, 31, 55, 89, 91], i.e.,

$$P^{\mathcal{T}}\left(Y^{\mathcal{T}}=k|\mathbf{x}_i\right)=\text{softmax}_k\left[g^{\mathcal{T}}(\mathbf{z}_i^{\mathcal{T}})/\tau\right], \tag{8}$$

where $\tau>0$ denotes a temperature hyper-parameter [31].

3.2 Deriving the Evidence Lower Bound

In practice, the conditional log-likelihood function in Eq. (4) is hard to be directly optimized [4,73]. To address this non-trivial problem, we start from the following evidence lower bound (ELBO) of the vanilla objective $\log P_{\boldsymbol{\theta}}^{\mathcal{S}}(Y^{\mathcal{S}}=y_i|\mathbf{x}_i)$:

$$\begin{aligned}
&\log P_{\boldsymbol{\theta}}^{\mathcal{S}}(Y^{\mathcal{S}}=y_i|\mathbf{x}_i)\\
=&\text{ELBO}(\hat{P},\mathbf{x}_i,y_i;\boldsymbol{\theta})+D_{\text{KL}}\left[\hat{P}(Y^{\mathcal{T}}=k|\mathbf{x}_i)||P_{\boldsymbol{\theta}}^{\mathcal{S}}(Y^{\mathcal{T}}=k|Y^{\mathcal{S}}=y_i,\mathbf{x}_i)\right]\\
\geq&\text{ELBO}(\hat{P},\mathbf{x}_i,y_i;\boldsymbol{\theta})\\
\triangleq&\sum_{k=1}^{K}\left[\hat{P}(Y^{\mathcal{T}}=k|\mathbf{x}_i)\log P_{\boldsymbol{\theta}}^{\mathcal{S}}(Y^{\mathcal{S}}=y_i|Y^{\mathcal{T}}=k,\mathbf{x}_i)\right]-D_{\text{KL}}\left[\hat{P}(Y^{\mathcal{T}}=k|\mathbf{x}_i)||P_{\boldsymbol{\theta}}^{\mathcal{S}}(Y^{\mathcal{T}}=k|\mathbf{x}_i)\right],
\end{aligned} \tag{9}$$

where $D_{\text{KL}}(\cdot)$ represents the Kullback–Leibler (KL) divergence and $\hat{P}(Y^{\mathcal{T}}=k|\mathbf{x}_i)$ denotes a distribution conditioned on $\mathbf{x}_i$. To make the inequality hold with equality so that the ELBO reaches its maximum value $\log P_{\boldsymbol{\theta}}^{\mathcal{S}}(Y=y_i|\mathbf{x}_i)$, we need to require:

$$D_{\text{KL}}\left[\hat{P}(Y^{\mathcal{T}}=k|\mathbf{x}_i)||P_{\boldsymbol{\theta}}^{\mathcal{S}}(Y^{\mathcal{T}}=k|Y^{\mathcal{S}}=y_i,\mathbf{x}_i)\right]=0. \tag{10}$$

By approximating the variational distribution $\hat{P}(Y^{\mathcal{T}}=k|\mathbf{x}_i)$ with $P_{\boldsymbol{\theta}}^{\mathcal{S}}(Y^{\mathcal{T}}=k|Y^{\mathcal{S}}=y_i,\mathbf{x}_i)$ and connecting Eq. (9) with Eq. (4), we are now ready to convert the optimization of Eq. (4) into :

$$\arg\max_{\boldsymbol{\theta}}\mathcal{R}_{\texttt{MoE-KD}}(\mathbf{x}_i,y_i;\boldsymbol{\theta})=\arg\max_{\boldsymbol{\theta}}\text{ELBO}(\hat{P},\mathbf{x}_i,y_i;\boldsymbol{\theta}). \tag{11}$$

3.3 Formulating MoE-KD as Expectation-Maximization

E-step. This step aims to estimate $\hat{P}_{t+1}(Y^{\mathcal{T}}=k|\mathbf{x}_i)$ with the fixed $\boldsymbol{\theta}_t$ at the iteration t to make $\hat{P}_{t+1}(Y^{\mathcal{T}}=k|\mathbf{x}_i)=P^{\mathcal{S}}_{\boldsymbol{\theta}_t}(Y^{\mathcal{T}}=k|Y^{\mathcal{S}}=y_i,\mathbf{x}_i)$, which is implied by Eq. (10). To this end, by applying the Bayes' theorem to $P^{\mathcal{S}}_{\boldsymbol{\theta}_t}(Y^{\mathcal{T}}=k|Y^{\mathcal{S}}=y_i,\mathbf{x}_i)$, we approximate $\hat{P}_{t+1}(Y^{\mathcal{T}}=k|\mathbf{x}_i)$ as:

$$\hat{P}_{t+1}(Y^{\mathcal{T}}=k|\mathbf{x}_i)=\frac{P^{\mathcal{S}}_{\boldsymbol{\theta}_t}(Y^{\mathcal{T}}=k|\mathbf{x}_i)P^{\mathcal{S}}_{\boldsymbol{\theta}_t}(Y^{\mathcal{S}}=y_i|Y^{\mathcal{T}}=k,\mathbf{x}_i)}{\sum_{j=1}^{K}P^{\mathcal{S}}_{\boldsymbol{\theta}_t}(Y^{\mathcal{T}}=j|\mathbf{x}_i)P^{\mathcal{S}}_{\boldsymbol{\theta}_t}(Y^{\mathcal{S}}=y_i|Y^{\mathcal{T}}=j,\mathbf{x}_i)} \tag{12}$$

M-step. With the sub-optimal $\hat{P}_{t+1}(Y^{\mathcal{T}}=k|\mathbf{x}_i)=P^{\mathcal{S}}_{\boldsymbol{\theta}_t}(Y^{\mathcal{T}}=k|Y^{\mathcal{S}}=y_i,\mathbf{x}_i)$ after E-step, we turn to maximize the ELBO in Eq. (9):

$$\boldsymbol{\theta}_{t+1}=\arg\max_{\boldsymbol{\theta}_t}\mathbb{E}_{(\mathbf{x}_i,y_i)\sim\mathbb{P}_{XY}}\left[\mathrm{ELBO}(\hat{P}_{t+1},\mathbf{x}_i,y_i;\boldsymbol{\theta}_t)\right]. \tag{13}$$

When integrating Eq. (13) into the batch-based training routine where we sample a mini-batch $\mathcal{B}$ from the dataset $\mathcal{D}$ at the beginning of each iteration, it is natural to build an efficient stochastic estimator of the ELBO over $\mathcal{D}$ to learn the parameters $\boldsymbol{\theta}_t$, which is given by:

$$\boldsymbol{\theta}_{t+1}=\arg\max_{\boldsymbol{\theta}_t}\frac{1}{|\mathcal{B}|}\sum_{(\mathbf{x}_i,y_i)\in\mathcal{B}}\mathrm{ELBO}(\hat{P}_{t+1},\mathbf{x}_i,y_i;\boldsymbol{\theta}_t). \tag{14}$$

Finally, the inference with the optimized parameters $\hat{\boldsymbol{\theta}}$ for a test-time sample $\mathbf{x}$ requires to compute $\arg\max_k\mathcal{R}_{\texttt{MoE-KD}}(\mathbf{x},k;\hat{\boldsymbol{\theta}})$. For clarity, we summarize the complete algorithm in Algorithm 1.

3.4 Convergence Analysis.

At the E-step of the iteration $t+1$, we estimate $\hat{P}_{t+1}(Y^{\mathcal{T}}=k|\mathbf{x}_i)$ to ensure $\mathrm{ELBO}(\hat{P}_{t+1},\mathbf{x}_i,y_i;\boldsymbol{\theta}_t)=\mathcal{R}_{\mathrm{MoE}}(\mathbf{x}_i,y_i;\boldsymbol{\theta}_t)$. At the M-step after the E-step, we have obtained $\boldsymbol{\theta}_{t+1}$ with a fixed variational distribution $\hat{P}_{t+1}(Y^{\mathcal{T}}=k|\mathbf{x}_i)$ to have $\mathrm{ELBO}(\hat{P}_{t+1},\mathbf{x}_i,y_i;\boldsymbol{\theta}_{t+1})\geq\mathrm{ELBO}(\hat{P}_{t+1},\mathbf{x}_i,y_i;\boldsymbol{\theta}_t)$. Therefore, we obtain the following sequence:

$$\begin{aligned}\mathcal{R}_{\texttt{MoE-KD}}(\mathbf{x}_i,y_i;\boldsymbol{\theta}_{t+1})&\geq\mathrm{ELBO}(\hat{P}_{t+1},\mathbf{x}_i,y_i;\boldsymbol{\theta}_{t+1})\\&\geq\mathrm{ELBO}(\hat{P}_{t+1},\mathbf{x}_i,y_i;\boldsymbol{\theta}_t)=\mathcal{R}_{\texttt{MoE-KD}}(\mathbf{x}_i,y_i;\boldsymbol{\theta}_t).\end{aligned} \tag{15}$$

Given that $\mathcal{R}_{\texttt{MoE-KD}}(\mathbf{x}_i,y_i;\boldsymbol{\theta}_{t+1})\geq\mathcal{R}_{\texttt{MoE-KD}}(\mathbf{x}_i,y_i;\boldsymbol{\theta}_t)$, $\mathrm{ELBO}(\hat{P},\mathbf{x}_i,y_i;\boldsymbol{\theta})$ is upper-bounded and converge to a certain value with the EM algorithm proposed above.

3.5 Relation to Existing Works

We recently find that SRRL [83] also comes with the reused teacher classifier to train the student model and can be regarded as a natural baseline of our method. We forge a mathematical connection between SRRL and the ELBO in Eq. (9) by showing that the latter intrinsically subsumes the former as a special exemplar of itself, which implies the theoretical superiority of our method.

Algorithm 1 `MoE-KD`

Input: Training dataset $\mathcal{D}$, pre-trained teacher $\mathcal{T}$, randomly initialized student parameters $\boldsymbol{\theta}$, SGD optimizer $\boldsymbol{\Gamma}(\cdot)$.
Output: Well-taught student $\mathcal{S}$
repeat
  /* **E-step** */
  Update $\hat{P}_{t+1}(Y^{\mathcal{T}} = k \mid \mathbf{x}_i)$ for each $\mathbf{x}_i \in \mathcal{D}$ via Eq. (12)
  /* **M-step** */
  Randomly select a batch $\mathcal{B}$ from $\mathcal{D}$
  for each $(\mathbf{x}_i, y_i) \in \mathcal{B}$ **do**
    Calculate $\text{ELBO}(\hat{P}, \mathbf{x}_i, y_i; \boldsymbol{\theta})$ via Eq. (9)
  end for
  $\mathcal{L} = -\frac{1}{|\mathcal{B}|} \sum_{(\mathbf{x}_i, y_i) \in \mathcal{B}} \text{ELBO}(\hat{P}, \mathbf{x}_i, y_i; \boldsymbol{\theta})$
  $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \boldsymbol{\Gamma}(\nabla_{\boldsymbol{\theta}} \mathcal{L})$
until convergence or reaching the maximum number of iterations

Assumption 1 (Collapsed Projection) *The projector $\Psi(\cdot)$ in Eq. (7) is completely collapsed such that, for all inputs $\boldsymbol{\mu} \in \mathcal{Z}^{\mathcal{T}}$, we have $\Psi(\boldsymbol{\mu}) = \mathbf{b}$.*

Lemma 1. *If Assumption 1 holds, the expert $P^{\mathcal{S}}_{\boldsymbol{\theta}}\left(Y^{\mathcal{S}} = y_i | Y^{\mathcal{T}} = k, x_i\right)$ in Eq. (6) will degenerate into a universal parametric softmax classifier, i.e.,*

$$P^{\mathcal{S}}_{\boldsymbol{\theta}}\left(Y^{\mathcal{S}} = y_i | Y^{\mathcal{T}} = k, x_i\right) = \text{softmax}_{y_i}\left[\mathbf{W}^{\top}\mathbf{z}^{\mathcal{S}}_i + \mathbf{b}\right], \quad \forall k \in [K]. \tag{16}$$

In addition to Assumption 1, let $\hat{P}(Y^{\mathcal{T}} = k|\mathbf{x}_i) = P^{\mathcal{T}}(Y^{\mathcal{T}} = k|\mathbf{x}_i)$, the ELBO in Eq. (9), i.e.,

$$\text{ELBO}(\hat{P}, \mathbf{x}_i, y_i; \boldsymbol{\theta}) = \underbrace{\log\left[\text{softmax}_{y_i}\left(\mathbf{W}^{\top}\mathbf{z}^{\mathcal{S}}_i + \mathbf{b}\right)\right] - \sum_{k=1}^{K} D_{\text{KL}}\left[P^{\mathcal{T}}(Y^{\mathcal{T}} = k|\mathbf{x}_i) || P^{\mathcal{S}}_{\boldsymbol{\theta}}(Y^{\mathcal{T}} = k|\mathbf{x}_i)\right]}_{\mathcal{R}_{\text{SRRL}}(\mathbf{x}_i, y_i; \boldsymbol{\theta})},$$

is mathematically equivalent to the optimization objective in SRRL regardless the hyper-parameter β that scales the second term of $\mathcal{R}_{\text{SRRL}}(\mathbf{x}_i, y_i; \boldsymbol{\theta})$. Nevertheless, it is worthwhile to point out that, as disclosed by the authors of SRRL, $\beta = 1$ contributes to the best knowledge transfer performance, which empirically epochs our analysis above.

Interestingly, if we treat ground-truth annotations as a noisy version of teacher predictions and the gating function in Eq. (4) as the clean class posterior, the experts in Eq. (4) share a similar working mechanism with the so-called noise transition matrix $\mathbf{T} \in [0, 1]^{K \times K}$ [57] in label-noise learning [65] such that $\mathbf{T}_{ij}(\mathbf{x}) = P^{\mathcal{S}}_{\boldsymbol{\theta}}(Y^{\mathcal{S}} = i|Y^{\mathcal{T}} = j, \mathbf{x})$. However, directly estimating the transition matrix is generally infeasible [79] without the rigorous anchor-point assumption [49]. As a result, existing label-noise learners [11, 78, 84] have been developed with a two-stage training routine: 1) pre-training the gating function to estimate experts (or the noise transition matrix $\mathbf{T}$) and 2) fine-tuning the gating function with the fixed estimated experts. By contrast, our method enables

to simultaneously learn both the gating function and experts. While the mostly recent works [12, 48] approach label-noise learning in an end-to-end way, they can be criticized for removing the dependency between $P_{\boldsymbol{\theta}}^{\mathcal{S}}(Y^{\mathcal{S}} = i|Y^{\mathcal{T}} = j, \mathbf{x})$ and $\mathbf{x}$ to have $P_{\boldsymbol{\theta}}^{\mathcal{S}}(Y^{\mathcal{S}} = i|Y^{\mathcal{T}} = j, \mathbf{x}) = P_{\boldsymbol{\theta}}^{\mathcal{S}}(Y^{\mathcal{S}} = i|Y^{\mathcal{T}} = j), \forall i, j \in [K]$.

Table 1: Top-1 ACC (%) on CIFAR-100, Homogenous Architecture. The best results are in **boldface** while the second best results are in underline.

Teacher	WRN-40-2	WRN-40-2	ResNet56	ResNet110	ResNet32x4	VGG13
	75.61	75.61	72.34	74.31	79.42	74.64
Student	WRN-16-2	WRN-40-1	ResNet20	ResNet32	ResNet8x4	VGG8
	73.26	71.98	69.06	71.14	72.50	70.36
KD	74.92	73.54	70.66	73.08	73.33	72.98
FitNet	73.58	72.24	69.21	71.06	73.50	71.02
CRD	75.48	74.14	71.16	73.48	75.51	73.94
WCoRD	75.88	74.73	71.56	73.81	75.95	74.55
IPWD	—	74.64	71.32	73.91	76.03	—
WSLD	—	74.48	72.15	74.12	76.05	—
SRRL	75.96	74.75	71.44	73.80	75.92	74.40
DKD	76.24	74.81	71.97	74.11	76.32	74.68
NORM	75.65	74.82	71.35	73.67	76.49	73.95
DIST	—	74.73	71.75	—	76.31	—
DiffKD	—	74.09	71.92	—	76.72	—
WTTM	76.37	74.58	71.92	74.13	76.06	74.44
SDD	75.86	74.53	71.52	73.97	75.09	73.91
LSKD	76.22	74.43	71.24	73.76	76.16	74.23
EAKD	—	74.38	—	—	75.46	74.08
EKD	76.15	74,43	71.48	73.68	**77.72**	74.71
TeKAP	75.21	73.80	71.71	73.50	74.79	73.80
Ours	**76.98**	**75.21**	**72.49**	**74.58**	77.10	**75.03**

4 Experiments

4.1 Setup

Datasets. We, following prior works [35,47,89], perform experiments on CIFAR-100 [43], ImageNet-1K [60], CUB [71] and Stanford Dogs [40].

Baselines. We compare `MoE-KD` with KD (NeurIPS14) [31], DKD (CVPR22) [89], IPWD (NeurIPS22) [55], WSLD (ICLR21) [91], ESKD (CVPR19) [13], TAKD (AAAI20) [53], SCKD (CVPR21) [94], ATS (NeurIPS22) [47], NKD (CVPR23) [85], DIST (NeurIPS22) [33], FitNets (ICLR15) [59], CRD (ICLR20) [69], WCoRD (CVPR21) [7], ReviewKD (CVPR21) [8], NORM (ICLR23) [50], DiffKD (NeurIPS23) [35], SRRL (ICLR21) [83], WTTM (ICLR24) [90], LSKD

Table 2: Top-1 ACC (%) on CIFAR-100, Heterogeneous Architecture. The best results are in **boldface** while the second best results are in underline.

Teacher	VGG13	ResNet50	ResNet32x4	ResNet32x4	WRN-40-2
	74.64	79.34	79.42	79.42	75.61
Student	MobileNetV2	MobileNetV2	ShuffleNetV1	ShuffleNetV2	ShuffleNetV1
	64.60	64.60	70.50	71.82	70.50
KD	67.37	67.35	74.07	74.45	74.83
FitNet	64.14	63.16	73.59	73.54	73.73
CRD	69.73	69.11	75.11	75.65	76.05
WCoRD	69.47	70.45	75.40	75.96	76.32
IPWD	—	70.25	76.03	—	76.44
WSLD	—	—	75.46	75.93	76.21
SRRL	69.14	69.45	75.66	76.40	76.61
DKD	69.71	70.35	76.45	77.07	76.70
NORM	68.94	<u>70.56</u>	<u>77.42</u>	<u>78.07</u>	<u>77.06</u>
DIST	—	68.66	76.34	77.35	—
DiffKD	—	69.21	76.57	77.52	—
WTTM	69.16	69.59	74.37	76.55	75.42
SDD	68.79	69.55	—	76.67	76.65
LSKD	68.61	69.02	—	75.56	—
EAKD	69.17	69.67	—	75.91	—
EKD	<u>69.94</u>	70.38	—	77.65	—
TeKAP	67.39	69.00	74.92	75.43	76.75
Ours	**70.54**	**71.38**	**78.23**	**78.69**	**77.52**

(CVPR24) [68], WKD (NeurIPS24) [52], OFA (NeurIPS23) [30], SDD (CVPR24) [75], TeKAP (ICLR25) [32], EAKD (ICLR25) [67], and EKD (ICCV25) [81].

Implementation Detail. We employ the last feature map and a three-layer bottleneck transformation for implementing the projector $\mathcal{G}(\cdot)$, which only incurs a less than 3% cost to the pruning ratio in teacher-to-student compression [6]. We design $\Psi(\cdot)$ as a two-layer MLP module [9].

4.2 Main Results on CIFAR-100

To evaluate the effectiveness of our method, we experiment on CIFAR100 with 11 student-teacher combinations. We consider a standard data augmentation scheme including padding 4 pixels before random cropping and horizontal flipping. We set the batch size as 64 and the initial learning rate as 0.01 (for ShuffleNet and MobileNet-V2) or 0.05 (for the other series). We train the model for 240 epochs, in which the learning rate is decayed by 10 every 30 epochs after 150 epochs. We use SGD as the optimizer with weight decay $5e-4$ and momentum 0.9, Table 1 and Table 2 compare the Top-1 accuracy under two different scenarios respectively: 1) the student and the teacher share the same network architecture and 2) the student and the teacher are of a different architectural

style. The results show that ours surpasses previous methods in all cases. Taking the ResNet32x4/ResNet8x4 and WRN-40-2/ShuffleNetV1 pairs as an example, our method outperforms the most recent WTTM by 1.04% and 2.10% for each.

4.3 Main Results on ImageNet-1K

To validate the scalability of our method, we employ the PyTorch-version student-teacher combinations to perform experiments on ImageNet. The standard PyTorch ImageNet practice is adopted except for 100 training epochs. We set the batch size as 256 and the initial learning rate as 0.1. The learning rate is divided by 10 for every 30 epochs. We use SGD as the optimizer with weight decay $1e-4$ and momentum 0.9. The Top-1 and Top-5 accuracy of different distillation methods are reported in Table 3. While our method slightly performs worse than the state-of-the-art DiffKD by 0.21% for the ResNet50/MobileNetV1 pair, we achieve significantly better performance than DiffKD by 0.82% and 0.64% for the ResNet34/ResNet18 pair regarding Top-1 and Top-5 accuracy.

Table 3: Top-1 and Top-5 ACC (%) on ImageNet-1K. The best results are in **boldface** while the second best results are in underline.

ResNet34 → ResNet18		
Method	Top-1 ACC	Top-5 ACC
Teacher	73.31	91.42
Student	69.75	89.07
KD	70.66	89.88
WSLD	72.04	90.70
NKD	71.96	90.48
DKD	71.70	90.41
DIST	72.07	90.42
CRD	71.17	90.13
ReviewKD	71.61	90.51
DiffKD	72.22	90.64
SRRL	71.73	90.60
WTTM	72.19	—
WKD-L	72.49	90.75
WKD-F	72.50	91.00
SDD	71.44	90.05
LSKD	71.42	90.29
EKD	71.73	90.69
TeKAD	72.87	91.05
Ours	**73.15**	**91.78**

ResNet50 → MobileNetV1		
Method	Top-1 ACC	Top-5 ACC
Teacher	76.16	92.87
Student	68.87	88.76
KD	70.68	90.30
WSLD	71.52	90.34
NKD	72.58	90.96
DKD	72.05	91.05
DIST	73.24	91.12
CRD	71.31	90.41
ReviewKD	72.56	91.00
DiffKD	**73.62**	91.34
SRRL	72.49	90.92
WTTM	73.09	—
WKD-L	73.17	91.32
WKD-F	73.12	91.39
SDD	72.24	90.71
LSKD	72.18	90.80
EKD	72.54	91.20
TeKAD	71.35	90.54
Ours	73.41	**91.35**

Table 4: Ablation study results on CIFAR-100. Each row shows the Top-1 ACC (%).

Teacher → Student	Baseline (i)	Baseline (ii)	Baseline (iii)	Baseline (iv)	Baseline (v)	Full model
WRN-40-2 → WRN-40-1	73.87	74.43	74.66	74.95	74.79	75.21
ResNet50 → MobileNetV2	69.28	70.62	70.65	71.16	70.94	71.38

4.4 Ablation Study

We conduct an ablation study to validate our motivation and design, with the following baselines.

(i) We validate the necessity of the gating function in our method by simplifying the gating function as a uniform one such that $P^{\mathcal{S}}_{\boldsymbol{\theta}}\left(Y^{\mathcal{T}}=k|\mathbf{x}_i\right)=1/K$.
(ii) We justify the use of the teacher's classifier for the gating function by learning the gating function from scratch with the student.
(iii) In analogy to (ii), we learn the subtask-specific embedding vector from scratch with the student.
(iv) We replace the soft aggregation strategy in Eq. (7) with the hard aggregation strategy based on the hard assignments produced by the teacher.
(v) As described in Section 3.3, we formulate our method as an EM algorithm where we keep estimating the Bayes-optimal variational distribution. In this baseline, we replace the Bayes-optimal estimation with the direct assignment of the teacher's predictive distribution.

Experimental results of the ablation study on CIFAR-100 are shown in Table 4. We note several interesting observations: 1) The performance drop in Baselines (ii) and (iii) show that introducing the teacher's knowledge from either outputs or architecture is beneficial to the student; 2) Baseline (iv) performs worse than the full model. An explanation is that, compared with the hard aggregation in Baseline (iv), the soft aggregation in the full model injects class relationship knowledge so that smoothness between classes is preserved in subtask-specific embedding for each expert; 3) Baseline (i) performs worst among the baselines, which could be attributed to that, with a uniform prior, the experts would take extra efforts to become specialized in a set of images with shared semantics; 4) Baseline (v) also performs worse than the full model, which implies that it is non-trivial to properly estimate the variational distribution.

4.5 Cross-architecture Knowledge Distillation

In practice, the teacher and student models should not belong to the same model family. To simulation this, We evaluate `MoE-KD` in the setting where the teacher is a CNN and the student is a Transformer or vice versa. We use CNN models including ResNet and ConvNeXt, and vision transformers that involve ViT and DeiT. Our results in Table 5 indicate the superior transferability of our method.

Table 5: Image classification resultson CIFAR-100 across CNNs and Transformers. We report Top-1 ACC (%). The best result is in **boldface**.

Teacher → Student	KD	DKD	FitNet	CRD	DIST	OFA	WKD-L	WKD-F	Ours
ViT-S → ResNet-18	77.26	78.10	77.71	74.26	76.49	80.15	80.81	81.12	**81.57**
ConvNeXt-T → DeiT-T	72.99	74.60	60.78	68.01	73.55	75.76	76.11	73.27	**76.62**

Table 6: Top-1 ACC (%) on CUB and Stanford Dogs compared to advanced knowledge distillers. We use the ResNeXt101-32-8d as the teacher for both datasets. The best result is shown in **boldface**.

Datasets	CUB			Stanford Dogs		
Student	AlexNet	ShuffleNetV2	MobileNetV2	AlexNet	ShuffleNetV2	MobileNetV2
Random Init.	55.66	71.24	74.49	50.20	68.72	68.67
KD	55.10	71.89	76.45	50.22	68.48	71.25
ESKD	55.64	72.15	76.87	50.39	69.02	71.56
TAKD	54.82	71.53	76.25	50.36	68.94	70.61
SCKD	56.78	71.99	75.13	51.78	68.80	70.13
KD+ATS	58.32	73.15	77.83	52.96	70.92	73.16
Ours	**59.46**	**74.09**	**78.68**	**53.85**	**72.15**	**74.23**

4.6 Knowledge Distillation for Fine-grained Classification

In practice, available training samples may be visually similar. We validate our method in the scenario of fine-grained classification. Table 6 reports the Top-1 accuracy of state-of-the-art methods on CUB and Stanford Dogs. It can be found that KD can help to improve the performance of a student network. The improvement can be further enhanced by the early-stopped teacher in ESKD [3], the teacher assistant in TAKD [14], the student-customized teacher in SCKD [61] and the asymmetric temperature scaling in ATS [47]. Even though, `MOE-KD` consistently contributes to the most significant improvement for various students.

4.7 Knowledge Distillation with Stronger Teachers.

To fully investigate the sensitivity of `MoE-KD` to the capacity gap between the teacher and the student, we conduct experiments on teachers with stronger training strategies following DIST. It can be observed from Table 7 that our method keeps achieving the best performance for both ResNet50/ResNet34 and ResNet50/EfficientNet-B0 pairs. In particular, while the state-of-the-art DiffKD improves Top-1 accuracy by 1.3% and 0.8% for ResNet34 and EfficientNet-B0 respectively, our proposed method enhances the two by 1.5% and 1.0%.

Table 7: Top-1 ACC (%) on ImageNet-1K with ResNet50 trained by [76] as a stronger teacher. Students are trained under a stronger strategy [33, 35].

Teacher → Student	Random Init.	KD	RKD	SRRL	DIST	DiffKD	Ours
ResNet50 → ResNet34	76.8	77.2	76.6	76.7	77.8	78.1	**78.3**
ResNet50 → EfficientNet-B0	78.0	77.4	77.5	77.3	78.6	78.8	**79.0**

5 Related Work

5.1 Knowledge Distillation

Knowledge distillation, which uses a teacher model to improve the performance of a student model, has also been shown to benefit downstream tasks [92, 93]. In its classical form, one trains the student to fit the teacher's predictive distribution. [31] popularizes this solution by formulating it as logit matching. MLD [39] extends logit matching not only at the instance level but also at the batch and class levels, DKD [89] decouples classical KD into distilling target and non-target class knowledge, and WSLD [91] provides a bias-variance trade-off perspective for the KL term. Besides, the teacher's knowledge can also be distilled in the form of features. One line of feature-based distillation is to mimic the intermediate representations of the teacher network in terms of Euclidean distance [59], mutual information [25, 69], Wassertein distance [7], and maximum mean discrepancy of the network activations [36] respectively. Another line of feature-based distillation occurs to explore transferring the relationship between features rather than the actual features themselves, where the feature correlation can be captured by the Gram matrix [86], Taylor series expansion [58], graph [51], or quantized visual word space [38].

The transfer gap between the teacher and the student is an emerging topic in KD. To mitigate the feature-level transfer gap, MasKD [34] distils the valuable information from receptive regions that contribute to the task precision; NORM [50] conducts feature matching in a many-to-one manner, and DiffKD [35] explicitly denoises and matches features using diffusion models. When it comes to the logit-level transfer gap, DIST [33] relaxes the KL divergence in logit-based distillation with a correlation-based loss; TAKD [53] introduces multiple middle-sized teaching assistant models to guide the student; DGKD [64] improves TAKD by densely gathering all the assistant models, and SFTN [56] provides the teacher with a snapshot of the student during training. Different from these prior works, our method is motivated by the observations that teacher predictions and ground-truth labels indeed behave differently [55], arguing that this largely overlooked transfer gap makes it problematic for classical KD to encourage student predictions to simultaneously mimic the ground-truth labels and teacher predictions. Facing this dilemma, this paper proceeds from a mixture-of-experts perspective by rethinking the role of teacher predictions as latent variables. This

contributes to a student-centered paradigm where students are expected to engage themselves in decisions about what kind of knowledge they need to deal with the task at hand. Differently, exisiting KD methods can be summarized as a teacher-centered paradigm where students are expected to passively receive the knowledge presented by the teacher and imitate the teacher's behaviors However,due to the network capacity gap, the teacher would find it hard to perceive whether the knowledge it presents is the most suitable for students to absorb, which would lead to sub-optimal knowledge transfer

5.2 Mixture of Experts

The MoE model was initially proposed by [37] as a technique to combine a series of sub-models and perform conditional computation [2, 3, 14] that aims at activating different subsets of a network for different inputs. To increase the model capacity in dealing with complex data, [1,27] extend the MoE structure to the deep neural networks by proposing a deep MoE model composed of multiple layers of routers and experts. Recently, [61] simplifies the MoE layer by making the output of the gating function sparse for each example, which greatly improves the training stability and reduces the computational cost. Since then, the MoE layer with different base neural network structures has achieved tremendous success in scene parsing [24], multi-task learning [28], deep clustering [5,41,70,88], domain generalization [17,45], data generation [80] and question answering [16].

KD and MoE tend to evolve mostly independently in the literature. To the best of our knowledge, the only exceptions are [15,82]. In essence, [15,82] exploit the benefits of KD to overcome over-fitting problems [23,77] of MoE models on downstream tasks with limited data. On the contrary, this paper formulates the student's classifier as a lightweight MoE layer with the teacher's knowledge to enhance the efficiency of knowledge transfer from the teacher to the student.

6 Conclusion

In this paper, we study knowledge distillation from a novel mixture-of-experts perspective, where we leverage the teacher's knowledge from outputs and learned parameters to tackle the classification within a partition-and-classify principle. Our method comprises an input-dependent gating function that distributes subtasks to one or a few specialized experts, and multiple experts that classify the subset of samples based on sample-level student representations and class-level teacher representations. Moreover, by deriving the ELBO, our model can be formulated as an expectation-maximization algorithm and trained without requiring any other loss or regularization terms. Extensive experiments show that our method is empirically effective in not only consistently boosting the student model classification performance in various distillation settings but also improving the feature transferability.

Limitations. This paper only explores one of the parameterization schemes for the proposed mixture-of-experts framework. It will be exciting to explore

more possibilities for parameterization in the future when promoting KD from the mixture-of-experts perspective.

References

1. Ahmed, K., Baig, M.H., Torresani, L.: Network of experts for large-scale image categorization. In: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part VII 14. pp. 516–532. Springer (2016)
2. Bengio, E., Bacon, P.L., Pineau, J., Precup, D.: Conditional computation in neural networks for faster models. arXiv preprint arXiv:1511.06297 (2015)
3. Bengio, Y., Léonard, N., Courville, A.: Estimating or propagating gradients through stochastic neurons for conditional computation. arXiv preprint arXiv:1308.3432 (2013)
4. Bishop, C.: Pattern recognition and machine learning. Springer google schola **2**, 5–43 (2006)
5. Chazan, S.E., Gannot, S., Goldberger, J.: Deep clustering based on a mixture of autoencoders. In: 2019 IEEE 29th International Workshop on Machine Learning for Signal Processing (MLSP). pp. 1–6. IEEE (2019)
6. Chen, D., Mei, J.P., Zhang, H., Wang, C., Feng, Y., Chen, C.: Knowledge distillation with the reused teacher classifier. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11933–11942 (2022)
7. Chen, L., Wang, D., Gan, Z., Liu, J., Henao, R., Carin, L.: Wasserstein contrastive representation distillation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 16296–16305 (2021)
8. Chen, P., Liu, S., Zhao, H., Jia, J.: Distilling knowledge via knowledge review. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5008–5017 (2021)
9. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: International conference on machine learning. pp. 1597–1607. PMLR (2020)
10. Chen, Z., Shen, Y., Ding, M., Chen, Z., Zhao, H., Learned-Miller, E.G., Gan, C.: Mod-squad: Designing mixtures of experts as modular multi-task learners. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11828–11837 (2023)
11. Cheng, D., Liu, T., Ning, Y., Wang, N., Han, B., Niu, G., Gao, X., Sugiyama, M.: Instance-dependent label-noise learning with manifold-regularized transition matrix estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16630–16639 (2022)
12. Cheng, D., Ning, Y., Wang, N., Gao, X., Yang, H., Du, Y., Han, B., Liu, T.: Class-dependent label-noise learning with cycle-consistency regularization. Advances in Neural Information Processing Systems **35**, 11104–11116 (2022)
13. Cho, J.H., Hariharan, B.: On the efficacy of knowledge distillation. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 4794–4802 (2019)
14. Cho, K., Bengio, Y.: Exponentially increasing the capacity-to-computation ratio for conditional computation in deep learning. arXiv preprint arXiv:1406.7362 (2014)
15. Dai, D., Dong, L., Ma, S., Zheng, B., Sui, Z., Chang, B., Wei, F.: Stablemoe: Stable routing strategy for mixture of experts. arXiv preprint arXiv:2204.08396 (2022)

16. Dai, D., Jiang, W., Zhang, J., Peng, W., Lyu, Y., Sui, Z., Chang, B., Zhu, Y.: Mixture of experts for biomedical question answering. arXiv preprint arXiv:2204.07469 (2022)
17. Dai, Y., Li, X., Liu, J., Tong, Z., Duan, L.Y.: Generalizable person re-identification with relevance-aware mixture of experts. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16145–16154 (2021)
18. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the em algorithm. Journal of the royal statistical society: series B (methodological) **39**(1), 1–22 (1977)
19. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
20. Dong, C., Liu, L., Shang, J.: Toward student-oriented teacher network training for knowledge distillation. In: The Twelfth International Conference on Learning Representations (2023)
21. Du, S., Lee, J.: On the power of over-parametrization in neural networks with quadratic activation. In: International conference on machine learning. pp. 1329–1338. PMLR (2018)
22. Du, S.S., Koushik, J., Singh, A., Póczos, B.: Hypothesis transfer learning via transformation functions. Advances in neural information processing systems **30** (2017)
23. Fedus, W., Zoph, B., Shazeer, N.: Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. The Journal of Machine Learning Research **23**(1), 5232–5270 (2022)
24. Fu, H., Gong, M., Wang, C., Tao, D.: Moe-spnet: A mixture-of-experts scene parsing network. Pattern Recognition **84**, 226–236 (2018)
25. Fu, S., Yang, H., Yang, X.: Contrastive consistent representation distillation. In: The British Machine Vision Conference (2023)
26. Gou, J., Yu, B., Maybank, S.J., Tao, D.: Knowledge distillation: A survey. International Journal of Computer Vision **129**, 1789–1819 (2021)
27. Gross, S., Ranzato, M., Szlam, A.: Hard mixtures of experts for large scale weakly supervised vision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6865–6873 (2017)
28. Gupta, S., Mukherjee, S., Subudhi, K., Gonzalez, E., Jose, D., Awadallah, A.H., Gao, J.: Sparsely activated mixture-of-experts are robust multi-task learners. arXiv preprint arXiv:2204.07689 (2022)
29. Hao, Z., Guo, J., Han, K., Hu, H., Xu, C., Wang, Y.: Revisit the power of vanilla knowledge distillation: from small scale to large scale. Advances in Neural Information Processing Systems **36** (2024)
30. Hao, Z., Guo, J., Han, K., Tang, Y., Hu, H., Wang, Y., Xu, C.: One-for-all: Bridge the gap between heterogeneous architectures in knowledge distillation. Advances in Neural Information Processing Systems **36**, 79570–79582 (2023)
31. Hinton, G., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531 (2015)
32. Hossain, M.I., Akhter, S., Hong, C.S., Huh, E.N.: Single teacher, multiple perspectives: Teacher knowledge augmentation for enhanced knowledge distillation. In: The Thirteenth International Conference on Learning Representations (2025)
33. Huang, T., You, S., Wang, F., Qian, C., Xu, C.: Knowledge distillation from a stronger teacher. Advances in Neural Information Processing Systems **35**, 33716–33727 (2022)
34. Huang, T., Zhang, Y., You, S., Wang, F., Qian, C., Cao, J., Xu, C.: Masked distillation with receptive tokens. arXiv preprint arXiv:2205.14589 (2022)

35. Huang, T., Zhang, Y., Zheng, M., You, S., Wang, F., Qian, C., Xu, C.: Knowledge diffusion for distillation. arXiv preprint arXiv:2305.15712 (2023)
36. Huang, Z., Wang, N.: Like what you like: Knowledge distill via neuron selectivity transfer. arXiv preprint arXiv:1707.01219 (2017)
37. Jacobs, R.A., Jordan, M.I., Nowlan, S.J., Hinton, G.E.: Adaptive mixtures of local experts. Neural computation **3**(1), 79–87 (1991)
38. Jain, H., Gidaris, S., Komodakis, N., Pérez, P., Cord, M.: Quest: Quantized embedding space for transferring knowledge. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXI 16. pp. 173–189. Springer (2020)
39. Jin, Y., Wang, J., Lin, D.: Multi-level logit distillation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 24276–24285 (2023)
40. Khosla, A., Jayadevaprakash, N., Yao, B., Li, F.F.: Novel dataset for fine-grained image categorization: Stanford dogs. In: Proc. CVPR workshop on fine-grained visual categorization (FGVC). vol. 2. Citeseer (2011)
41. Kopf, A., Fortuin, V., Somnath, V.R., Claassen, M.: Mixture-of-experts variational autoencoder for clustering and generating from similarity-based representations on single cell data. PLoS computational biology **17**(6), e1009086 (2021)
42. Krizhevsky, A., Sutskever, I., Hinton, G.: Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems **25**(2) (2012)
43. Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images (2009)
44. Lee, J., Lee, Y., Kim, J., Kosiorek, A., Choi, S., Teh, Y.W.: Set transformer: A framework for attention-based permutation-invariant neural networks. In: International conference on machine learning. pp. 3744–3753. PMLR (2019)
45. Li, B., Shen, Y., Yang, J., Wang, Y., Ren, J., Che, T., Zhang, J., Liu, Z.: Sparse mixture-of-experts are domain generalizable learners. arXiv preprint arXiv:2206.04046 (2022)
46. Li, L., Dong, P., Li, A., Wei, Z., Yang, Y.: Kd-zero: Evolving knowledge distiller for any teacher-student pairs. Advances in Neural Information Processing Systems **36** (2024)
47. Li, X.C., Fan, W.S., Song, S., Li, Y., Yunfeng, S., Zhan, D.C., et al.: Asymmetric temperature scaling makes larger networks teach well again. Advances in Neural Information Processing Systems **35**, 3830–3842 (2022)
48. Li, X., Liu, T., Han, B., Niu, G., Sugiyama, M.: Provably end-to-end label-noise learning without anchor points. In: International conference on machine learning. pp. 6403–6413. PMLR (2021)
49. Liu, T., Tao, D.: Classification with noisy labels by importance reweighting. IEEE Transactions on pattern analysis and machine intelligence **38**(3), 447–461 (2015)
50. Liu, X., Li, L., Li, C., Yao, A.: Norm: Knowledge distillation via n-to-one representation matching. arXiv preprint arXiv:2305.13803 (2023)
51. Liu, Y., Cao, J., Li, B., Yuan, C., Hu, W., Li, Y., Duan, Y.: Knowledge distillation via instance relationship graph. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7096–7104 (2019)
52. Miles, R., Rodríguez, A.L., Mikolajczyk, K.: Information theoretic representation distillation. arXiv preprint arXiv:2112.00459 (2021)
53. Mirzadeh, S.I., Farajtabar, M., Li, A., Levine, N., Matsukawa, A., Ghasemzadeh, H.: Improved knowledge distillation via teacher assistant. In: Proceedings of the AAAI conference on artificial intelligence. vol. 34, pp. 5191–5198 (2020)

54. Müller, R., Kornblith, S., Hinton, G.E.: When does label smoothing help? Advances in neural information processing systems **32** (2019)
55. Niu, Y., Chen, L., Zhou, C., Zhang, H.: Respecting transfer gap in knowledge distillation. Advances in Neural Information Processing Systems **35**, 21933–21947 (2022)
56. Park, D.Y., Cha, M.H., Kim, D., Han, B., et al.: Learning student-friendly teacher networks for knowledge distillation. Advances in neural information processing systems **34**, 13292–13303 (2021)
57. Patrini, G., Rozza, A., Krishna Menon, A., Nock, R., Qu, L.: Making deep neural networks robust to label noise: A loss correction approach. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1944–1952 (2017)
58. Peng, B., Jin, X., Liu, J., Li, D., Wu, Y., Liu, Y., Zhou, S., Zhang, Z.: Correlation congruence for knowledge distillation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5007–5016 (2019)
59. Romero, A., Ballas, N., Kahou, S.E., Chassang, A., Gatta, C., Bengio, Y.: Fitnets: Hints for thin deep nets. arXiv preprint arXiv:1412.6550 (2014)
60. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. International journal of computer vision **115**, 211–252 (2015)
61. Shazeer, N., Mirhoseini, A., Maziarz, K., Davis, A., Le, Q., Hinton, G., Dean, J.: Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. arXiv preprint arXiv:1701.06538 (2017)
62. Silver, D., Huang, A., Maddison, C.J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., et al.: Mastering the game of go with deep neural networks and tree search. nature **529**(7587), 484–489 (2016)
63. Soltanolkotabi, M., Javanmard, A., Lee, J.D.: Theoretical insights into the optimization landscape of over-parameterized shallow neural networks. IEEE Transactions on Information Theory **65**(2), 742–769 (2018)
64. Son, W., Na, J., Choi, J., Hwang, W.: Densely guided knowledge distillation using multiple teacher assistants. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9395–9404 (2021)
65. Song, H., Kim, M., Park, D., Shin, Y., Lee, J.G.: Learning from noisy labels with deep neural networks: A survey. IEEE Transactions on Neural Networks and Learning Systems (2022)
66. Stanton, S., Izmailov, P., Kirichenko, P., Alemi, A.A., Wilson, A.G.: Does knowledge distillation really work? arXiv preprint arXiv:2106.05945 (2021)
67. Su, C.P., Tseng, C.H., Pu, B., Zhao, L., Yang, J., Chen, Z., Lee, S.J.: Ea-kd: Entropy-based adaptive knowledge distillation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 731–740 (2025)
68. Sun, S., Ren, W., Li, J., Wang, R., Cao, X.: Logit standardization in knowledge distillation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15731–15740 (2024)
69. Tian, Y., Krishnan, D., Isola, P.: Contrastive representation distillation. arXiv preprint arXiv:1910.10699 (2019)
70. Tsai, T.W., Li, C., Zhu, J.: Mice: Mixture of contrastive experts for unsupervised image clustering. In: International conference on learning representations (2021)
71. Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S.: The caltech-ucsd birds-200-2011 dataset (2011)

72. Wang, L., Yoon, K.J.: Knowledge distillation and student-teacher learning for visual intelligence: A review and new outlooks. IEEE transactions on pattern analysis and machine intelligence **44**(6), 3048–3068 (2021)
73. Wang, Q., Han, B., Liu, T., Niu, G., Yang, J., Gong, C.: Tackling instance-dependent label noise via a universal probabilistic model. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 35, pp. 10183–10191 (2021)
74. Wang, X., Wu, Z., Lian, L., Yu, S.X.: Debiased learning from naturally imbalanced pseudo-labels. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14647–14657 (2022)
75. Wei, S., Luo, C., Luo, Y.: Scaled decoupled distillation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15975–15983 (2024)
76. Wightman, R., Touvron, H., Jégou, H.: Resnet strikes back: An improved training procedure in timm. arXiv preprint arXiv:2110.00476 (2021)
77. Wu, L., Liu, M., Chen, Y., Chen, D., Dai, X., Yuan, L.: Residual mixture of experts. arXiv preprint arXiv:2204.09636 (2022)
78. Xia, X., Liu, T., Han, B., Wang, N., Gong, M., Liu, H., Niu, G., Tao, D., Sugiyama, M.: Part-dependent label noise: Towards instance-dependent label noise. Advances in Neural Information Processing Systems **33**, 7597–7610 (2020)
79. Xia, X., Liu, T., Wang, N., Han, B., Gong, C., Niu, G., Sugiyama, M.: Are anchor points really indispensable in label-noise learning? Advances in neural information processing systems **32** (2019)
80. Xia, X., Yang, W., Ren, J., Li, Y., Zhan, Y., Han, B., Liu, T.: Pluralistic image completion with probabilistic mixture-of-experts. arXiv preprint arXiv:2205.09086 (2022)
81. Xiang, L., Gao, J., Xu, C.: Evidential knowledge distillation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2814–2824 (2025)
82. Xue, F., He, X., Ren, X., Lou, Y., You, Y.: One student knows all experts know: From sparse to dense. arXiv preprint arXiv:2201.10890 (2022)
83. Yang, J., Martinez, B., Bulat, A., Tzimiropoulos, G., et al.: Knowledge distillation via softmax regression representation learning. International Conference on Learning Representations (ICLR) (2021)
84. Yang, S., Yang, E., Han, B., Liu, Y., Xu, M., Niu, G., Liu, T.: Estimating instance-dependent bayes-label transition matrix using a deep neural network. In: International Conference on Machine Learning. pp. 25302–25312. PMLR (2022)
85. Yang, Z., Zeng, A., Yuan, C., Li, Y.: From knowledge distillation to self-knowledge distillation: A unified approach with normalized loss and customized soft labels. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 17185–17194 (2023)
86. Yim, J., Joo, D., Bae, J., Kim, J.: A gift from knowledge distillation: Fast optimization, network minimization and transfer learning. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4133–4141 (2017)
87. Zaheer, M., Kottur, S., Ravanbakhsh, S., Poczos, B., Salakhutdinov, R.R., Smola, A.J.: Deep sets. Advances in neural information processing systems **30** (2017)
88. Zhang, D., Sun, Y., Eriksson, B., Balzano, L.: Deep unsupervised clustering using mixture of autoencoders. arXiv preprint arXiv:1712.07788 (2017)
89. Zhao, B., Cui, Q., Song, R., Qiu, Y., Liang, J.: Decoupled knowledge distillation. In: Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition. pp. 11953–11962 (2022)

90. Zheng, K., YANG, E.H.: Knowledge distillation based on transformed teacher matching. In: The Twelfth International Conference on Learning Representations (2024), https://openreview.net/forum?id=MJ3K7uDGGl
91. Zhou, H., Song, L., Chen, J., Zhou, Y., Wang, G., Yuan, J., Zhang, Q.: Rethinking soft labels for knowledge distillation: A bias-variance tradeoff perspective. arXiv preprint arXiv:2102.00650 (2021)
92. Zhu, W., Peng, B., Yan, W.Q.: Dual knowledge distillation on multiview pseudo labels for unsupervised person re-identification. IEEE Transactions on Multimedia **26**, 7359–7371 (2024)
93. Zhu, W., Peng, B., Yan, W.Q.: Deep inductive and scalable subspace clustering via nonlocal contrastive self-distillation. IEEE Transactions on Circuits and Systems for Video Technology (2025)
94. Zhu, Y., Wang, Y.: Student customized knowledge distillation: Bridging the gap between student and teacher. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5057–5066 (2021)