Knowledge Distillation with Auxiliary Variable

Bo Peng [1] **Zhen Fang** [1] **Guangquan Zhang** [1] **Jie Lu** [1]

Abstract

Knowledge distillation (KD) provides an efficient framework for transferring knowledge from a teacher model to a student model by aligning their predictive distributions. The existing KD methods adopt the same strategy as the teacher to formulate the student's predictive distribution. However, employing the same distribution-modeling strategy typically causes sub-optimal knowledge transfer due to the discrepancy in model capacity between teacher and student models. Designing student-friendly teachers contributes to alleviating the capacity discrepancy, while it requires either complicated or student-specific training schemes. To cast off this dilemma, we propose to introduce an auxiliary variable to promote the ability of the student to model predictive distribution. The auxiliary variable is defined to be related to target variables, which will boost the model prediction. Specifically, we reformulate the predictive distribution with the auxiliary variable, deriving a novel objective function of KD. Theoretically, we provide insights to explain why the proposed objective function can outperform the existing KD methods. Experimentally, we demonstrate that the proposed objective function can considerably and consistently outperform existing KD methods.

1. Introduction

Over the past decades, deep learning has shown its significance by boosting the performance of various real-world tasks (Hassaballah & Awad, 2020; Hupkes et al., 2023). The effectiveness of deep learning generally comes at the expense of huge computational complexity and massive storage requirements. This restricts the deployment of large-scale models (teachers) in real-time applications where lightweight models (students) are preferable due to limited resources (Li et al., 2023). In this context, knowledge distillation (KD) (Gou et al., 2021; Wang & Yoon, 2021) is introduced to transfer knowledge from a teacher to a student model. Conventionally, KD is approached by minimizing the Kullback-Leibler (KL) divergence between predictive distributions of the teacher and student (Hinton et al., 2015). To implement this vision, an intuitive yet commonly accepted approach, initially introduced in Hinton et al. (2015), is that the student follows the pre-trained teacher to formulate predictive posterior probabilities with logit outputs. Consequently, knowledge can be distilled from the teacher to the student by matching their logit outputs.

This logit-matching approach, however, is challenged by the counter-intuitive observations (Cho & Hariharan, 2019; Stanton et al., 2021). Specifically, a larger teacher does not necessarily increase a student's accuracy compared to a relatively smaller teacher. This is attributed to the capacity gap between the two models (Huang et al., 2022a; Mirzadeh et al., 2020) since the discrepancy between their predictions can be significantly large. Thus, directly aligning their predictive distributions would lead to sub-optimal knowledge transfer and even disturb the training of the student.

Advanced methods introduce a novel direction to go beyond the logit-matching approach. These methods develop student-friendly teachers to shrink the capacity gap. For instance, TAKD (Mirzadeh et al., 2020) introduces multiple middle-sized teaching assistant models to guide the student; DGKD (Son et al., 2021) improves TAKD by densely gathering all the assistant models; SFTN (Park et al., 2021) provides the teacher with a snapshot of the student during training. Despite remarkable progress, these methods need to resort to either sophisticated or student-specific training schemes, which weakens the practicability and universality.

In order to get out of this dilemma, this paper proceeds from a different perspective and raises the following important yet under-explored question: *is it possible to compensate for the weaker capacity of the student with a stronger ability to model predictive distribution*? We give an affirmative answer to this question by introducing an auxiliary variable for modeling predictive distributions. The insight is to leverage a suitable auxiliary variable to promote the ability of the student to model predictive distribution.

[1]Faculty of Engineering & Information Technology, University of Technology Sydney, Sydney, Australia. Correspondence to: Zhen Fang <Zhen.Fang@uts.edu.au>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

At a high level, promoting the ability of the student to model predictive distribution requires the auxiliary variable to bring external knowledge. Inspired by the success of contrastive clustering (Shen et al., 2021; Tsai et al., 2021), we introduce instance membership as the auxiliary variable that is defined to be related to labels. Thanks to the correlation between labels and the auxiliary variable, we reformulate predictive posterior probabilities of the student model (see Eqn. (2)), deriving a novel learning framework for KD (see Eqn. (5)). Consequently, the novel learning framework can exploit instance-level semantics as a stepping stone to guide the student to model the predictive distribution. Thus, the derived novel framework can promote the ability of the student to model predictive distribution thanks to the external knowledge introduced by the auxiliary variable.

To realize the derived learning framework, we design an effective parameterization to generalize KD into a methodologically unified paradigm that elegantly conjoins logit-level and feature-level knowledge via a single objective function (see Eqn. (10)). The parameterization can be theoretically supported by Theorems 3.5 and 3.6. Surprisingly, our realization forges a connection between logit matching and feature matching by showing that they can be unified into the same optimization objective, though they appear to be different regarding motivation and methodology.

Theoretically, we justify the proposed parameterization via Theorems 3.5 and 3.6, which draw a connection to the mutual information neural estimator (MINE) (Belghazi et al., 2018) and the evidence lower bound (ELBO) of the log-likelihood, respectively. The theoretical insights are twofold. In particular, our method provably learns a student feature space that conforms to a deterministic distribution. This enables the student to predict class membership via a Bayes-based rule without resorting to a parametric classifier that implicitly makes a strong distributional assumption (Grathwohl et al., 2019). Due to the absence of the classification layer, knowledge is transferred to learn representations, which would make the learned representations more generalizable. Fortunately, this is consistent with our experimental results (see Section 4.3). Besides, our method intrinsically blends hard positive/negative mining into knowledge distillation, where the hard positive/negative teacher features are automatically mined to dominate the optimization of student features.

Empirically, extensive experiments demonstrate that our method establishes state-of-the-art performance. For instance, we achieve 73.04% Top-1 accuracy with ResNet18 student and ResNet34 teacher on ImageNet, surpassing DiffKD (Huang et al., 2023) by 0.82%; while on linear probing, ours outperforms DKD (Yeh et al., 2022) by 1.5% and 1.9% w.r.t Top-1 accuracy on STL-10 and Tiny-ImageNet respectively. We also validate our method on the self-distillation setting, and ours significantly outperforms IPWD (Niu et al., 2022) by 1.24% with DenseNet121 on CIFAR-100.

2. Related Work

Knowledge distillation is the process of using a teacher model to improve the performance of a student model. In its conventional form, one trains the student to fit the teacher's predictive distribution. Hinton et al. (2015) popularizes this solution by formulating it as logit matching. MLD (Jin et al., 2023) extends logit matching not only at the instance level but also at the batch and class levels. Besides distillation on logits, some works aim at transferring knowledge from intermediate features. FitNet (Romero et al., 2014) mimics the intermediate features of a teacher network in the Euclidean metric space, which opens a door to feature matching (Chen et al., 2021b; Lin et al., 2022; Heo et al., 2019; Zagoruyko & Komodakis, 2016; Tian et al., 2019; Chen et al., 2021a). To the best of our knowledge, the ideas of "logits as knowledge" and "features as knowledge" in the literature are either explored separately or conjoined in a decoupled manner. This paper unifies the transfer of logit-level and feature-level knowledge into a unified probabilistic framework both methodologically and theoretically.

The transfer gap between the teacher and the student is an emerging topic in KD. DIST (Huang et al., 2022a) relaxes the KL divergence in logit matching with a correlation-based loss. However, the Pearson correlation in DIST only has shift and scale invariances. On the other hand, Mirzadeh et al. (2020); Park et al. (2021); Wang et al. (2022a); Li et al. (2021) attribute the transfer gap to the capacity gap between the two models. They address this by making a teacher network hold better transferable knowledge, which suffers from complex training schemes and heavy computational costs. Different from Huang et al. (2022b); Liu et al. (2023); Huang et al. (2023), this paper addresses the transfer gap of the predictive distribution. From this perspective, SimKD (Chen et al., 2022) and SRRL (Yang et al., 2021), to some extent, can be viewed as two earlier attempts in this direction. Orthogonal to them, our work shows that the capacity gap can be effectively reduced by equipping the student with a stronger ability to model the predictive distribution.

Our work is also related to KD decomposition. Li et al. (2022) decompose the efficacy of KD into three parts: correct guidance, smooth regularization, and class discriminability. Tang et al. (2020) decomposes the knowledge into universal knowledge, domain knowledge, and gradient rescaling. Zhou et al. (2021) utilizes bias-variance decomposition to analyze KD and discovers regularization samples that increase bias and decrease variance. Yeh et al. (2022) rewrites KD as a decoupled sum of target class knowledge distillation and non-target class knowledge distillation. Different from these methods, this paper reformulates KD by

introducing the label-related auxiliary variable.

3. Methodology

3.1. Preliminaries

Notations. We write vectors as bold lowercase characters. The operation $\circ$ calculates the cosine similarity between two vectors. Considering K-way multi-class classification as a case study, we are given a training dataset $\mathcal{D} = \{\mathcal{X}, \mathcal{Y}\} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{M}$ that contains $\mathbf{x}_i$ as the i-th sample independently drawn from $\mathcal{X}$ and $y_i \in \mathcal{Y} := \{1, .., K\}$ as the corresponding class membership. We define $p(k|\mathbf{x}_i)$ and $\mathbf{z}_i$ as the posterior probability and intermediate feature of the sample $\mathbf{x}_i$ produced by a neural network respectively.

Knowledge Distillation involves transferring dark knowledge from a teacher model to a student model. Conventional KD (Hinton et al., 2015) proposes to use the soft labels produced by the teacher as an additional target for the student to match, which is approached by minimizing the KL divergence between the predictive distributions of the student and the teacher. Since the teacher model is typically pre-trained and fixed in the KD context, the conventional KD objective can be reduced to the following form:

$$\mathcal{L}_i^{\mathrm{KD}} = -\mathbb{E}_{p^{\mathcal{T}}(k|\mathbf{x}_i)} \left[\log p^{\mathcal{S}}(k|\mathbf{x}_i)\right]. \tag{1}$$

In Eqn. (1), we have used $\mathcal{T}$ and $\mathcal{S}$ as superscripts to indicate the teacher and student model respectively, which, unless explicitly stated, is a default setting in the rest of this paper.

3.2. Knowledge Distillation with Auxiliary Variable

To mitigate the transfer gap between the teacher and the student, we propose to promote the ability of the student to model predictive distribution, relaxing assumptions about the teacher. We introduce a suitable auxiliary variable to promote the ability of the student to model predictive distribution, sharing the same spirit as Zhang et al. (2022). The insight is straightforward. The auxiliary variable can be employed as a stepping stone to guide the student in modeling the predictive distribution. Thus, the auxiliary variable for KD is defined as being related to labels.

In this work, we realize the auxiliary variable as the instance membership s_i, inspired by Shen et al. (2021) and Tsai et al. (2021). Specifically, we reformulate the conventional KD objective defined in Eqn. (1) from a probabilistic perspective where the instance membership s_i serves as a latent variable. In this way, according to Bayes and total probability laws, we can naturally rewrite $p^{\mathcal{S}}(k|\mathbf{x}_i)$ as follows:

$$p^{\mathcal{S}}(k|\mathbf{x}_i) = \frac{p^{\mathcal{S}}(s_i|\mathbf{x}_i)p^{\mathcal{S}}(k|\mathbf{x}_i, s_i)}{p^{\mathcal{S}}(s_i|\mathbf{x}_i, k)}, \tag{2}$$

which is built upon the correlation between the label k and the instance membership s_i.

Combining Eqn. (2) with Eqn. (1), we can reformulate the objective function of conventional KD as:

$$\mathcal{L}_i^{\mathrm{KD}} = -\log p^{\mathcal{S}}(s_i|\mathbf{x}_i) - \mathbb{E}_{p^{\mathcal{T}}(k|\mathbf{x}_i)} \left[\log \frac{p^{\mathcal{S}}(k|\mathbf{x}_i, s_i)}{p^{\mathcal{S}}(s_i|\mathbf{x}_i, k)}\right], \tag{3}$$

where the second term in Eqn. (3) makes the back-propagation through the discrete entries k and s_i infeasible. Consequently, we view the logarithm as a whole and calculate it as:

$$\log \frac{p^{\mathcal{S}}(k|\mathbf{x}_i, s_i)}{p^{\mathcal{S}}(s_i|\mathbf{x}_i, k)} = \log \frac{p^{\mathcal{S}}(\mathbf{x}_i|k)p^{\mathcal{S}}(k)}{p^{\mathcal{S}}(\mathbf{x}_i|s_i)p^{\mathcal{S}}(s_i)}. \tag{4}$$

The detailed derivation of Eqn. (4) is given in Appendix A. Same as Jiang et al. (2023), we denote $p^{\mathcal{S}}(k) = |\mathcal{D}_k|/M$ where $\mathcal{D}_k = \{(\mathbf{x}_i, y_i)|(\mathbf{x}_i, y_i) \in \mathcal{D}, y_i = k\}$. Thanks to the fact that $p^{\mathcal{T}}(k|\mathbf{x}_i)$ is fixed in the KD task, the constant term $p^{\mathcal{S}}(s_i)$ can be omitted during optimization though the true distribution $p^{\mathcal{S}}(s_i)$ is unknown. Consequently, we arrive at a mathematical equivalence to the conventional KD objective (up to a constant), i.e.,

$$\hat{\mathcal{L}}_i^{\mathrm{KD}} = -\log \frac{p^{\mathcal{S}}(s_i|\mathbf{x}_i)}{p^{\mathcal{S}}(\mathbf{x}_i|s_i)} - \mathbb{E}_{p^{\mathcal{T}}(k|\mathbf{x}_i)} \left[\log p^{\mathcal{S}}(\mathbf{x}_i|k)\right]. \tag{5}$$

In Section 3.3, we will elaborate on how we effectively parameterize each term in Eqn. (5) to fit the KD task.

3.3. Effective Parameterization

Parameterizing $p^{\mathcal{S}}(s_i|\mathbf{x}_i)$. Drawing inspiration from Shen et al. (2021); Tsai et al. (2021), we organize $p^{\mathcal{S}}(s_i|\mathbf{x}_i)$ as an instance discrimination task (Wu et al., 2018; Chen et al., 2020; He et al., 2020; Yeh et al., 2022) where the sample $\mathbf{x}_i$ discriminates itself from negative candidates with the identity s_i as the identifier. Since we have access to a teacher model, it is tempting to adopt teacher features and class labels for unbiased negative sampling. Formally, we implement this vision by formulating $p^{\mathcal{S}}(s_i|\mathbf{x}_i)$ as:

$$p^{\mathcal{S}}(s_i|\mathbf{x}_i) \triangleq \frac{\exp \phi_\alpha(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_i^{\mathcal{S}})}{\exp \phi_\alpha(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_i^{\mathcal{S}}) + \sum_{j \in \mathcal{N}_i} \exp \phi_\alpha(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_j^{\mathcal{T}})}, \tag{6}$$

where $\mathcal{N}_i = \left\{j|\mathbf{z}_n^{\mathcal{T}} \in \mathcal{Z}^{\mathcal{T}} := \left\{\mathbf{z}_1^{\mathcal{T}}, \cdots, \mathbf{z}_M^{\mathcal{T}}\right\}, y_j \neq y_i\right\}$ stores the index of all the negative teacher features of $\mathbf{x}_i$. The pair-wise similarity measure $\phi_\alpha(\cdot, \cdot)$ is defined by:

$$\phi_\alpha(\mathbf{z}^{\mathcal{S}}, \mathbf{z}^{\mathcal{T}}) \triangleq \mathbf{h}^{\mathcal{S}} \circ \mathbf{z}^{\mathcal{T}}/\alpha, \quad \mathbf{h}^{\mathcal{S}} = g(\mathbf{z}^{\mathcal{S}}), \tag{7}$$

where $\alpha > 0$ and a projector $g(\cdot)$ is introduced to match feature dimensions at a relatively small cost.

Parameterizing $p^{\mathcal{S}}(\mathbf{x}_i|s_i)$. Given that $p^{\mathcal{S}}(\mathbf{x}_i|s_i)$ reflects the dependence of the identification of $\mathbf{x}_i$ on its instance membership s_i, a desirable parameterization of $p^{\mathcal{S}}(\mathbf{x}_i|s_i)$ should serve as a regularizer to avoid the student from

naively maximizing $p^{\mathcal{S}}(s_i|\mathbf{x}_i)$ by encoding only instance-specific information into the feature space. With classification as the target task, an intuition is to encourage the learned features to respect the underlying inter-class data structures, which can be easily achieved by de-differentiating the sample $\mathbf{x}_i$ from its positive candidates. In analogy to Eqn. (6), $p^{\mathcal{S}}(\mathbf{x}_i|s_i)$ takes the following form:

$$P^{\mathcal{S}}(\mathbf{x}_i|s_i) \triangleq \frac{\exp \phi_\beta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_i^{\mathcal{S}})}{\exp \phi_\beta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_i^{\mathcal{S}}) + \sum_{j\in\mathcal{P}_i} \exp \phi_\beta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_j^{\mathcal{T}})}, \quad (8)$$

where $\mathcal{P}_i = \left\{j|\mathbf{z}_j^{\mathcal{T}} \in \mathcal{Z}^{\mathcal{T}}, y_j = y_i\right\}$ denotes the index set of all the positive features for the sample $\mathbf{x}_i$.

Parameterizing $p^{\mathcal{S}}(\mathbf{x}_i|k)$. This is motivated by the fact that $p^{\mathcal{S}}(\mathbf{x}_i) = \sum_{k=1}^{K} p^{\mathcal{S}}(\mathbf{x}_i|k)p^{\mathcal{S}}(k)$. We then define $p^{\mathcal{S}}(\mathbf{x}_i|k)$ as a class-conditional probability density function. Note that the parameterization of $p^{\mathcal{S}}(\mathbf{x}_i|k)$ is generic to the choice of distributional assumptions for the feature space while this paper, following Ming et al. (2022), focuses on an exemplar based on the von Mises-Fisher (vMF) distribution, i.e.,

$$p^{\mathcal{S}}(\mathbf{x}_i|k) \triangleq C_d(\kappa^{-1}) \exp\left(\mathbf{z}_i^{\mathcal{S}} \circ \boldsymbol{\mu}_k/\kappa\right), \quad (9)$$

where $\kappa > 0$ and the class prototype $\boldsymbol{\mu}_k \in \mathbb{R}^d$ denotes the mean vector of the class k with $\|\boldsymbol{\mu}_k\|_2 = 1$. The normalization constant $C_d(\kappa)$ is calculated based on κ and d: $C_d(\kappa) = \kappa^{d/2-1}/\left[(2\pi)^{d/2} I_{d/2-1}(\kappa)\right]$ where I_d denotes the modified Bessel function of the first kind and order d.

Benefiting from the parameterization above, we have the following as the objective function of our proposed AuxKD[1]:

$$\mathcal{L}_i^{\text{AuxKD}} = -\log \frac{\exp \phi_\beta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_i^{\mathcal{S}}) + \sum_{j\in\mathcal{P}_i} \exp \phi_\beta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_j^{\mathcal{T}})}{\exp \phi_\alpha(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_i^{\mathcal{S}}) + \sum_{j\in\mathcal{N}_i} \exp \phi_\alpha(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_j^{\mathcal{T}})} - \mathbb{E}_{p^{\mathcal{T}}(k|\mathbf{x}_i)}\left[\mathbf{z}_i^{\mathcal{S}} \circ \boldsymbol{\mu}_k/\kappa\right]. \quad (10)$$

For a fair comparison, we formulate $p^{\mathcal{T}}(k|\mathbf{x}_i)$ in accordance with prior works (Hinton et al., 2015; Zhao et al., 2022; Zhou et al., 2021; Niu et al., 2022; Hao et al., 2023), i.e.,

$$p^{\mathcal{T}}(k|\mathbf{x}_i) \triangleq \frac{\exp(\mathbf{e}_{i,k}^{\mathcal{T}}/\sigma)}{\sum_{j=1}^{K} \exp(\mathbf{e}_{i,j}^{\mathcal{T}}/\sigma)}, \quad (11)$$

where $\sigma > 0$ and $\mathbf{e}_{i,k}^{\mathcal{T}}$ denotes to the teacher's logit of the k-th class for the sample $\mathbf{x}_i$.

Interestingly, Eqn. (10) presents a methodologically unified KD paradigm that simultaneously exploits feature-level and logit-level knowledge from the teacher as guidance via a single objective function. Notably, in the current KD literature, logit-based distillation and feature-based distillation have evolved mostly independently. As we will show in Section. 3.4, with Eqn. (10) as the steppingstone, the two tracks of distillation actually work in a similar mechanism though they seem to be distinct in their designed objective functions, which results in a unified insight into KD.

[1]Since constant terms do not contribute to backpropagation, we have omitted them in Eqn. (10) for brevity.

3.4. A Unified Insight into Knowledge Distillation

Definition 3.1 (Logit Matching (Hinton et al., 2015))**.** Let $\mathbf{e}_{i,k}^{\mathcal{S}}$ denotes the student's logit of the k-th class for the sample $\mathbf{x}_i$, logit matching (LM) can be formally expressed as:

$$\mathcal{L}_i^{\text{LM}} = -\mathbb{E}_{p^{\mathcal{T}}(k|\mathbf{x}_i)}\left[\log \frac{\exp(\mathbf{e}_{i,k}^{\mathcal{S}}/\sigma)}{\sum_{j=1}^{K} \exp(\mathbf{e}_{i,j}^{\mathcal{S}}/\sigma)}\right]. \quad (12)$$

Remark 3.2 (Relation to Logit Matching). Logit matching can be considered as another instantiation of conventional KD in Eqn. (5), where, orthogonal to our implementation, $p^{\mathcal{S}}(k|\mathbf{x}_i)$ in Eqn. (1) is parameterized with the student's logit outputs in the same manner as $p^{\mathcal{T}}(k|\mathbf{x}_i)$ in Eqn. (11).

Definition 3.3 (Feature Matching)**.** Formally, let $w^{\mathcal{S}}(\cdot)$ and $w^{\mathcal{T}}(\cdot)$ be the feature transform functions for the student and teacher respectively, according to Liu et al. (2023) and Heo et al. (2019), feature matching (FM) can be generalized as:

$$\mathcal{L}_i^{\text{FM}} = d(w^{\mathcal{S}}(\mathbf{z}_i^{\mathcal{S}}), w^{\mathcal{T}}(\mathbf{z}_i^{\mathcal{T}})), \quad (13)$$

where the distance metrics $d(\cdot,\cdot)$ can be ℓ_2-norm distance (Romero et al., 2014; Huang et al., 2023), ℓ_1-norm distance (Li et al., 2021), mutual information (Tian et al., 2019; Fu et al., 2023), and Wasserstein distance (Chen et al., 2021a).

Theorem 3.4 (Relation to Feature Matching)**.** *Without loss of generalization, let us use $\ell_2(\cdot)$ to indicate the ℓ_2 normalization operation and define*

$$d(w^{\mathcal{S}}(\mathbf{z}_i^{\mathcal{S}}), w^{\mathcal{T}}(\mathbf{z}_i^{\mathcal{T}})) = \frac{1}{2\beta}\left\|\ell_2(g(\mathbf{h}_i^{\mathcal{S}})) - \ell_2(\mathbf{z}_i^{\mathcal{T}})\right\|_2^2 = \frac{1}{\beta}(1 - \mathbf{h}_i^{\mathcal{S}} \circ \mathbf{z}_i^{\mathcal{T}}).$$

In the extreme case where $\alpha \to +\infty$ (Assumption A1), $\kappa \to +\infty$ (Assumption A2), and the class-level information in Eqn. (8) is omitted (Assumption A3), we then have

$$\mathcal{L}_i^{\text{FM}} \geq \mathcal{L}_i^{\text{AuxKD}} + const.$$

To keep the main content concise, we detail the derivation in Appendix B. Theorem 3.4 shows that our method subsumes feature matching as a special case of itself regardless of the specific form of the distance metrics $d(\cdot,\cdot)$. With the help of Remark 3.2 and Theorem 3.4, one could conclude that logit matching and feature matching essentially optimize the conventional KD objective in Eqn. (1) though they indeed seem to be quite distinct regarding their objective functions.

3.5. Theoretical Analysis

In this section, we provide theoretical justification for our proposed parameterization in Section 3.3. As an overview, we show that minimizing $\mathcal{L}_i^{\text{AuxKD}}$ in Eqn. (10) provably maximizes the mutual information between the student's and teacher's features (Theorem 3.5) while shaping the student feature space towards a vMF mixture distribution (Theorem 3.6). The proof of the two theorems is given in Appendix C.

Theorem 3.5. *Let $\eta(\mathbf{z}^{\mathcal{S}}, \mathbf{z}^{\mathcal{T}}) = \mathbf{h}^{\mathcal{S}} \circ \mathbf{z}^{\mathcal{T}}/\tau$. If $\alpha = \beta = \tau$, then the first term of $\mathcal{L}_i^{\text{AuxKD}}$ in Eqn. (10), i.e.,*

$$\log \frac{\exp \phi_\beta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_i^{\mathcal{S}}) + \sum_{j\in\mathcal{P}_i} \exp \phi_\beta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_j^{\mathcal{T}})}{\exp \phi_\alpha(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_i^{\mathcal{S}}) + \sum_{j\in\mathcal{N}_i} \exp \phi_\alpha(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_j^{\mathcal{T}})}$$
$$\leq \frac{1}{|\mathcal{P}_i|} \sum_{j\in\mathcal{P}_i} \eta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_j^{\mathcal{T}}) - \log \frac{1}{|\mathcal{N}_i|} \sum_{j\in\mathcal{N}_i} \exp \eta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_j^{\mathcal{T}}),$$

is equivalent to the MINE estimator (Belghazi et al., 2018).

In addition to implying that our method maximizes a lower bound on the mutual information neural estimation (MINE) (Belghazi et al., 2018), Theorem 3.5 formally supports the rationality of simply assigning the same value to α and β.

Theorem 3.6. *The second term of $\mathcal{L}_i^{\text{AuxKD}}$ in Eqn. (10) is an evidence lower bound (ELBO) on the marginal likelihood of the sample $\mathbf{x}_i$ (up to a constant), which can be written as*

$$\begin{aligned}\log p^{\mathcal{S}}(\mathbf{x}_i) &= \log \mathbb{E}_{p^{\mathcal{S}}(k)} \left[p^{\mathcal{S}}(\mathbf{x}_i|k)\right] \\ &= \log \mathbb{E}_{p^{\mathcal{S}}(k)} \left[C_d(\kappa^{-1}) \exp\left(\mathbf{z}_i^{\mathcal{S}} \circ \boldsymbol{\mu}_k/\kappa\right)\right] \\ &\geq \mathbb{E}_{p^{\mathcal{T}}(k|\mathbf{x}_i)} \left[\mathbf{z}_i^{\mathcal{S}} \circ \boldsymbol{\mu}_k/\kappa\right] + const.\end{aligned}$$

Theorem 3.6 states that optimizing the second term of Eqn. (10) drives the deep features to follow the pre-defined distribution. This means that the strong distributional assumption behind Eqn. (9) can be naturally satisfied during optimization without requiring explicit constraints. Moreover, since this ELBO is built upon the teacher's predictions $p^{\mathcal{T}}(k|\mathbf{x}_i)$, we have injected the class relationship prior learned by the teacher into the distribution modelling of the student feature space to preserve smoothness between classes.

3.6. Training and Inference

Our overall training objective for the sample $\mathbf{x}_i$ linearly combines the classification objective $\mathcal{L}_i^{\text{CLS}}$ and the reformulated KD objective $\mathcal{L}_i^{\text{AuxKD}}$ in Eqn. (10):

$$\mathcal{L}_i^{\text{Overall}} = \mathcal{L}_i^{\text{CLS}} + \lambda \mathcal{L}_i^{\text{AuxKD}}, \tag{14}$$

where $\lambda > 0$ is a balancing factor. Since we have shown in Theorem 3.6 that $\mathcal{L}_i^{\text{AuxKD}}$ helps to shape the student feature space towards a vMF Mixture distribution, the classification for a sample $\mathbf{x}_i$ can take place with a Bayes-based rule instead of a parametric softmax layer. As such, the classification objective $\mathcal{L}_i^{\text{CLS}}$ naturally turns to be:

$$\begin{aligned}\mathcal{L}_i^{\text{CLS}} &= -\log \frac{p^{\mathcal{S}}(y_i) p^{\mathcal{S}}(\mathbf{x}_i|y_i)}{\sum_{k=1}^K p^{\mathcal{S}}(k) p^{\mathcal{S}}(\mathbf{x}_i|k)} \\ &= -\log \frac{p^{\mathcal{S}}(y_i) \exp\left(\mathbf{z}_i \circ \boldsymbol{\mu}_{y_i}/\kappa\right)}{\sum_{k=1}^K p^{\mathcal{S}}(k) \exp\left(\mathbf{z}_i \circ \boldsymbol{\mu}_k/\kappa\right)}.\end{aligned} \tag{15}$$

Inference with the trained student for a test-time sample $\mathbf{x} \in \mathcal{X}$ only requires to compute the predicted label $\hat{y}$:

$$\begin{aligned}\hat{y} &= \arg\max_{j\in\mathcal{Y}} \frac{p^{\mathcal{S}}(j) p^{\mathcal{S}}(\mathbf{x}|j)}{\sum_{k=1}^K p^{\mathcal{S}}(k) p^{\mathcal{S}}(\mathbf{x}|k)} \\ &= \arg\max_{j\in\mathcal{Y}} p^{\mathcal{S}}(j) \exp\left(\mathbf{z} \circ \boldsymbol{\mu}_j/\kappa\right).\end{aligned} \tag{16}$$

Algorithm 1 knowledge distillation with auxiliary variable

Input: Training dataset $\mathcal{D}$, Pre-trained teacher $\mathcal{T}$, Randomly initialized student parameters $\boldsymbol{\theta}$, SGD optimizer $\boldsymbol{\Gamma}(\cdot)$ and empty queue $\mathcal{Q}$
Output: Well-taught student $\mathcal{S}$
repeat
  Randomly select a batch $\mathcal{B}$ from $\mathcal{D}$
  if $\mathcal{Q}$ is full **then**
    Dequeue the oldest batch of teacher features in $\mathcal{Q}$
  end if
  Enqueue teacher features of samples in $\mathcal{B}$
  for each $(\mathbf{x}_i, y_i)$ in $\mathcal{B}$ **do**
    Construct $\mathcal{N}_i$ and $\mathcal{P}_i$ from teacher features in $\mathcal{Q}$
    $\mathcal{L}_i^{\text{Overall}} \leftarrow$ Eqn. (14)
  end for
  $\mathcal{L} = \frac{1}{|\mathcal{B}|} \sum_{(\mathbf{x}_i, y_i)\in\mathcal{B}} \mathcal{L}_i^{\text{Overall}}$
  $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \boldsymbol{\Gamma}(\nabla_{\boldsymbol{\theta}} \mathcal{L})$
until convergence or reaching max iteration

Versatility. We emphasize that AuxKD can be effortlessly integrated into existing student models with only removing the parametric softmax layer. However, this minimal architecture change, as a direct consequence of Theorem 3.6, contributes to a clear explanation of the classification as the proximity of test data to the class prototypes in the feature space without slowing the inference speed. By contrast, parametric classifiers, whose parameters are abstract and detached from the physical nature of the problem being modelled (Tang et al., 2020), could not lend to an explanation that humans can process (Li et al., 2018).

Implementation. We observe that it is impractical for Eqns. (6) and (8) to cache the training dataset $\mathcal{X}$ to build negative and positive feature sets respectively. To address the problem, we implement a L-sized ($L \ll M$) queue $\mathcal{Q}$ that stores the teacher features of samples in previous batches, following MoCo (He et al., 2020) closely. The overall training algorithm is summarized in Algorithm 1.

Table 1: Top-1 accuracy (%) on CIFAR-100, Homogenous Architecture. The best result is highlighted in boldface.

Distillation Manner	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
Logit	DKD	76.24	74.81	71.97	74.11	76.32	74.68
	IPWD	—	74.64	71.32	73.91	76.03	—
	WSLD	—	74.48	72.15	74.12	76.05	—
	MLD	76.63	75.35	72.19	74.11	77.08	75.18
	DIST	—	74.73	71.75	—	76.31	—
Feature	ReviewKD	76.12	75.09	71.89	73.89	75.63	74.84
	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
	CoCoRD	75.48	75.17	71.74	74.10	75.29	73.99
	NROM	75.65	74.82	71.35	73.67	76.49	73.95
	DiffKD	—	74.09	71.92	—	76.72	—
Logit + Feature	SSRL	75.96	74.75	71.40	73.80	75.92	74.40
	SSKD	76.04	76.13	71.49	—	76.20	75.33
	Ours	**77.50**	**76.68**	**73.21**	**75.33**	**77.47**	**75.65**

3.7. More Discussions

This section explains why we prefer the methodologically unified KD paradigm in Eqn. (10) to a linear combination of logit matching and feature matching that has been widely accepted in the state-of-the-art, mainly in three aspects.

1) Free from Assumptions. The linear combination tends to involve the use of a parametric classifier, which implicitly makes a strong distributional assumption of the learned feature space being Gibbs-Boltzmann (Grathwohl et al., 2019; LeCun et al., 2006). This problem can not be addressed by naively replacing the parametric classifier with the nonparametric one in Eqn. (16). As implied by Theorem 3.6, optimizing $\mathcal{L}_i^{\text{AuxKD}}$ makes the learned features to conform to the distributional assumption behind Eqn. (16).

2) Hard Positive/Negative Mining We show in Appendix D that $\mathcal{L}_i^{\text{AuxKD}}$ in Eqn. (10) induces a gradient structure that gives rise to implicit hard positive/negative mining, where the gradient contributions from hard positives/negatives (i.e., ones against which continuing to contrast the anchor greatly benefits the student) are large while those for easy positives/negatives (i.e., ones against which continuing to contrast the anchor only weakly benefits the student) are small.

3) Completeness. The linear combination leverages logit matching to mainly improve the relative relation among logit outputs (Pang et al., 2019; Zhang et al., 2020; Wang et al., 2022b), therefore storing most of the logit-level knowledge in the parametric classifier. However, the parametric classifier has to be abandoned during transfer learning since different visual recognition tasks typically have distinct label spaces. By contrast, due to the nonparametric nature of Eqn. (15), the student trains all network parameters only for data representations so that all the lessons learnt from the teacher can be completely transferred for target tasks.

4. Experiments

Baselines. We compare our method with mainstream knowledge distillers, including, KD (Hinton et al., 2015), DKD (Yeh et al., 2022), IPWD (Niu et al., 2022), WSLD (Zhou et al., 2021), CS-KD (Yun et al., 2020), TF-KD (Yuan et al., 2020), PS-KD (Kim et al., 2021), NKD (Yang et al., 2023), MLD (Jin et al., 2023), DIST (Huang et al., 2022a), FitNets (Romero et al., 2014), CRD (Tian et al., 2019), WCoRD (Chen et al., 2021a), ReviewKD (Chen et al., 2021b), NORM (Liu et al., 2023), CoCoRD (Fu et al., 2023), DiffKD (Huang et al., 2023), SRRL (Yang et al., 2021) and SSKD (Xu et al., 2020).

Settings. We conduct experiments on multiple benchmarks for knowledge transfer: CIFAR-100 (Krizhevsky et al., 2009), ImageNet-1K (Russakovsky et al., 2015), STL-10 (Coates et al., 2011), Tiny-ImageNet (Chrabaszcz et al., 2017), PASCAL-VOC (Everingham et al., 2009) and MS-COCO (Lin et al., 2014). Following Chen et al. (2022), we employ the last feature map and a three-layer bottleneck transformation for implementing the projector $g(\cdot)$. As suggested by Theorem 3.5, we take $\alpha = \beta = \tau$. The reported results of our method are averaged over 5 runs. The implementation details are attached in Appendix E.

Table 2: Top-1 accuracy (%) on CIFAR-100, Heterogeneous Architecture. The best results are highlighted in boldface.

Distillation Manner	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
Logit	DKD	69.71	70.35	76.45	77.07	76.70
	IPWD	—	70.25	76.03	—	76.44
	WSLD	—	—	75.46	75.93	76.21
	MLD	70.57	71.04	77.18	78.44	76.21
	DIST	—	68.66	76.34	77.35	—
Feature	ReviewKD	70.37	69.89	77.45	77.78	77.14
	CRD	69.73	69.11	75.11	75.65	76.05
	WCoRD	69.47	70.45	75.40	75.96	76.32
	CoCoRD	69.86	70.22	75.99	77.28	76.42
	NORM	68.94	70.56	77.42	78.07	77.06
	DiffKD	—	69.21	76.57	77.52	—
Logit + Feature	SSRL	69.14	69.45	75.66	76.40	76.61
	SSKD	71.53	**72.57**	78.44	78.61	77.40
	Ours	**72.26**	71.78	**78.92**	**79.39**	**78.54**

Table 3: Top-1 accuracy (%) on CIFAR-100 in the self-distillation setting. The best results are shown in boldface.

Method	ResNet18	ResNet101	DenseNet121	ResNeXt29
CS-KD	78.70	79.24	79.53	81.74
TF-KD	77.12	79.87	80.12	82.67
PS-KD	79.18	80.57	81.27	82.72
IPWD	79.82	81.39	81.60	83.30
Ours	**81.01**	**82.53**	**82.84**	**84.24**

4.1. Classification on CIFAR-100

To evaluate the effectiveness of our method, we experiment on CIFAR100 with 11 student-teacher combinations. 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 most cases. Taking the resnet56/resnet20 and ResNet32x4/MobileNetV2 pairs as an example, CoCoRD outperforms the second best by 1.01% and 0.95% for each.

Distillation without Teachers. To investigate the practicability of our method, we deploy it as a plug-in technique on PS-KD (Kim et al., 2021). We strictly adopt the training details of PS-KD for a fair comparison except that, same as IPWD (Niu et al., 2022), we apply our method on PS-KD at the last $1/4$ of the total training epochs. Table 3 shows the classification performance of different teacher-free distillers on CIFAR100. It can be found that our method consistently outperforms others across fou architectures, further improving the Top-1 accuracy by 0.86%-1.24%.

4.2. Classification on ImageNet-1K

To validate the scalability of our proposed AuxKD, we employ the PyTorch-version student-teacher combinations to perform experiments on ImageNet, the Top-1 and Top-5 accuracy rates of different distillation methods are reported in Table 4. It can be observed that our method keeps achieving the best performance on ImageNet. In particular, while the state-of-the-art DiffKD reduces the gap of Top-1 and Top-5 accuracy rate between the teacher and the student by 2.47% and 1.57% respectively for the ResNet34/ResNet18 pair, our method narrows the two by 3.29% and 1.81%.

4.3. Transfer Learning

To study the generalization of our method, we evaluate our distilled model on downstream tasks, i.e., image classification and object detection. For image classification, we employ linear probing on STIL-10 and Tiny-ImageNet. We freeze the student and train a linear classifier on the global average pooling features. Our results in Table 5 indicate the outstanding transferability of features learned with our method on both datasets. For object detection, we train a ResNet-50 student by distilling from a ResNet-101 teacher on ImageNet, followed by initializing the backbone of Faster R-CNN (Ren et al., 2015) and Mask R-CNN (He et al., 2017)

Table 4: Top-1 and Top-5 accuracy (%) on ImageNet-1K validation set. The best results are highlighted in boldface.

Teacher: ResNet34 → Student: 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
MLD	71.90	90.55
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
Ours	**73.04**	**90.88**

Teacher: ResNet50 → Student: 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
MLD	73.01	91.42
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
Ours	**73.90**	**91.51**

Table 5: Linear probing on STL-10 and Tiny-ImageNet for image classification: We use the combination of teacher WRN-40-2 and student WRN-16-2. We report Top-1 accuracy (%). The best results are highlighted in boldface.

Source → Target	Student	KD	DKD	FitNet	ReviewKD	CRD	CoCoRD	SSKD	Ours
CIFAR-100 → STL-10	69.7	70.9	72.9	70.3	72.4	71.6	73.6	72.4	**74.4**
CIFAR-100 → Tiny-ImageNet	33.7	33.9	37.1	33.5	36.6	35.6	38.4	36.2	**39.0**

with the pre-trained student model before a fine-tuning on PASCAL-VOC and MS-COCO respectively. Our results in Tables 6 indicate that our method contributes to more transferable features than the CRD-initialized, SSKD-initialized, and CoCoRD-initialized counterparts.

Table 6: Fine-tuning on PASCAL-VOC and MS-COCO for object detection. The best results are shown in boldface.

Method	PASCAL-VOC			MS-COCO		
	AP	AP_{50}	AP_{75}	AP	AP_{50}	AP_{75}
CRD	54.2	81.7	60.0	40.7	60.5	43.9
CoCoRD	55.0	82.0	61.1	41.0	60.9	44.5
SSKD	54.4	81.8	60.5	40.5	60.3	43.8
Ours	**56.1**	**82.6**	**62.3**	**41.9**	**61.3**	**45.0**

5. Conclusion

This paper rethinks KD from a probabilistic perspective, where we reformulate the conventional KD objective by introducing instance membership as a latent variable. We show that a parameterization of the reformulation offers a theoretically unified insight into logit matching and feature matching, both of which evolve independently in the KD literature. Second, the parameterization presents a methodologically Unified KD paradigm that provably maximizes the mutual information between the student's and teacher's features while performing distributional modelling of the student feature space with the teacher's class relationship knowledge. We hope our work can motivate future research on unifying KD either methodologically or theoretically.

Limitations. This paper only explores one type of realization of the introduced auxiliary variable and one type of parameterization schemes. It would be exciting to explore more possibilities for the realization and parameterization.

Acknowledgements

This work is supported by the ARC Australian Laureate Fellowship (FL190100149).

Impact Statement

Investigating the efficacy of the proposed method would consume considerable computing resources, which can lead to increased carbon emissions, which could raise environmental concerns. However, AuxKD can improve the performance of lightweight compact models, therefore saving energy consumption, and it is necessary to validate AuxKD adequately. This work does not raise ethical impacts.

References

Belghazi, M. I., Baratin, A., Rajeswar, S., Ozair, S., Bengio, Y., Courville, A., and Hjelm, R. D. Mine: mutual information neural estimation. *arXiv preprint arXiv:1801.04062*, 2018.

Chen, D., Mei, J.-P., Zhang, H., Wang, C., Feng, Y., and 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.

Chen, L., Wang, D., Gan, Z., Liu, J., Henao, R., and Carin, L. Wasserstein contrastive representation distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16296–16305, 2021a.

Chen, P., Liu, S., Zhao, H., and Jia, J. Distilling knowledge via knowledge review. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5008–5017, 2021b.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PMLR, 2020.

Cho, J. H. and Hariharan, B. On the efficacy of knowledge distillation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4794–4802, 2019.

Chrabaszcz, P., Loshchilov, I., and Hutter, F. A downsampled variant of imagenet as an alternative to the cifar datasets. *arXiv preprint arXiv:1707.08819*, 2017.

Coates, A., Ng, A., and Lee, H. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pp. 215–223. JMLR Workshop and Conference Proceedings, 2011.

Everingham, M., Van Gool, L., Williams, C. K., Winn, J., and Zisserman, A. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88: 303–308, 2009.

Fu, S., Yang, H., and Yang, X. Contrastive consistent representation distillation. In *The British Machine Vision Conference*, 2023.

Gou, J., Yu, B., Maybank, S. J., and Tao, D. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129:1789–1819, 2021.

Grathwohl, W., Wang, K.-C., Jacobsen, J.-H., Duvenaud, D., Norouzi, M., and Swersky, K. Your classifier is secretly an energy based model and you should treat it like one. *arXiv preprint arXiv:1912.03263*, 2019.

Hao, Z., Guo, J., Han, K., Hu, H., Xu, C., and Wang, Y. Vanillakd: Revisit the power of vanilla knowledge distillation from small scale to large scale. *arXiv preprint arXiv:2305.15781*, 2023.

Hassaballah, M. and Awad, A. I. *Deep learning in computer vision: principles and applications*. CRC Press, 2020.

He, K., Gkioxari, G., Dollár, P., and Girshick, R. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pp. 2961–2969, 2017.

He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

Heo, B., Kim, J., Yun, S., Park, H., Kwak, N., and Choi, J. Y. A comprehensive overhaul of feature distillation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1921–1930, 2019.

Hinton, G., Vinyals, O., and Dean, J. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Huang, T., You, S., Wang, F., Qian, C., and Xu, C. Knowledge distillation from a stronger teacher. *Advances in Neural Information Processing Systems*, 35:33716–33727, 2022a.

Huang, T., Zhang, Y., You, S., Wang, F., Qian, C., Cao, J., and Xu, C. Masked distillation with receptive tokens. *arXiv preprint arXiv:2205.14589*, 2022b.

Huang, T., Zhang, Y., Zheng, M., You, S., Wang, F., Qian, C., and Xu, C. Knowledge diffusion for distillation. *arXiv preprint arXiv:2305.15712*, 2023.

Hupkes, D., Giulianelli, M., Dankers, V., Artetxe, M., Elazar, Y., Pimentel, T., Christodoulopoulos, C., Lasri, K., Saphra, N., Sinclair, A., et al. A taxonomy and review of generalization research in nlp. *Nature Machine Intelligence*, 5(10):1161–1174, 2023.

Jiang, X., Liu, F., Fang, Z., Chen, H., Liu, T., Zheng, F., and Han, B. Detecting out-of-distribution data through in-distribution class prior. In *International Conference on Machine Learning*, pp. 15067–15088. PMLR, 2023.

Jin, Y., Wang, J., and Lin, D. Multi-level logit distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24276–24285, 2023.

Kim, K., Ji, B., Yoon, D., and Hwang, S. Self-knowledge distillation with progressive refinement of targets. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6567–6576, 2021.

Kingma, D. P. and Welling, M. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images. 2009.

LeCun, Y., Chopra, S., Hadsell, R., Ranzato, M., and Huang, F. A tutorial on energy-based learning. *Predicting structured data*, 1(0), 2006.

Li, H., Wu, X., Lv, F., Liao, D., Li, T. H., Zhang, Y., Han, B., and Tan, M. Hard sample matters a lot in zero-shot quantization. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, pp. 24417–24426, 2023.

Li, O., Liu, H., Chen, C., and Rudin, C. Deep learning for case-based reasoning through prototypes: A neural network that explains its predictions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

Li, X., Li, S., Omar, B., Wu, F., and Li, X. Reskd: Residual-guided knowledge distillation. *IEEE Transactions on Image Processing*, 30:4735–4746, 2021.

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.

Lin, S., Xie, H., Wang, B., Yu, K., Chang, X., Liang, X., and Wang, G. Knowledge distillation via the target-aware transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10915–10924, 2022.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pp. 740–755. Springer, 2014.

Liu, X., Li, L., Li, C., and Yao, A. Norm: Knowledge distillation via n-to-one representation matching. *arXiv preprint arXiv:2305.13803*, 2023.

Ming, Y., Sun, Y., Dia, O., and Li, Y. How to exploit hyperspherical embeddings for out-of-distribution detection? *arXiv preprint arXiv:2203.04450*, 2022.

Mirzadeh, S. I., Farajtabar, M., Li, A., Levine, N., Matsukawa, A., and Ghasemzadeh, H. Improved knowledge distillation via teacher assistant. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 5191–5198, 2020.

Niu, Y., Chen, L., Zhou, C., and Zhang, H. Respecting transfer gap in knowledge distillation. *Advances in Neural Information Processing Systems*, 35:21933–21947, 2022.

Pang, T., Xu, K., Dong, Y., Du, C., Chen, N., and Zhu, J. Rethinking softmax cross-entropy loss for adversarial robustness. *arXiv preprint arXiv:1905.10626*, 2019.

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.

Ren, S., He, K., Girshick, R., and Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015.

Romero, A., Ballas, N., Kahou, S. E., Chassang, A., Gatta, C., and Bengio, Y. Fitnets: Hints for thin deep nets. *arXiv preprint arXiv:1412.6550*, 2014.

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.

Shen, Y., Shen, Z., Wang, M., Qin, J., Torr, P., and Shao, L. You never cluster alone. *Advances in Neural Information Processing Systems*, 34:27734–27746, 2021.

Son, W., Na, J., Choi, J., and 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.

Stanton, S., Izmailov, P., Kirichenko, P., Alemi, A. A., and Wilson, A. G. Does knowledge distillation really work? *arXiv preprint arXiv:2106.05945*, 2021.

Tang, J., Shivanna, R., Zhao, Z., Lin, D., Singh, A., Chi, E. H., and Jain, S. Understanding and improving knowledge distillation. *arXiv preprint arXiv:2002.03532*, 2020.

Tian, Y., Krishnan, D., and Isola, P. Contrastive representation distillation. *arXiv preprint arXiv:1910.10699*, 2019.

Tsai, T. W., Li, C., and Zhu, J. Mice: Mixture of contrastive experts for unsupervised image clustering. *arXiv preprint arXiv:2105.01899*, 2021.

Wang, C., Yang, Q., Huang, R., Song, S., and Huang, G. Efficient knowledge distillation from model checkpoints. *Advances in Neural Information Processing Systems*, 35: 607–619, 2022a.

Wang, L. and 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.

Wang, W., Han, C., Zhou, T., and Liu, D. Visual recognition with deep nearest centroids. *arXiv preprint arXiv:2209.07383*, 2022b.

Wu, Z., Xiong, Y., Yu, S. X., and Lin, D. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3733–3742, 2018.

Xu, G., Liu, Z., Li, X., and Loy, C. C. Knowledge distillation meets self-supervision. In *European Conference on Computer Vision*, pp. 588–604. Springer, 2020.

Yang, J., Martinez, B., Bulat, A., Tzimiropoulos, G., et al. Knowledge distillation via softmax regression representation learning. International Conference on Learning Representations (ICLR), 2021.

Yang, Z., Zeng, A., Yuan, C., and 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.

Yeh, C.-H., Hong, C.-Y., Hsu, Y.-C., Liu, T.-L., Chen, Y., and LeCun, Y. Decoupled contrastive learning. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVI*, pp. 668–684. Springer, 2022.

Yuan, L., Tay, F. E., Li, G., Wang, T., and Feng, J. Revisiting knowledge distillation via label smoothing regularization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3903–3911, 2020.

Yun, S., Park, J., Lee, K., and Shin, J. Regularizing class-wise predictions via self-knowledge distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 13876–13885, 2020.

Zagoruyko, S. and Komodakis, N. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. *arXiv preprint arXiv:1612.03928*, 2016.

Zhang, X., Zhao, R., Qiao, Y., and Li, H. Rbf-softmax: Learning deep representative prototypes with radial basis function softmax. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVI 16*, pp. 296–311. Springer, 2020.

Zhang, Y., Gong, M., Liu, T., Niu, G., Tian, X., Han, B., Schölkopf, B., and Zhang, K. Causaladv: Adversarial robustness through the lens of causality. *ICLR*, 2022.

Zhao, B., Cui, Q., Song, R., Qiu, Y., and Liang, J. Decoupled knowledge distillation. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, pp. 11953–11962, 2022.

Zhou, H., Song, L., Chen, J., Zhou, Y., Wang, G., Yuan, J., and Zhang, Q. Rethinking soft labels for knowledge distillation: A bias-variance tradeoff perspective. *arXiv preprint arXiv:2102.00650*, 2021.

A. Deviation of Eqn. (4)

Given the facts that

$$p^{\mathcal{S}}(k, \mathbf{x}_i, s_i) = p^{\mathcal{S}}(k|\mathbf{x}_i, s_i)p^{\mathcal{S}}(\mathbf{x}_i|s_i)p^{\mathcal{S}}(s_i) \tag{17}$$
$$p^{\mathcal{S}}(k, \mathbf{x}_i, s_i) = p^{\mathcal{S}}(s_i|\mathbf{x}_i, k)p^{\mathcal{S}}(\mathbf{x}_i|k)p^{\mathcal{S}}(k) \tag{18}$$

we have

$$\begin{aligned} & p^{\mathcal{S}}(k|\mathbf{x}_i, s_i)p^{\mathcal{S}}(\mathbf{x}_i|s_i)p^{\mathcal{S}}(s_i) = p^{\mathcal{S}}(s_i|\mathbf{x}_i, k)p^{\mathcal{S}}(\mathbf{x}_i|k)p^{\mathcal{S}}(k) \\ \Leftrightarrow & \frac{p^{\mathcal{S}}(k|\mathbf{x}_i, s_i)}{p^{\mathcal{S}}(s_i|\mathbf{x}_i, k)} = \frac{p^{\mathcal{S}}(\mathbf{x}_i|k)p^{\mathcal{S}}(k)}{p^{\mathcal{S}}(\mathbf{x}_i|s_i)p^{\mathcal{S}}(s_i)} \end{aligned} \tag{19}$$

B. Proof of Theorem 3.4 in Section 3.4

Assumption B.1. The temperature $\alpha \to +\infty$ such that, for all s_i,

$$p^{\mathcal{S}}(s_i|\mathbf{x}_i) = \frac{1}{1 + |\mathcal{N}_i|}, \quad \forall i. \tag{20}$$

Assumption B.2. The temperature $\kappa \to +\infty$ such that, for all k, each class-conditional vMF distribution is uniform on the sphere, i.e,

$$p^{\mathcal{S}}(\mathbf{x}_i|k) = const, \quad \forall i. \tag{21}$$

As the teacher is fixed in knowledge distillation, the term $\sum_{k=1}^{K} p^{\mathcal{T}}(k|\mathbf{x}_i) \log p^{\mathcal{S}}(\mathbf{x}_i|k)$ is actually a constant value when $\kappa \to +\infty$. Therefore, for brevity, we will take $\sum_{k=1}^{K} p^{\mathcal{T}}(k|\mathbf{x}_i) \log p^{\mathcal{S}}(\mathbf{x}_i|k) = C$ when Assumption B.2 holds.

Assumption B.3. The class-wise information is out of consideration for selecting positive candidates such that, for all i,

$$\mathcal{P}_i = \{i\}. \tag{22}$$

Without loss of generalization, let us use $\ell_2(\cdot)$ to indicate the ℓ_2 normalization operation and define

$$\begin{aligned} d(w^{\mathcal{S}}(\mathbf{z}_i^{\mathcal{S}}), w^{\mathcal{T}}(\mathbf{z}_i^{\mathcal{T}})) &= \frac{1}{2\beta} \left\| \ell_2(g(\mathbf{h}_i^{\mathcal{S}})) - \ell_2(\mathbf{z}_i^{\mathcal{T}}) \right\|_2^2 \\ &= \frac{1}{\beta}(1 - \mathbf{h}_i^{\mathcal{S}} \circ \mathbf{z}_i^{\mathcal{T}}) \\ &= \frac{1}{\beta} - \phi_\beta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_i^{\mathcal{T}}). \end{aligned} \tag{23}$$

If Assumption B.1, Assumption B.2 and Assumption B.3 hold, we then have:

$$\begin{aligned} \mathcal{L}_i^{\text{AuxKD}} &= -\log \frac{1}{1 + |\mathcal{N}_{s_i}|} + \log \frac{\exp \phi_\beta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_i^{\mathcal{S}})}{\exp \phi_\beta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_i^{\mathcal{S}}) + \exp \phi_\beta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_i^{\mathcal{T}})} + C \\ &= -\log \frac{1}{1 + |\mathcal{N}_{s_i}|} + \log \frac{\exp -d(w^{\mathcal{S}}(\mathbf{z}_i^{\mathcal{S}}), w^{\mathcal{S}}(\mathbf{z}_i^{\mathcal{S}}))}{\exp -d(w^{\mathcal{S}}(\mathbf{z}_i^{\mathcal{S}}), w^{\mathcal{S}}(\mathbf{z}_i^{\mathcal{S}})) + \exp -d(w^{\mathcal{S}}(\mathbf{z}_i^{\mathcal{S}}), w^{\mathcal{T}}(\mathbf{z}_i^{\mathcal{T}}))} + C \\ &= \log(1 + |\mathcal{N}_{s_i}|) - \log \left\{1 + \exp[d(w^{\mathcal{S}}(\mathbf{z}_i^{\mathcal{S}}), w^{\mathcal{S}}(\mathbf{z}_i^{\mathcal{S}})) - d(w^{\mathcal{S}}(\mathbf{z}_i^{\mathcal{S}}), w^{\mathcal{T}}(\mathbf{z}_i^{\mathcal{T}}))]\right\} + C \\ &\leq -\log \left\{\exp[d(w^{\mathcal{S}}(\mathbf{z}_i^{\mathcal{S}}), w^{\mathcal{S}}(\mathbf{z}_i^{\mathcal{S}})) - d(w^{\mathcal{S}}(\mathbf{z}_i^{\mathcal{S}}), w^{\mathcal{T}}(\mathbf{z}_i^{\mathcal{T}}))]\right\} + \log(1 + |\mathcal{N}_i|) + C \\ &= -d(w^{\mathcal{S}}(\mathbf{z}_i^{\mathcal{S}}), w^{\mathcal{S}}(\mathbf{z}_i^{\mathcal{S}})) + d(w^{\mathcal{S}}(\mathbf{z}_i^{\mathcal{S}}), w^{\mathcal{T}}(\mathbf{z}_i^{\mathcal{T}})) + \log(1 + |\mathcal{N}_i|) + C \\ &= \underbrace{d(w^{\mathcal{S}}(\mathbf{z}_i^{\mathcal{S}}), w^{\mathcal{T}}(\mathbf{z}_i^{\mathcal{T}}))}_{\mathcal{L}_i^{\text{FM}}} + \log(1 + |\mathcal{N}_i|) + C. \end{aligned} \tag{24}$$

C. Proof of Theorems 3.5 and 3.6 in Section 3.5

C.1. Proof of Theorem 3.5

Without loss of generalization, let us define $\eta(\mathbf{z}^{\mathcal{S}}, \mathbf{z}^{\mathcal{T}}) = \mathbf{h}^{\mathcal{S}} \circ \mathbf{z}^{\mathcal{T}}/\tau$. If $\alpha = \beta = \tau$, we then have

$$
\begin{aligned}
&\log \frac{\exp \phi_\beta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_i^{\mathcal{S}}) + \sum_{p\in\mathcal{P}_i} \exp \phi_\beta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_j^{\mathcal{T}})}{\exp \phi_\alpha(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_i^{\mathcal{S}}) + \sum_{n\in\mathcal{N}_i} \exp \phi_\alpha(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_j^{\mathcal{T}})} \\
=&\log \frac{\exp \eta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_i^{\mathcal{S}}) + \sum_{p\in\mathcal{P}_i} \exp \eta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_j^{\mathcal{T}})}{\exp \eta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_i^{\mathcal{S}}) + \sum_{n\in\mathcal{N}_i} \exp \eta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_j^{\mathcal{T}})} \\
\leq&\frac{1}{|\mathcal{P}_i|} \sum_{p\in\mathcal{P}_i} \log \frac{|\mathcal{P}_i| \exp \eta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_i^{\mathcal{S}}) + \exp \eta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_p^{\mathcal{T}})}{\exp \eta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_i^{\mathcal{S}}) + \sum_{n\in\mathcal{N}_i} \exp \eta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_n^{\mathcal{T}})} \\
\leq&\frac{1}{|\mathcal{P}_i|} \sum_{p\in\mathcal{P}_i} \log \frac{|\mathcal{N}_i| \exp \eta(\mathbf{v}_i^{\mathcal{S}}, \mathbf{z}_p^{\mathcal{T}})}{\sum_{n\in\mathcal{N}_i} \exp \eta(\mathbf{v}_i^{\mathcal{S}}, \mathbf{z}_n^{\mathcal{T}})} \\
=&\frac{1}{|\mathcal{P}_i|} \sum_{p\in\mathcal{P}_i} \log \frac{|\mathcal{N}_i| \exp \eta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_p^{\mathcal{T}})}{\sum_{n\in\mathcal{N}_i} \exp \eta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_n^{\mathcal{T}})} \\
=&\frac{1}{|\mathcal{P}_i|} \sum_{p\in\mathcal{P}_i} \eta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_p^{\mathcal{T}}) - \log \frac{1}{|\mathcal{N}_i|} \sum_{n\in\mathcal{N}_i} \exp \eta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_n^{\mathcal{T}}).
\end{aligned}
\tag{25}
$$

Note that the first inequality holds thanks to the fact that $\eta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_i^{\mathcal{S}}) \geq \eta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_p^{\mathcal{T}}), \forall p \in \mathcal{P}_i$, and the second inequality holds by mildly assuming the representation of a student anchor and a teacher positive is more aligned than that of the anchor and most of the teacher negatives such that $\exp \eta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_p^{\mathcal{T}}) \geq \sum_{n\in\mathcal{N}_i} \exp \eta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_n^{\mathcal{T}})/|\mathcal{N}_i|, \forall p \in \mathcal{P}_i$.

C.2. Proof of Theorem 3.6

Similar to Kingma & Welling (2013), our ELBO can be derived with the Jensen's inequality $\log \mathbb{E}\left[\cdot\right] \geq \mathbb{E}\left[\log(\cdot)\right]$:

$$
\begin{aligned}
\log p^{\mathcal{S}}(\mathbf{x}_i) &= \log \mathbb{E}_{p^{\mathcal{S}}(k)} \left[p^{\mathcal{S}}(\mathbf{x}_i|k)\right] \\
&= \log \mathbb{E}_{p^{\mathcal{S}}(k)} \left[C_d(\kappa^{-1}) \exp\left(\mathbf{z}_i^{\mathcal{S}} \circ \boldsymbol{\mu}_k/\kappa\right) \cdot \frac{p^{\mathcal{T}}(k|\mathbf{x}_i)}{p^{\mathcal{T}}(k|\mathbf{x}_i)}\right] \\
&\geq \mathbb{E}_{p^{\mathcal{T}}(k|\mathbf{x}_i)} \left[\log \left[C_d(\kappa^{-1}) \exp\left(\mathbf{z}_i^{\mathcal{S}} \circ \boldsymbol{\mu}_k/\kappa\right)\right]\right] - \mathrm{KL}(p^{\mathcal{T}}(k|\mathbf{x}_i)||p^{\mathcal{S}}(k)) \\
&= \mathbb{E}_{p^{\mathcal{T}}(k|\mathbf{x}_i)} \left[\mathbf{z}_i^{\mathcal{S}} \circ \boldsymbol{\mu}_k/\kappa\right] + \underbrace{\log C_d(\kappa^{-1}) - \mathrm{KL}(p^{\mathcal{T}}(k|\mathbf{x}_i)||p^{\mathcal{S}}(k))}_{const}.
\end{aligned}
\tag{26}
$$

D. Intrinsic Hard Positive and Negative Mining Properties

Without loss of generalization, we denote $\hat{\mathbf{h}}^{\mathcal{S}} = \mathbf{h}^{\mathcal{S}}/\left\|\mathbf{h}^{\mathcal{S}}\right\|_2$, $\hat{\mathbf{z}}^{\mathcal{T}} = \mathbf{z}^{\mathcal{T}}/\left\|\mathbf{z}^{\mathcal{T}}\right\|_2$. Let $\langle \mathbf{a}, \mathbf{b} \rangle = \mathbf{a}^\top \mathbf{b}$, we have $\mathbf{h}^{\mathcal{S}} \circ \mathbf{z}^{\mathcal{T}} = \left\langle \hat{\mathbf{h}}^{\mathcal{S}}, \hat{\mathbf{z}}^{\mathcal{T}} \right\rangle$ and

$$
\begin{aligned}
\frac{\partial \hat{\mathbf{h}}^{\mathcal{S}}}{\partial \mathbf{h}^{\mathcal{S}}} &= \frac{\partial}{\partial \mathbf{h}^{\mathcal{S}}} \left(\frac{\mathbf{h}^{\mathcal{S}}}{\|\mathbf{h}^{\mathcal{S}}\|_2}\right) \\
&= \frac{1}{\|\mathbf{h}^{\mathcal{S}}\|_2} \mathbf{I} - \mathbf{h}^{\mathcal{S}} \left(\frac{\partial \left(1/\left\|\mathbf{h}^{\mathcal{S}}\right\|_2\right)}{\partial \mathbf{h}^{\mathcal{S}}}\right)^\top \\
&= \frac{1}{\|\mathbf{h}^{\mathcal{S}}\|_2} \left(\mathbf{I} - \frac{\mathbf{h}^{\mathcal{S}} \left(\mathbf{h}^{\mathcal{S}}\right)^\top}{\|\mathbf{h}^{\mathcal{S}}\|_2^2}\right) \\
&= \frac{1}{\|\mathbf{h}^{\mathcal{S}}\|_2} \left(\mathbf{I} - \hat{\mathbf{h}}^{\mathcal{S}} \left(\hat{\mathbf{h}}^{\mathcal{S}}\right)^\top\right).
\end{aligned}
\tag{27}
$$

For convenience, we print below the expressions for the first term of $\mathcal{L}_i^{\mathrm{AuxKD}}$ in Eqn. (10):

$$\mathcal{L}_i = -\log \frac{\exp \phi_\beta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_i^{\mathcal{S}}) + \sum_{j\in\mathcal{P}_i} \exp \phi_\beta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_j^{\mathcal{T}})}{\exp \phi_\alpha(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_i^{\mathcal{S}}) + \sum_{j\in\mathcal{N}_i} \exp \phi_\alpha(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_j^{\mathcal{T}})}. \tag{28}$$

The gradient of $\mathcal{L}_i$ with respect to $\mathbf{h}_i^{\mathcal{S}}$ is related to that with respect to $\hat{\mathbf{h}}_i^{\mathcal{S}}$ via the chain rule:

$$\frac{\partial \mathcal{L}_i}{\partial \mathbf{h}_i^{\mathcal{S}}} = \frac{\partial \mathcal{L}_i}{\partial \hat{\mathbf{h}}_i^{\mathcal{S}}} \cdot \frac{\partial \hat{\mathbf{h}}_i^{\mathcal{S}}}{\partial \mathbf{h}_i^{\mathcal{S}}}. \tag{29}$$

As suggested by Theorem 3.5, we take $\alpha = \beta = \tau$ and

$$\begin{aligned}
\frac{\partial \mathcal{L}_i}{\partial \hat{\mathbf{h}}_i^{\mathcal{S}}} &= \frac{\partial}{\partial \hat{\mathbf{h}}_i^{\mathcal{S}}} \left(-\log \frac{\exp \phi_\beta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_i^{\mathcal{S}}) + \sum_{j\in\mathcal{P}_i} \exp \phi_\beta(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_j^{\mathcal{T}})}{\exp \phi_\alpha(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_i^{\mathcal{S}}) + \sum_{j\in\mathcal{N}_i} \exp \phi_\alpha(\mathbf{z}_i^{\mathcal{S}}, \mathbf{z}_j^{\mathcal{T}})} \right) \\
&= \frac{\partial}{\partial \hat{\mathbf{h}}_i^{\mathcal{S}}} \left(-\log \frac{\exp \left\langle \frac{1}{\tau}\hat{\mathbf{h}}_i^{\mathcal{S}}, \hat{\mathbf{h}}_i^{\mathcal{S}} \right\rangle + \sum_{j\in\mathcal{P}_i} \exp \left\langle \frac{1}{\tau}\hat{\mathbf{h}}_i^{\mathcal{S}}, \hat{\mathbf{z}}_j^{\mathcal{T}} \right\rangle}{\exp \left\langle \frac{1}{\tau}\hat{\mathbf{h}}_i^{\mathcal{S}}, \hat{\mathbf{h}}_i^{\mathcal{S}} \right\rangle + \sum_{j\in\mathcal{N}_i} \exp \left\langle \frac{1}{\tau}\hat{\mathbf{h}}_i^{\mathcal{S}}, \hat{\mathbf{z}}_j^{\mathcal{T}} \right\rangle} \right) \\
&= -\frac{\partial}{\partial \hat{\mathbf{h}}_i^{\mathcal{S}}} \left[\log \left(\exp \frac{1}{\tau} + \sum_{j\in\mathcal{P}_i} \exp \left\langle \frac{1}{\tau}\hat{\mathbf{h}}_i^{\mathcal{S}}, \hat{\mathbf{z}}_j^{\mathcal{T}} \right\rangle \right) \right] + \frac{\partial}{\partial \hat{\mathbf{h}}_i^{\mathcal{S}}} \left[\log \left(\exp \frac{1}{\tau} + \sum_{j\in\mathcal{N}_i} \exp \left\langle \frac{1}{\tau}\hat{\mathbf{h}}_i^{\mathcal{S}}, \hat{\mathbf{z}}_j^{\mathcal{T}} \right\rangle \right) \right] \\
&= -\frac{1}{\tau} \frac{\sum_{j\in\mathcal{P}_i} \mathbf{z}_j^{\mathcal{T}} \exp \left\langle \frac{1}{\tau}\hat{\mathbf{h}}_i^{\mathcal{S}}, \hat{\mathbf{z}}_j^{\mathcal{T}} \right\rangle}{\exp \frac{1}{\tau} + \sum_{j\in\mathcal{P}_i} \exp \left\langle \frac{1}{\tau}\hat{\mathbf{h}}_i^{\mathcal{S}}, \hat{\mathbf{z}}_j^{\mathcal{T}} \right\rangle} + \frac{1}{\tau} \frac{\sum_{j\in\mathcal{N}_i} \mathbf{z}_j^{\mathcal{T}} \exp \left\langle \frac{1}{\tau}\hat{\mathbf{h}}_i^{\mathcal{S}}, \hat{\mathbf{z}}_j^{\mathcal{T}} \right\rangle}{\exp \frac{1}{\tau} + \sum_{j\in\mathcal{N}_i} \exp \left\langle \frac{1}{\tau}\hat{\mathbf{h}}_i^{\mathcal{S}}, \hat{\mathbf{z}}_j^{\mathcal{T}} \right\rangle} \\
&= -\frac{1}{\tau} \sum_{j\in\mathcal{P}_i} \mathbf{z}_j^{\mathcal{T}} P_{ij} + \frac{1}{\tau} \sum_{j\in\mathcal{N}_i} \mathbf{z}_j^{\mathcal{T}} N_{ij},
\end{aligned} \tag{30}$$

where

$$P_{ij} = \frac{\exp \left\langle \frac{1}{\tau}\hat{\mathbf{h}}_i^{\mathcal{S}}, \hat{\mathbf{z}}_j^{\mathcal{T}} \right\rangle}{\exp \frac{1}{\tau} + \sum_{j\in\mathcal{P}_i} \exp \left\langle \frac{1}{\tau}\hat{\mathbf{h}}_i^{\mathcal{S}}, \hat{\mathbf{z}}_j^{\mathcal{T}} \right\rangle}, \qquad \forall j \in \mathcal{P}_i. \tag{31}$$

$$N_{ij} = \frac{\exp \left\langle \frac{1}{\tau}\hat{\mathbf{h}}_i^{\mathcal{S}}, \hat{\mathbf{z}}_j^{\mathcal{T}} \right\rangle}{\exp \frac{1}{\tau} + \sum_{j\in\mathcal{N}_i} \exp \left\langle \frac{1}{\tau}\hat{\mathbf{h}}_i^{\mathcal{S}}, \hat{\mathbf{z}}_j^{\mathcal{T}} \right\rangle}, \qquad \forall j \in \mathcal{N}_i. \tag{32}$$

Combing Eqns. (27, 30) with Eqn. (29), we have:

$$\begin{aligned}
\frac{\partial \mathcal{L}_i}{\partial \mathbf{h}_i^{\mathcal{S}}} &= \frac{1}{\tau \left\| \mathbf{h}_i^{\mathcal{S}} \right\|_2} \left(\mathbf{I} - \hat{\mathbf{h}}_i^{\mathcal{S}} \left(\hat{\mathbf{h}}_i^{\mathcal{S}} \right)^\top \right) \left(\sum_{j\in\mathcal{N}_i} \mathbf{z}_j^{\mathcal{T}} N_{ij} - \sum_{j\in\mathcal{P}_i} \mathbf{z}_j^{\mathcal{T}} P_{ij} \right) \\
&= \frac{1}{\tau \left\| \mathbf{h}_i^{\mathcal{S}} \right\|_2} \left[\sum_{j\in\mathcal{N}_i} \left(\mathbf{z}_j^{\mathcal{T}} - \left\langle \hat{\mathbf{h}}_i^{\mathcal{S}}, \hat{\mathbf{z}}_j^{\mathcal{T}} \right\rangle \hat{\mathbf{h}}_i^{\mathcal{S}} \right) N_{ij} - \sum_{j\in\mathcal{P}_i} \left(\mathbf{z}_j^{\mathcal{T}} - \left\langle \hat{\mathbf{h}}_i^{\mathcal{S}}, \hat{\mathbf{z}}_j^{\mathcal{T}} \right\rangle \hat{\mathbf{h}}_i^{\mathcal{S}} \right) P_{ij} \right] \\
&= \frac{1}{\tau \left\| \mathbf{h}_i^{\mathcal{S}} \right\|_2} \left(\left. \frac{\partial \mathcal{L}_i}{\partial \mathbf{h}_i^{\mathcal{S}}} \right|_{\mathcal{N}_i} - \left. \frac{\partial \mathcal{L}_i}{\partial \mathbf{h}_i^{\mathcal{S}}} \right|_{\mathcal{P}_i} \right),
\end{aligned} \tag{33}$$

where

$$\left. \frac{\partial \mathcal{L}_i}{\partial \mathbf{h}_i^{\mathcal{S}}} \right|_{\mathcal{P}_i} = \sum_{j\in\mathcal{P}_i} \left(\mathbf{z}_j^{\mathcal{T}} - \left\langle \hat{\mathbf{h}}_i^{\mathcal{S}}, \hat{\mathbf{z}}_j^{\mathcal{T}} \right\rangle \hat{\mathbf{h}}_i^{\mathcal{S}} \right) P_{ij}. \tag{34}$$

$$\left. \frac{\partial \mathcal{L}_i}{\partial \mathbf{h}_i^{\mathcal{S}}} \right|_{\mathcal{N}_i} = \sum_{j\in\mathcal{N}_i} \left(\mathbf{z}_j^{\mathcal{T}} - \left\langle \hat{\mathbf{h}}_i^{\mathcal{S}}, \hat{\mathbf{z}}_j^{\mathcal{T}} \right\rangle \hat{\mathbf{h}}_i^{\mathcal{S}} \right) N_{ij}. \tag{35}$$

We now show that easy positives and negatives have small gradient contributions while hard positives and negatives have large ones. For an easy positive $\mathbf{z}_j^{\mathcal{T}} \in \mathcal{P}_i$ (i.e., one against which contrasting the anchor only weakly benefits the encoder) such that $\left\langle \hat{\mathbf{h}}_i^{\mathcal{S}}, \hat{\mathbf{z}}_j^{\mathcal{T}} \right\rangle \approx 1$. Thus (See Eqn. (34)):

$$\left\| \mathbf{z}_j^{\mathcal{T}} - \left\langle \hat{\mathbf{h}}_i^{\mathcal{S}}, \hat{\mathbf{z}}_j^{\mathcal{T}} \right\rangle \hat{\mathbf{h}}_i^{\mathcal{S}} \right\|_2 = \sqrt{1 - \left\langle \hat{\mathbf{h}}_i^{\mathcal{S}}, \hat{\mathbf{z}}_j^{\mathcal{T}} \right\rangle^2} \approx 0. \tag{36}$$

For a hard positive $\mathbf{z}_j^{\mathcal{T}} \in \mathcal{P}_i$ (i.e., one against which contrasting the anchor greatly benefits the encoder) such that $\left\langle \hat{\mathbf{h}}_i^{\mathcal{S}}, \hat{\mathbf{z}}_j^{\mathcal{T}} \right\rangle \approx 0$. Thus (See Eqn. (34)):

$$\left\| \mathbf{z}_j^{\mathcal{T}} - \left\langle \hat{\mathbf{h}}_i^{\mathcal{S}}, \hat{\mathbf{z}}_j^{\mathcal{T}} \right\rangle \hat{\mathbf{h}}_i^{\mathcal{S}} \right\|_2 = \sqrt{1 - \left\langle \hat{\mathbf{h}}_i^{\mathcal{S}}, \hat{\mathbf{z}}_j^{\mathcal{T}} \right\rangle^2} \approx 1. \tag{37}$$

Thus, for weak positives $\mathbf{z}_j^{\mathcal{T}} \in \mathcal{P}_i$ (since $\left\langle \hat{\mathbf{h}}_i^{\mathcal{S}}, \hat{\mathbf{z}}_j^{\mathcal{T}} \right\rangle \approx 1$) the contribution to the gradient is small while for hard positives $\mathbf{z}_j^{\mathcal{T}} \in \mathcal{P}_i$ the contribution is large (since $\left\langle \hat{\mathbf{h}}_i^{\mathcal{S}}, \hat{\mathbf{z}}_j^{\mathcal{T}} \right\rangle \approx 0$). Similarly, analysing Eqn. (35)) for weak negatives $\mathbf{z}_j^{\mathcal{T}} \in \mathcal{N}_i$ ($\left\langle \hat{\mathbf{h}}_i^{\mathcal{S}}, \hat{\mathbf{z}}_j^{\mathcal{T}} \right\rangle \approx -1$) vs hard negatives $\mathbf{z}_j^{\mathcal{T}} \in \mathcal{N}_i$ ($\left\langle \hat{\mathbf{h}}_i^{\mathcal{S}}, \hat{\mathbf{z}}_j^{\mathcal{T}} \right\rangle \approx 0$), we conclude that the gradient contribution is large for hard negatives and small for weak negatives.

E. Implementation Details

Classification on CIFAR-100. We consider a standard data augmentation scheme including padding 4 pixels prior to 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.

Classification on ImageNet-1K. The standard PyTorch ImageNet practice is adopted except for 100 training epochs. We set the batch size as 512 and the initial learning rate as 0.2. 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.

Linear Probing. We utilize an SGD optimizer with a momentum of 0.9, a batch size of 64 and a weight decay of 0. The initial learning rate starts at 0.1 and is decayed by 10 at the 30-th, 60-th and 90-th epochs within the total 100 epochs.

Fine-tuning. Both Faster R-CNN and Mask R-CNN are equipped with ResNet50-C4, which is available in Detectron2, as the backbone. The backbone ends with the conv4 stage and the box prediction head consists of the conv5 (including global pooling) followed by a BN layer. The Faster R-CNN is fine-tuned on VOC trainval07+12 for 24k iterations in an end-to-end manner. The image scale is $[480, 800]$ pixels during training and 800 at inference. The image scale is in $[640, 800]$ pixels during training and is 800 at inference. We fine-tune Mask R-CNN on the COCO train2017 in an end-to-end manner and evaluate on COCO val2017. The schedule is $2\times$ as in (Yang et al., 2023). The image scale is in $[640, 800]$ pixels during training and is 800 at inference.

F. Distillation with Stronger Teachers.

To investigate the efficacy of our method on reducing the transfer gap, we further conduct experiments on much stronger teachers following Huang et al. (2022a). From the results in Table 7, we can see that ours outperforms the most advanced DIST and DiffKD. We kindly note that DiffKD requires considerably more parameters to train a latent diffusion model.

Table 7: Top-1 accuracy (%) on ImageNet-1K, ResNet50 trained by Yuan et al. (2020) is used as a stronger teacher. The best results are shown in boldface.

Model	KD	SRRL	DIST	DiffKD	Ours
ResNet34	77.2	76.7	77.8	78.1	**78.6**
MobileNetV2	71.7	69.2	74.4	74.9	**75.2**