# DINO笔记

> Emerging Properties in Self-Supervised Vision Transformers

## 问题与贡献

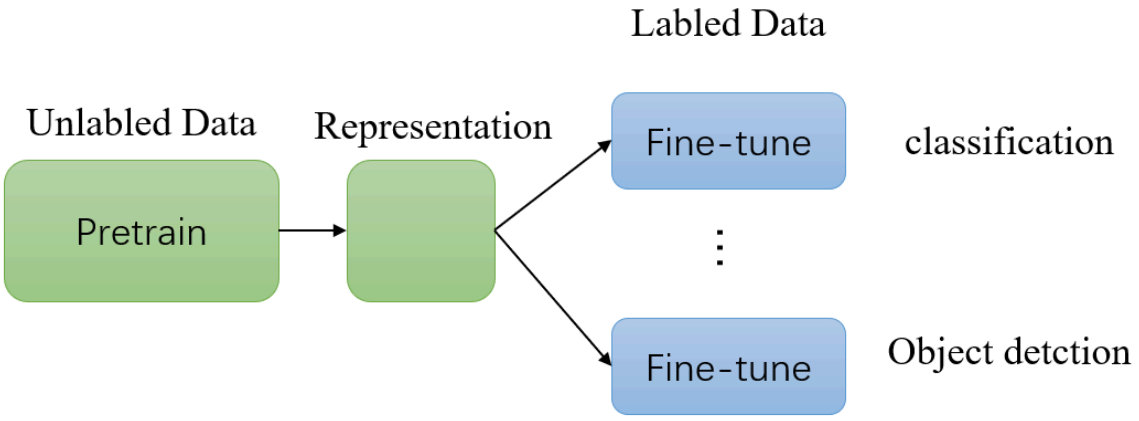
作者认为self-supervised learning自监督学习结合vision transformer（ViT），相对于卷积神经网络，能挖掘更多目标特性，更具象化的表达目标。本文有如下两点贡献：

- 提出了一种新的自监督学习方法，DINO（self-**di**stillation with **no** labels），结合ViT-Base在ImageNet的linear evaluation上达到了80.1%的Top-1指标；
- 自监督学习结合ViT得到的特征包含一张图像**更显式的语义分割信息**，远远超过带监督信息的ViTs或者卷积神经网络；如下图所示，自监督的ViTs可以自动学习目标类别的特征信息实现无监督的目标分割。


## 前置概念和理论

### 自监督学习

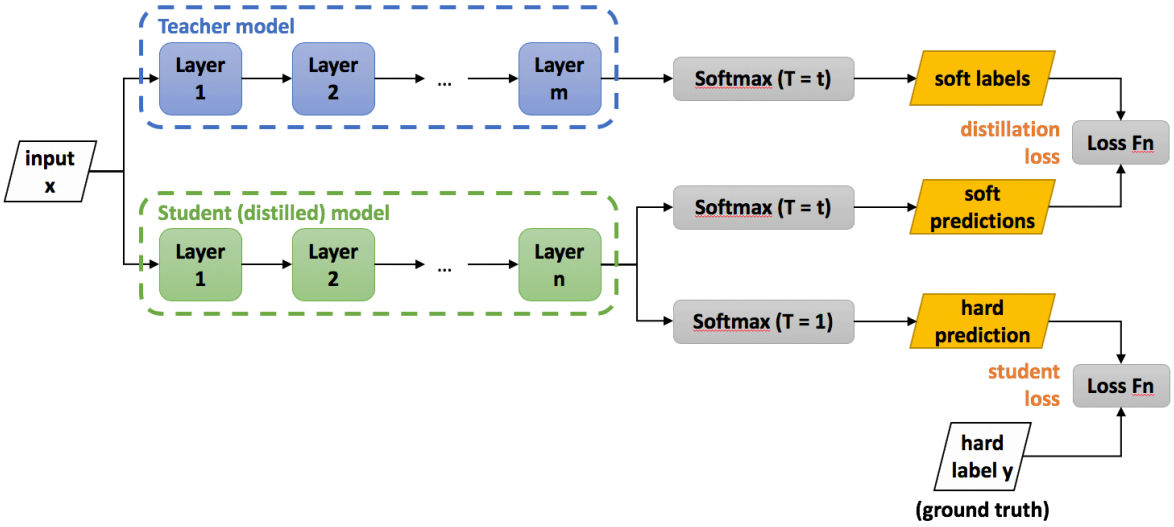
在机器学习中，根据是否有标签的训练数据可以分为有监督学习和无监督学习。其中，self-supervise learning，又称自监督学习，是无监督学习中的一种，主要是希望能够学习到一种通用的特征表达用于下游任务。
自监督学习模型训练分为两个阶段，在预训练阶段使用大量无标签的数据集进行训练，学习到通用的特征表达，然后根据下游任务的不同用带有标签的数据进行FineTune。




在自监督学习中，有两类主流的方法：基于Generative的方法和基于Contrastive的方法。基于Generative的方法主要关注重建误差，如一个句子中间盖住一个token，让模型去预测，计算预测结果与真实的token之间的误差更新模型。基于Contrastive方法不要求模型能重建原始输入，而是希望模型能够在特征空间上对不同的输入进行分辨。

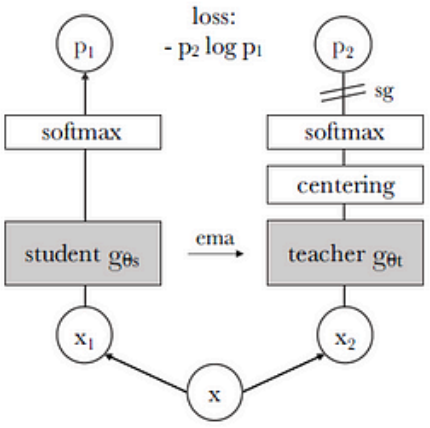
## 知识蒸馏

知识蒸馏是一种模型压缩方法，是一种基于"教师-学生网络思想"的训练方法。其核心思想是将已经训练好的模型包含的知识蒸馏提取到另一个模型里面去。
如下图所示，教师模型的知识可以通过不同的方式传递给学生模型，包括软标签、特征表示和模型输出的概率分布等。学生模型通过与教师模型的输出进行比较和对齐，以最小化它们之间的差异，从而学习到教师模型的知识。




# 模型、理论和方法

## DINO框架




**Algorithm 1** DINO PyTorch pseudocode w/o multi-crop.

```
# gs, gt: student and teacher networks
# C: center (K)
# tps, tpt: student and teacher temperatures
# l, m: network and center momentum rates
gt.params = gs.params
for x in loader: # load a minibatch x with n samples
    x1, x2 = augment(x), augment(x) # random views

    s1, s2 = gs(x1), gs(x2) # student output n-by-K
    t1, t2 = gt(x1), gt(x2) # teacher output n-by-K

    loss = H(t1, s2)/2 + H(t2, s1)/2
    loss.backward() # back-propagate

    # student, teacher and center updates
    update(gs) # SGD
    gt.params = l*gt.params + (1-l)*gs.params
    C = m*C + (1-m)*cat([t1, t2]).mean(dim=0)

def H(t, s):
    t = t.detach() # stop gradient
    s = softmax(s / tps, dim=1)
    t = softmax((t - C) / tpt, dim=1) # center + sharpen
    return - (t * log(s)).sum(dim=1).mean()
```

DINO本质上是一种自监督学习方法，其核心思想是通过在大规模的无标签数据集上进行对比学习，期待学习到更好的视觉通用表征。该方法采用自蒸馏的方式，即将一个student和teacher的输出进行比较，以学习出更好的表示。

## 主要特点

- 整个框架包含 teacher 和 student 模型，并且两者的网络结构相同，但参数 $\theta_t$ 和 $\theta_s$ 不同；
- 对于输入图像，生成一系列不同视角的 patches，其中包含两个全局视图 $x_1^g$ 和 $x_2^g$，以及一些局部视角的 crops。所有的 crops 都是会输入到 student 模型中，而只有全局视角的 crops 会送入到 teacher 中；

- teacher 和 student 模型的输出会使用 softmax 函数进行归一化，公式如下，得到 $K$ 维度的概率分布，分别用 $P_t$ 和 $P_s$ 表示：

$$P(x)^{(i)} = \frac{\exp(g\theta(x)/\tau)}{\sum_{k=1}^{K}\exp\left(g\theta(x)^{(k)}/\tau\right)}$$

- student 训练，更新参数，通过 SGD 优化函数来最小化目标，损失函数公式如下：

$$\min H(P_t(x), P_s(x))$$

# 对比学习

teacher和student的网络结构相同，采用知识蒸馏的方式来使得student的输出拟合teacher的输出，但是没有真实标签，两者之间是如何进行对比学习？主要采用的是如下集中方式。

## multi-crop learning

DINO中会对输入图像进行不同尺度的裁剪采样，这个也是自监督学习领域应用非常广泛的策略，裁剪后的图像可以分为两种：

- local views：局部视角，也称为small crops，指的是crop图像的面积小于原图的50%；
- global views：全局视角，也成为global crops，指的是crop图像的面积大于原图的50%；

在DINO中，student模型接收的是所有的crops图，而teacher模型接收的只是global views的裁剪图。
通过这种方式，**监督student模型学习到从局部到全局的响应。**
此外，为了增强网络的鲁棒性，采用了其他的数据增强手段，如：颜色扰动、高斯模糊和曝光增强。

## momentum teacher

teacher模型的权重参数更新不是基于反向传播更新的，而是通过指数移动平均法，将student模型学习到的权重参数更新给teacher。teacher模型权重的更新公式如下：

$$\theta_t \leftarrow \lambda\theta_t + (1-\lambda)\theta_s$$

其中，$\theta_t$ 和 $\theta_s$ 分别表示teacher和student模型的参数，$\lambda$ 在训练过程中采用余弦学习率衰减策略从0.996变化为1。

# centering and sharpening

在DINO中，采用centering和sharpening来防止model collapse模型坍塌。在自监督学习中，模型坍塌指的是网络学习过程中出现了多样性减少的现象。具体而言，当模型把多个输入数据映射到相同的特征表示时，只考虑了一部分数据的表示，而忽略了其他数据样本的特征，从而导致多样性缺失，对模型的鲁棒性会产生很大的负面影响。

DINO使用centering来避免一部分特征占据主导地位，具体是通过添加一个偏差项到teacher模型中，表示如下：

$$g_t(x) \leftarrow g_t(x) + c$$

其中的更新策略采用的指数移动平均，可以减少batch size对训练效果的影响，公式如下：

$$c \leftarrow mc + (1-m)\frac{1}{B}\sum_{i=1}^{B} g_{\theta_t}(x_i)$$

其中，$0 < m \leq 1$ 是一个比例项，$B$ 表示batch size。

sharpening操作是通过在softmax函数中加入一个temperature参数，来强制让模型将概率分布跟价锐化。使得小的差异会引起较大的变化，搭配centering使用，可以使得激活值不断变化。

# 实验与结论

## 训练细节

- 训练集：不包含标签的ImageNet数据集
- 优化函数：Adamw optimizer
- Batch size： 1024
- 学习率在最初的10epochs相对于base值呈线性增长，按照如下规则: lr=0.0005 * batchsize / 256，在warmup之后，使用余弦策略衰减学习率
- 权重衰减也是按照余弦衰减策略从0.04到0.4
- 在最开始的30epochs，温度系数设置为0.1，使用linear warm-up从0.04到0.07
- 使用BYOL数据增强：color jittering、Gaussian blur和solarization，multi-crop使用bicubic interpolation适应position embedding

## 对比实验

实验的评价方式有两种：linear 和k-NN 评估，其中linear评估指的是冻结预训练模型的权重，仅训练linear层；k-NN分类评估先使用预训练模型计算和保存数据集的特征，然后使用k-NN基于提取的特征对输入图像进行分类。

### 相同模型下比较

| Method | Arch. | Param. | im/s | Linear | $k$-NN |
|---|---|---|---|---|---|
| Supervised | RN50 | 23 | 1237 | 79.3 | 79.3 |
| SCLR [12] | RN50 | 23 | 1237 | 69.1 | 60.7 |
| MoCov2 [15] | RN50 | 23 | 1237 | 71.1 | 61.9 |
| InfoMin [67] | RN50 | 23 | 1237 | 73.0 | 65.3 |
| BarlowT [81] | RN50 | 23 | 1237 | 73.2 | 66.0 |
| OBoW [27] | RN50 | 23 | 1237 | 73.8 | 61.9 |
| BYOL [30] | RN50 | 23 | 1237 | 74.4 | 64.8 |
| DCv2 [10] | RN50 | 23 | 1237 | 75.2 | 67.1 |
| SwAV [10] | RN50 | 23 | 1237 | **75.3** | 65.7 |
| DINO | RN50 | 23 | 1237 | **75.3** | **67.5** |
| Supervised | ViT-S | 21 | 1007 | 79.8 | 79.8 |
| BYOL* [30] | ViT-S | 21 | 1007 | 71.4 | 66.6 |
| MoCov2* [15] | ViT-S | 21 | 1007 | 72.7 | 64.4 |
| SwAV* [10] | ViT-S | 21 | 1007 | 73.5 | 66.3 |
| DINO | ViT-S | 21 | 1007 | **77.0** | **74.5** |

以ResNet-50为基础模型，在相同设置下，DINO的效果都是最优的。在ViT框架下，DINO相对于BYOL，MoCov2和SwAV在linear和k-NN分类上分别至少提升了3.5%和7.9%。

### 不同框架比较

| *Comparison across architectures* | | | | | |
|---|---|---|---|---|---|
| SCLR [12] | RN50w4 | 375 | 117 | 76.8 | 69.3 |
| SwAV [10] | RN50w2 | 93 | 384 | 77.3 | 67.3 |
| BYOL [30] | RN50w2 | 93 | 384 | 77.4 | – |
| DINO | ViT-B/16 | 85 | 312 | 78.2 | 76.1 |
| SwAV [10] | RN50w5 | 586 | 76 | 78.5 | 67.1 |
| BYOL [30] | RN50w4 | 375 | 117 | 78.6 | – |
| BYOL [30] | RN200w2 | 250 | 123 | 79.6 | 73.9 |
| DINO | ViT-S/8 | 21 | 180 | 79.7 | **78.3** |
| SCLRv2 [13] | RN152w3+SK | 794 | 46 | 79.8 | 73.1 |
| DINO | ViT-B/8 | 85 | 63 | **80.1** | 77.4 |

ViT-B/8，指的是ViT-Base模型patch size为8×8，其linear分类取得80.1%的Top-1，77.4%的k-NN分类，相较于之前的方法参数量减少了至少10倍，运行速度提升了1.4倍。

## 消融实验

### 不同模块的影响

| | Method | Mom. | SK | MC | Loss | Pred. | $k$-NN | Lin. |
|---|---|---|---|---|---|---|---|---|
| 1 | DINO | ✓ | ✗ | ✓ | CE | ✗ | 72.8 | 76.1 |
| 2 | | ✗ | ✗ | ✓ | CE | ✗ | 0.1 | 0.1 |
| 3 | | ✓ | ✓ | ✓ | CE | ✗ | 72.2 | 76.0 |
| 4 | | ✓ | ✗ | ✗ | CE | ✗ | 67.9 | 72.5 |
| 5 | | ✓ | ✗ | ✓ | MSE | ✗ | 52.6 | 62.4 |
| 6 | | ✓ | ✗ | ✓ | CE | ✓ | 71.8 | 75.6 |
| 7 | BYOL | ✓ | ✗ | ✗ | MSE | ✓ | 66.6 | 71.4 |
| 8 | MoCov2 | ✓ | ✗ | ✗ | INCE | ✗ | 62.0 | 71.6 |
| 9 | SwAV | ✗ | ✓ | ✓ | CE | ✗ | 64.7 | 71.8 |

SK: Sinkhorn-Knopp, MC: Multi-Crop, Pred.: Predictor
CE: Cross-Entropy, MSE: Mean Square Error, INCE: InfoNCE

上图展示了通过添加不同模块到DINO结果的变化，同时也比较了这些模块在BYOL、MoCov2和SwAV上的影响。可以看到最好的组合是momentum encoder、multicrop augmentation和交叉熵损失函数。

### Patch size的影响

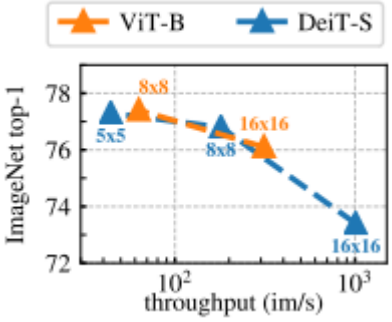
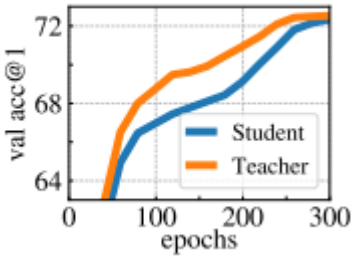



Figure 5: **Effect of Patch Size.** $k$-NN evaluation as a function of the throughputs for different input patch sizes with ViT-B and ViT-S. Models are trained for 300 epochs.

上图研究了ViT在不同patch size下模型性能的变化，可以看到patch size越小，性能越好，虽然参数量没有变换，但是推理速度变慢了。当使用5×5的patch size时，速率为44im/s，而8×8为180im/s。

### 不同teacher模型的影响




| Teacher | Top-1 |
|---|---|
| Student copy | 0.1 |
| Previous iter | 0.1 |
| Previous epoch | 66.6 |
| Momentum | 72.8 |

上图左边比较了student和teacher模型在训练过程中k-NN分类性能，可以看到teacher的效果要好于student。右边为teacher模型的不同参数更新策略，可以看到momentum的方式是最优的。

### 训练时batch size的影响

| bs | 128 | 256 | 512 | 1024 |
|---|---|---|---|---|
| top-1 | 57.9 | 59.1 | 59.6 | 59.9 |

Table 9: **Effect of batch sizes.** Top-1 with $k$-NN for models trained for 100 epochs without multi-crop.

上图研究了训练阶段不同batch size的影响，表明即使使用小的batch size也可以取得很好的效果。

## 思考

本文提出了一种自监督学习方法，DINO，展示了采用自监督学习结合ViT也可以取得和精心设计的卷积神经网络类型的效果。但是DINO具有如下两个重要的特点：

- 采用的是无标签的数据进行训练；
- 学习了更高阶层的通用特征，可以直接用于非监督语义分割；