# Dirichlet-Based Prediction Calibration for Learning with Noisy Labels 论文复现

**摘要**

本文聚焦于带噪声标签学习问题，对“Dirichlet - Based Prediction Calibration for Learning with Noisy Labels”论文进行复现。文中首先介绍了噪声标签的类别及相关研究现状，详细阐述了基于狄利克雷的预测校准（DPC）方法，该方法通过提出校准的 softmax 函数解决模型预测过自信问题，并借助证据深度学习构建训练损失，同时采用大间隔示例选择准则及 MixMatch 训练框架进行模型训练。在复现过程中，将原代码迁移至 DISC 框架并重构，在特定实验环境下进行实验。结果表明，复现结果与论文数据基本相符，但在高噪声率非对称噪声场景下存在误差。通过系列实验进一步验证了 DPC 方法及校准的 softmax 函数的有效性。最后，对研究进行总结并提出未来可从优化数据划分、推广校准的 softmax 函数及融合更多先进方法等方面深入探索的展望。

**关键词：**噪声标签；损失校准；机器学习；半监督学习

## 1 引言

带噪声标签学习（Noise Label Learning，NLL）是指在机器学习模型训练过程中，由于标签数据存在误标或噪声（如错误的标签），导致模型训练受到影响的问题。这类问题通常出现在实际应用中，尤其是在大规模数据集或标注成本高昂的情况下，人工标注可能存在错误或不一致，进而造成标签噪声。带噪声标签学习的目标是设计有效的算法，使得模型在面对标签噪声时，依然能够学习到有用的知识，并提高模型的鲁棒性和泛化能力。

噪声标签又分为三种类别,对称噪声(Symmetric Noise)和非对称噪声(Asymmetric Noise)是两种常见的标签噪声类型。对称噪声指的是在错误标注时具有相同的概率，即每个标签都有相等的机会被污染成任何其他标签。而非对称噪声则是错误标签具有不同的概率，并且噪声会被标记为相似的类，这通常反映了更现实的情境，即某些类别更可能与其他类别混淆。实例依赖噪声（Instance-Dependent Noise，IDN）[18] 是指错误标签的概率取决于特定实例的特征，这种噪声更难以处理。实例依赖噪声假设扩展了之前的设定，认为加噪过程不仅取决于真实标签和噪声标签，还取决于输入样本。这意味着对于每个样本，其标签噪声的模式可能都是独特的，这使得设计鲁棒的学习算法变得更加复杂。

## 2 相关工作

在机器学习和深度学习领域，带噪声标签的问题一直是一个重要且具有挑战性的研究方向。噪声标签通常由数据标注过程中的人为错误、标签的不一致性或自动标注系统的缺陷引起。噪声标签会严重影响模型的训练，导致模型性能下降，在存在噪声标签的情况下，已知训练 DNN 容易受到噪声标签的影响，因为大量的模型参数使 DNN 过度拟合到甚至具有学习任何复杂函数能力的损坏标签 [6]。因此，如何处理带噪声标签的数据，成为了一个亟待解决的问题。近年来，国内外学者针对带噪声标签的学习问题提出了多种解决方法，取得了显著进展。

Zhang 等人 [19] 证明，DNN 可以很容易地适应具有任何比例的损坏标签的整个训练数据集，这最终导致测试数据集的泛化能力较差。不幸的是，流行的正则化技术，如数据增强、权重衰减、dropout 和批归一化已被广泛应用，但它们本身并不能完全克服过拟合问题。即使激活了上述所有正则化技术，在干净和有噪声数据上训练的模型之间的测试精度差距仍然很大。此外，标签噪声导致的精度下降被认为比输入噪声等其他噪声更有害 [24]。因此，在存在噪声标签的情况下实现良好的泛化能力是一个关键挑战。现有的用于应对噪声标签问题的深度学习方法可以归类为以下 4 种类别 [14]。

### 2.1 基于模型结构的方法

在底层 DNN 的顶部添加噪声自适应层以学习标签转换过程，或开发专用架构以可靠地支持更多类型的标签噪声。通过设计特殊的网络结构，可以增强模型对噪声的鲁棒性，但需要对具体任务有深入理解，设计难度较高。添加噪声自适应层的一个共同缺点是他们无法识别错误标记的例子，对所有例子一视同仁。因此，当只使用有噪声的训练数据或噪声率很高时，转移矩阵的估计误差通常很大 [17]。与噪声自适应层相比，开发专用架构方法显著提高了对更多类型标签噪声的鲁棒性，但通常不能轻易扩展到其他架构 [1–3]。

### 2.2 基于正则的方法

强制 DNN 显式或隐式地减少对错误标记示例的过度拟合。通过避免模型训练中的过拟合，使用广泛使用的正则化技术，如数据增强、权重衰减、dropout 和批归一化，可以提高对标签噪声的鲁棒性。该系列方法的主要优势在于其与其他方向协作的灵活性，因为它只需要简单的修改。但显式正则化通常会引入敏感的依赖于模型的超参数，或者需要更深的架构来补偿容量的减少，但如果对其进行优化调整，它可以带来显著的性能提升。隐式正则化在不降低表示能力的情况下提高了 DNN 的泛化能力。它也不会引入敏感的模型相关超参数，因为它应用于训练数据。然而，扩展的特征或标签空间会减缓训练的收敛速度 [5, 15, 20, 21]。

### 2.3 基于损失函数的方法

根据其调整原理，与之相关的方法可分为三组 [10–12]：1）损失校正，估计噪声转移矩阵以校正前向或后向损失；2）对加权训练方案的每个示例施加不同重要性的损失重新加权；3）标签翻新，使用从噪声和预测标签的凸组合中获得的翻新标签来调整损失；这些方法的鲁棒性在理论上得到了很好的支持。然而，它们只在简单的情况下表现良好，即学习容易或类数

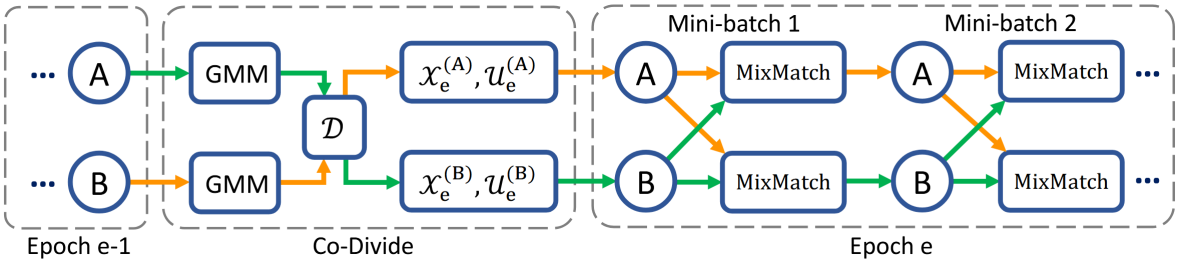



图 1. DivideMix 训练流程图

量少。此外，损失函数的修改增加了收敛所需的训练时间。损失校正方法的鲁棒性高度依赖于如何精确地估计转移矩阵。为了获得这样的转换矩阵，通常需要先验知识，比如锚点或者干净的验证数据。损失重加权方法需要人工预先指定加权函数和额外的超参数，这在实际应用中相当困难，因为合适的加权方案依赖于噪声类型和训练数据的显著变化。如果数据中没有太多的易混淆类，标签翻新方法通过高精度地重新整理噪声标签，效果很好。相反，DNN 可能会过度拟合错误翻新的标签。

### 2.4 基于样本选择的方法

使用样本选择进行学习具有很好的动机，并且总体上效果良好，但这种方法会因不正确的选择而产生累积误差，特别是在训练数据中存在许多模糊类的情况下。因此，最近的方法通常利用多个 DNN 相互合作或进行多轮训练。此外，为了从错误标记的示例中受益，损失校正或半监督学习最近与样本选择策略相结合 [4, 8, 16]。联合训练方法有助于减少确认偏差，这是一种倾向于在训练开始时选择的示例的风险，而可学习参数数量的增加使其学习流程效率低下。此外，当真标记和假标记示例的损耗分布在很大程度上重叠时，小损失技巧效果不佳。多轮学习方法，所选择的干净集随着迭代精化而不断扩展和纯化，但训练的计算成本随着训练轮数的增加而线性增加。通过与其他技术相结合，噪声鲁棒性得到了显著提高。然而，这些技术引入的超参数使 DNN 更容易受到数据和噪声类型变化的影响，导致计算成本的增加是不可避免的。

## 3 本文方法

### 3.1 本文方法概述

本文提出了基于狄利克雷的预测校准（DPC）方法 [25] 来解决带噪标签学习问题。首先针对 softmax 函数的平移不变性导致模型预测过自信的问题，提出校准的 softmax 函数，在指数项添加常数打破平移不变性，以获得更可靠的输出概率。接着引入证据深度学习（EDL）[13]，将概率视为随机变量，用 Dirichlet 分布表示概率密度，通过最小化负对数边际似然和采用 KL 散度项构建训练损失，解决校准带来的梯度缩小问题。基于校准后 logits 更具区分性，提出大间隔示例选择准则，根据给定类与互补标签最大可能类的预测 logits 之差定义间隔，用高斯混合模型拟合间隔分布来划分示例。最后在训练时将数据集分为干净子集和错误标记子集，采用 MixMatch 作为半监督学习框架，并使用双头网络架构分别处理监督损失和无监督损失，计算总体损失进行模型训练。

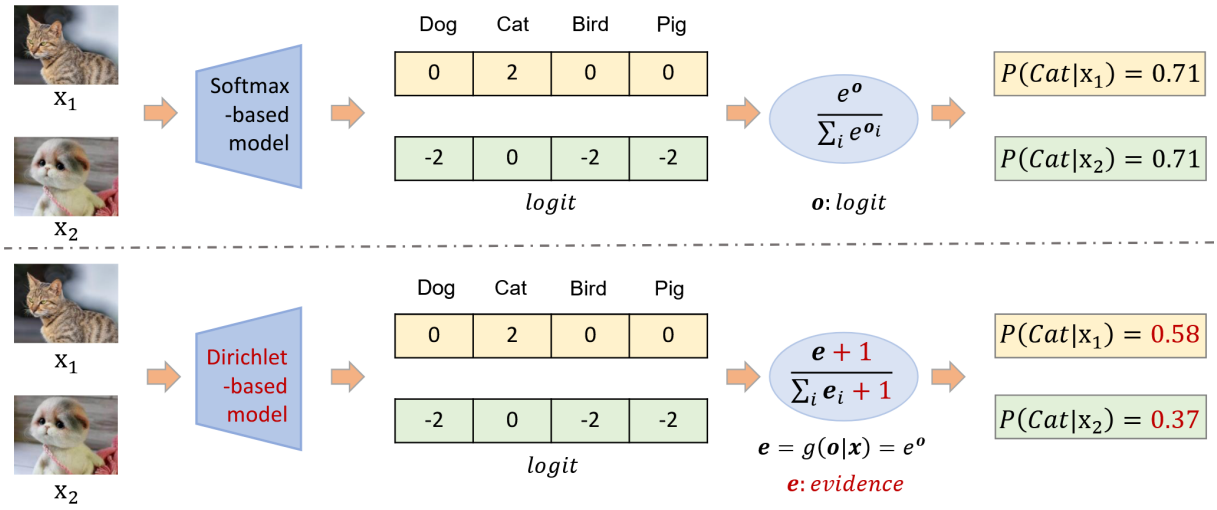



图 2. softmax 的平移不变性示例图

### 3.2 基线方法介绍

DPC 方法是在 DivideMix [7] 基础上改进完成的，并使用了与其相同的训练框架。具体的训练流程如图 1所示。DivideMix 同时训练两个网络 (A 和 B)。在每个 epoch，一个网络用 GMM 对其每个样本的损失分布进行建模，将数据集分为有标记集 (大部分是干净的) 和无标记集 (大部分是噪声的)，然后将其作为另一个网络的训练数据 (即 Co-Divide)。在每个小批量下，网络使用改进的 MixMatch 方法进行半监督训练。在有标签样本上执行 Co-refinement (对真实值和预测值做线性组合，然后做锐化得到标签)，在无标签样本上执行 Co-guess（采用两个网络预测结果的均值作为无标签样本的猜测值)。

### 3.3 过置信现象分析

在 DPC 中提到使用分治混合策略，如上述 DivideMix 方法训练的模型经常会出现基于期望校准误差（ ECE ）的过度自信问题。猜测现有的样本选择和标签校正方法都高度依赖于模型预测的质量，如果预测的概率更接近真实概率，则模型将更精确地选择干净的样本。同时发现了 softmax 的平移不变性，如图 2所示。

图中给出了一个具体的案例比较了基于 softmax 的模型和作者提出的基于狄利克雷校准的模型。softmax 函数具有平移不变性，即只能反映 logits 之间的相对关系，对 $x_1$ 和 $x_2$ 给出相同的预测，这与我们的主观直觉相矛盾。作者通过在指数项上放置一个合适的常数，并提出相应的基于狄利克雷的训练方法来打破平移不变性。

### 3.4 基于狄利克雷的预测校准

如前所述，将 logit 向量转换为概率向量的 softmax 算子往往会导致模型预测过度自信。为了获得更可靠的输出概率，作者提出了一个校准的 softmax 函数，并将 $xi$ 的预测概率表示为:

$$\hat{\rho}_{ic} = \frac{e^{o_{ic}} + \gamma}{\sum_{j=1}^{C} (e^{o_{ij}} + \gamma)}, \quad c = 1, 2, \ldots, C, \tag{1}$$

其中 $\gamma$ 为常数。然而，由于校准前后的梯度差在互补标签 (除给定标签外的其他标签) 上始终大于 0，校准导致常用的交叉熵损失存在梯度收缩问题，这一点可以表现为：

$$\begin{aligned}&\frac{\partial \mathcal{L}_{ce|\boldsymbol{\rho}_i}}{\partial o_{ic}}|_{\boldsymbol{x}_i}-\frac{\partial \mathcal{L}_{ce|\hat{\boldsymbol{\rho}}_i}}{\partial o_{ic}}|_{\boldsymbol{x}_i}\\&=\frac{\gamma C e^{o_{ic}}}{\sum_{j=1}^{C} e^{o_{ij}} \sum_{j=1}^{C}\left(e^{o_{ij}}+\gamma\right)}>0, \forall y_{ic}=0.\end{aligned} \tag{2}$$

然而简单地在模型输出中添加正则化项（例如，KL 散度或 L2 正则化）会提供过强的约束。因此引入证据深度学习（EDL），将证据项作为从数据中收集的支持量的度量，有利于将示例分类到某个类别。

与给出 $\rho$ 的点估计的传统 DNN 不同，EDL 将 $\rho$ 视为随机变量，并在 $\rho$ 上放置狄利克雷分布以表示每个可能的 $\rho$ 的概率密度：

$$\begin{aligned}&p\left(\boldsymbol{\rho}_i|\boldsymbol{x}_i, \boldsymbol{\theta}\right)=Dir\left(\boldsymbol{\rho}_i|\boldsymbol{\alpha}_i\right)\\&=\begin{cases}\frac{\Gamma\left(\sum_{j=1}^{C} \alpha_{ij}\right)}{\Pi_{j=1}^{C} \Gamma\left(\alpha_{ij}\right)} \Pi_{j=1}^{C} \rho_{ij}^{\alpha_{ic}-1}, & \text{if } \boldsymbol{\rho}_i \in \triangle^C, \\ 0, & \text{otherwise },\end{cases}\end{aligned}$$

式中：$\alpha_i$ 为 $x_i$ 的 Dirichlet 分布参数，$\Gamma(\cdot)$ 为 Gamma 函数，$\triangle^C=\{\boldsymbol{\rho}_i \mid \sum_{j=1}^{C} \rho_{ij}=1$ and $\forall j 0 \leq \rho_{ij} \leq 1\}$ 是一个 C 维单位单形。进一步得到模型对于 $c$ 类的预测概率：

$$\begin{aligned}P(y=c|\boldsymbol{x}_i, \boldsymbol{\theta})&=\int p(y=c|\boldsymbol{\rho}_i) p(\boldsymbol{\rho}_i|\boldsymbol{x}_i, \boldsymbol{\theta}) d\boldsymbol{\rho}_i\\&=\frac{\alpha_{ic}}{\sum_{j=1}^{C} \alpha_{ij}}=\frac{g(o_{ic})+\gamma}{\sum_{j=1}^{C}\left(g(o_{ij})+\gamma\right)}.\end{aligned} \tag{3}$$

### 3.5 损失函数设计

通过驱动 EDL 模型为所有标记数据生成位于 $\triangle^C$ 角处的尖锐狄利克雷分布来训练 EDL 模型。为了确保这一点，一方面，最小化边际似然的负对数（$\mathcal{L}_{nll}$）以确保预测的正确性：

$$\begin{aligned}\mathcal{L}_{nll}&=-\frac{1}{N} \sum_{i=1}^{N} \log\left[P(y=c|\boldsymbol{x}_i, \boldsymbol{\theta})\right]\\&=\frac{1}{N} \sum_{i=1}^{N} \sum_{c=1}^{C} y_{ic}\left[\log\left(\sum_{j=1}^{C} \alpha_{ij}\right)-\log \alpha_{ic}\right].\end{aligned} \tag{4}$$

另一方面，为了解决方程2提出的问题并规则化预测分布，采用 KL 散度项 Lkl，通过惩罚互补标签的证据使其趋近于 0：

$$\begin{aligned}\mathcal{L}_{kl}&=\frac{1}{NC} \sum_{i=1}^{N} D_{KL}\left(Dir\left(\boldsymbol{\rho}_i|\tilde{\boldsymbol{\alpha}}_i\right) \| \; Dir\left(\boldsymbol{\rho}_i|\mathbf{1}\right)\right)\\&=\frac{1}{NC} \sum_{i=1}^{N}\left[\log\left[\frac{\Gamma\left(\sum_{j=1}^{C} \tilde{\alpha}_{ij}\right)}{\Gamma(C) \Pi_{j=1}^{C} \Gamma\left(\tilde{\alpha}_{ij}\right)}\right]\right.\\&\left.+\sum_{c=1}^{C}\left(\tilde{\alpha}_{ic}-1\right)\left[\psi\left(\tilde{\alpha}_{ic}\right)-\psi\left(\sum_{j=1}^{C} \tilde{\alpha}_{ij}\right)\right]\right],\end{aligned} \tag{5}$$

其中 $\tilde{\boldsymbol{\alpha}}_i = \boldsymbol{y}_i + (1 - \boldsymbol{y}_i) \odot \boldsymbol{\alpha}_i$ 可以看做从预测参数 $\alpha$ 中移除给定标签证据后的 Dirichlet 参数，1 是一个由 $C$ 个参数组成的向量，$\psi(\cdot)$ 表示 digamma 函数. 然后，总体训练损失可以表示为：

$$\mathcal{L}_{edl} = \mathcal{L}_{nll} + \beta \mathcal{L}_{kl}, \tag{6}$$

其中 $\beta$ 是用来平衡这两个项的。

### 3.6 大间隔样本选择标准

为了在单个算例上达到相同的预测概率，校准后的 softmax 需要提供比常用 softmax 函数具有更大区分度的 logit 分布。这使得所提出的校准方法产生的输出对数分布更具有区分性。因此, 作者提出了一个大间隔的样本选择标准，并将给定样本 xi 的间隔定义为给定类与互补标签的最大可能类之间的预测 logits 之差：

$$margin\left(\boldsymbol{x}_i\right) = o_{ic} - \max_{j \neq c} o_{ij}, \text{ where } c = arg\max \boldsymbol{y}_i.$$

## 4 复现细节

### 4.1 与已有开源代码对比

源论文提供了相关源代码，具体代码地址如下：DPC 源码地址，在此代码基础上，我将其迁移到 DISC [9] 框架上，其源代码地址如下：DISC 源码地址上，并重构了代码，使其能适应新的训练框架以及新的 pytorch 版本。同时 DISC 框架上不支持分治混合策略的训练方式，因此重构了其主函数和数据加载等部分代码，使 DPC 能正常训练。后续尝试融合其他方法，包括 L2B [23],C2D [22] 方法，其源代码地址如下：L2B 源码地址,C2D 源码地址。由于 L2B 方法训练成本过大，后续放弃该方法。

### 4.2 实验环境搭建

使用 Pycharm2024.2.4 进行本地开发与部署，使用 Anaconda3 在 linux 服务器上进行环境搭建，CUDA 版本为 12.0，pytorch 版本为 2.4.1，训练使用单张 RTX3090 显卡。

### 4.3 创新点

- 将原来简单的代码迁移到统一的框架，使其能更好的与其他方法集成以及在不同类型噪声下训练与测试，并能通过 bash 脚本进行训练，同时生成更丰富的训练日志。
- 尝试选用不同的 backbone 网络训练策略，验证不同网络结构对性能的影响。
- 尝试融合其他先进方法，增强网络泛化能力和准确性。

表 1. 结果复现表

| Dataset | CIFAR-10 | | | | | | CIFAR-100 | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Symmetric | | | Asymmetric | | | Symmetric | | | Asymmetric | | |
| Noise Rate | 20% | 50% | 80% | 10% | 30% | 40% | 20% | 50% | 80% | 10% | 30% | 40% |
| DivideMix | 95.7 | 94.4 | 92.9 | 93.8 | 92.5 | 91.7 | 76.9 | 74.2 | 59.6 | 71.6 | 69.5 | 55.1 |
| DPC | 96.1 | **95.2** | **93.5** | **95.5** | **94.5** | **93.6** | **79.4** | **76.5** | **63.0** | 79.0 | **77.6** | 74.1 |
| DPC* | **96.2** | 95.0 | 93.0 | 95.4 | - | 92.9 | 79.2 | 76.4 | 62.7 | **79.1** | - | **75.4** |
| △ | 0.1 | -0.2 | -0.5 | -0.1 | - | -0.7 | -0.2 | -0.1 | -0.3 | 0.1 | - | 1.3 |

表 2. 实例依赖噪声测试结果表

| Dataset | CIFAR-10 | | | CIFAR-100 | | |
|---|---|---|---|---|---|---|
| | Ins | | | Ins | | |
| Noise Rate | 20% | 40% | 60% | 20% | 40% | 60% |
| DivideMix | 93.47 | 95.18 | 86.21 | 79.25 | 76.43 | 48.04 |
| DPC* | **96.37** | **95.49** | **95.16** | **79.93** | **79.48** | **76.72** |
| △ | 2.90 | 0.31 | 8.95 | 0.68 | 3.05 | 28.68 |

## 5 实验结果分析

所有实验在单张 RTX3090 进行，如无特殊说明网络选择 18-layer PreAct Resnet，并使用 SGD 进行训练，超参按照原论文所给出的进行设置。训练数据集使用 CIFAR-10 和 CIFAR-100。

### 5.1 数据集

CIFAR-10 数据集由 60000 张 32x32 的彩色图像组成，分为 10 个类别，每个类别有 6000 张图像。其中，50000 张图像用作训练集，10000 张图像用作测试集。这些类别包括飞机、汽车、鸟类、猫、鹿、狗、青蛙、马、船和卡车。CIFAR-10 的数据集中，每个图像的标签是一个 0-9 范围内的数字，代表 10 个不同的类别。CIFAR-100 数据集与 CIFAR-10 类似，但它包含了 100 个类别，每个类别有 600 张图像，总共 60000 张图像。其中，50000 张图像用作训练集，10000 张图像用作测试集。CIFAR-100 的类别是层次化的，100 个类别被分为 20 个超类别（coarse labels），每个超类别包含 5 个细类别（fine labels）。

### 5.2 结果复现

表1显示 CIFAR-10 和 CIFAR-100 在最后 10 个 epoch 内具有不同水平的对称和非对称标签噪声的平均测试准确率。其中，DPC* 为复现数据，DivideMix 与 DPC 数据均来自于原

表 3. 消融实验结果表

| Dataset | CIFAR-10 | | | CIFAR-100 | | |
|---|---|---|---|---|---|---|
| | sym | | | sym | | |
| Noise Rate | 20% | 50% | 80% | 20% | 50% | 80% |
| DPC | **96.3** | **95.3** | **93.6** | **79.9** | **76.9** | **63.3** |
| DPC w/o $\mathcal{L}_{edl}$ | 96.1 | 94.9 | 93.3 | 79.4 | 75.6 | 60.3 |
| DPC* | 96.4 | 95.2 | 93.2 | 79.6 | 76.7 | 62.7 |
| DPC w/o softmax* | 96.2 | - | 92.1 | 79.5 | - | 59.8 |
| △ | -0.2 | - | -1.1 | -0.1 | - | -2.9 |

论文。复现结果基本符合论文给出的数据，但是在高噪声率的非对称噪声情况下有较大误差，可能是实验次数不够导致的误差，也可能是因为高噪声率的非对称噪声数据比较复杂，模型训练准确率波动较大。

此外，原论文只在对称噪声和非对称噪声上进行了测试，我通过迁移代码测试了 DPC 方法在实例依赖噪声上的效果，实例依赖噪声更容易被拟合，也更容易过拟合，因此可以更真实的反应模型在真实环境下的精度具体测试结果，具体数据如表2所示。

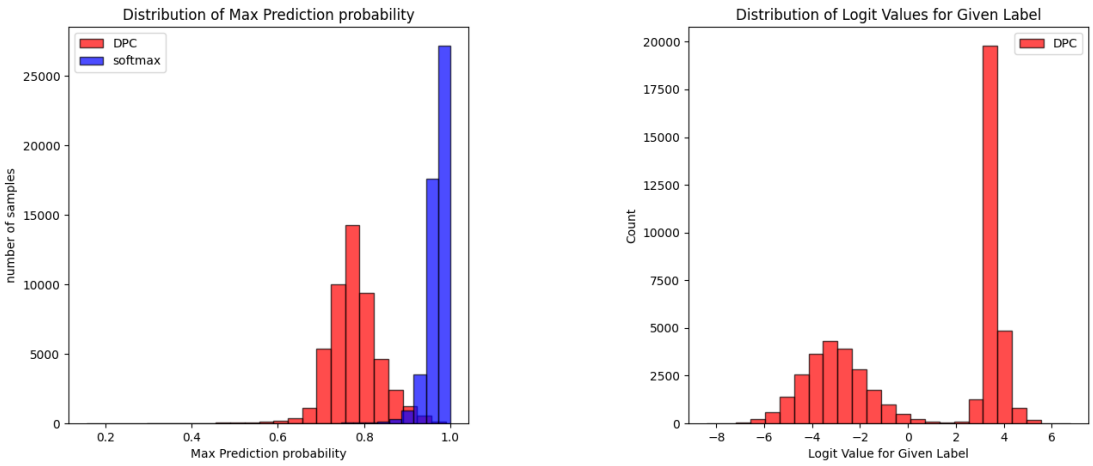
表中数据为训练过程中最好的结果。其中，DPC* 为复现数据，△ 表示与基线方法最好结果的差值，测试结果表示 DPC 在复杂的实例依赖噪声环境下依然有非常较好的性能，尤其是在高噪声率的 CIFAR-100 数据集上，较 baseline 方法有较大提高。

随后对校准的 softmax 函数进行测试，验证其是否能降低模型预测概率，以及验证 DPC 方法的 logit 是否更易区分。具体结果如图3所示，图 (a) 给出了训练实例的最大预测概率的分布，图 (b) 给出了训练样本的标签 logit 的分布。可以看到，使用校准后的 softmax 函数可以降低模型的最大预测概率，并且 DPC 方法能产生更可分的 logit，并且为 logit 赋予了更具体的含义，即 logit 越大/越小表明模型越有/越没有概率相信样本属于某一类。为了进一步验证校准的 softmax 函数是否真的有效果，补充了消融实验，具体如表3所示。

表中数据为训练过程中最好结果。其中 DPC w/o $\mathcal{L}_{edl}$ 是原论文去掉整个狄利克雷训练方案的结果，可以看到其对模型精度影响较大，为了进一步验证校准的 softmax 函数的有效性，我补充了 DPC w/o softmax*，即只消去校准后的 softmax 函数方法，保留其他模块，为了保证公平性，测试结果只与我复现结果进行对比，△ 表示去掉校准 softmax 函数后的数据与复现数据的差值，测试数据在不同噪声率的对称噪声集上都有不同程度的下降，尤其是在高噪声率的情况下，结果验证了提出的校准 softmax 的函数的有效性，说明基于 softmax 的预测概率确实存在偏差，有必要对现有的噪声标签学习方法进行校准。

### 5.3 替换 backbone 网络

为了提高网络性能，首先尝试使用不同的 backbone 进行测试，具体的测试结果如表4所示。表中数据为训练过程中最好结果。首先将网络更换为 Resnet34，以增强网络的特征提取能力，但是测试结果在多数情况下都会下降，只有在低噪声率的 CIFAR-100 上有细微提升。




(a) (b)

图 3. 校准的 softmax 预测概率与 DPC 输出 logit 结果图

表 4. 替换 backbone 后的测试结果表

| Dataset | CIFAR-10 | | | CIFAR-100 | | |
|---|---|---|---|---|---|---|
| | sym | | | sym | | |
| Noise Rate | 20% | 50% | 80% | 20% | 50% | 80% |
| DivideMix | 95.7 | 94.4 | 92.9 | 76.9 | 74.2 | 59.6 |
| DPC | 96.1 | **95.2** | **93.5** | 79.4 | **76.5** | **63.0** |
| DPC* | **96.2** | 95.0 | 93 | 79.2 | - | 62.7 |
| DPC_r34 | 94.7 | - | 91.9 | **80.2** | - | 56.4 |
| DPC_r14 | 94.8 | 93.6 | 91.7 | - | - | - |
| DPC_r34_d | 95.0 | - | 92.7 | 78.5 | - | 51.6 |

我猜测是替换更深层的网络提升了模型的特征抽取能力，但是因为噪声数据的干扰，导致模型在训练过程中对噪声数据过拟合导致性能的下降。以此出发，为了降低模型特征抽取能力，我将 Resnet18 网络去掉几层，以及在 Resnet34 的层间增加 dropout 抑制网络能力测试降低网络特征提取能力时的效果，但是结果并不理想。并且发现修改网络后准确率都有所下降，而且下降程度相差不多。可见降低特征提取能力也会降低模型拟合正确样本的能力，导致训练效果不佳。除此之外，也可能是更换网络后需要调整 warmup 阶段以及相关超参设置。

### 5.4 融合其他方法

为了克服 warmup 阶段的障碍，减少网络对噪声标签的拟合并提升网络性能，在 DPC 方法上融合了 C2D [22] 方法，这是通过以自监督的方式预训练特征提取器，来克服热身阶段障碍的方法，监督对比学习有效地利用标签信息，将属于同一类的点簇在嵌入空间中拉到一起，同时将来自不同类的样本簇推开，在表示和对比损失之间引入可学习的非线性变换，可以提

表 5. 融合方法测试结果表

| Dataset | CIFAR-10 | | | CIFAR-100 | | |
|---|---|---|---|---|---|---|
| | sym | | | sym | | |
| Noise Rate | 20% | 50% | 80% | 20% | 50% | 80% |
| DivideMix | 95.7 | 94.4 | 92.9 | 76.9 | 74.2 | 59.6 |
| DPC | 96.1 | 95.2 | 93.5 | 79.4 | 76.5 | 63.0 |
| DivideMix+C2D | 96.50 | 95.44 | 94.44 | 78.86 | **76.68** | 68.08 ± 0.30 |
| DPC+C2D | 96.9 | **95.98** | 94.73 | 79.93 | 76.67 | 68.00 |
| DPC+C2D_r50 | **97.23** | - | **95.46** | **82.64** | - | **68.96** |

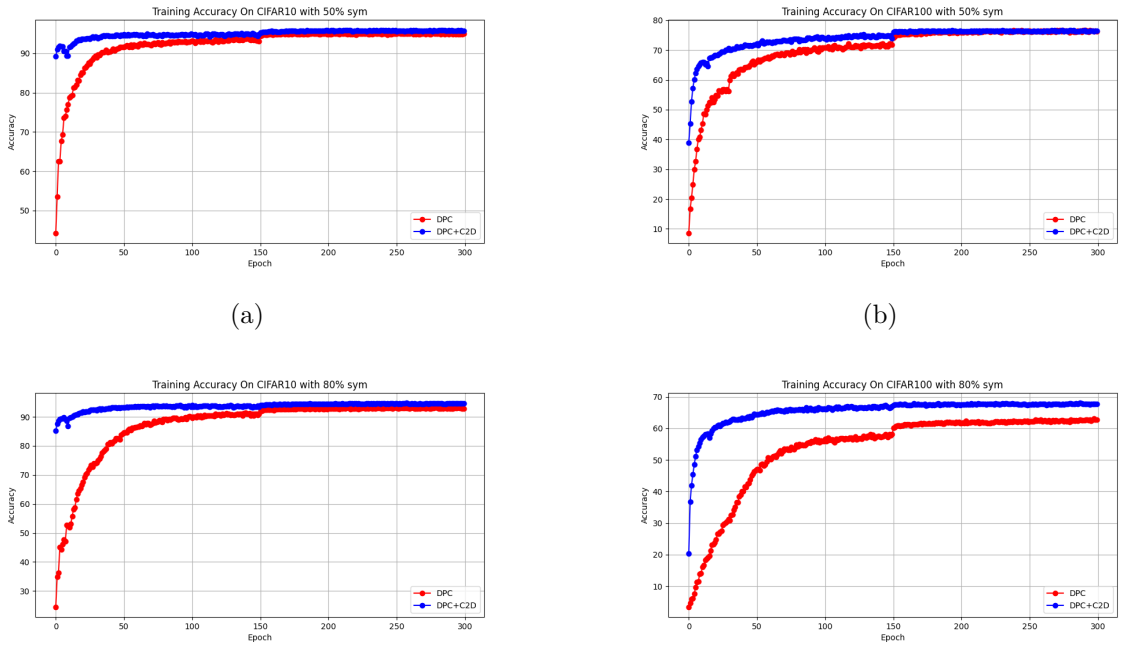
高学习表示的质量。融合 C2D 方法后测试结果如表5所示。

从表中结果可以看到，在融合了 C2D 方法后，DivideMix 方法和 DPC 方法的性能都有所提升，其中 DPC 方法提升幅度更大，也验证了 DPC 方法的有效性。并测试了使用更深层网络的结果，使用 Resnet50 作为 backbone，在结合 C2D 方法后可以明显的提升网络性能，同时避免了更换深层网络导致训练效果下降的问题，同时使用 C2D 方法后网络只需要更少的 warmup 就可以达到较好的效果。图4给出了更直观的训练过程精度结果果展示。从图中可以直观的看到，在融合 C2D 方法后，模型的精度有所提升，并且只需更短的 warmup 就可以达到较好的效果。

## 6 总结与展望

本文围绕 “Dirichlet - Based Prediction Calibration for Learning with Noisy Labels” 论文展开复现研究，旨在解决带噪标签学习问题。通过对相关工作的梳理，明确了现有应对噪声标签方法的四类方向及其优缺点。在此基础上，详细阐述了基于狄利克雷的预测校准（DPC）方法。该方法针对 softmax 函数平移不变性导致的模型预测过自信问题，提出校准的 softmax 函数，通过在指数项添加常数打破平移不变性，以获取更可靠的输出概率。同时引入证据深度学习（EDL），用 Dirichlet 分布表示概率密度，构建训练损失解决校准带来的梯度缩小问题。基于校准后 logits 更具区分性，提出大间隔示例选择准则划分示例，并采用 MixMatch 半监督学习框架及双头网络架构进行模型训练。在复现细节方面，将原论文代码迁移到 DISC 框架，重构部分代码以适应新框架与 PyTorch 版本，并尝试融合其他方法。实验结果表明，复现结果基本符合论文数据，但在高噪声率的非对称噪声下存在一定误差。通过对校准的 softmax 函数测试、消融实验、替换 backbone 网络以及融合其他方法等进一步研究，验证了 DPC 方法的有效性。

尽管 DPC 方法在处理带噪标签学习问题上取得了一定成果，但仍存在提升空间。未来可进一步探索如何更精准地划分干净集和噪声集，以提升模型性能。当前 DPC 方法对两者的区分效果有待提高，更准确的数据划分有助于模型更好地学习，减少噪声干扰。虽然校准的 softmax 函数在本研究中展现出降低模型过置信问题的优势，但能否在更广泛的场景中有效应用仍需进一步验证。未来可尝试在不同类型的数据集和任务中应用，以验证其通用性和




(a) (b)

(c) (d)

图 4. 在 50% 对称噪声情况 CIFAR-10 和 CIFAR-100 训练结果图

稳定性。继续探索与其他先进方法的融合，以增强网络的泛化能力和准确性。例如，结合更多无监督学习方法或针对噪声标签的新型处理技术，进一步提升模型在复杂噪声环境下的表现。

## 参考文献

[1] Alan Joseph Bekker and Jacob Goldberger. Training deep neural-networks based on unreliable labels. In *Proc. Int. Conf. on Acoustics, Speech & Signal Processing*, pages 2682–2686. IEEE, 2016.

[2] Xinlei Chen and Abhinav Gupta. Webly supervised learning of convolutional networks. In *Proc. Int. Conf. on Computer Vision*, pages 1431–1439, 2015.

[3] Jacob Goldberger and Ehud Ben-Reuven. Training deep neural-networks using a noise adaptation layer. In *Proc. Int. Conf. on Learning Representations*, 2017.

[4] Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor Tsang, and Masashi Sugiyama. Co-teaching: Robust training of deep neural networks with extremely noisy labels. *Advances in neural information processing systems*, 31, 2018.

[5] Simon Jenni and Paolo Favaro. Deep bilevel learning. In *Proc. Euro. Conf. on Computer Vision*, pages 618–633, 2018.

[6] Jonathan Krause, Benjamin Sapp, Andrew Howard, Howard Zhou, Alexander Toshev, Tom Duerig, James Philbin, and Li Fei-Fei. The unreasonable effectiveness of noisy data

for fine-grained recognition. In *Proc. Euro. Conf. on Computer Vision*, pages 301–320. Springer, 2016.

[7] J. Li, R. Socher, and S. C. Hoi. Dividemix: Learning with noisy labels as semi-supervised learning. In *Proc. Int. Conf. on Learning Representations*, 2020.

[8] Junnan Li, Richard Socher, and Steven CH Hoi. Dividemix: Learning with noisy labels as semi-supervised learning. In *Proc. Int. Conf. on Learning Representations*, 2020.

[9] Yifan Li, Hu Han, Shiguang Shan, and Xilin Chen. Disc: Learning from noisy labels via dynamic instance-specific selection and correction. In *Proc. IEEE/CVF Conf. on Computer Vision & Pattern Recognition*, pages 24070–24079, 2023.

[10] Tongliang Liu and Dacheng Tao. Classification with noisy labels by importance reweighting. *IEEE Transactions on pattern analysis and machine intelligence*, 38(3):447–461, 2015.

[11] Giorgio Patrini, Alessandro Rozza, Aditya Krishna Menon, Richard Nock, and Lizhen Qu. Making deep neural networks robust to label noise: A loss correction approach. In *Proc. IEEE/CVF Conf. on Computer Vision & Pattern Recognition*, pages 1944–1952, 2017.

[12] Scott Reed, Honglak Lee, Dragomir Anguelov, Christian Szegedy, Dumitru Erhan, and Andrew Rabinovich. Training deep neural networks on noisy labels with bootstrapping. In *Proc. Int. Conf. on Learning Representations*, 2015.

[13] Murat Sensoy, Lance Kaplan, and Melih Kandemir. Evidential deep learning to quantify classification uncertainty. *Proc. Conf. on Neural Information Processing Systems*, 31, 2018.

[14] Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. Learning from noisy labels with deep neural networks: A survey. *IEEE transactions on neural networks and learning systems*, 34(11):8135–8153, 2022.

[15] Ryutaro Tanno, Ardavan Saeedi, Swami Sankaranarayanan, Daniel C Alexander, and Nathan Silberman. Learning from noisy labels by regularized estimation of annotator confusion. In *Proc. IEEE/CVF Conf. on Computer Vision & Pattern Recognition*, pages 11244–11253, 2019.

[16] Yisen Wang, Weiyang Liu, Xingjun Ma, James Bailey, Hongyuan Zha, Le Song, and Shu-Tao Xia. Iterative learning with open-set noisy labels. In *Proc. IEEE/CVF Conf. on Computer Vision & Pattern Recognition*, pages 8688–8696, 2018.

[17] X. Xia, B. Han, N. Wang, J. Deng, J. Li, Y. Mao, and T. Liu. Extended $t$ t: Learning with mixed closed - set and open - set noisy labels. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3):3047 – 3058, 2022.

[18] Xiaobo Xia, Tongliang Liu, Bo Han, Nannan Wang, Mingming Gong, Haifeng Liu, Gang Niu, Dacheng Tao, and Masashi Sugiyama. Part-dependent label noise: Towards instance-dependent label noise. In *Proc. Conf. on Neural Information Processing Systems*, volume 33, pages 7597–7610, 2020.

[19] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3):107–115, 2021.

[20] Hongyi Zhang. mixup: Beyond empirical risk minimization. In *Proc. Int. Conf. on Learning Representations*, 2018.

[21] Shengnan Zhang, Yuexian Hou, Benyou Wang, and Dawei Song. Regularizing neural networks via retaining confident connections. *Entropy*, 19(7):313, 2017.

[22] Evgenii Zheltonozhskii, Chaim Baskin, Avi Mendelson, Alex M Bronstein, and Or Litany. Contrast to divide: Self-supervised pre-training for learning with noisy labels. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1657–1667, 2022.

[23] Yuyin Zhou, Xianhang Li, Fengze Liu, Qingyue Wei, Xuxi Chen, Lequan Yu, Cihang Xie, Matthew P Lungren, and Lei Xing. L2b: Learning to bootstrap robust models for combating label noise. In *Proc. IEEE/CVF Conf. on Computer Vision & Pattern Recognition*, pages 23523–23533, 2024.

[24] Xingquan Zhu and Xindong Wu. Class noise vs. attribute noise: A quantitative study. *Artificial intelligence review*, 22:177–210, 2004.

[25] Chen-Chen Zong, Ye-Wen Wang, Ming-Kun Xie, and Sheng-Jun Huang. Dirichlet-based prediction calibration for learning with noisy labels. In *Proc. AAAI Conf. on Artificial Intelligence*, volume 38, pages 17254–17262, 2024.