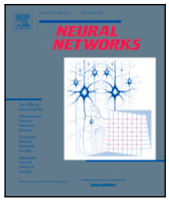
Neural Networks 173 (2024) 106160

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet



Full Length Article

Knowledge distillation under ideal joint classifier assumption

Huayu Li [a], Xiwen Chen [b], Gregory Ditzler [c], Janet Roveda [a,d,e], Ao Li [a,e,*]

[a] Department of Electrical & Computer Engineering at the University of Arizona, Tucson, 85721, AZ, USA
[b] School of Computing at Clemson University, Clemson, 29634, SC, USA
[c] EpiSys Science, Philadelphia, PA, 19128, USA
[d] Department of Biomedical Engineering, The University of Arizona, Tucson, 85721, AZ, USA
[e] BIO5 Institute, The University of Arizona, Tucson, 85721, AZ, USA

ARTICLE INFO

Keywords:
Knowledge distillation
Teacher-student learning
Deep learning

ABSTRACT

Knowledge distillation constitutes a potent methodology for condensing substantial neural networks into more compact and efficient counterparts. Within this context, softmax regression representation learning serves as a widely embraced approach, leveraging a pre-established teacher network to guide the learning process of a diminutive student network. Notably, despite the extensive inquiry into the efficacy of softmax regression representation learning, the intricate underpinnings governing the knowledge transfer mechanism remain inadequately elucidated. This study introduces the 'Ideal Joint Classifier Knowledge Distillation' (IJCKD) framework, an overarching paradigm that not only furnishes a lucid and exhaustive comprehension of prevailing knowledge distillation techniques but also establishes a theoretical underpinning for prospective investigations. Employing mathematical methodologies derived from domain adaptation theory, this investigation conducts a comprehensive examination of the error boundary of the student network contingent upon the teacher network. Consequently, our framework facilitates efficient knowledge transference between teacher and student networks, thereby accommodating a diverse spectrum of applications.

1. Introduction

The advancements in deep neural networks in the field of computer vision are undeniable. Deep neural networks outperform classical machine learning algorithms in various tasks such as image recognition (He, Zhang, Ren, & Sun, 2016; Krizhevsky, Sutskever, & Hinton, 2017), semantic segmentation (Chen, Papandreou, Kokkinos, Murphy, & Yuille, 2017; Long, Shelhamer, & Darrell, 2015; Zhao, Shi, Qi, Wang, & Jia, 2017), and object detection (Girshick, Donahue, Darrell, & Malik, 2014; Redmon, Divvala, Girshick, & Farhadi, 2016). Unfortunately, the network's performance is usually determined by the number of parameters it has. Although more parameters result in a higher network capacity, they also lead to an increase in computational complexity and storage costs, which limits the use of deep neural networks on resource-limited hardware. As a result, reducing the size of a deep neural network while retaining the high performance of the large network is a critical to many applications.

Lightweight models, such as those discussed in Howard et al. (2017) and Zhang, Zhou, Lin, and Sun (2018), aim to reduce computational complexity through efficient network architecture design. Model compression techniques have been developed to address the complexity issue beyond architecture, including parameter pruning (Han, Mao, & Dally, 2015), low-rank factorization (Tai, Xiao, Zhang, Wang, et al., 2015), and knowledge distillation (Buciluǎ, Caruana, & Niculescu-Mizil, 2006; Hinton, Vinyals, & Dean, 2015). The goal of knowledge distillation through model compression is to transfer the "dark" knowledge from a large, highly-accurate pre-trained teacher network to a small, high-speed student network with lower accuracy. Under knowledge distillation, the student's network is trained using the teacher's soft labels for guidance (Hinton et al., 2015), or using "hints" from the teacher's hidden layers (Romero et al., 2014) to produce higher performance from the student than training the student's network with hard target labels alone.

An intuitive approach to improve knowledge distillation is to have the student model mimic the teacher model as closely as possible. Recent works in knowledge distillation can be divided into two main categories: representation and logits distillation. Representation-based techniques aim to extract richer information from the intermediate layers of the teacher model (Chen, Liu, Zhao & Jia, 2021; Chen, Mei et al., 2021; Zagoruyko & Komodakis, 2016), or to better align the teacher and student features (Tian, Krishnan, & Isola, 2019; Tung & Mori, 2019). On the other hand, logits-based distillation techniques (Zhao, Cui, Song, Qiu, & Liang, 2022; Zhou et al., 2021) focus more on the

* Corresponding author at: Department of Electrical & Computer Engineering at the University of Arizona, Tucson, 85721, AZ, USA.
E-mail address: aoli1@arizona.edu (A. Li).

https://doi.org/10.1016/j.neunet.2024.106160
Received 3 October 2023; Received in revised form 5 December 2023; Accepted 1 February 2024
Available online 3 February 2024

statistical significance of the output logits and probabilities between the teacher and student networks.

Softmax Regression Representation Learning (SRRL) (Yang, Martinez, Bulat, Tzimiropoulos, et al., 2021) is a type of logits-based distillation that matches the teacher and student logits, while using the teacher's classifier applied to the student's feature (penultimate layer output) and teacher's feature at the same time. SRRL offers a new perspective to improve the student's performance by not only using the teacher's logits, but also teacher's classifier. A related method, Simple Knowledge Distillation (SimKD) (Chen et al., 2022), further improves SRRL by allowing the student to use a frozen, pre-trained teacher classifier for additional supervision. Despite the improvement achieved by SRRL-based methods, the mechanism that achieves this performance is not well understood because they are based on intuition rather than a solid theoretical explanation.

This paper explains the reasoning for using the teacher classifier as supervision in softmax regression-based distillation via a theory of domain adaptation. We summarized the existing methods and provides a theoretical analysis to understand the regularization effect of these methods. Our work presents an error bound that limits the student network's error by the teacher's error and two disagreement terms. As the error bounds are derived, the central motivation for SRRL and SimKD can be explained. We also introduce the concept of the "Ideal Joint Classifier assumption" to tighten the upper bound and better understand how to translate the theory of the upper bound to algorithms that improve the task of knowledge distillation. Using this bound, we presents a unified framework called Ideal Joint Classifier Knowledge Distillation (IJCKD), to connect SRRL and SimKD. The error bound is derived using a proof scheme from a theory of domain adaptation (Ben-David et al., 2010).

In conclusion, the authors' main contributions are:

- Introducing a theoretical examination of techniques rooted in softmax regression-based methods, providing insights into their operation within the context of knowledge distillation.
- Establishing an error bound that establishes a connection between teacher and student errors in the softmax regression setting, thereby offering a holistic approach to existing methods.
- Presenting the IJCKD framework, which unifies various softmax regression-based methods and opens avenues for future research extensions.

2. Related works

The history of compressing machine models via training smaller models under the supervision of large models can be traced back to Buciluǎ et al. (2006). In Buciluǎ et al. (2006), MUNGE presented a method train a compact neural network to mimic the large and complex ensembles of classifiers. Taking advantage of the universal approximation property of neural networks (Hornik, Stinchcombe, & White, 1989; Scarselli & Tsoi, 1998), MUNGE "compresses" the ensemble into smaller size networks. The concept of knowledge distillation was formally defined in Hinton et al. (2015), which defined the softmax output of the teacher network as the "knowledge". The student is supervised by hard labels from the oracle, and the soft labels from teacher outputs to achieve higher accuracy than a network trained without distillation. The knowledge distillation objective can be formulated as: $l^{all} = l^{ce} + \lambda \cdot l^{kd}$, where l^{ce} is the cross entropy between student output probabilities and the hard label, and the l^{kd} is the Kullback–Leibler (KL) divergence between the student and teacher output probabilities weighted by a factor λ. Typically, the l_{kd} term has a temperature factor τ that softens the output probabilities. The following distillation works focus on better use of the teacher's logits. The work proposed in Kim, Oh, Kim, Cho, and Yun (2021) comprehensively compared the difference and association between KL divergence between the softened output probabilities and mean square error (MSE) between the output logits prior to the softmax activation. Based upon their findings, Kim et al. proposed replacing the original KL divergence term with the MSE between the student and teacher logits (Kim et al., 2021). Their experimental results showed that the logits loss provided better distillation and performance. Further, decoupled knowledge distillation (DKD) (Zhao et al., 2022) divides the KL divergence into the target and non-target loss according to the ground truth label. Weighted soft label distillation (WSLD) (Zhou et al., 2021) analyzed the regularization effect of knowledge distillation through a bias–variance decomposition and proposed to rescale the knowledge distillation loss with respect to the regularization samples. Gou et al. (2023) proposed MTKD-SSR to use multi-stage learning and self-reflection in knowledge distillation. Their results showed significant improvements over existing methods in various computer vision tasks.

Feature matching (e.g., feature based distillation or representation distillation) aims to have the student network approximate intermediate features of the teacher's network. FitNet (Romero et al., 2014) directly uses the teacher's features as hints to supervise the student's feature representations. A generic feature distillation optimization task can be defined as: $l^{all} = l^{ce} + \lambda \cdot l^{fm}$, where l^{fm} is typically the distance between the intermediate features from teachers and students. Several representative feature based distillation methodologies were proposed after FitNet. Overhaul of Feature Distillation (OFD) (Heo et al., 2019) investigated the effect of the distillation feature position and distance function. Relational Knowledge Distillation (RKD) (Park, Kim, Lu, & Cho, 2019) designed a new loss from distance- and angle-wise perspectives to extract relations between the feature representations. Contrastive Representation Distillation (CRD) (Tian et al., 2019) deployed contrastive learning to maximize the mutual information between the student and teacher representations.

The focus of this paper is on a specific type of knowledge distillation, which we refer to as Softmax Regression-based Distillation. SRRL (Yang et al., 2021) uses the teacher's classifier as additional supervision to regularize the student's features. This approach ensures that the student features produce the same outputs as the teacher features when scored by the teacher classifier. By being supervised by the teacher classifier, SRRL achieves outstanding performance with simply two MSE loss terms between the student and teacher penultimate layer features, and the logits produced by the student and teacher features when passed through the teacher classifier. Following this, SimKD (Chen et al., 2022) adopts a more aggressive approach known as the reused teacher classifier. SimKD abandons the student classifier and lets the student use the frozen pre-trained teacher classifier as its own classifier. The student's network is trained with the feature matching loss to learn the same features as the teacher, and the teacher's network is frozen. SimKD also proposes a multi-layer convolutional module with little computational overhead to better align the teacher and student features.

3. Knowledge distillation under ideal joint classifier assumption

Given two deep neural networks f_t and f_s as the teacher and student networks, respectively. The knowledge distillation task aims to train a student network, f_s, from the output of a significantly larger teacher network f_t. More generally, we wish the student network to mimic the teacher. Therefore, upper-bounding the student network's error with the teacher's error is essential to study the learning framework of knowledge distillation. Let the dataset $\mathcal{D} = \{\mathcal{X}, \mathcal{Y}\}$ contain n data pairs of inputs $\mathcal{X} = \{x_i\}_{i=1}^n$ and the ground truth labels $\mathcal{Y} = \{y_i\}_{i=1}^n$. A deep neural network f typically consists of a feature extractor $\Phi : \mathcal{X} \to \mathcal{Z}$ to map the inputs x to the latent representations z and a linear layer (a classifier) $g : \mathcal{Z} \to \mathcal{O}$ to classify the latent representations z into their output logits o. We also denote p as the output probability of the classifier after softmax activation, where $p_k = \frac{exp(o_k)}{\sum_j exp(o_j)}$.

The error (also called risk) of a neural network f is measured by the disagreement between its predictions and the ground truth labels. Let us define an error $\epsilon(f)$:

$$\epsilon(f) = \mathbb{E}_D[|f(x) - y|] = \mathbb{E}_D[|g \circ \boldsymbol{\Phi}(x) - y|] = \epsilon(g \circ \boldsymbol{\Phi}). \tag{1}$$

To guarantee student performance under the supervision of both the ground truth labels and the teacher, we seek to limit student error in terms of teacher error. For a student network f_s and teacher f_t, we have a basic assumption that the teacher network has a larger capacity than the student, e.g., the teacher has a more significant number of parameters which let the teacher performs better than the student on dataset D. Thus, we could bound the student error in terms of the teacher error.

Theorem 1. *For a teacher network $f_t = g_t \circ \boldsymbol{\Phi}_t$, a student network $f_s = g_s \circ \boldsymbol{\Phi}_s$, the student error can be bounded as:*

$$\epsilon(f_s) \le \epsilon(f_t) + \underbrace{\mathbb{E}_D[|g_s \circ \boldsymbol{\Phi}_s(x) - g_t \circ \boldsymbol{\Phi}_s(x)|]}_{\Delta_1} + \underbrace{\mathbb{E}_D[|g_t \circ \boldsymbol{\Phi}_s(x) - g_t \circ \boldsymbol{\Phi}_t(x)|]}_{\Delta_2} \tag{2}$$

Proof.

$$\begin{aligned}
\epsilon(f_s) &= \epsilon(f_s) + (\epsilon(f_t) - \epsilon(f_t)) + (\epsilon(g_t \circ \boldsymbol{\Phi}_s) - \epsilon(g_t \circ \boldsymbol{\Phi}_s)) \\
&\le \epsilon(f_t) + |\epsilon(g_s \circ \boldsymbol{\Phi}_s) - \epsilon(g_t \circ \boldsymbol{\Phi}_s)| + |\epsilon(g_t \circ \boldsymbol{\Phi}_s) - \epsilon(g_t \circ \boldsymbol{\Phi}_t)| \\
&\le \epsilon(f_t) + \underbrace{\mathbb{E}_D[|g_s \circ \boldsymbol{\Phi}_s(x) - g_t \circ \boldsymbol{\Phi}_s(x)|]}_{\Delta_1} \\
&\quad + \underbrace{\mathbb{E}_D[|g_t \circ \boldsymbol{\Phi}_s(x) - g_t \circ \boldsymbol{\Phi}_t(x)|]}_{\Delta_2} \quad \square
\end{aligned}$$

This error bound consists of three distinct components: (a) the first term represents the teacher's risk on the dataset D, (b) the second term, Δ_1, captures the difference between the teacher and student classifiers, and (c) the third term, Δ_2, measures the discrepancy between the teacher and student models when scoring the same inputs. This bound reveals the requirement for training a good student network under knowledge distillation frameworks. First, the Δ_1 term tells us the learned student's classifier should to be similar to the teacher's classifier, which means that the student's and teacher's classifiers are supposed to have close outputs with respect to the same input features. The Δ_2 term represents that a suitable student feature is expected to lead to similar predictions under the teacher's classifier as the teacher representation. Further, this bound limits the error, but more importantly, it shows us three individual terms that we can seek to minimize to improve the task of knowledge distillation.

In practice, a pairwise loss function l is used for capturing the disagreements, including the MSE loss and the softmax cross-entropy loss, which are defined as:

$$l^{\text{mse}}(y_i, f(x_i)) = \left\|y_i - f(x_i)\right\|_2^2, \quad l^{\text{ce}}(y_i, f(x_i)) = -\log p(Y = y_i|x_i)$$

With a minor deviation from the original notation, the RHS of the inequality of Theorem 1 can be (approximately) rewritten as:

$$\epsilon(f_t) + \underbrace{\mathbb{E}_D\left[l(g_s \circ \boldsymbol{\Phi}_s(x), g_t \circ \boldsymbol{\Phi}_s(x))\right]}_{\Delta_1} + \underbrace{\mathbb{E}_D\left[l(g_t \circ \boldsymbol{\Phi}_s(x), g_t \circ \boldsymbol{\Phi}_t(x))\right]}_{\Delta_2}. \tag{3}$$

Note that we are not claiming a strict inequality here as we did in Theorem 1, and we have substituted the absolute loss in Eq. (1) with a generic loss $l(\cdot)$. At this point, we can make two observations regarding how SRRL and SimKD address the error bound. SRRL addresses the Δ_2 term by using a softmax regression loss, $l^{sr}(g_t \circ \boldsymbol{\Phi}_s(x), g_t \circ \boldsymbol{\Phi}_t(x))$. On the other hand, SimKD mitigates the Δ_1 term by allowing the student to share the teacher's pre-trained classifier, then minimizes the feature matching loss $l^{fm}(\boldsymbol{\Phi}_s(x), \boldsymbol{\Phi}_t(x))$. Based on these observations, we first address the Δ_1 term as our core assumption. The disparity between the teacher and student classifiers should be small. Hence, we can assume that there exists an ideal joint classifier that achieves the lowest risk for both the student and teacher representations.

Definition 1. The ideal joint classifier of the student and teacher representations on a dataset D is

$$\hat{g} = \arg\min_g \mathbb{E}_D[l(y, g \circ \boldsymbol{\Phi}_t(x)) + l(y, g \circ \boldsymbol{\Phi}_s(x))] \tag{4}$$

We assume that there exists an ideal joint classifier that could reach a low risk with both the teacher's and student's representations. The ideal joint classifier assumption provides insight into analyzing the student's performance. The student will be unlikely to learn effectively under the teacher's guidance when there is a large discrepancy between the student's and the teacher's classifiers. Under the ideal joint classifier assumption, the student's risk can be bounded by the teacher's risk and the discrepancy between the outputs of the ideal joint classifier over the student and teacher representations. SimKD provides a natural choice for reusing the teacher's classifier as the ideal joint classifier, which we use in our approach. Then, the bound is re-expressed with the ideal joint classifier. This highlights the significance of the ideal joint classifier assumption in understanding the student's performance and serves as a foundation for further analysis.

Now that we can substitute the ideal joint classifier into the RHS of the modified inequality of Theorem 1 described in Eq. (3) and demonstrate how it allows us to bound the student's error. Given a teacher and a student feature extractor $\boldsymbol{\Phi}_s$ and $\boldsymbol{\Phi}_t$, respectively, and an ideal joint classifier $\hat{g}$, the RHS of the inequality of Theorem 1 can be further rewritten as:

$$\epsilon(\hat{g} \circ \boldsymbol{\Phi}_t) + \mathbb{E}_D\left[l(\hat{g} \circ \boldsymbol{\Phi}_s(x), \hat{g} \circ \boldsymbol{\Phi}_t(x))\right]. \tag{5}$$

We can look at the SRRL learning objective. Recall that SRRL defines its learning objective as follows:

$$f_s = g_s \circ \boldsymbol{\Phi}_s = \arg\min_{g, \boldsymbol{\Phi}} \mathbb{E}_D[l^{ce}(y, g \circ \boldsymbol{\Phi}(x)) + \alpha l^{sr}(g_t \circ \boldsymbol{\Phi}_t(x), g_t \circ \boldsymbol{\Phi}(x))]. \tag{6}$$

The softmax regression loss, denoted as l^{sr}, is the key component of the SRRL's objective, and α controls the trade-off between the losses. We rewrite the SRRL objective function with the ideal joint classifier setting the softmax regression loss to cross-entropy loss:

$$\boldsymbol{\Phi}_s = \arg\min_{\boldsymbol{\Phi}} \mathbb{E}_D[l^{ce}(y, \hat{g} \circ \boldsymbol{\Phi}(x)) + \alpha l^{ce}(\hat{g} \circ \boldsymbol{\Phi}_t(x), \hat{g} \circ \boldsymbol{\Phi}(x))], \tag{7}$$

The objective function can be seen as training the student's feature extractor $\boldsymbol{\Phi}_t$ with the smoothed labels based on the teacher's prior over each class. If we followed the original implementations of SRRL (Yang et al., 2021), and set the l^{sr} be mean square error loss l^{mse}, the objective function is defined as:

$$\boldsymbol{\Phi}_s = \arg\min_{\boldsymbol{\Phi}} \mathbb{E}_D[l^{ce}(y, \hat{g} \circ \boldsymbol{\Phi}(x)) + \alpha l^{mse}(o^s, o^t)], \tag{8}$$

where o^s and o^t denote the logits produced by the teacher and student models, respectively. Recent work has observed that using MSE loss between the logits is more effective than cross-entropy loss (Kim et al., 2021; Yang et al., 2021). Then SRRL, which is closely related to SimKD through the incorporation of the ideal joint classifier assumption, can be formalized into the IJCKD framework. However, there is a key difference in the learning objective of IJCKD compared to SimKD, as Eq. (5) and the ideal joint classifier assumption suggest the minimization of both cross-entropy loss between the student outputs and the hard label, as well as the feature/logits matching loss simultaneously. Algorithms 1, 2, and 3 show the implementations of SRRL, SimKD, and IJCKD, respectively. Further, Fig. 1 provides an intuitive illustration of distinctions between SRRL, SimKD, and IJCKD.

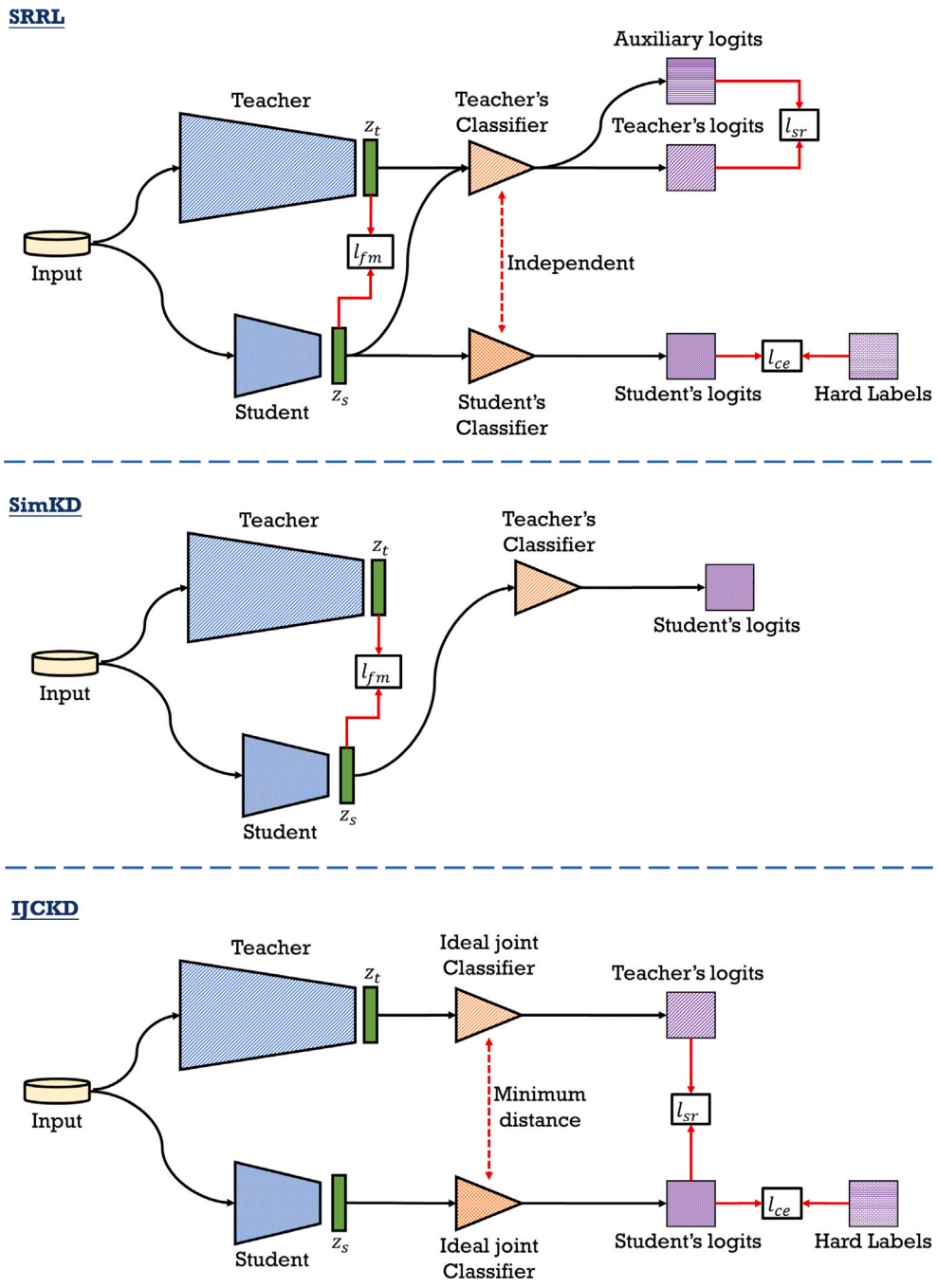


Fig. 1. An illustration to show the differences between the SRRL, SimKD, and IJCKD frameworks..

4. Experiments

This section presents experiments that demonstrate the effectiveness of our proposed IJCKD approach. We first present the results on standard benchmark datasets and compare them with several representative state-of-the-art approaches (CIFAR-100 (Krizhevsky, Hinton, et al., 2009), and ImageNet (Deng et al., 2009)). We compare IJCKD with SimKD in a separate subsection since the connector's design is a key factor in SimKD's performance. Additionally, we conduct experiments where a student network with only a pre-trained teacher classifier to verify whether the pre-trained teacher classifier can be the optimal classifier for the student network. We implemented the IJCKD in Python and the Pytorch framework (Paszke et al., 2019). The experiments were conducted on a workstation with an Intel i9-12900 processor, and two Nvidia RTX 3090 graphics cards with 24 GB RAM.

4.1. Results on CIFAR-100

We followed the standard training procedure adopted by previous works (Tian et al., 2019; Zhao et al., 2022; Zhou et al., 2021). Specifically, we train the networks for 240 epochs, and the learning rate is decayed multiplying 0.1 at the 150th, 180th, and 210th epochs, respectively. The initial learning rate is set to 0.01 when ShuffleNet is the backbone (Ma, Zhang, Zheng, & Sun, 2018; Zhang et al., 2018), and an initial learning rate to 0.05 for all other backbones. We used SGD with Nesterov's momentum of 0.9, a mini-batch size of 64, and a weight decay of 5×10^{-4}. To align the teacher and student feature channels, we applied a 1×1 convolutional layer followed by batch normalization (Ioffe & Szegedy, 2015), and ReLU activations (Nair & Hinton, 2010) to the student's features. We report the top-1 classification accuracy in Table 1, and each result is reported over five

Algorithm 1 PyTorch Code for SRRL

```
#input: x, hard label: y
#student network: net_s, teacher network: net_t
#connector for align teacher and student channels
logits_s, feat_s = net_s(x)
feat_s = connector(feat_s)
logits_t, feat_t = net_t(x)
logits_aux = net_t.fc(avg_pool(feat_s))

loss_fm = F.mse_loss(feat_s, feat_t)
loss_lm = F.mse_loss(logits_aux, logits_t)
loss_ce = F.cross_entropy(logits_s,y)

#alpha, beta are used to scale the losses

loss_srrl = loss_ce + alpha*loss_lm + beta*loss_fm
```

Algorithm 2 PyTorch Code for SimKD

```
#input: x
#student network: net_s, teacher network: net_t
#connector for align teacher and student channels
_, feat_s = net_s(x)
feat_s = connector(feat_s)
_, feat_t = net_t(x)

loss_fm = F.mse_loss(feat_s, feat_t)

loss_simkd = loss_fm
```

Algorithm 3 PyTorch Code for IJCKD

```
#input: x, hard label: y
#student network: net_s, teacher network: net_t
#connector for align teacher and student channels
_, feat_s = net_s(x)
feat_s = connector(feat_s)
logits_t, feat_t = net_t(x)
logits_s = net_t.fc(avg_pool(feat_s))

loss_lm = F.mse_loss(logits_s, logits_t)
loss_ce = F.cross_entropy(logits_s,y)

#alpha is used to scale the losses

loss_ijckd = loss_ce + alpha*loss_lm
```

run average. Our study primarily focuses on IJCKD and SRRL, with additional evaluations of several other recent state-of-the-art methods including FitNet (Romero et al., 2014), KD (Hinton et al., 2015), VID (Ahn, Hu, Damianou, Lawrence, & Dai, 2019), RKD (Park et al., 2019), PKT (Passalis & Tefas, 2018), OFD (Heo et al., 2019), CRD (Tian et al., 2019), WSLL (Zhou et al., 2021), and DKD (Zhao et al., 2022). We also present the reproduced results of SimKD with 1×1 connector. We compare distillation approaches across models with both the same and different architectures. To ensure a fair comparison and showcase the versatility of IJCKD, we set $\alpha = 1$ without an exhaustive search as the scaling factor for the logits matching loss in all teacher-student pairs.

Our experimental results offer valuable insights into the effectiveness of IJCKD compared to the SRRL method. Focusing on the differences between these two approaches, it becomes evident that IJCKD consistently outperforms SRRL across various teacher–student pairs evaluated in our study, as detailed in Table 1. For instance, let us consider the ResNet32x4→ResNet8x4 teacher–student pair, a particularly challenging scenario. Here, IJCKD achieves an impressive top-1 accuracy of 76.52%, surpassing SRRL by a substantial margin of 0.6%. These results underscore the remarkable efficacy of IJCKD in knowledge transfer and student network improvement. A key element contributing to IJCKD's superior performance lies in the combination of logits matching and cross-entropy losses under the Ideal Joint Classifier Assumption. By aligning the output logits of the student with those of the teacher, IJCKD enhances the student's ability to capture intricate details and fine-grained information from the teacher model. This effectively reduces the information gap between the teacher and student, resulting in more accurate predictions and higher classification accuracy. It is noteworthy that these improvements are not limited to specific teacher–student pairs; rather, IJCKD consistently demonstrates its superiority across the range of teacher architectures tested in this work.

This consistency highlights the robustness and general applicability of IJCKD as a knowledge distillation technique. Furthermore, when comparing IJCKD to other knowledge distillation methods in Table 1, we can see that IJCKD often achieves top-1 accuracy that surpasses or closely rivals the best-performing alternatives. This suggests that IJCKD holds great promise for enhancing the performance of student networks across a wide array of architectures and domains. In summary, our experimental findings strongly support the notion that IJCKD, by leveraging the Ideal Joint Classifier Assumption, consistently outperforms SRRL and other knowledge distillation techniques in improving student network accuracy.

4.2. Compare with SimKD

In this section, we conduct an in-depth analysis comparing the performance of IJCKD and SimKD across various connector architectures. Our primary goal is to assess the adaptability and effectiveness of these methods concerning different connector designs. To this end, we consider three distinct connector architectures: 1×1Conv, 1×1Conv-1×1Conv, and 1×1Conv-3×3Conv-1×1Conv, where the notation denotes the presence and size of convolutional layers in the connector. The results presented in Table 2 offer valuable insights into the comparative performance of IJCKD and SimKD with varying connector architectures. Notably, these findings highlight the versatility and adaptability of IJCKD, emphasizing its ability to excel across different connector designs.

IJCKD consistently outperforms SimKD, achieving superior performance even with a simpler 1×1Conv connector. This result underscores the robustness of the IJCKD approach, as it demonstrates its effectiveness in distillation across various architectures, even when minimal convolutional layers are involved. Conversely, SimKD struggles to match the adaptability of IJCKD. Specifically, SimKD fails to achieve competitive results when confronted with 1×1Conv and 1×1Conv-1×1Conv connectors. This limitation is indicative of SimKD's primary approach, which focuses on training the student to mimic the teacher's representation through a regression-like process. When paired with connectors that differ significantly from the teacher's architecture, SimKD faces challenges in achieving satisfactory performance. In contrast, IJCKD showcases itself as a robust and reliable knowledge distillation method capable of effectively handling varying connector architectures. The improved performance consistently observed across different connector designs reaffirms IJCKD's adaptability and highlights its potential as a versatile tool for knowledge transfer in diverse scenarios.

Table 1
Top-1 accuracy (%) on CIFAR-100.

	Same architecture style					Different architecture style		
Teacher	WRN-40-2	ResNet56	ResNet110	ResNet110	ResNet32x4	ResNet32x4	ResNet32x4	WRN-40-2
	75.61	72.34	74.31	74.31	79.42	79.42	79.42	75.61
Student	WRN-40-1	ResNet20	ResNet20	ResNet32	ResNet8x4	ShuffleNetV1	ShuffleNetV2	ShuffleNetV1
	71.98	69.06	69.06	71.14	72.50	70.50	71.82	70.50
FitNet	72.24	69.21	68.99	71.06	73.50	73.59	73.54	73.73
KD	73.54	70.66	70.67	73.08	73.33	74.07	74.45	74.83
VID	73.30	70.38	70.16	72.61	73.09	73.38	73.40	73.61
RKD	72.22	69.61	69.25	71.82	71.9	72.28	73.21	72.21
PKT	73.45	70.34	70.25	72.61	73.64	74.10	74.69	73.89
OFD	72.38	69.47	69.53	70.98	73.17	73.55	74.31	73.34
CRD	74.14	71.16	71.46	73.48	75.51	75.11	75.65	76.05
WSLL	74.48	72.15	72.19	74.12	76.05	75.46	75.93	76.21
DKD	74.81	71.97	72.31	74.11	76.32	76.45	77.07	76.70
SRRL	74.75	71.44	71.51	73.80	75.92	75.66	76.40	76.61
SimKD	71.76	68.86	69.98	72.85	75.91	76.48	76.55	75.65
IJCKD	75.14	71.73	71.76	73.98	76.52	76.51	76.56	76.84

Table 2
Comparison with SimKD with different connector architecture.

Teacher/Student	WRN-40-2 WRN-40-1		resnet32x4 resnet8x4	
Methods	IJCKD	SimKD	IJCKD	SimKD
1×1Conv	75.14	71.76	76.52	75.91
1×1Conv-1×1Conv	75.33	72.23	77.10	76.21
1×1Conv-3×3Conv-1×1Conv	75.57	75.48	77.76	77.46

Table 3
Top-1 accuracy for different loss scale and combination.

α_{ce}	SR only	0.1	0.2	0.5	1.0	CE only
top-1	75.91	76.36	76.61	76.09	76.52	74.46
Loss comb	$l^{ce}+l^{sr}$		$l^{ce}+l^{fm}$		$l^{ce}+l^{sr}+l^{fm}$	
top-1	76.52		76.47		76.86	

Table 4
Top-1 accuracy for different logits matching loss.

Teacher	Student	Logits matching loss		
		MSE	Cos Sim	CE
ResNet32x4	ResNet8x4	76.52	76.86	75.24
WRN-40-2	WRN-40-1	75.14	75.51	74.27

In summary, our comprehensive evaluation of IJCKD and SimKD across different connector architectures underscores IJCKD's superior adaptability and performance. These findings indicate that IJCKD is well-suited for scenarios involving a wide range of connector designs, making it a valuable choice for knowledge distillation tasks with varying model architectures and complexities.

4.3. Ablation study

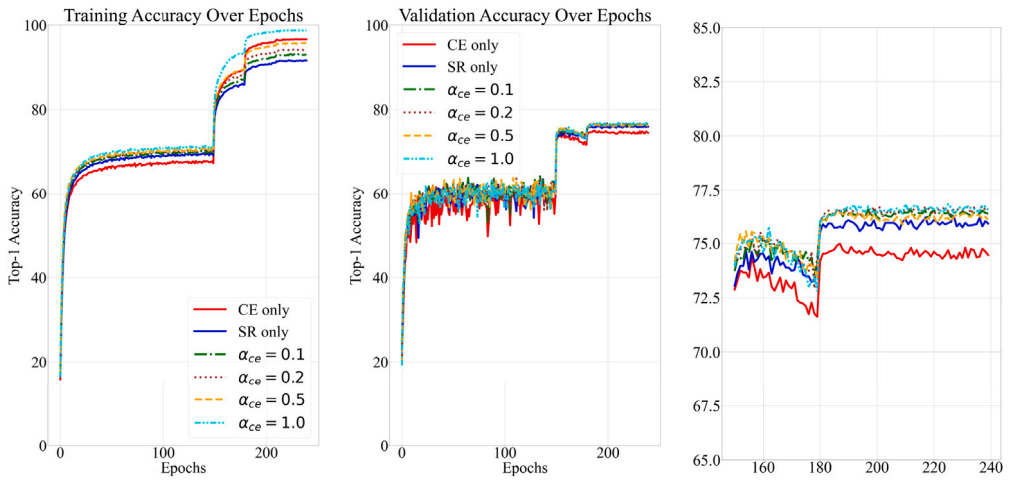
In this ablation study, the weight assigned to the cross-entropy loss and logits matching loss was assessed. Initially, we set the logits matching loss (α_{lm}) weight to 1 and then we adjust the cross-entropy loss (α_{ce}) weight from 0 to 1. When $\alpha_{ce} = 0$, the student network is trained exclusively under the teacher's logits supervision, a scenario labeled as "SR only". With increasing α_{ce}, the influence of the hard ground truth label becomes more pronounced in the learning process. This approach aligns with the ideal joint classifier assumption, which posits that the student network should be trained under both teacher and ground truth supervision. Additionally, the study explored the scenario of learning without the teacher's logits, termed 'CE only', where α_{lm} is set to 0. The results, specifically the top-1 accuracy for the ResNet32x4-ResNet8x4 teacher–student pair on the CIFAR-100 dataset, are presented in Table 3. Fig. 2 illustrates the corresponding training and validation top-1 accuracy curves. Notably, both 'SR only' and 'CE only' conditions were less effective than a linear combination of both losses, corroborating the proposed assumption and error bound. Optimal results were achieved with $\alpha_{ce} = 0.2$ and $\alpha_{ce} = 1.0$ for the best validation and training accuracy, respectively. However, it is crucial to acknowledge that the ideal balance of these losses varies depending on the teacher–student pair and dataset. Thus, future research should focus on hyperparameter tuning to identify the most effective loss scales for specific contexts, aiming to enhance distillation performance.

Further, in addition to exploring the impact of α, we delved into the influence of different combinations of feature and logits matching losses in our study. Our default choice, the naive softmax regression loss l^{sr} (MSE loss between teacher and student logits), was compared against the feature matching loss l^{fm} (MSE loss between teacher and student features). Additionally, we examined the combination of both losses. These experiments were conducted with the ResNet32x4-ResNet8x4 pair on CIFAR-100, and the resulting top-1 accuracy outcomes are presented in Table 3, while corresponding training and validation curves can be found in Fig. 3. Our results illuminate the significant impact of the chosen matching loss on IJCKD's performance. Intriguingly, the combination of feature and logits matching losses consistently outperformed the individual losses, highlighting the benefits of jointly optimizing both loss components in practical scenarios. This finding underscores the importance of considering and carefully selecting the matching losses when employing IJCKD, as this choice can significantly affect the distillation process's overall effectiveness.

Our study further explored various logits matching losses, with results presented in Table 4. This comparison encompassed MSE, negative cosine similarity, and cross-entropy as logits matching losses. The scale for both MSE and cross-entropy was set to 1, whereas for negative cosine similarity, we assigned an α_{lm} value of 10. Our findings indicated that negative cosine similarity, with its adjusted scale factor, consistently surpassed MSE and cross-entropy in achieving higher top-1 accuracy for both teacher-student pairings. Notably, the cross-entropy loss underperformed compared to the other two methods, aligning with observations reported in the SRRL paper (Yang et al., 2021). In summary, this ablation study underscores the significance of the appropriate combination of logits matching loss and cross-entropy loss. It also highlights the effectiveness of varying logits matching losses in different distillation contexts.

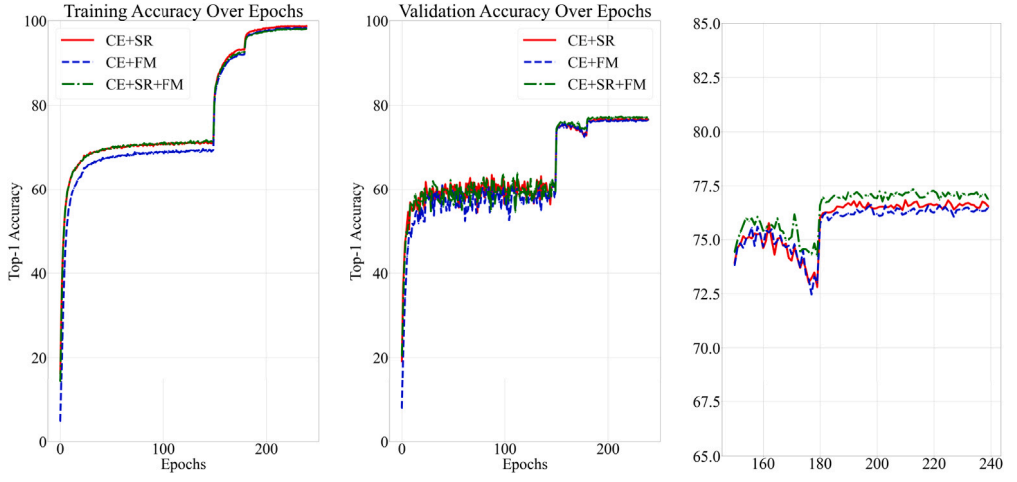
4.4. Results on ImageNet

In this section, we present the results on the ImageNet (Deng et al., 2009) dataset. We used the SGD optimizer with the same momentum parameter as we did for the CIFAR-100 dataset, and applied a weight



(a) training (b) validation (c) Zoom-in view of validation after 150 epochs.

Fig. 2. Top-1 accuracy curves for different α_{ce}.



(a) training (b) validation (c) Zoom-in view of validation after 150 epochs.

Fig. 3. Top-1 accuracy curves for different loss combination.

decay factor of 1×10^{-4}. For the training process, we utilized a batch size of 512 and commenced with a learning rate of 0.2. Our learning rate schedule involved reducing the learning rate by a factor of 10 at predefined epochs: 30th, 60th, and 90th, completing the training at the 120th epoch. we employed a linear combination of the cross-entropy loss (l^{ce}) and the softmax regression loss (l^{sr}) as we did for CIFAR-100 training.

The chosen architecture for the connector layers was a 1×1Conv-3×3Conv-1×1Conv configuration. The selection of distinct settings for the CIFAR-100 and ImageNet experiments was informed by insights from the SimKD settings (Chen et al., 2022). A single-layer transformation may be insufficient for precise alignment due to the significant capability disparity between the teacher and student models. Considering the intricacy and size of the ImageNet dataset, the 1×1Conv-3×3Conv-1×1Conv connector was selected for its ability to achieve accurate alignment between the teacher and student models, thereby enhancing performance. Additionally, as indicated in Chen et al. (2022), this configuration does not impose considerable computational overhead.

The experimental results for IJCKD, SRRL, and other comparative methods are detailed in Table 5, which illustrates performance metrics such as top-1 and top-5 accuracy. Notably, IJCKD consistently outperformed SRRL, demonstrating its superior efficacy in enhancing model performance on large-scale datasets, including ImageNet. For the ResNet-18 model, IJCKD achieved significant improvements, with increases of 0.51% in top-1 accuracy and 0.46% in top-5 accuracy. In the context of the MobileNet model, the improvements were even more pronounced, with IJCKD leading to a 2.18% enhancement in top-1 accuracy and a 1.46% rise in top-5 accuracy. These results emphatically affirm the effectiveness of the IJCKD approach, especially in the realm of complex, high-dimensional datasets.

4.5. Validation of the ideal joint classifier assumption

4.5.1. Can teacher classifier be the ideal joint classifier?

To evaluate the ideal joint classifier assumption that states the joint classifier should achieve lower risks on both teacher and student representations, we compared the performance of a student trained with the teacher's classifier to a student trained with its own classifier. Specifically, we evaluated four teacher-student pairs on CIFAR-100 and reported the top-1 accuracy in Table 6. The student networks were only trained with hard labels but using the pre-trained teacher classifier.

Table 5
Top-1 and top-5 accuracy (%) on ImageNet.

	T: ResNet-34 S: ResNet-18		T: ResNet-50 S: MobileNet-v1	
	Top-1	Top-5	Top-1	Top-5
Teacher	73.31	91.42	76.16	92.87
Student	69.75	89.07	68.87	88.76
KD	70.67	90.04	70.49	89.92
AT	71.03	90.04	70.18	89.68
CRD	71.17	90.13	69.07	88.94
WSLL	72.04	90.70	71.52	90.34
DKD	71.70	90.41	72.05	91.05
SRRL	71.73	90.60	72.49	90.92
IJCKD	**72.24**	**91.06**	**74.67**	**92.38**

Table 6
Top-1 accuracy for the student backbone with teacher and its own classifier.

Teacher	Student	Teacher's classifer	Student's classifer
WRN-40-2	WRN-40-1	72.32	71.98
ResNet56	ResNet20	68.91	69.06
ResNet110	ResNet32	70.98	71.14
ResNet32x4	ResNet8x4	74.46	72.50

The results show that training the student with the teacher and its own classifier achieved close results across all four teacher-student pairs.

This supports our assumption of the existence of the ideal joint classifier. However, we also observed that in some cases, the student trained with a teacher classifier obtained significantly higher accuracy. For example, the ResNet8x4 trained with ResNet32x4's classifier achieved 1.96% higher accuracy. This suggests that other factors may be at play and there are still unexplored mechanisms behind reusing the pre-trained teacher classifier. Overall, our experiments on CIFAR-100 provide evidence that the ideal joint classifier is a valuable concept in knowledge distillation. Further research is needed to explore its properties and potential applications.

4.5.2. Evolution of classifier alignment during SRRL training

In order to provide additional compelling evidence in support of the Ideal Joint Classifier Assumption, we conducted a series of experimental investigations aimed at observing the evolution of the relationship between the teacher's classifier and the student's classifier, particularly when the student network was trained under the SRRL framework. The objective was to elucidate whether the alignment between these classifiers dynamically evolves during the course of training, thereby further substantiating the Ideal Joint Classifier Assumption.

Fig. 4, illustrates the Frobenius Norm of the difference in weights between the teacher's classifier and the student's classifier over the course of SRRL training. The observed trends unveil a noteworthy pattern: as the performance of the student network progressively improves, there is a discernible trend towards a reduction in the distance between the teacher's classifier and the student's classifier. This empirical observation suggests that, indeed, as the student becomes more proficient, there is a growing convergence towards a shared classifier, corroborating the Ideal Joint Classifier Assumption.

These findings, in conjunction with our previous comparative analysis of student performance when trained with teacher classifiers versus their own classifiers, collectively underscore the validity of the Ideal Joint Classifier Assumption. It is apparent that the alignment of classifiers between the teacher and student networks benefits the performance of knowledge distillation algorithms, further reinforcing the central concept underlying our framework.

4.5.3. Alternative implementations of IJCKD

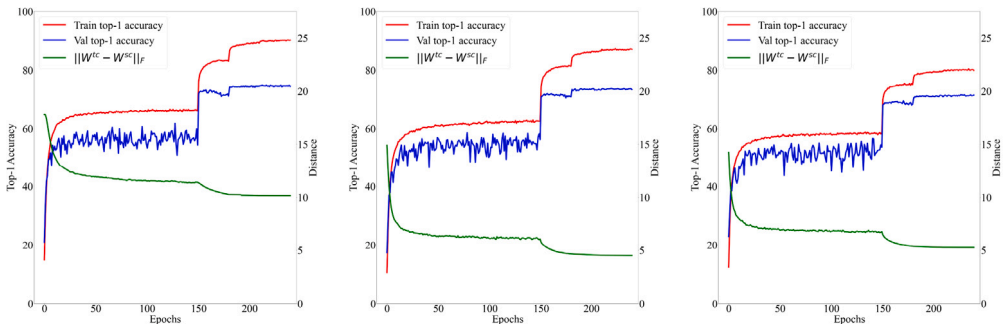
To further assess the versatility of the IJCKD framework, two alternative implementations are presented, extending beyond the mere reuse of the teacher's classifier. Each variant possesses unique characteristics and specific implementation details. The first alternative involves the online training of a joint classifier, as detailed in Algorithm 4. This method emphasizes the use of a jointly trained classifier to align teacher and student network representations. The procedure encompasses several critical steps: extracting features from both the student and teacher networks, generating logits using the joint classifier for both networks, and computing logits matching loss between the student's and teacher's logits. Additionally, the cross-entropy losses for both sets of logits are calculated and scaled by the factor α. In this implementation, α is set to 0.2, and β for scaling the logits matching loss is set to 1.0. This value was chosen based on the premise that the teacher's representations are already well-trained, and a higher scale of the teacher's cross-entropy loss would better balance the overall scale.

The second variant, as illustrated in Algorithm 5, incorporates a distance penalty to bridge the gap between the teacher and student classifiers in the distillation process. This implementation consists of several key components: extracting logits and features directly from both the student and teacher networks, computing logits matching loss between the two, and calculating the cross-entropy loss based on the student's logits and the provided hard label. A distinctive element of this approach is the introduction of a distance penalty, determined by the norm of the discrepancy in weights of the final fully connected layers of both networks. We simply set both the two scaling factors α and β to 1. This method focuses on reducing the differences between the teacher and student classifiers, guiding the student classifier towards an approximation of the ideal joint classifier.

Table 7 presents a comparative analysis of three different implementations of IJCKD across two network pairs, ResNet32x4-ResNet8x4 and WRN-40-2-WRN-40-1, on the CIFAR-100 dataset. For the ResNet32x4-ResNet8x4 pairing, the method utilizing the teacher's classifier reached a top-1 accuracy of 76.52%, with the joint training approach marginally surpassing it at 76.65%. The distance penalty variant was also effective, achieving a top-1 accuracy of 76.55%. In the case of the WRN-40-2-WRN-40-1 pair, the teacher's classifier strategy led with a top-1 accuracy of 75.14%, followed by the joint training method at 75.02% and the distance penalty method at 74.69%. These results indicate that IJCKD, as a comprehensive framework, is adaptable to various implementations, each showing promising results.

Algorithm 4 PyTorch Code for IJCKD with joint training.

```
#input: x, hard label: y
#student network: net_s, teacher network: net_t
#connector for align teacher and student channels
#joint_classifier get by online joint training
_, feat_s = net_s(x)
feat_s = connector(feat_s)
_, feat_t = net_t(x)

logits_t = joint_classifier(avg_pool(feat_t))
logits_s = joint_classifier(avg_pool(feat_s))

loss_lm = F.mse_loss(logits_s, logits_t)

#alpha is used to scale the teacher and student cross entropy losses

loss_ce = alpha*F.cross_entropy(logits_s,y) + (1-alpha)*F.cross_entropy(logits_t,y)

#beta is used to scale the losses
loss_ijckd = loss_ce + beta*loss_lm
```



(a) Teacher: ResNet32x4; Student: ResNet8x4 (b) Teacher: ResNet110; Student: ResNet32 (c) Teacher: ResNet110; Student: ResNet20

Fig. 4. Frobenius Norm of the difference of the weight of teacher's classifier (W^{tc}) and student's classifier (W^{sc}) during SRRL training.

Algorithm 5 PyTorch Code for IJCKD with distance penalty.

```
#input: x, hard label: y
#student network: net_s, teacher network: net_t

logits_s, feat_s = net_s(x)
logits_t, feat_t = net_t(x)

loss_lm = F.mse_loss(logits_s, logits_t)
loss_ce = F.cross_entropy(logits_s,y)
dist_penalty = torch.norm(net_t.fc.weight-net_s.fc.weight) # Default: Frobenius norm

#alpha and beta are used to scale the losses

loss_ijckd = loss_ce + alpha*loss_lm + beta*dist_penalty
```

Table 7
Comparison of three different implementations of IJCKD.

Teacher	Student	Teacher's classifier	Joint training	Distance penalty
ResNet32x4	ResNet8x4	76.52	76.65	76.55
WRN-40-2	WRN-40-1	75.14	75.02	74.69

5. Discussion

5.1. Scope

This paper presents the development and examination of the proposed IJCKD framework, which is built on the observation of previous works, SRRL and SimKD. Its main contribution is the ideal joint classifier assumption, which refines error bounds into optimization objectives and underscores the effectiveness of using the teacher's classifier in knowledge distillation. The IJCKD's adaptability is illustrated through three different implementations based on the ideal joint classifier assumption, showcasing its potential as a broad framework for new distillation algorithms. The IJCKD framework offers a novel perspective in knowledge distillation research but also encourages a more systematic, theory-based approach to developing distillation methods. Acknowledging the foundational nature of this assumption, the paper highlights the IJCKD framework's potential for further exploration and diverse applications, emphasizing its strength in integrating new ideas and deepening the understanding of knowledge distillation.

5.2. Understanding the 'Ideal Joint Classifier Assumption'

The 'Ideal Joint Classifier Assumption' refers to a classifier that achieves the lowest risk for the student and teacher representations, as described in Eq. (4) of our paper. In our work, we first adopt the setting of SimKD, where the teacher's classifier is reused as the ideal joint classifier. Note that the classifier in this context refers to an output layer of the neural network (e.g., the layer before the softmax). This choice using the SimKD setting is rooted in the teacher's classifier has already minimized the risk on their own representations; however, we extend this concept by recognizing that the ideal joint classifier should also jointly minimize the risk on the student representations. To achieve this, we include the cross-entropy loss (the first term in Eq. (8)) to minimize the risk of the teacher's classifier over the student representation with respect to the corresponding hard labels. Since the teacher's classifier is fixed in this context, we adapt the student representation with respect to the teacher's classifier to ensure that the teacher's classifier achieve the lowest risk on the student representations.

5.3. IJCKD as a corrective step

The IJCKD framework can be viewed as a corrective step in comparison to SimKD and SRRL. Where IJCKD distinguishes itself is by acknowledging that the ideal joint classifier should not solely minimize the risk of the teacher's classifier over the teacher representation but should also jointly minimize the risk of the student representation. To achieve this, we introduce the cross-entropy loss (the first term in Eq. (8)) to minimize the risk of the teacher's classifier concerning the student representation using corresponding hard labels. In this context, the teacher's classifier remains fixed, and the student representation adapts to ensure that the teacher's classifier achieves the lowest risk concerning the student representations. It enforces the reused teacher classifier to become the ideal joint classifier by introducing the cross-entropy loss term. This distinction becomes particularly significant when dealing with scenarios where the reused teacher is not inherently the ideal joint classifier, necessitating a more potent connector to effectively align the teacher and student representations.

5.4. Limitations and future research

Although the experimental results validate the IJCKD framework, there are concerns to consider. While IJCKD shares basic principles with SRRL and SimKD, its distinct approach, centered on the ideal joint classifier assumption, leads to unique challenges. A key success factor for these methods is the shared classifier between teacher and student networks. However, merely reusing the teacher's classifier might restrict IJCKD's applicability in certain contexts. The proposed alternative implementations partially mitigate this, yet the intricate balance between classifier discrepancy and adaptability remains less explored. It

is essential to delve deeper into this aspect and examine the nuances of classifier sharing and its impact on the versatility and effectiveness of the IJCKD framework. Thus, further investigations could focus on optimizing the balance between maintaining model accuracy and ensuring sufficient flexibility for diverse applications, potentially broadening the scope and utility of IJCKD in various knowledge distillation scenarios.

Also, there is a perception that the framework might be overly idealized, with concerns about the potential looseness of the error bound it proposes. However, it is important to emphasize that the significance of this error bound transcends its immediate numerical implications. The value of this error bound lies in its role as a fundamental concept, aiding in the comprehension of the complex interactions between teacher and student networks in the knowledge distillation process. It provides a theoretical lens through which the effectiveness and efficiency of knowledge transfer can be analyzed, offering insights into how different network architectures and training strategies might influence the distillation outcome. This understanding is crucial for refining the distillation techniques and advancing the field, pushing beyond mere empirical results to a more nuanced, theory-driven approach. However, future work should aim to refine and tighten this error bound, particularly in scenarios where there is a large discrepancy between the teacher and student representations. This would enhance the precision and applicability of the IJCKD framework, enabling more effective knowledge transfer in diverse and challenging distillation contexts.

The potential of the IJCKD framework extends beyond its current applications, heralding a new era of research that could revolutionize knowledge distillation across diverse sectors, including industrial inspection and beyond, as referenced in Hu, Dong, Shao, Zhang, and Wang (2023), Hu and Wang (2020) and Zhao, Hu, Shao, Wang, and Wang (2023). Its inherent flexibility showcases its applicability across a myriad of learning environments, fostering a fertile ground for innovative research. This adaptability not only opens new vistas for the development of more advanced and effective distillation methodologies but also invites the exploration of IJCKD's applicability in more specific and targeted areas. Future research should focus on tailoring the IJCKD framework to specific domains, thereby unlocking its full potential in specialized applications. This could involve adapting the framework to unique challenges and data characteristics of different fields, ranging from healthcare and finance to autonomous systems and beyond. Such targeted exploration could lead to highly specialized distillation techniques that are more efficient and effective in their respective areas. The adaptability and versatility of the IJCKD framework make it an ideal candidate for this kind of focused research, promising to yield significant advancements in both the theory and practice of knowledge distillation.

6. Conclusion

In summary, this paper theoretically analyzed the softmax regression-based representation learning, including two representative methods, SRRL and SimKD. We established an error bound which upper bounded the student's error by the teacher's error with the disagreement term between student and teacher output logits under the proposed ideal joint classifier assumption. Further, IJCKD, a novel knowledge distillation framework was proposed that unifies the previous works based on the idea of softmax regression. Our experiments demonstrate the effectiveness of IJCKD, which consistently outperformed state-of-the-art methods on a variety of benchmarks. Our results suggest that IJCKD is a versatile and effective method for knowledge distillation, which can be adapted to various architectures and datasets. Overall, we believe that our work provides a valuable contribution to the field of knowledge distillation and can be applied to a wide range of practical applications.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

CIFAR-100 and ImageNet are publicly available datasets.

Acknowledgments

This work was supported by grants from the National Heart, Lung, and Blood Institute, United States (#R21HL159661), and the National Science Foundation, United States (IUCRC #2052528 and CAREER #1943552). G. Ditzler was affiliated with Rowan University when this work was performed.

References

Ahn, S., Hu, S. X., Damianou, A., Lawrence, N. D., & Dai, Z. (2019). Variational information distillation for knowledge transfer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 9163–9171).

Ben-David, S., Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., & Vaughan, J. W. (2010). A theory of learning from different domains. *Machine Learning*, *79*(1), 151–175.

Buciluǎ, C., Caruana, R., & Niculescu-Mizil, A. (2006). Model compression. In *Proceedings of the 12th ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 535–541).

Chen, P., Liu, S., Zhao, H., & Jia, J. (2021). Distilling knowledge via knowledge review. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 5008–5017).

Chen, D., Mei, J.-P., Zhang, H., Wang, C., Feng, Y., & Chen, C. (2022). Knowledge distillation with the reused teacher classifier. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 11933–11942).

Chen, D., Mei, J.-P., Zhang, Y., Wang, C., Wang, Z., Feng, Y., et al. (2021). Cross-layer distillation with semantic calibration. *Vol. 35*, In *Proceedings of the AAAI conference on artificial intelligence* (pp. 7028–7036).

Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., & Yuille, A. L. (2017). Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *40*(4), 834–848.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition* (pp. 248–255). IEEE.

Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 580–587).

Gou, J., Xiong, X., Yu, B., Du, L., Zhan, Y., & Tao, D. (2023). Multi-target knowledge distillation via student self-reflection. *International Journal of Computer Vision*, 1–18.

Han, S., Mao, H., & Dally, W. J. (2015). Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. arXiv preprint arXiv:1510.00149.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770–778).

Heo, B., Kim, J., Yun, S., Park, H., Kwak, N., & Choi, J. Y. (2019). A comprehensive overhaul of feature distillation. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 1921–1930).

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

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, *2*(5), 359–366.

Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., et al. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861.

Hu, C., Dong, B., Shao, H., Zhang, J., & Wang, Y. (2023). Toward purifying defect feature for multilabel sewer defect classification. *IEEE Transactions on Instrumentation and Measurement*, *72*, 1–11.

Hu, C., & Wang, Y. (2020). An efficient convolutional neural network model based on object-level attention mechanism for casting defect detection on radiography images. *IEEE Transactions on Industrial Electronics*, *67*(12), 10922–10930.

Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning* (pp. 448–456). PMLR.

Kim, T., Oh, J., Kim, N., Cho, S., & Yun, S.-Y. (2021). Comparing kullback-leibler divergence and mean squared error loss in knowledge distillation. arXiv preprint arXiv:2105.08919.

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

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, *60*(6), 84–90.

Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3431–3440).

Ma, N., Zhang, X., Zheng, H.-T., & Sun, J. (2018). Shufflenet v2: Practical guidelines for efficient cnn architecture design. In *Proceedings of the European conference on computer vision* (pp. 116–131).

Nair, V., & Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning* (pp. 807–814).

Park, W., Kim, D., Lu, Y., & Cho, M. (2019). Relational knowledge distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 3967–3976).

Passalis, N., & Tefas, A. (2018). Learning deep representations with probabilistic knowledge transfer. In *Proceedings of the European conference on computer vision* (pp. 268–284).

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., et al. (2019). Pytorch: An imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems*, *32*.

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 779–788).

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

Scarselli, F., & Tsoi, A. C. (1998). Universal approximation using feedforward neural networks: A survey of some existing methods, and some new results. *Neural Networks*, *11*(1), 15–37.

Tai, C., Xiao, T., Zhang, Y., Wang, X., et al. (2015). Convolutional neural networks with low-rank regularization. arXiv preprint arXiv:1511.06067.

Tian, Y., Krishnan, D., & Isola, P. (2019). Contrastive representation distillation. arXiv preprint arXiv:1910.10699.

Tung, F., & Mori, G. (2019). Similarity-preserving knowledge distillation. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 1365–1374).

Yang, J., Martinez, B., Bulat, A., Tzimiropoulos, G., et al. (2021). Knowledge distillation via softmax regression representation learning. In *International conference on learning representations*.

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

Zhang, X., Zhou, X., Lin, M., & Sun, J. (2018). Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 6848–6856).

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

Zhao, C., Hu, C., Shao, H., Wang, Z., & Wang, Y. (2023). Towards trustworthy multi-label sewer defect classification via evidential deep learning. In *ICASSP 2023-2023 IEEE international conference on acoustics, speech and signal processing* (pp. 1–5). IEEE.

Zhao, H., Shi, J., Qi, X., Wang, X., & Jia, J. (2017). Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2881–2890).

Zhou, H., Song, L., Chen, J., Zhou, Y., Wang, G., Yuan, J., et al. (2021). Rethinking soft labels for knowledge distillation: A bias-variance tradeoff perspective. arXiv preprint arXiv:2102.00650.