两种条件熵正则（CEL）实现的数学机制详解

——CEM-main 与 gated-att 的逐步推导与对比

2025 年 10 月 30 日

目录

1 研究场景与总体目标

我们考虑协同推理（split inference）环境：客户端持有输入图像 x 和前半段模型（编码器）$f_{\boldsymbol{\theta}}$；服务器端持有后半段模型 $g_{\boldsymbol{\phi}}$。编码器输出的中间表示（*smashed data*）$z = f_{\boldsymbol{\theta}}(x)$ 会发送到服务器端继续推理，也因此成为攻击者重建原图的突破口。为了降低模型反演攻击的成功率，我们在训练期间向分类损失 $\mathcal{L}_{\text{CE}}$ 之外加入条件熵正则项（Conditional Entropy Loss，CEL），鼓励同类 smashed data 在特征空间中彼此靠近、分布集中。

本文分别对 `CEM-main` 与 `gated-att` 两种实现中 CEL 的计算流程做细致说明，目标是让熟悉机器学习与数学但未接触项目代码的读者也能够完整复现两条管线。

2 统一符号与训练结构

- $x \in \mathcal{X}$：输入样本；$y \in \{1, \dots, C\}$：类别标签。
- $f_{\boldsymbol{\theta}}$：客户端编码器；$g_{\boldsymbol{\phi}}$：服务器端模型（尾部 + 分类器）。
- $z = f_{\boldsymbol{\theta}}(x)$：smashed data，维度 d。
- $\mathcal{B}$：当前 mini-batch；$Z_{\mathcal{B}} = \{z_i\}_{i=1}^{|\mathcal{B}|}$。
- $Z_{\mathcal{B}}^{(c)} = \{z_i \mid y_i = c\}$：batch 内类别 c 的子集，大小 $m_c = |Z_{\mathcal{B}}^{(c)}|$。
- 训练总损失
$$\mathcal{L} = \mathcal{L}_{\text{CE}} + \lambda \mathcal{L}_{\text{CEL}}.$$
- 记 $\varepsilon > 0$ 为数值稳定的平滑项；$\tau > 0$ 为控制方差上限的阈值；$\gamma \in (0, 1]$ 为缩放系数。

3 `CEM-main`：基于聚类的条件熵正则

3.1 高层流程概览

`CEM-main` 采用“**跨 epoch 汇总 + 聚类 + 方差门控**”的思路。每个 epoch 结束时，收集当轮全部 smashed data，并按类别执行 K-means 或高斯混合模型（GMM）聚类，用以估计真实条件分布 $p(z \mid y = c)$ 的多个模式。下一轮训练时，mini-batch 内的 smashed data 将借助聚类结果获得类内方差估计，从而构造 CEL。

3.2 步骤 0：初始化与 warm-up

0.1 选择簇数 K、阈值 τ、平滑项 ε。

0.2 初始化聚类缓存 $\mathcal{S}_c = \varnothing$；若设定 warm-up 轮数 T_{warm}，则在 $t \le T_{\text{warm}}$ 时暂不引入 CEL。

3.3 步骤 1：全量特征收集与聚类

在第 t 个训练 epoch 结束时：

1.1 遍历本轮的所有 mini-batch，将 smashes 以及标签打包到集合

$$\mathcal{Z}^{(t)} = \{(z_i, y_i)\}_{i=1}^{N_t}, \quad z_i = f_{\boldsymbol{\theta}}(x_i).$$

1.2 对每个类别 c 取出其样本集合 $Z_c^{(t)} = \{z_i \mid y_i = c\}$，如果上一轮已有聚类结果，可将其中心 $\mu_{c,k}^{(t-1)}$ 作为暖启动。

1.3 执行 K-means（或 GMM）：

$$\begin{aligned}
S_{c,k} &= \{z \in Z_c^{(t)} \mid k = \arg\min_{k'} \|z - \mu_{c,k'}\|_2^2\}, \\
\mu_{c,k} &= \frac{1}{|S_{c,k}|} \sum_{z \in S_{c,k}} z, \\
v_{c,k} &= \frac{1}{|S_{c,k}|} \sum_{z \in S_{c,k}} \|z - \mu_{c,k}\|_2^2, \\
\pi_{c,k} &= \frac{|S_{c,k}|}{|Z_c^{(t)}|}.
\end{aligned}$$

如需更丰富的统计，可令协方差 $\Sigma_{c,k} = \operatorname{diag}\big(\frac{1}{|S_{c,k}|} \sum (z - \mu_{c,k})(z - \mu_{c,k})^\top\big)$。

1.4 将 $(\pi_{c,k}, \mu_{c,k}, v_{c,k})_{k=1}^K$ 缓存在 $\mathcal{S}_c$ 中，供下一轮使用。

3.4 步骤 2：mini-batch 内的 CEL 评估

对第 $t+1$ 轮训练中的任意 mini-batch $\mathcal{B}$：

2.1 计算 smashed data $z_i = f_{\boldsymbol{\theta}}(x_i)$。

2.2 对每个类别 c，根据上一轮的统计 $\mathcal{S}_c$，对 batch 内的样本执行簇分配：

$$k^\star(z) = \arg\min_k \|z - \mu_{c,k}\|_2^2, \quad z \in Z_{\mathcal{B}}^{(c)}.$$

2.3 在 mini-batch 数据上重新估计类内方差：

$$\hat{v}_c = \sum_{k=1}^{K} \pi_{c,k} \cdot \hat{v}_{c,k}, \qquad \hat{v}_{c,k} = \frac{1}{|S_{c,k}^{(\mathcal{B})}|} \sum_{z \in S_{c,k}^{(\mathcal{B})}} \|z - \mu_{c,k}\|_2^2,$$

其中 $S_{c,k}^{(\mathcal{B})} = \{z \in Z_{\mathcal{B}}^{(c)} \mid k^\star(z) = k\}$，若该集合为空则跳过该簇。

2.4 定义门控函数：

$$\mathcal{R}_{\text{lin}}(c) = \hat{v}_c, \tag{1}$$

$$\mathcal{R}_{\log}(c) = \max\Big(0, \log(\hat{v}_c + \varepsilon) - \log(\tau + \varepsilon)\Big). \tag{2}$$

根据实验设置选择其一，得到

$$\mathcal{L}_{\text{CEL}} = \sum_{c=1}^{C} \beta_c \mathcal{R}(c), \qquad \beta_c = \frac{m_c}{|\mathcal{B}|}.$$

3.5 步骤 3：损失合成与梯度传播

计算总损失 $\mathcal{L} = \mathcal{L}_{\text{CE}} + \lambda \mathcal{L}_{\text{CEL}}$。在原实现中，为了确保 CEL 只对客户端编码器产生影响，会按以下顺序处理梯度：

3.1 仅对 $\mathcal{L}_{\text{CEL}}$ 做一次反向传播，保留编码器梯度 $\nabla_{\boldsymbol{\theta}} \mathcal{L}_{\text{CEL}}$。

3.2 清零梯度，再对 $\mathcal{L}_{\text{CE}}$ 反向传播。

3.3 将两者梯度相加，更新 $\boldsymbol{\theta}$；服务器端参数 $\boldsymbol{\phi}$ 仅由 $\mathcal{L}_{\text{CE}}$ 更新。

3.6 机制总结

聚类法的关键优势在于：它通过离线统计刻画每个类别的多峰结构，能够利用跨 batch 的全局信息；缺点是内存消耗与聚类开销较大，并需要额外的簇数与阈值调参。

4 gated-att：基于门控注意力的条件熵正则

4.1 设计出发点

`gated-att` 抛弃离线聚类，改用“**批内门控注意力 + 加权二阶统计**”构造 CEL。该方案来自多实例学习中的 gated attention [2]：通过学习两个投影 V、U，用门控信号 $\sigma(Uz)$ 调节 $\tanh(Vz)$，从而得到既可以区分类内关键样本又保持稳定的注意力权重。核心思想是：让模型自己决定哪些 *smashed data* 更值得关注。

4.2 步骤 0：初始化与 warm-up

0.1 设定 warm-up 轮数 T_{warm}，在前 T_{warm} 轮仅优化分类器。

0.2 初始化注意力模块参数 $\boldsymbol{\phi} = \{W_V, W_U, \boldsymbol{w}\}$：

$$W_V \in \mathbb{R}^{h\times d}, \quad W_U \in \mathbb{R}^{h\times d}, \quad \boldsymbol{w} \in \mathbb{R}^h,$$

其中 h 为注意力隐层维度，可取 $d/4$ 或固定常数（如 128）。

0.3 可同时初始化 LayerNorm 或 BatchNorm，用于标准化 smashed data。

4.3 步骤 1：批内预处理

对任意 mini-batch $\mathcal{B}$（假设 $t > T_{\text{warm}}$ 已启用 CEL）：

1.1 计算 smashed data $z_i = f_{\boldsymbol{\theta}}(x_i)$。

1.2 若特征是卷积张量（$C \times H \times W$），则在编码器端已有展平操作；若没有，可在进入注意力模块前使用 reshape。

1.3 对 z_i 执行规范化，如 LayerNorm：

$$\tilde{z}_i = \text{LayerNorm}(z_i).$$

规范化的目的是让注意力网络处理的输入具有稳定尺度。

4.4 步骤 2：门控注意力权重计算

对于类别 c 的特征集合 $Z_{\mathcal{B}}^{(c)}$，依次进行：

2.1 双投影与门控：对每个样本，计算两个线性投影并进行门控：

$$\begin{aligned}
\boldsymbol{v}_i &= \tanh(W_V \tilde{z}_i), \\
\boldsymbol{u}_i &= \sigma(W_U \tilde{z}_i), \\
\boldsymbol{s}_i &= \boldsymbol{v}_i \odot \boldsymbol{u}_i,
\end{aligned}$$

其中 σ 为 Sigmoid，$\odot$ 表示逐元素乘。$\boldsymbol{u}_i$ 对应门控通道：它控制 V 投影的每个元素是否被放大或抑制；$\boldsymbol{v}_i$ 则提供非线性刻画能力。

2.2 求注意力 logit：通过一个向量 $\boldsymbol{w}$ 将门控后的特征压缩为标量：

$$\alpha_i' = \boldsymbol{w}^\top \boldsymbol{s}_i.$$

2.3 归一化：使用 softmax 得到注意力权重：

$$\alpha_i = \frac{\exp(\alpha'_i)}{\sum_{j=1}^{m_c} \exp(\alpha'_j)}, \qquad \sum_{i=1}^{m_c} \alpha_i = 1.$$

注意力权重满足两个作用：

(i) 通过 softmax 强制权重归一，有利于稳定梯度；

(ii) 通过门控结构，在 $\boldsymbol{u}_i$ 对某些维度给出较小值时，可以让输入较“异常”的样本获得更低权重。

4.5 步骤 3：加权一阶、二阶统计与 CEL

3.1 加权均值：

$$\bar{z}_c = \sum_{i=1}^{m_c} \alpha_i z_i.$$

3.2 加权方差（类内散度）：

$$v_c = \sum_{i=1}^{m_c} \alpha_i \|z_i - \bar{z}_c\|_2^2.$$

注意：若方差矩阵而非标量很重要，也可以保留加权协方差矩阵

$$\Sigma_c = \sum_{i=1}^{m_c} \alpha_i (z_i - \bar{z}_c)(z_i - \bar{z}_c)^\top,$$

但在实际实现中，为简洁通常只取 trace（即上式的平方和）。

3.3 门控函数（与前述相同）：

$$\mathcal{R}_{\text{lin}}(c) = v_c, \tag{3}$$

$$\mathcal{R}_{\log}(c) = \max\Big(0, \log(v_c + \varepsilon) - \log(\tau + \varepsilon)\Big). \tag{4}$$

3.4 CEL 聚合：

$$\mathcal{L}_{\text{CEL}} = \sum_{c=1}^{C} \beta_c \mathcal{R}(c), \qquad \beta_c = \frac{m_c}{|\mathcal{B}|}.$$

4.6 步骤 4：损失合成与梯度流向

总损失写作

$$\mathcal{L} = \mathcal{L}_{\text{CE}} + \lambda\gamma \mathcal{L}_{\text{CEL}}.$$

其梯度同时作用于：

- 客户端编码器参数 $\boldsymbol{\theta}$：通过 v_c 中的 z_i 依赖关系得到；

- 注意力模块参数 $\boldsymbol{\phi}$：通过 α_i 的 softmax 和门控结构得到；
- 服务器端参数 $\boldsymbol{\phi}$：只受 $\mathcal{L}_{\text{CE}}$ 影响，与 `CEM-main` 一致。

首次启用 CEL 时，需要将注意力参数加入优化器。由于注意力结构简单（几层仿射映射），其额外计算成本约为 $O(|\mathcal{B}|hd)$。

4.7 门控注意力发挥作用的直观解释

门控注意力之所以适合作为 CEL surrogate，是因为它提供了三个层面的信息过滤：

1. **尺度归一化**：通过 LayerNorm 或 BatchNorm 控制输入幅度，让注意力网络聚焦于方向性差异而非尺度差异。
2. **双分支投影**：W_V 与 W_U 分别对应“候选特征”和“门控信号”。若某些 smashed data 在 W_U 投影下激活较低，则其对应元素被 sigmiod 抑制，相当于告诉网络“这条样本不可靠”。
3. **Softmax 权重分配**：softmax 将所有注意力归一，其梯度能够自适应地放大贡献较小、但对降低方差更有效的样本。

在训练早期，由于 W_V、W_U 尚未学到有效区分模式的能力，方差项可能出现偏大或梯度不稳定的现象。因此 warm-up 和缩放系数 γ 十分必要：先保证分类准确率，再逐步收紧 smashed data 的类内分布。

4.8 完整伪代码

Algorithm 1 `gated-att` 中的 CEL 计算（单个 mini-batch）

Require: smashed data $\{(z_i, y_i)\}_{i=1}^{|\mathcal{B}|}$，注意力参数 $W_V, W_U, \boldsymbol{w}$，阈值 τ

Ensure: $\mathcal{L}_{\text{CEL}}$

for 每个类别 c **do**

  $Z_c \leftarrow \{z_i \mid y_i = c\}$，$m_c \leftarrow |Z_c|$

  对 Z_c 中每个 z 做规范化 $\tilde{z} \leftarrow \text{LayerNorm}(z)$

  计算 $\boldsymbol{v} = \tanh(W_V \tilde{z})$，$\boldsymbol{u} = \sigma(W_U \tilde{z})$，$\boldsymbol{s} = \boldsymbol{v} \odot \boldsymbol{u}$

  $\alpha' = \boldsymbol{w}^\top \boldsymbol{s}$；对所有样本做 softmax 得 α

  $\bar{z}_c = \sum_{z \in Z_c} \alpha(z)\, z$

  $v_c = \sum_{z \in Z_c} \alpha(z)\, \|z - \bar{z}_c\|_2^2$

  $\mathcal{R}(c) = \max\big(0, \log(v_c + \varepsilon) - \log(\tau + \varepsilon)\big)$

end for

$\mathcal{L}_{\text{CEL}} = \sum_c \frac{m_c}{|\mathcal{B}|} \mathcal{R}(c)$

5 两种方法的对比与思考

表 1: 聚类法与门控注意力法的关键差异

维度	`CEM-main`（聚类）	`gated-att`（门控注意力）
统计方式	跨 epoch 聚类估计多峰结构	批内注意力即时估计散度
数据缓存	需保存所有 smashed data	仅需当前 batch
额外参数	无（仅统计量）	$W_V, W_U, \boldsymbol{w}$ 等注意力参数
计算复杂度	聚类 $O(NKd)$，内存 $O(Nd)$	每 batch $O(\|\mathcal{B}\|hd)$
控制力度	依赖聚类分配的准确性	依赖注意力学习到正确权重
梯度流向	只约束编码器	同时约束编码器与注意力模块
调参要点	簇数、阈值、聚类频率	warm-up、注意力维度、缩放系数

6 详尽示例：三类二维特征

设 smashed data 维度为 2，共三类，每类两个样本：

$$Z^{(1)} = \{(0.0, 0.0), (0.2, 0.1)\},$$
$$Z^{(2)} = \{(1.0, 1.0), (1.2, 1.1)\},$$
$$Z^{(3)} = \{(1.0, -1.0), (0.8, -0.9)\}.$$

令 $\tau = 0.05, \varepsilon = 10^{-6}$。

6.1 聚类法

1. $K = 1$，簇中心分别为 $\mu_{1,1} = (0.1, 0.05)$、$\mu_{2,1} = (1.1, 1.05)$、$\mu_{3,1} = (0.9, -0.95)$。
2. 方差约为 $v_{1,1} \approx 0.03125$，$v_{2,1} \approx 0.01$，$v_{3,1} \approx 0.01$。
3. log-entropy 下三者均低于阈值，CEL 为 0；线性形式则 $\mathcal{L}_{\text{CEL}}^{\text{lin}} = \frac{1}{3}(0.03125+0.01+0.01) = 0.0171$。

6.2 门控注意力法

1. 若注意力未训练，假设 $\alpha_i = 0.5$，则 $\bar{z}_c = \mu_{c,1}$。
2. 加权方差同样得到 v_1, v_2, v_3，与聚类法数值一致。
3. 当注意力网络学习到“偏离中心的样本应被赋予更高权重”时，例如将类别 1 的权重调整为 $\alpha = (0.7, 0.3)$，方差估计会更准确地捕捉到散布情况，并通过梯度促使较远样本向中心收缩。

7 实践建议

针对 `CEM-main`

- 合理设置簇数 K：类别越复杂，K 应适当提高。
- 聚类频率：可选择每轮或每几轮执行一次以平衡时间开销。
- 阈值 τ 应与噪声注入（如高斯噪声）协同调整。

针对 `gated-att`

- warm-up 至少 5–10 轮，有利于避免训练初期不稳定。
- 注意力隐层维度 h 不宜过大；常见选择为 $d/4$ 或 128。
- 缩放系数 γ 可以从 0.1 逐步增大，观察其对准确率和重建质量的影响。

8 总结

两种 CEL 方案服务于同一目标：降低 smashed data 的条件熵，从而防御模型反演攻击。聚类法擅长聚合跨 batch 的全局统计，适合有充足内存与时间的环境；门控注意力法提供端到端、低内存开销的替代方案，并通过可学习的门控机制动态决定样本的重要性。实践中可根据资源与需求选择其一，亦可考虑结合两者（例如用注意力结果初始化聚类中心）。

附：常见符号

符号	含义
z	smashed data 特征向量
K	聚类簇数
$\mu_{c,k}$	类别 c 的第 k 个簇中心
v_c	类别 c 的加权方差
W_V, W_U	门控注意力的两个线性投影矩阵
$\boldsymbol{w}$	注意力权重投影向量
α_i	softmax 注意力权重

参考文献

参考文献

[1] Xia, S., Yu, Y., Yang, W., Ding, M., Chen, Z., Duan, L.-Y., Kot, A. C., & Jiang, X. Theoretical Insights in Model Inversion Robustness and Conditional Entropy Maximization for Collaborative Inference Systems.
In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

[2] Ilse, M., Tomczak, J. M., & Welling, M.
Attention-based Deep Multiple Instance Learning.
In *Proceedings of the International Conference on Machine Learning (ICML)*, 2018.