

# 基于用户指导的深度学习分类系统

温俊芳, 宋庆增\*

天津工业大学, 计算机科学与技术学院, 天津

收稿日期: 2022年2月23日; 录用日期: 2022年3月22日; 发布日期: 2022年3月29日

## 摘要

目前, 深度学习模型在现实中得到了广泛的应用, 当这些模型应用于不同的环境时, 可以利用环境中样本分布等经验来进一步提高分类的准确率。基于此, 本文提出了基于用户指导的深度学习分类系统 DLC-UG, 该方法有效地利用了环境中的样本分布来提高分类精度。首先, 系统在训练集上训练深度学习模型, 接着, 用户可以根据自己的经验或他人的建议, 在相应的环境中输入每个标签的分布, 之后, 系统可以通过这个训练模型得到每个样本标签的概率, 最后, 分布信息有选择地与测试样本上标签的概率相结合。论文在三个真实数据集上选择了六种流行的深度学习模型进行评估, 实验结果表明, 提出的方法能够提高分类任务的准确率, 且明显高于现有的分类方法。

## 关键词

标签分布, 标签概率, 用户指导

# Deep Learning Classification System Based on User Guidance

Junfang Wen, Qingzeng Song\*

School of Computer Science and Technology, Tiangong University, Tianjin

Received: Feb. 23<sup>rd</sup>, 2022; accepted: Mar. 22<sup>nd</sup>, 2022; published: Mar. 29<sup>th</sup>, 2022

## Abstract

Currently, deep learning models are widely used in many applications. When utilizing these models in different environments, the experience like the distribution of objects in an environment can be used to further increase the accuracy of classification. In this paper, we carried out user guidance based deep learning classification task that is named DLC-UG, which efficiently utilizes

\*通讯作者。

this kind of distribution in an environment to increase the accuracy of classification. Firstly, we train the deep learning model on the training set. Secondly, the user can input the distribution of each label in the corresponding environment on the experience of himself or advice from others. Thirdly, we can get the probability of labels on each sample by this trained model. Finally, the distribution information selectively cooperates with the probability of labels on testing samples. We select six popular deep learning models on three real datasets for the evaluation. The experimental results show that our method can increase the accuracy of classification tasks, which is obviously higher than state-of-art methods.

## Keywords

Distribution of Labels, Probability of Labels, User Guidance

Copyright © 2022 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

## 1. 引言

现如今, 已经证明了深度学习模型在许多应用中是有效的[1] [2] [3] [4] [5]。通常, 基于深度学习的分类模型的性能取决于捕获特征的能力[6] [7] [8]。在测试样本上使用深度学习模型时, 会输出每个标签的概率, 然后选择具有概率最大值的标签作为最终结果。

有时, 当样本具有属于不同标签的特征时, 错误结果的概率可能高于正确结果的概率。为了捕获更多特征以获得更高的精度, 模型的结构可能会变得更大, 而这会受到许多因素的限制, 例如消失梯度问题[9] [10] [11]。因此, 应该设计另一种方法来提高深度学习模型在实际应用中的性能。

与深度学习模型不同, 人类不仅可以通过目标的特征, 而且还可以通过目标的分布等因素对目标进行分类。如图 1 展示了这种例子, 可以很容易地发现, 该样本中的对象同时具有马和鹿的特征, 这可能会增加分类的难度, 但如果可以知道在这种环境中没有鹿(或马)时, 就可以很容易地对其进行分类。



Figure 1. The introduction of problem. A sample that may be deer or horse

图 1. 问题示例: 一个可能是鹿或马的样本

这种信息可以通过生活经验或他人的建议获得, 这是人类的一种特殊能力, 可以用来提高环境中分

类的准确性。

本文建立了基于用户指导的深度学习分类系统 DLC-UG, 以提高分类的准确性, 本文的贡献可以总结如下: 1) 系统使用户能够引导深度学习模型在环境中提高分类准确性。通过系统, 用户体验可以很容易地应用到深度学习模型中, 以获得更高的精度。2) 系统提高了深度学习模型在新环境中的鲁棒性。标签的分布在不同的环境中可能会有所不同, 在这个系统中, 用户只需要在相应的环境中更新标签的分布, 这比模型迁移或重新训练的成本更低。

论文在 CIFAR-10 [12] [13] [14]、CIFAR-100 [15] [16] [17] 和 Mini-ImageNet [18] [19] [20] 的样本上验证了本文方法和现有方法, 所有这些评估都证明了本文系统的有效性。

论文分为以下几个部分: 第一节介绍了背景和本文的贡献, 第二节介绍了现有的方法及其存在的问题, 第三节介绍了本文系统和相关分析, 实验安排在第四节, 第五节做出了结论和未来的工作规划。

## 2. 相关工作

**VoVNet-57:** VoVNet [21] 的组成首先是一个 stem 块, 由 3 个卷积层组成, 之后是 4 级 OSA 模块, 输出步幅 32。OSA 模块用于最小化 MAC, MAC 由 5 个具有相同输入/输出通道的卷积层组成。VoVNet-57 在第四和第五阶段包含更多的 OSA 模块, 因此模型更大, 常用于目标检测。

**VGG16:** 是用于图像识别的 VGG 模型[22]的变体。图像通过一堆卷积层, 滤波器使用一个非常小的  $3 \times 3$  的感受野, 卷积步长固定为 1 像素, 五个最大化池层位于卷积层之后, 步幅为 2, 三个完全连接层紧跟一组卷积层之后, 最后一层是 softmax 层, 所有隐藏层都配备了(ReLU)非线性函数。

**ResNeSt:** 特点是由 Split-Attention 块构成[23], 它有四个版本, 从 ResNeSt50 到 ResNeSt269, 结构变得越来越复杂, 当训练样本越来越多时, 可以获得更高的精度。由于测试样本的大小和计算资源的耗费, 本文实验使用了 ResNeSt50。

**RepVGG:** 是在 VGG 网络的基础上进行了改进[24]。主要改进包括: 1) 在 VGG 网络的 Block 块中增加了 Identity 和残差分支, 相当于将 ResNet 网络的精华应用于 VGG 网络; 2) 在模型推理阶段, 通过 OP 融合策略将所有网络层转换为  $3 \times 3$  卷积层, 便于部署和加速模型。

**ResNet:** 在 ILSVRC 2015 分类任务中获得了第一名[25]。它允许将原始输入信息直接转移到输出, 从而简化了神经网络学习的目标和难度。使用 ResNet 主干模型可以在多个任务中实现最先进的性能, 即: 图像分类、对象检测、实例分割和语义分割等。

**DenseNet:** 是一个具有紧密连接的卷积神经网络[26]。在这个网络中, 任意两层之间都有一个直接连接, 也就是说, 每一层的输入是之前所有层的输出的并集, 由该层学习的特征图将直接传输到所有后续层作为输入。

这些模型在现实中都得到了广泛的应用, 这也证明了它们的有效性。为了提高模型的准确性, 必须增加训练样本的数量, 这在现实应用中是一项艰巨的工作, 此外, 模型的结构必须足够大, 并使用一些特殊的调整技术才能实现, 而人类在环境中对分布标签(样本的种类)的经验可以用来提高准确率。

## 3. 系统框架

首先, 先做一些定义。令  $S_n$  为样本,  $L_k$  为样本的标签,  $G_n$  为  $S_n$  的真实标签[27] [28], 可以得到  $G_n \in \{L_k\}$ 。标签可用于计算, 通常设置为数字[29] [30], 例如, 如果有 10 个样本需要分类, 则标签的范围为 0 到 9, 同时,  $M$  为一个经过训练的模型。

### 3.1. 整体框架介绍

如图 2 介绍了系统的整体框架, 本文将其命名为 DLC-UG (基于用户指导的深度学习分类系统)。首

先, 系统在训练集上训练深度学习模型, 然后在输入样本上运行训练模型, 得到相应的输出, 即标签的概率, 接着, 用户可以通过输入环境中标签的分布来引导系统, 对于一个测试集, 用户只需输入一次标签的分布, 最后, 系统在用户输入和标签概率之间进行选择性的结合, 以提高准确性, 当有验证集时, 系统还可以进一步的优化。

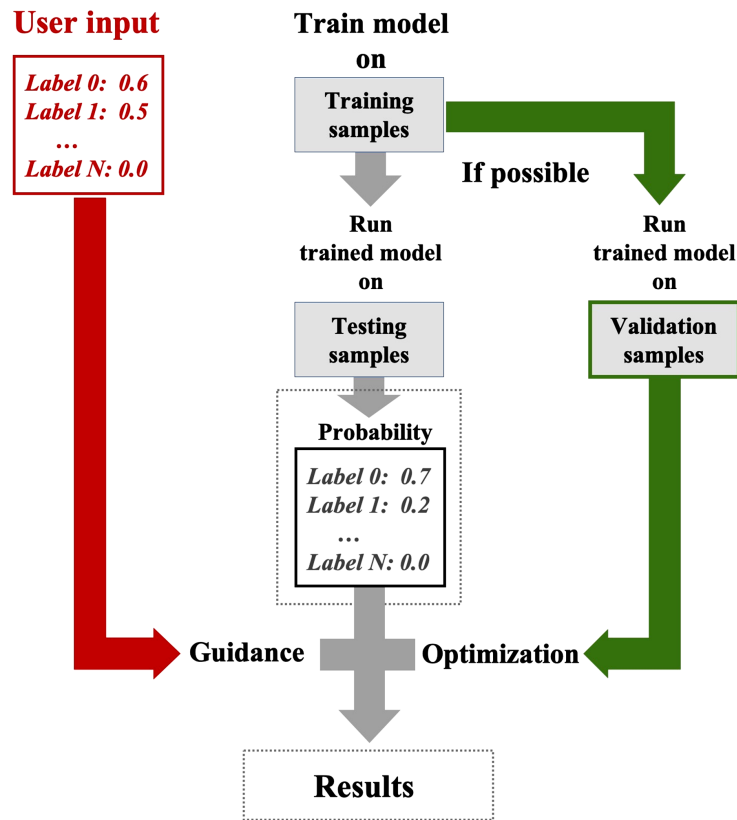


Figure 2. Overall system framework  
图 2. 系统整体框架

### 3.2. 根据标签分布进行用户指导

在本文中, 用户通过输入标签的分布来引导系统, 因此, 用户输入和标签的实际分布之间可能存在误差, 标签的用户输入分布可以定义如下:

$$\tilde{P}(L_k) = P(L_k) + \varepsilon \quad (1)$$

其中,  $\tilde{P}(L_k)$  是用户输入分布,  $P(L_k)$  是真实分布,  $\varepsilon$  代表这两种分布之间的误差, 在一般情况下, 正确预测标签的分布对用户来说是很艰难的, 即  $\varepsilon \neq 0$ 。

### 3.3. 训练模型得出概率

令  $P(M(S_n) = L_k)$  是训练模型  $M$  在样本  $S_n$  上得到标签  $L_k$  的概率, 通常, 一个经过训练的深度学习模型通过以下公式选择最可能的结果:

$$L_x = \arg \max_{L_k} P(M(S_n) = L_k) \quad (2)$$

正如图 3 的例子所示:

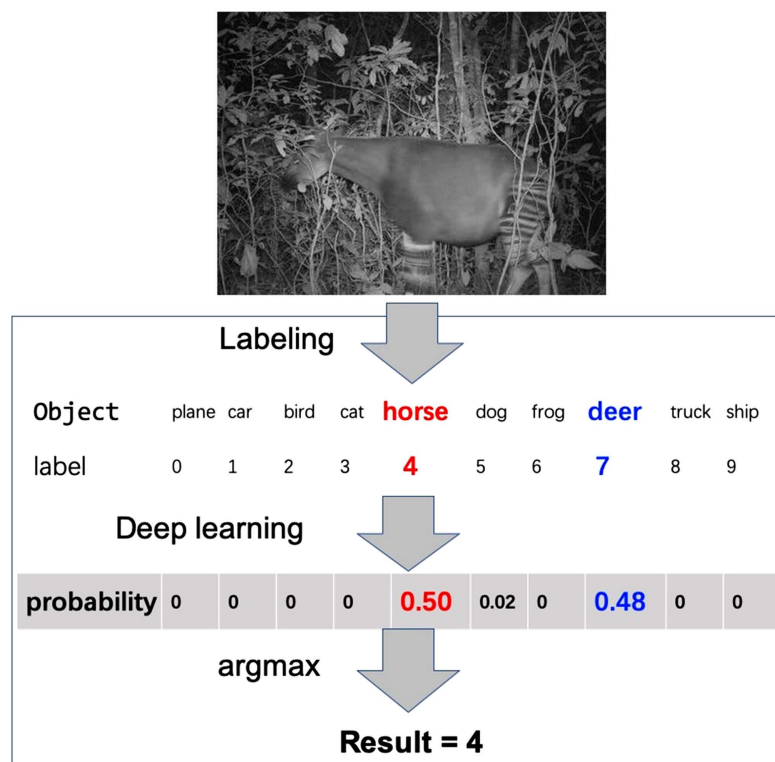


Figure 3. Process example of predicting results through tag probability  
图 3. 通过标签概率预测结果的过程示例

### 3.4. 通过后验分布优化

正如上面介绍的, 用户输入是不完全准确的, 当存在验证集时, 可以使用验证集上的后验分布进一步优化输入分布。首先, 定义  $\hat{P}(L_k)$  是标签  $L_k$  在验证集上的后验分布, 接着使用  $\hat{P}(L_k)$  按照下面的公式来优化输入分布  $\tilde{P}(L_k)$ :

$$f(\tilde{P}(L_k), \hat{P}(L_k)) = \tilde{P}(L_k) \times \varphi + \hat{P}(L_k) \times (1 - \varphi) \quad (3)$$

其中,  $\varphi$  表示权重, 且满足  $0 \leq \varphi \leq 1$ , 通过选择  $\varphi$  的值, 可以使经过训练的模型在验证集上达到最高的精度的方法, 本文将这种优化方法命名为 DLC-UG-opt。

### 3.5. 选择性结合

本小节将在用户输入分布和标签概率之间进行选择性结合, 需要建立了两条选择规则, 如下所示:

$$\begin{aligned} \text{Select}_1 &: \{L_x \mid P(M(S_n) = L_x) < \delta\} \\ \text{Select}_2 &: \{L_k \mid P(M(S_n) = L_k) > \gamma\} \end{aligned} \quad (4)$$

其中,  $L_x = \arg \max_{L_k} P(M(S_n) = L_k)$ , 在所有的最终结果中,  $\text{Select}_1$  重新考虑了那些概率小于  $\delta$  的结果, 然后注意这些结果的相应样本, 在每个样本上, 可以得到模型输出的标签的概率, 然后选择概率大于  $\gamma$  的标签作为最终结果的潜在集合。 $\delta$  是决定是否重新考虑结果的阈值,  $\gamma$  这个参数意味着只能选择一部分标签作为可能的最终结果, 这是为了避免标签概率  $P(M(S_n) = L_k) \approx 0$  通过结合被选为了最终的结果。

可以使用图 3 中的例子来解释这个过程, 在这个图中, 经过训练的模型输出  $L_x = L_4$  作为样本的最终



结果, 因为该标签在所有标签中的概率最大。如果提前设定  $\delta = 0.6$ , 可以得到  $P(M(S_n) = L_4) = 0.5 < \delta$ , 然后根据  $\text{Select}_1$  重新考虑这个样本的结果,  $\text{Select}_2$  选择概率大于预设值  $\gamma = 0.1$  的那些标签, 这样, 系统就会选择并重新考虑  $L_4$ 、 $L_7$  作为最终结果的潜在集合。

### 3.6. 本文方法

接下来, 可以根据论文的系统执行以下方法: 对于最终结果  $L_x \in \text{Select}_1$ , 可以通过以下公式重新考虑输出:

$$F(S_n, L_k) = P(M(S_n) = L_k) + \omega \times f(\tilde{P}(L_k), \hat{P}(L_k)) \tag{5}$$

$$L_x = \arg \max_{L_k \in \text{Select}_2} F(S_n, L_k)$$

当使用这种方法时, 我们可以设置参数的默认值如  $\omega$ , 用户输入分布的影响由参数  $\omega$  控制, 当存在验证集时, 我们还可以优化该参数。

下面是论文系统比现有系统性能更好的原因。第一个原因: 样本  $S_n$  上可能存在以下关系:

$$P(M(S_n) = L_q \neq G_n) > P(M(S_n) = L_k = G_n) \tag{6}$$

这意味着经过训练的模型预测出了错误的结果, 在这种情况下, 有  $P(M(S_n) = L_k = G_n) \neq 0$ 。特别的, 当  $P(M(S_n) = L_q) < \delta$  时, 有  $P(M(S_n) = L_k = G_n) \gg 0$ , 这表示正确结果可能是其他的。例如, 在图 4 中,  $P(M(S_n) = L_4 = G_n) = 0.50$ , 同时  $P(M(S_n) = L_7 = G_n) = 0.48$ , 在这种情况下, 如果存在以下关系:

$$\tilde{P}(L_7) = 0.67 \gg \tilde{P}(L_4) = 0.01 \tag{7}$$

就可以使用用户输入来获得最终结果。例如, 我们可以很容易地在图 4 中选择  $L_7$  是鹿。第二个原因, 在深度学习模型中, 捕获的特征在分类中起着重要作用, 捕获特征的数量取决于层的结构[31] [32], 深度学习的训练过程是选择能够代表样本的特征, 然后, 获得标签的概率也是基于这些特征。所以, 当存在以下关系时, 标签  $L_k$  比标签  $L_q$  更有可能是正确的结果:

$$P(M(S_n) = L_k) > \gamma > P(M(S_n) = L_q) \tag{8}$$

因此, 在重新考虑结果时, 选择具有高概率的标签是合理的。

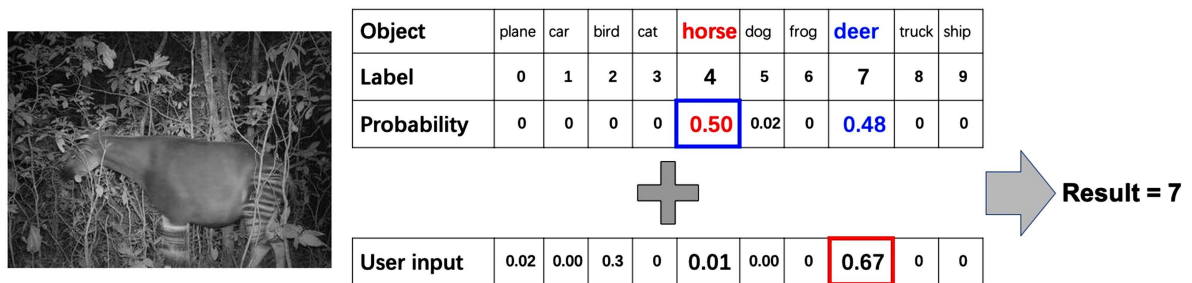


Figure 4. The result of the model is 4 (horse), and the correct result 7 can be obtained through the distribution of labels  
 图 4. 模型得出的结果是 4 (马), 而通过标签的分布, 可以得到正确结果 7

## 4. 实验结果

论文在一些真实数据集上评估了所提出的方法和现有的方法。首先, 在训练集上对已有的深度学习模型进行训练, 生成训练模型, 在默认设置下训练所有深度学习模型(不改变结构或调整超参数), 在训练这些模型时, 将迭代次数 epoch [33] [34] 设置为 10。接着, 根据样本分布从数据集中选取测试样本, 以

模拟环境中的真实分布, 然后在这些样本上评估本文的方法和现有的方法, 当随机化这些参数时, 实验评估 1000 次并计算平均值。

#### 4.1. 在 CIFAR-10 上的评估结果

CIFAR-10 [12] [13] [14] 有 50,000 个训练样本和 10,000 个测试样本, 属于 10 个标签。每个样本都是一个带有三个通道的 RGB 图像: 红色、绿色和蓝色。实验使用训练集中的 50,000 个训练样本来训练模型, 另外还有 10000 个测试样本, 根据不同的样本分布从这 10,000 个样本中选择样本, 以模拟环境中的真实分布。

实验使用三种样本分布来评估这些方法, Zero20 表示 20% 的标签上没有样本, 然后以同样的方式定义 Zero40 (有 40% 的标签是零样本) 和 Zero80 (有 80% 的标签是零样本), 要分配零样本的标签是随机选择的。图 5 显示了这些分布的示例, 在这些情况下, 测试集的样本数量少于 10,000 个。例如, Zero20 案例中大约有 8000 个样本。

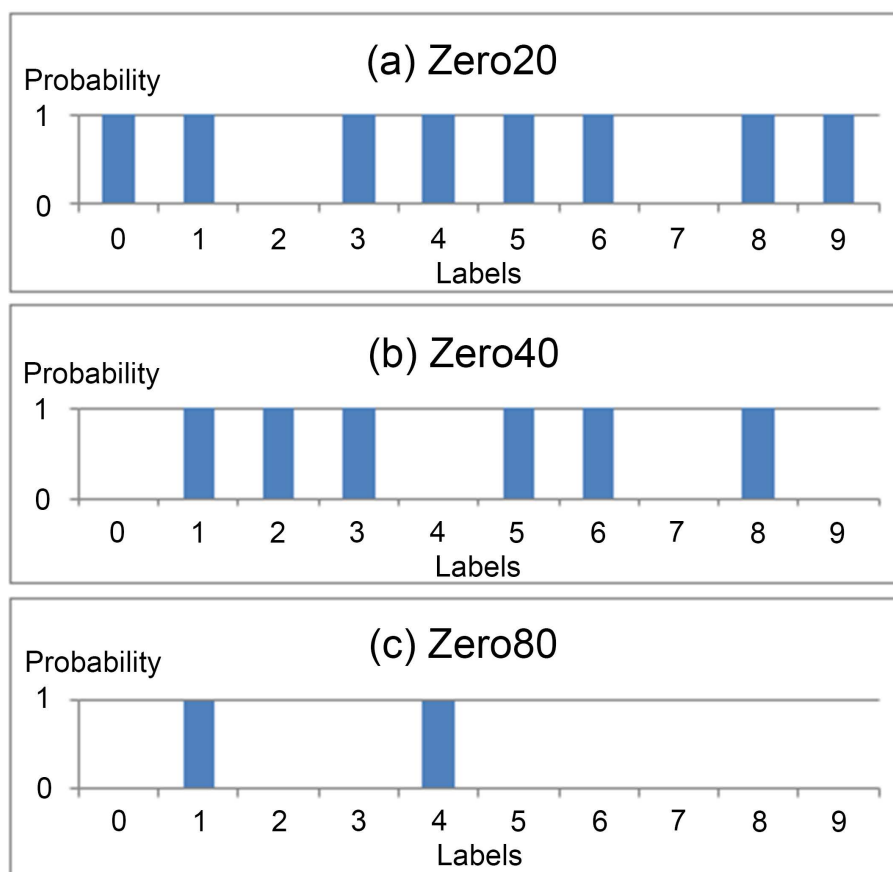


Figure 5. Examples of Zero20, Zero40, and Zero80 on CIFAR-10

图 5. 在 CIFAR-10 上的 Zero20、Zero40 和 Zero80 示例

在这些情况下, 假设用户输入是正确的, 这意味着  $\varepsilon = 0$  (公式(1)所述)。如表 1 所示, 与现有方法相比, 本文的方法可以将准确度至少提高 1.29%, 最多提高 24.34%。

#### 4.2. 在 CIFAR-100 上的评估结果

该数据集与 CIFAR-10 类似, 只是它有 100 个类, 每个类包含 600 个图像: 包括 500 个训练图像和

100 个测试图像[15] [16] [17]。实验使用训练集中的 50,000 个训练样本来训练模型, 另外还有 10,000 个样本。根据不同的样本分布从这 10,000 个样本中选择样本, 以模拟环境中的真实分布。

在这些情况下, 假设用户输入是正确的, 即  $\varepsilon = 0$ , 分布 Zero20、Zero40 和 Zero80 的含义与 4.1 小节描述相同。如表 2 所示, 与现有方法相比, 论文提出的方法可以将准确度至少提高 2.55%, 最多提高 23.75%。

**Table 1.** Experimental results on CIFAR-10

**表 1.** CIFAR-10 上的实验结果

分布	By DLC-UG					
	Zero20	Zero40	Zero80	Zero20	Zero40	Zero80
VoVNet-57 [21]	81.19%	81.74%	81.44%	84.29%	86.99%	96.16%
VGG16 [22]	73.54%	73.22%	73.46%	75.53%	79.06%	92.68%
ResNeSt50 [23]	90.85%	90.22%	90.32%	92.14%	93.79%	98.06%
RepVGG [24]	80.85%	80.73%	80.20%	83.46%	86.42%	95.25%
ResNet [25]	68.81%	68.87%	68.31%	72.72%	77.80%	92.65%
DenseNet [26]	75.62%	75.64%	75.63%	78.52%	82.39%	93.95%

**Table 2.** Experimental results on CIFAR-100

**表 2.** CIFAR-100 上的实验结果

分布	By DLC-UG					
	Zero20	Zero40	Zero80	Zero20	Zero40	Zero80
VoVNet-57 [21]	64.25%	64.37%	64.84%	66.85%	70.47%	82.53%
VGG16 [22]	45.96%	46.14%	46.11%	49.12%	53.25%	68.75%
ResNeSt50 [23]	66.10%	66.57%	66.68%	68.65%	72.33%	83.93%
RepVGG [24]	63.29%	63.89%	63.80%	66.20%	70.25%	82.48%
ResNet [25]	41.77%	42.49%	42.48%	44.79%	49.55%	66.23%
DenseNet [26]	54.37%	54.17%	54.27%	57.52%	61.44%	77.03%

### 4.3. 在 Mini-ImageNet 上的评估结果

Mini-ImageNet [18] [19] [20]数据集常用于小样本学习评估。由于使用了 ImageNet 中的图像, 它类型比较复杂, 但与在完整的 ImageNet 数据集上运行相比, 需要的资源和基础设施更少。数据集共有 100 个类别, 每个类别有 600 个样本和 8484 幅彩色图像。实验使用训练集中的 48,000 个训练样本来训练模型, 另外还有 12,000 个测试样本, 根据不同的样本分布从这 12,000 个样本中选择样本, 以模拟环境中的真实分布。

在这些情况下, 我们假设用户输入是正确的, 这意味着  $\varepsilon = 0$ 。分布 Zero20、Zero40 和 Zero80 的含义与 4.1 小节描述相同。如表 3 所示, 与现有方法相比, 论文的方法可以将准确度至少提高 1.86%, 最多提高 22.60%。

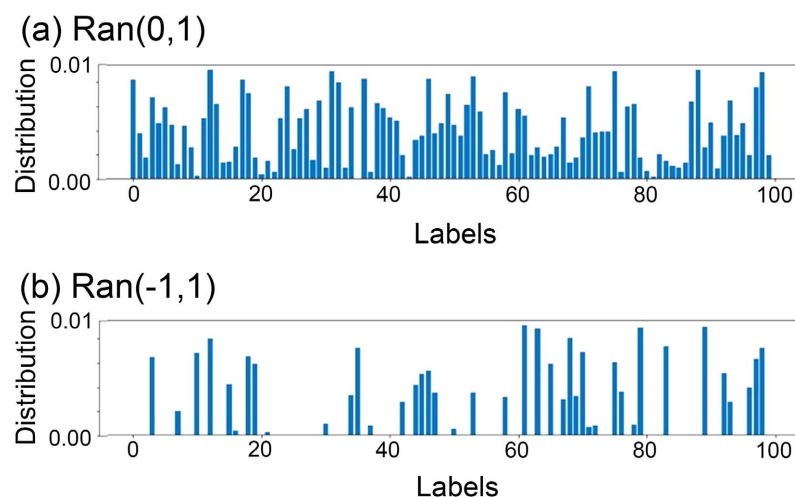


**Table 3.** Experimental results on Mini-ImageNet  
**表 3.** Mini-ImageNet 上的实验结果

分布	By DLC-UG					
	Zero20	Zero40	Zero80	Zero20	Zero40	Zero80
VoVNet-57 [21]	72.93%	72.83%	72.85%	75.22%	78.08%	86.75%
VGG16 [22]	43.28%	42.95%	44.27%	46.14%	50.32%	66.87%
ResNeSt50 [23]	69.86%	70.81%	70.65%	72.10%	75.22%	85.71%
RepVGG [24]	70.50%	70.52%	71.00%	72.78%	75.65%	86.32%
ResNet [25]	41.15%	41.59%	41.48%	43.01%	48.24%	63.08%
DenseNet [26]	51.19%	51.29%	51.11%	53.90%	58.37%	72.07%

#### 4.4. 两个数据集上的随机情况

在本小节中, 实验随机将示例分布分配给 CIFAR-100 和 Mini-ImageNet, 具体来说, 实验随机选择标签并设置随机分布来评估这些方法。然后, 本文将提出的方法与这些数据集上现有的方法进行了比较, 如图 6 所示,  $\text{Ran}(\cdot)$  是将随机值乘以分布的函数, 如果随机值小于 0, 则用 0 代替这个值, 然后通过这个函数生成样本的分布。例如, 如果对对象标签的原始样本数为 100, 且  $\text{Ran}(0,1) = 0.9$ , 则有大约 90 个用于评估的该标签样本, 图 6 显示了  $\text{Ran}(0,1)$  和  $\text{Ran}(-1,1)$  的示例。



**Figure 6.** The example random distributions, (a)  $\text{Ran}(0,1)$ ; (b)  $\text{Ran}(-1,1)$

**图 6.** 随机分布示例, (a)  $\text{Ran}(0,1)$ ; (b)  $\text{Ran}(-1,1)$

如表 4 所示, 论文的方法比现有的最佳方法平均提高了约 0.54% (在  $\text{Ran}(0,1)$  情况下)和 8.61% (在  $\text{Ran}(-1,1)$  情况下)的平均准确率。

#### 4.5. 在两个数据集上均错误的情况

在本小节中, 实验假设用户输入不正确, 即  $\varepsilon \neq 0$ , 定义  $\varepsilon = (\cdot)$  意味着在分布中加入了一个随机误差, 例如, 当标签的分布如图 6 所示介于 0 和 0.01 之间时,  $\varepsilon = (-0.2, 0.2)$  意味着向分布中添加了一个随机误

差  $(-0.2 \times 0.01, 0.2 \times 0.01)$ 。当使用本文的优化方法 DLC-UG-opt 时, 随机将验证集中的样本数设置在 200 到 500 之间。

**Table 4.** Random distribution results on CIFAR-100 and Mini-ImageNet  
**表 4.** CIFAR-100 和 Mini-ImageNet 上的随机分布结果

模型	Cifar-100				Mini-ImageNet			
	By DLC-UG				By DLC-UG			
分布	Ran(0, 1)	Ran(-1, 1)	Ran(0, 1)	Ran(-1, 1)	Ran(0, 1)	Ran(-1, 1)	Ran(0, 1)	Ran(-1, 1)
VoVNet-57 [21]	64.32%	64.05%	64.85%	72.68%	72.61%	72.56%	73.03%	79.38%
VGG16 [22]	45.70%	46.12%	46.45%	56.59%	43.67%	43.52%	44.39%	53.32%
ResNeSt50 [23]	66.35%	66.73%	66.91%	74.69%	69.82%	69.87%	70.21%	77.67%
RepVGG [24]	63.38%	63.94%	63.87%	72.66%	70.76%	70.80%	71.17%	77.54%
ResNet [25]	41.47%	41.25%	42.08%	51.07%	40.39%	41.50%	40.93%	50.83%
DenseNet [26]	54.89%	54.79%	55.71%	65.22%	50.80%	51.04%	51.45%	61.60%

如表 5 所示, 与现有方法相比, DLC-UG 方法平均可提高约 7.80% (当  $\varepsilon = (-0.0, 0.0)$  时) 的平均精度, 3.30% (当  $\varepsilon = (-0.2, 0.2)$  时) 和 1.53% (当  $\varepsilon = (0.4, 0.4)$  时) 的平均精度。本文的方法 DLC-UG-opt 比现有方法平均提高了 8.10% ( $\varepsilon = (-0.0, 0.0)$ )、7.03% ( $\varepsilon = (0.2, 0.2)$ ) 和 5.92% ( $\varepsilon = (-0.4, 0.4)$ )。这些结果表明, 当标签的输入分布存在误差时, 论文提出的方法也可以具有更高的精度。与本文的方法 DLC-UG 相比, 优化方法 DLC-UG-opt 可以减少误差的影响, 从而获得更高的精度。

**Table 5.** Experimental results on CIFAR-100 and Mini-ImageNet when user input error  
**表 5.** 用户输入错误时在 CIFAR-100 和 Mini-ImageNet 上的实验结果

模型	分布: Ran(-1, 1)	Cifar-100		Mini-ImageNet	
		VoVNet-57 [21]	ResNeSt50 [23]	VoVNet-57 [21]	ResNeSt50 [23]
现有方法		64.05%	66.73%	72.56%	69.87%
DLC-UG 的方法	$\varepsilon = (-0.0, 0.0)$	72.68%	74.69%	79.38%	77.67%
	$\varepsilon = (-0.2, 0.2)$	67.36%	70.10%	75.78%	73.15%
	$\varepsilon = (-0.4, 0.4)$	66.04%	68.49%	74.42%	70.38%
DLC-UG-opt 的方法	$\varepsilon = (-0.0, 0.0)$	73.08%	74.99%	79.58%	77.97%
	$\varepsilon = (-0.2, 0.2)$	72.08%	74.23%	78.78%	76.24%
	$\varepsilon = (-0.4, 0.4)$	71.72%	72.12%	78.07%	74.97%

#### 4.6. 结果分析

论文已经在不同分布的真实数据集上对提出的方法进行了评估, 这些模型的精确度从低(<50%)到高(>90%)的结果都证明了本文的系统在这些情况下的有效性。在某些情况下, 也可以通过另一种方式获得分布, 例如, 当地居民的建议类似于“这个地区没有马”, 可以用来从候选集中消除错误的标签。本文的系统可以有效地利用用户的体验来提高准确性, 使用这一方法的另一个优点是, 应用中不需要对模型

进行重新训练,也不需要将模型迁移到各种环境中来加强鲁棒性,本文中的系统实现了这种对基于深度学习的分类的鲁棒性。

## 5. 总结

本文提出了一个新的系统,它可以提高基于深度学习模型分类任务的准确性,这个系统使用模型输出的样本的概率,使该概率有条件地与用户输入的标签分布相结合,以获得更高的精度,所提出的系统可以有效提高环境中深度学习分类模型的鲁棒性。在未来的工作中,我们将研究标签的概率和分布之间的深度结合,例如,可以使用多个模型的概率[35]和用户输入的概率相结合,这能够包含对象特征的更多信息,这些信息可以正确表示经过训练的模型从样本中捕获到的内容。

## 参考文献

- [1] Kim, M., Han, D. and Rhee, J.-K.K. (2021) Multiview Variational Deep Learning with Application to Practical Indoor Localization. *IEEE Internet of Things Journal*, **8**, 12375-12383. <https://doi.org/10.1109/JIOT.2021.3063512>
- [2] Li, J.L., Luo, G.Y., Cheng, N., Yuan, Q., Wu, Z.H., Gao, S. and Liu, Z.H. (2019) An End-to-End Load Balancer Based on Deep Learning for Vehicular Network Traffic Control. *IEEE Internet of Things Journal*, **6**, 953-966. <https://doi.org/10.1109/JIOT.2018.2866435>
- [3] Wu, Y.F., Mendis, G.J. and Wei, J. (2021) DDLPF: A Practical Decentralized Deep Learning Paradigm for Internet-of-Things Applications. *IEEE Internet of Things Journal*, **8**, 9740-9752. <https://doi.org/10.1109/JIOT.2020.3033482>
- [4] Jin, G., Liu, F., Wu, H. and Song, Q. (2020) Deep Learning-Based Framework for Expansion, Recognition and Classification of Underwater Acoustic Signal. *Journal of Experimental & Theoretical Artificial Intelligence*, **32**, 205-218. <https://doi.org/10.1080/0952813X.2019.1647560>
- [5] Jain, D.K., et al. (2020) Multi Angle Optimal Pattern-Based Deep Learning for Automatic Facial Expression Recognition. *Pattern Recognition Letters*, **139**, 157-165. <https://doi.org/10.1016/j.patrec.2017.06.025>
- [6] Bakhshipour, A. (2021) Cascading Feature Filtering and Boosting Algorithm for Plant Type Classification Based on Image Features. *IEEE Access*, **9**, 82021-82030. <https://doi.org/10.1109/ACCESS.2021.3086269>
- [7] Derrouz, H., El Bouziady, A., Ait Abdelali, H., Thami, R.O.H., El Fkihi, S. and Bourzeix, F. (2019) Moroccan Video Intelligent Transport System: Vehicle Type Classification Based on Three-Dimensional and Two-Dimensional Features. *IEEE Access*, **7**, 72528-72537. <https://doi.org/10.1109/ACCESS.2019.2920740>
- [8] Liao, J.S. and Wang, L.G. (2020) Adaptive Hyperspectral Image Classification Based on the Fusion of Manifolds Filter and Spatial Correlation Features. *IEEE Access*, **8**, 90390-90409. <https://doi.org/10.1109/ACCESS.2020.2993864>
- [9] Ahn, S.M. (2016) Deep Learning Architectures and Applications. *Journal of Intelligence & Information Systems*, **22**, 127-142. <https://doi.org/10.13088/jiis.2016.22.2.127>
- [10] Sharma, K., Alsadoon, A., Prasad, P., Al-Dala'In, T. and Pham, D. (2020) A Novel Solution of Using Deep Learning for Left Ventricle Detection: Enhanced Feature Extraction. *Computer Methods and Programs in Biomedicine*, **197**, Article ID: 105751. <https://doi.org/10.1016/j.cmpb.2020.105751>
- [11] Dai, Z.W. and Heckel, R. (2019) Channel Normalization in Convolutional Neural Network Avoids Vanishing Gradients. CoRR abs/1907.09539.
- [12] Ho-Phuoc, T. (2018) CIFAR10 to Compare Visual Recognition Performance between Deep Neural Networks and Humans. CoRR abs/1811.07270.
- [13] Su, J., Vargas, D.V. and Kouichi, S. (2019) One Pixel Attack for Fooling Deep Neural Networks. *IEEE Transactions on Evolutionary Computation*, **23**, 828-841.
- [14] Wu, D.X., Xu, J. and Liu, H. (2020) Analysis of the Influence of Stylized-CIFAR10 Dataset on ResNet. *ML4CS 2020*, Guangzhou, 8-10 October 2020, 416-426. [https://doi.org/10.1007/978-3-030-62460-6\\_37](https://doi.org/10.1007/978-3-030-62460-6_37)
- [15] Termritthikun, C., et al. (2020) An Improved Residual Network Model for Image Recognition Using a Combination of Snapshot Ensembles and the Cutout Technique. *Multimedia Tools and Applications*, **79**, 1475-1495. <https://doi.org/10.1007/s11042-019-08332-3>
- [16] Chen, J. and Liu, Z. (2020) Mask Dynamic Routing to Combined Model of Deep Capsule Network and U-Net. *IEEE Transactions on Neural Networks and Learning Systems*, **31**, 2653-2664. <https://doi.org/10.1109/TNNLS.2020.2984686>

- [17] Sharma, N., Jain, V. and Mishra, A. (2018) An Analysis of Convolutional Neural Networks for Image Classification. *Procedia Computer Science*, **132**, 377-384. <https://doi.org/10.1016/j.procs.2018.05.198>
- [18] Wang, P., Cheng, J., Hao, F., Wang, L. and Feng, W. (2020) Embedded Adaptive Cross-Modulation Neural Network for Few-Shot Learning. *Neural Computing and Applications*, **32**, 5505-5515. <https://doi.org/10.1007/s00521-019-04605-y>
- [19] Xiang, W., et al. (2020) FMixCutMatch for Semi-Supervised Deep Learning. *Neural Networks*, **133**, 166-176. <https://doi.org/10.1016/j.neunet.2020.10.018>
- [20] Zhang, L.L., et al. (2019) Scheduled Sampling for One-Shot Learning via Matching Network. *Pattern Recognition*, **96**, Article ID: 106962. <https://doi.org/10.1016/j.patcog.2019.07.007>
- [21] Lee, Y., Hwang, J.W., Lee, S., Bae, Y. and Park, J. (2019) An Energy and GPU-Computation Efficient Backbone Network for Real-Time Object Detection. 2019 *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Long Beach, 16-17 June 2019, 752-760. <https://doi.org/10.1109/CVPRW.2019.00103>
- [22] Qu, Z., Mei, J., Liu, L. and Zhou, D.-Y. (2020) Crack Detection of Concrete Pavement with Cross-Entropy Loss Function and Improved VGG16 Network Model. *IEEE Access*, **8**, 54564-54573. <https://doi.org/10.1109/ACCESS.2020.2981561>
- [23] Wang, S., Liu, Y., Qing, Y., Wang, C., Lan, T. and Yao, R. (2020) Detection of Insulator Defects with Improved ResNeSt and Region Proposal Network. *IEEE Access*, **8**, 184841-184850. <https://doi.org/10.1109/ACCESS.2020.3029857>
- [24] Feng, X., Gao, X. and Luo, L. (2021) X-sdd: A New Benchmark for Hot Rolled Steel Strip Surface Defects Detection. *Symmetry*, **13**, 706. <https://doi.org/10.3390/sym13040706>
- [25] Zhao, S.F., Qiu, L.C., Qi, P. and Sun, Y. (2020) A Novel Image Classification Model Jointing Attention and ResNet for Scratch. 16th *International Wireless Communications and Mobile Computing Conference, IWCMC 2020*, Limassol, 15-19 June 2020, 1498-1503. <https://doi.org/10.1109/IWCMC48107.2020.9148279>
- [26] Zhang, Z., Liang, X., Xu, D., Xie, Y. and Cao, G. (2018) A Sparse-View CT Reconstruction Method Based on Combination of Densenet and Deconvolution. *IEEE Transactions on Medical Imaging*, **37**, 1407-1417. <https://doi.org/10.1109/TMI.2018.2823338>
- [27] Khodja, A., Zheng, Z.L., Mo, J.S.Z., Zhang, D.W. and Chen, L.Y. (2021) Rain to Rain: Learning Real Rain Removal without Ground Truth. *IEEE Access*, **9**, 57325-57337. <https://doi.org/10.1109/ACCESS.2021.3072687>
- [28] Zaatour, R., Bouzidi, S. and Zagrouba, E. (2020) Unsupervised Image-Adapted Local Fisher Discriminant Analysis to Reduce Hyperspectral Images without Ground Truth. *IEEE Transactions on Geoscience and Remote Sensing*, **58**, 7931-7941. <https://doi.org/10.1109/TGRS.2020.2985260>
- [29] Leo, L.M. and Reddy, T.K. (2020) Dental Caries Classification System Using Deep Learning Based Convolutional Neural Network. *Journal of Computational and Theoretical Nanoscience*, **17**, 4660-4665. <https://doi.org/10.1166/jctn.2020.9295>
- [30] Song, D., Vold, A., Madan, K. and Schilder, F. (2021) Multi-Label Legal Document Classification: A Deep Learning-Based Approach with Label-Attention and Domain-Specific Pre-Training. *Information Systems*, **106**, Article ID: 101718. <https://doi.org/10.1016/j.is.2021.101718>
- [31] Chen, Y.T., Liu, L.W., Tao, J.J., Chen, X., Xia, R.L., Zhang, Q., Xiong, J., Yang, K. and Xie, J.B. (2021) The Image Annotation Algorithm Using Convolutional Features from Intermediate Layer of Deep Learning. *Multimedia Tools and Applications*, **80**, 4237-4261. <https://doi.org/10.1007/s11042-020-09887-2>
- [32] Saraee, E., Jalal, M. and Betke, M. (2020) Visual Complexity Analysis Using Deep Intermediate-Layer Features. *Computer Vision and Image Understanding*, **195**, Article ID: 102949. <https://doi.org/10.1016/j.cviu.2020.102949>
- [33] Lim, G., Cheng, Y., Hsu, W. and Lee, M.L. (2016) Integrated Optic Disc and Cup Segmentation with Deep Learning. *IEEE International Conference on Tools with Artificial Intelligence*, San Jose, 6-8 November 2016, 162-169. <https://doi.org/10.1109/ICTAI.2015.36>
- [34] Song, W., Li, W., Hua, Z. and Zhu, F. (2021) A New Deep Auto-Encoder Using Multiscale Reconstruction Errors and Weight Update Correlation. *Information Sciences*, **559**, 130-152. <https://doi.org/10.1016/j.ins.2021.01.064>
- [35] Jin, G., Hu, Y., Jiao, Y., Wen, J. and Song, Q. (2021) Improving the Performance of Deep Learning Model-Based Classification by the Analysis of Local Probability. *Complexity*, **2021**, Article ID: 5534385. <https://doi.org/10.1155/2021/5534385>