

# 深度解析：对比学习 (Contrastive Learning) & 自编码器 (Autoencoder)

我们将详细讲解：

## 1. 对比学习 (Contrastive Learning)

- 介绍对比学习的核心思想
- 逐个解析 SimCLR, SwAV, BYOL, SimSiam, CLIP
- 给出每种方法的损失函数

## 2. 自编码器 (Autoencoder)

- 介绍自编码器的基本原理
- 解析 标准自编码器、去噪自编码器 (DAE)、Masked 自编码器 (MAE)
- 介绍自编码器的损失函数

---

## ◆ 1. 对比学习 (Contrastive Learning)

### (a) 核心思想

对比学习的目标是学习样本之间的相似性，使得：

- 正样本 (Positive Samples) 更接近
- 负样本 (Negative Samples) 更远离

常见方法：

1. 基于负样本的方法 (如 SimCLR)
2. 不需要负样本的方法 (如 BYOL, SimSiam)
3. 多模态方法 (如 CLIP)

---

### (b) SimCLR

#### ✦ 介绍

- 负样本：批次中的其他样本
- 网络结构：单网络，使用 投影头 (MLP Projection Head)
- 训练目标：最大化同一图像的不同增强版本的相似度

#### ✦ 损失函数

使用 InfoNCE (Normalized Temperature-scaled Cross-Entropy Loss)：

$$\mathcal{L} = -\log \frac{\exp(\text{sim}(z_i, z_j)/\tau)}{\sum_{k \neq i} \exp(\text{sim}(z_i, z_k)/\tau)}$$

- $\text{sim}(z_i, z_j)$  表示余弦相似度：  $\text{sim}(z_i, z_j) = \frac{z_i \cdot z_j}{\|z_i\| \|z_j\|}$
- $\tau$  是温度参数 (Temperature)

---

### (c) SwAV (Swapping Assignments between Views)

## ✦ 介绍

- 不使用负样本
- 使用聚类 (Clustering) 来学习特征

## ✦ 损失函数

SwAV 采用聚类损失:  $\mathcal{L} = \sum_i -q_i^T \log p_i$  其中:

- ( $q_i$ ) 是 Softmax 归一化后的聚类标签。
  - ( $p_i$ ) 是模型输出的概率分布。
- 

## (d) BYOL (Bootstrap Your Own Latent)

### ✦ 介绍

- 无负样本
- 双网络 (Student & Target)
- 目标网络无梯度

### ✦ 损失函数

$$\mathcal{L} = 2 - 2 \cdot \text{sim}(z_{\text{student}}, z_{\text{target}})$$

---

## (e) SimSiam

### ✦ 介绍

- 无负样本
- 单网络
- 通过梯度阻断 (Stop-Gradient) 避免崩溃

### ✦ 损失函数

$$\mathcal{L} = 1 - \text{cosine\_similarity}(z_{\text{proj}}, \text{stop\_grad}(z_{\text{pred}}))$$

---

## (f) CLIP

### ✦ 介绍

- 多模态 (图片-文本)
- 跨模态对比学习
- 目标: 让文本-图像对匹配

### ✦ 损失函数

$$\mathcal{L} = -\frac{1}{N} \sum_{i=1}^N \log \frac{\exp(\text{sim}(I_i, T_i)/\tau)}{\sum_{j=1}^N \exp(\text{sim}(I_i, T_j)/\tau)}$$

- $I_i$  和  $T_i$  是匹配的图像和文本。
-

## ◆ 2. 自编码器 (Autoencoder)

### (a) 标准自编码器

#### ✦ 介绍

- 目标：通过压缩数据学习隐藏特征。
- 结构：
  - 编码器 (Encoder):  $h = f(x)$
  - 解码器 (Decoder):  $x' = g(h)$

#### ✦ 损失函数

$$\mathcal{L} = \|x - x'\|^2 \text{ (均方误差, MSE)}$$

---

### (b) 去噪自编码器 (Denoising Autoencoder, DAE)

#### ✦ 介绍

- 输入数据被添加噪声
- 目标：恢复原始数据
- 训练方式：
  - 输入:  $\tilde{x} = x + \text{Noise}$
  - 目标:  $x' \approx x$

#### ✦ 损失函数

$$\mathcal{L} = \|x - x'\|^2$$

---

### (c) Masked Autoencoder (MAE)

#### ✦ 介绍

- 输入数据部分被遮蔽
- 模型需要重建遮蔽部分
- BERT 采用 Masked Language Modeling (MLM)

#### ✦ 损失函数

$$\mathcal{L} = \sum_{m \in \text{masked}} (x_m - x'_m)^2 \text{ 只计算被遮蔽部分的误差。}$$

---

## ✦ 总结

### 1 对比学习

方法	是否使用负样本?	网络结构	损失函数
SimCLR	✓	单网络	InfoNCE
SwAV	✗	单网络	聚类损失

方法	是否使用负样本?	网络结构	损失函数
BYOL	✗	双网络	余弦损失
SimSiam	✗	单网络	余弦损失
CLIP	☑ (多模态)	双网络	InfoNCE (跨模态)

## 2 自编码器

方法	核心思想	损失函数
标准自编码器	通过低维编码重建输入	MSE
去噪自编码器 (DAE)	训练模型恢复被噪声破坏的数据	MSE
Masked 自编码器 (MAE)	训练模型恢复被遮蔽的数据	MSE (仅计算被遮蔽部分)

## 重点理解

- 对比学习中的负样本作用：
  - SimCLR 依赖负样本，BYOL/SimSiam 通过梯度阻断避免崩溃。
- CLIP 是跨模态对比学习 (图像-文本)
- 自编码器通过信息压缩、去噪、遮蔽进行学习

☑ 如果你有任何问题，欢迎继续提问! 🚀