arXiv:2303.00181v1 [cs.CV] 1 Mar 2023

用于缓解图文匹配中梯度消失的选择性困难负样本挖掘

Zheng Li[1], Caili Guo[1], 王鑫[1],Zerun Feng[1],杜中天[21]北京邮电大学[2]中国电信数字智能科技有限公司

摘要

近年来，一系列图文匹配（*ITM*）方法取得了令人瞩目的性能。然而，我们发现，大多数现有图文匹配模型在训练初期都会出现梯度消失，因而容易陷入局部极小值。大多数图文匹配模型采用困难负样本挖掘（*HN*）和三元组损失作为优化目标。我们发现，仅使用困难负样本来优化图文匹配模型很容易导致梯度消失。本文推导了训练过程中梯度消失的条件：当正样本对与负样本对之间的相似度差接近 *0* 时，图像编码器和文本编码器上的梯度都会趋近于 *0*。为缓解梯度消失问题，我们提出了一种选择性困难负样本挖掘（*SelHN*）策略，根据梯度消失条件决定是否挖掘困难负样本。*SelHN* 可即插即用于现有图文匹配模型，从而改善其训练行为。为进一步确保梯度反向传播，我们构建了一个结合 *SelHN* 的残差视觉语义嵌入模型，记为 *RVSE++*。在两个图文匹配基准数据集上进行的大量实验展示了 *RVSE++* 的优势，并取得了最先进的性能。

1. 引言

图文匹配（ITM） [4, 18, 23] 旨在建立图像与文本之间的对应关系，这是各种视觉语言理解任务的基础。图文匹配面临的挑战在于图像与文本之间存在异构差距。图文匹配模型需要准确学习图像与文本之间的语义对应关系。现有的图文匹配方法通常遵循一个通用框架，其中包含一个由图像编码器和文本编码器组成的双分支神经网络，并以三元组损失 [30] 作为优化目标。三元组由三部分组成：锚点、正样本和负样本。我们将训练集中的每个样本都视为锚点。与锚点在语义上相关的样本

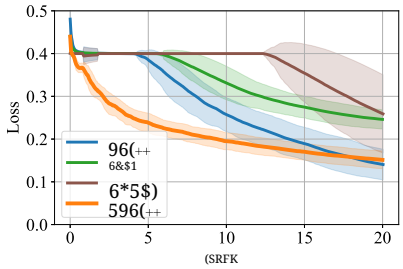


图1. 在 Flickr30K 数据集上绘制损失随训练轮次变化的曲线。实线表示损失的平滑均值，阴影区域表示其标准差。

称为正样本，而与锚点无关的样本称为负样本。锚点与对应的正样本构成正样本对。同样地，锚点与负样本构成负样本对。实现图文匹配的主流方法是学习一个能够衡量图像与文本之间匹配分数的模型。我们分别将正样本对和负样本对的匹配分数记为 s_p 和 s_n。三元组损失可以引导模型实现 $s_p > s_n$。

图文匹配方法主要分为两类，即视觉语义嵌入（VSE）方法（如 VSE++ [9]）和交叉注意力（CA）方法（如 SCAN [17]）。VSE 方法将整幅图像和整个句子嵌入联合嵌入空间。匹配分数可以通过简单的相似度度量计算得出。CA 方法通过计算图像区域与词语之间的交叉注意力来获得匹配分数。还有一些方法结合了 VSE 和 CA 方法进行全局和局部匹配，例如 SGRAF [8]。对于图文匹配模型的优化目标，Faghri等人 [9] 提出了困难负样本挖掘（HN）三元组损失。HN 会在每个批次中挖掘具有最大s_n 的困难负样本，从而显著提升匹配性能。大多数现有图文匹配方法 [8, 17, 43] 都采用含困难负样本的三元组损失作为优化目标。

近年来，一系列图文匹配方法取得了令人瞩目的表现。然而，我们发现，大多数图文匹配模型的训练行为不佳。如图1所示，我们

在 Flickr30K 数据集上绘制了若干经典图文匹配模型在各训练轮次的损失 [41] ，包括视觉语义嵌入++、SCAN 和 SGRAF。它们都采用含困难负样本的三元组损失作为优化目标。这三个模型在训练初期都经历了一段损失不下降的时期。尤其是模型结构最复杂的 SGRAF，其损失在前 12 个训练轮次中都没有下降。本文通过实验观察到，训练行为不佳的原因是图文匹配模型遭遇了梯度消失，导致模型容易陷入局部极小值。因此，模型无法充分发挥其表征能力。

在深度度量学习（DML）中，一些研究 [30, 39]关注使用三元组损失进行训练所导致的不良训练行为，并将其归因于对困难负样本进行优化。Schroff 等人 [30] 将满足 $s_n > s_p$ 的负样本定义为困难负样本。多项研究表明，在训练早期对困难负样本进行优化会导致不良的局部极小值 [27, 30, 39]。为缓解这一问题，研究者提出了多种负样本挖掘策略 [30, 39], ，例如半难负样本挖掘（SHN） [30], 。这类方法通常会放弃对包含困难负样本的三元组进行优化。但困难负样本对于学习判别模型至关重要 [39]。

本文针对图文匹配模型训练过程中的梯度消失问题提出了一种解决方案。通过对含困难负样本的三元组损失进行梯度分析，我们得到了梯度消失的条件。当（$s_n - s_p$）接近 0 时，图像编码器和文本编码器上的梯度将趋近于 0。为缓解梯度消失，我们提出了一种选择性困难负样本挖掘（SelHN）策略。满足该条件时，SelHN 使用所有负样本进行优化；不满足时，则挖掘困难负样本用于训练。SelHN 可即插即用地应用于现有图文匹配模型，从而改善其训练行为。为进一步确保梯度能够反向传播，我们构建了一个结合 SelHN 的残差视觉语义嵌入模型，记为 RVSE++。我们的主要贡献总结如下：

- 我们推导出了图文匹配模型训练过程中发生梯度消失的条件，这可用于指导困难负样本挖掘策略的设计。

- 我们提出了SelHN策略，以缓解梯度消失问题。SelHN可以即插即用地应用于现有图文匹配模型，从而使其具有更好的训练行为。

- 我们构建了一个 RVSE++模型。在两个图文匹配基准数据集上进行的大量实验表明，RVSE++性能强劲，达到了最先进水平。

2. 相关工作

2.1. 图文匹配模型

图文匹配旨在建立图像与文本之间的对应关系，这是多种视觉与语言理解任务的基础，例如视觉语言检索、图像/视频描述 [11, 25], 以及视觉问答 [3, 12]。现有的图文匹配方法主要分为两类：即视觉语义嵌入（VSE）方法 [5, 18, 32], 和交叉注意力（CA）方法 [4, 23, 44]。

视觉语义嵌入。 视觉语义嵌入方法将整幅图像和句子嵌入联合嵌入空间 [20, 40, 45]中。图像与文本之间的匹配分数可以通过简单的相似度度量（例如余弦相似度）计算。因此，视觉语义嵌入方法具有更快的推理速度，适用于实际应用场景。Frome等人 [10] 提出了首个视觉语义嵌入模型DeViSE，该模型利用卷积神经网络和 Skip-Gram [26] 将图像和句子投影到联合嵌入空间中。Faghri等人 [9]提出了VSE++，该方法使用卷积神经网络对图像进行编码，并使用门控循环单元（GRU） [7]对句子进行编码，同时在三元组损失中引入了困难负样本挖掘。Chen等人 [5]提出了广义池化算子（GPO），用于学习最佳池化策略以生成全局嵌入。我们提出的RVSE++属于视觉语义嵌入方法。

交叉注意力。CA 方法通过计算图像区域与词语之间的交叉注意力来获得匹配分数 [4, 23, 28, 33, 38, 44]。Lee 等人 [17]提出了 SCAN，通过对齐图像区域与词语来衡量图文相似度。交叉注意力方法能够匹配更细粒度的图文对应关系，但匹配速度较慢。部分工作将视觉语义嵌入方法与 CA 方法相结合。Diao 等人 [8] 提出了 SGRAF，其中包含用于局部和全局对齐的相似度图推理网络和注意力过滤网络。尽管这些图文匹配方法通过学习丰富的嵌入表示取得了令人瞩目的性能，我们观察到，大多数现有图文匹配模型在训练过程中都会出现梯度消失。

2.2. 图文匹配的损失函数

近年来，研究者提出了多种用于图文匹配的损失函数 [6, 9, 36] 。基于铰链的三元组损失 [10]被广泛用作优化目标，通过设置间隔，促使正样本对的匹配分数高于负样本对。Faghri等人 [9] 提出了含困难负样本的三元组损失，将困难负样本纳入三元组损失，从而显著提升了匹配性能。大多数最先进的图文匹配方法 [8, 17, 18, 43] 都采用含困难负样本的三元组损失作为优化目标。一些研究提出了更复杂的样本挖掘 [6] 或样本加权 [35–37] 策略—

这些策略进一步提升了图文匹配模型的匹配性能。这些方法都专注于构建优化目标，以学习具有判别性的特征表示，却忽视了神经网络训练过程中容易出现的梯度消失问题。本文提出的 SelHN 通过选择性困难负样本挖掘，在匹配性能上优于这些复杂的样本加权或挖掘策略。

一些深度度量学习领域的研究工作 [30, 39] 关注困难负样本挖掘带来的不良训练行为。SHN [30] 不优化困难负样本，而只挖掘带有 $s_n < s_p$ 的负样本来优化三元组损失。Xuan 等人 [39]提出了选择性对比三元组损失（SCT），使用对比损失 [13] 优化困难负样本，以避免不良训练行为，并使用三元组损失优化其余样本。这些方法改善了三元组损失的训练行为，但放弃了对由困难负样本构成的三元组进行优化。困难负样本对于学习具有判别能力的模型至关重要 [39]。本文推导了图文匹配模型训练过程中梯度消失的条件。所提出的 SelHN 根据该条件决定是否挖掘困难负样本。SelHN 不仅改善了图文匹配模型的训练行为，还充分发挥了困难负样本挖掘的作用。

3. 方法

3.1. 预备知识

图文匹配。我们首先介绍图文匹配的相关背景。给定一组图像 $\boldsymbol{V}$和相应的一组句子 $\boldsymbol{T}$。令 V_i 为一幅图像，T_i 为一个句子，(V_i, T_i)为一个图文对。我们将正样本对表示为 (V_i, T_i)，将负样本对表示为 $(V_i, T_j)_{i\neq j}$。现有的图文匹配方法通常遵循一个通用框架，其中包含一个作为图像编码器和文本编码器的双分支神经网络，并以三元组损失作为优化目标。我们分别将图像编码器和文本编码器表示为 $\boldsymbol{f}_{vis}(\cdot)$ 和 $\boldsymbol{f}_{text}(\cdot)$。该方法背后的核心思想是，存在一个映射函数 $s(V_i, T_i; \boldsymbol{W}) = \boldsymbol{f}_{vis}(V_i)^\top \boldsymbol{W} \boldsymbol{f}_{text}(T_i)$，用于度量视觉特征 $\boldsymbol{f}_{vis}(V_i)$ 与文本特征 $\boldsymbol{f}_{text}(T_i)$ 之间的匹配分数，其中 $\boldsymbol{W}$ 表示 $s(\cdot,\cdot)$ 的参数。对于视觉语义嵌入方法，$s(\cdot,\cdot)$ 是一种简单的相似度度量（例如余弦相似度）。对于 CA 方法，$s(\cdot,\cdot)$ 是一个基于注意力的匹配分数生成器。

模型架构。为便于开展实验分析图文匹配中的梯度消失问题，我们构建了三个简单的视觉语义嵌入模型。

VSE（FC）。对于图像编码，我们采用大多数图文匹配方法 [17, 18]所使用的自下而上和自上而下（BUTD）注意力区域特征 [1]。给定图像 V_i，我们使用 Faster-RCNN [29] 获取 M 区域级特征 $\{\boldsymbol{r}_i^m\}_{m=1}^M$。然后，我们使用全连接（FC）层将 $\{\boldsymbol{r}_i^m\}_{m=1}^M$ 的维度映射到 d，其中 d 是嵌入维度。最后，我们使用 GPO [5]将图像特征集聚合为图像嵌入 $\boldsymbol{v}_i \in \mathbb{R}^d$。VSE（FC）的图像编码器架构如图2a所示。对于文本编码，给定句子 T_i，我们采用双向门控循环单元 [2] 来获取 N 词级特征 $\{\boldsymbol{w}_i^n\}_{n=1}^N$。然后，我们同样使用 GPO 将文本特征集聚合为文本嵌入 $\boldsymbol{t}_i \in \mathbb{R}^d$。

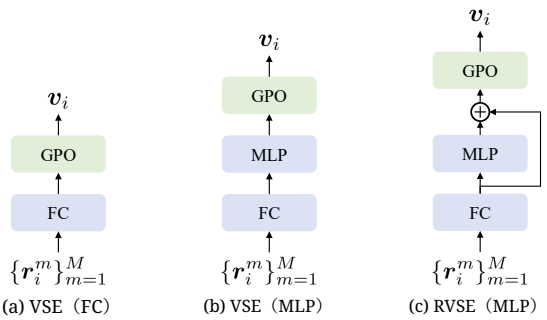


图 2. 几种图像编码器架构。

VSE（MLP）。为了增强图像编码器的表征能力，我们在 VSE（FC）的基础上，在全连接层和 GPO 之间添加了一个两层多层感知机（MLP），如图2b所示。除图像编码器外，VSE（MLP）的所有模型架构都与 VSE（FC）相同。

RVSE（MLP）。残差连接是缓解神经网络训练过程中梯度消失的一种有效方法。因此，我们在 VSE（MLP）的基础上，在全连接层和 GPO 之间添加了一个残差连接，记为 RVSE（MLP），如图2c所示。

对于这三种 VSE 模型，我们分别用 $\boldsymbol{\theta}_{vis}$ 和 $\boldsymbol{\theta}_{text}$ 表示图像编码器和文本编码器的可训练参数。$\boldsymbol{v}_i$ 和 $\boldsymbol{t}_i$ 可表示为：

$$\boldsymbol{v}_i = \boldsymbol{f}_{vis}\left(V_i; \boldsymbol{\theta}_{vis}\right), \boldsymbol{t}_i = \boldsymbol{f}_{text}\left(T_i; \boldsymbol{\theta}_{text}\right), \tag{1}$$

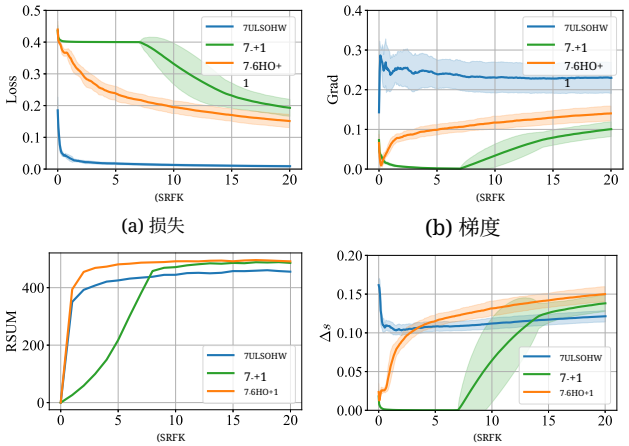
注意，$\boldsymbol{v}_i$ 和 $\boldsymbol{t}_i$ 都是经过 L_2 归一化的单位向量。最后，我们使用余弦相似度计算图像与文本之间的匹配分数：$s(V_i, T_i) = \boldsymbol{v}_i^\top \boldsymbol{t}_i$。

损失函数。为进行后续实验和梯度分析，我们重新审视两种经典损失函数。

三元组损失 [30] 是深度度量学习中的经典损失函数，广泛应用于人脸识别 [30]，图像检索 [27]，等。应用于图文匹配的三元组损失包含图像到文本和文本到图像两个方向，可表示为：

$$\begin{aligned}\mathcal{L}_{\text{Triplet}} = \sum_{i=1}^{B} \sum_{j=1, i\neq j}^{B} &\left([s(V_i, T_j) - s(V_i, T_i) + \lambda]_+ \right. \\ &\left. + [s(V_j, T_i) - s(V_i, T_i) + \lambda]_+\right),\end{aligned} \tag{2}$$

其中，B 为批大小，λ 为用于更好地区分相似度的间隔，$[x]_+ = \max(x, 0)$。



(a) 损失 (b) 梯度

(c) 召回率之和 (d) Δs

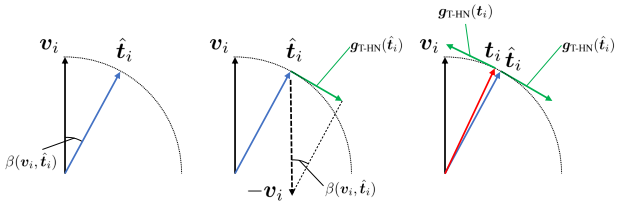
图3. VSE（FC）在Flickr30K上的训练性能。

困难负样本挖掘三元组损失（HN） [9] 将困难负样本纳入三元组损失，从而显著提升匹配性能。大多数最先进的图文匹配方法 [8, 17, 18, 43] 采用含困难负样本的三元组损失作为优化目标，其形式如下：

$$\mathcal{L}_{\text{T-HN}} = \sum_{i=1}^{B} \Big([s(V_i, \hat{T}_i) - s(V_i, T_i) + \lambda]_+ + [s(\hat{V}_i, T_i) - s(V_i, T_i) + \lambda]_+ \Big), \tag{3}$$

其中 $\hat{T}_i = \arg\max_{j=1, i\neq j}^{B} s(V_i, T_j)$ 和 $\hat{V}_i = \arg\max_{j=1, j\neq i}^{B} s(V_j, T_i)$ 是困难负样本。

模型训练。 如图 1所示，现有图文匹配模型通常具有不良的训练行为。为了探究原因，我们使用 VSE（FC）、VSE（MLP）和 RVSE（MLP）进行实验分析。我们在 Flickr30K 数据集上训练这三个模型，采用 $\mathcal{L}_{\text{Triplet}}$ 和 $\mathcal{L}_{\text{T-HN}}$ 作为优化目标。这些模型使用 AdamW [24] 训练 20 个训练轮次，批大小为 128。嵌入维度 d 设为 1,024。模型的学习率设为 0.0005。间隔 λ 设为 0.2。图 3a 展示了 VSE （FC）训练过程中各训练轮次对应的损失。可以看出， $\mathcal{L}_{\text{Triplet}}$ 在训练过程中稳步下降，没有出现不良训练行为。但 $\mathcal{L}_{\text{T-HN}}$ 在第 0 至 7 个训练轮次并未下降。如图 3b所示，我们统计了 VSE（FC）模型训练过程中全连接层的梯度。使用 $\mathcal{L}_{\text{T-HN}}$ 训练时，全连接层的梯度在第 0 至 7 个训练轮次接近 0 。而使用 $\mathcal{L}_{\text{Triplet}}$ 训练则不会出现梯度消失。这与训练过程中损失的下降趋势一致。因此，我们推断，不良训练行为的原因是梯度消失，这使模型无法充分发挥其表征能力。即使 $\mathcal{L}_{\text{T-HN}}$ 具有不良训练行为，如图 3c所示，困难负样本挖掘仍能提升模型的匹配性能，因此大多数



(a) (b) (c)

图4. 图文匹配模型使用困难负样本进行训练时嵌入的梯度示意图。

图文匹配模型仍使用 $\mathcal{L}_{\text{T-HN}}$ 作为优化目标。

3.2. 图文匹配中的梯度消失

为了探究困难负样本挖掘为何会导致梯度消失，我们分析了神经网络训练过程中 $\mathcal{L}_{\text{T-HN}}$ 的梯度，并得出了训练期间梯度消失的条件。我们考虑$\boldsymbol{f}_{vis}$(·)和 $\boldsymbol{f}_{text}$(·)将图像和文本嵌入单位超球面。为简化表示，我们仅分析图像到文本方向的损失，该损失与文本到图像方向的损失对称。以 $\boldsymbol{v}_i$ 为锚点，含困难负样本的三元组损失形式如下：

$$\mathcal{L}_{\text{T-HN}}(\boldsymbol{v}_i) = \max\left(\boldsymbol{v}_i^\top \hat{\boldsymbol{t}}_i - \boldsymbol{v}_i^\top \boldsymbol{t}_i + \lambda, 0\right), \tag{4}$$

其中， $\boldsymbol{t}_i$ 是正样本， $\hat{\boldsymbol{t}}_i$ 是困难负样本。当 $(\boldsymbol{v}_i^\top \hat{\boldsymbol{t}}_i - \boldsymbol{v}_i^\top \boldsymbol{t}_i + \lambda) \leq 0$时， $\mathcal{L}_{\text{T-HN}}(\boldsymbol{v}_i) = 0$。因此，我们只考虑 $(\boldsymbol{v}_i^\top \hat{\boldsymbol{t}}_i - \boldsymbol{v}_i^\top \boldsymbol{t}_i + \lambda) > 0$的情况。我们推导出损失关于 $\boldsymbol{\theta}_{vis}$ 和 $\boldsymbol{\theta}_{text}$的梯度：

$$\frac{\partial \mathcal{L}_{\text{T-HN}}(\boldsymbol{v}_i)}{\partial \boldsymbol{\theta}_{vis}} = \frac{\partial \mathcal{L}_{\text{T-HN}}(\boldsymbol{v}_i)}{\partial \boldsymbol{v}_i} \frac{\partial \boldsymbol{v}_i}{\partial \boldsymbol{\theta}_{vis}}, \tag{5}$$

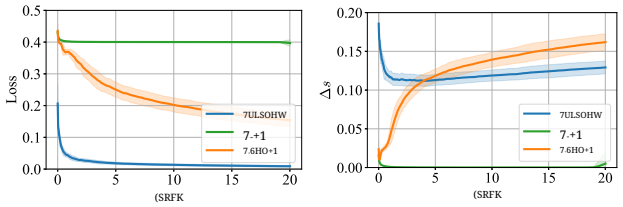
$$\frac{\partial \mathcal{L}_{\text{T-HN}}(\boldsymbol{v}_i)}{\partial \boldsymbol{\theta}_{text}} = \frac{\partial \mathcal{L}_{\text{T-HN}}(\boldsymbol{v}_i)}{\partial \hat{\boldsymbol{t}}_i} \frac{\partial \hat{\boldsymbol{t}}_i}{\partial \boldsymbol{\theta}_{text}} + \frac{\partial \mathcal{L}_{\text{T-HN}}(\boldsymbol{v}_i)}{\partial \boldsymbol{t}_i} \frac{\partial \boldsymbol{t}_i}{\partial \boldsymbol{\theta}_{text}}. \tag{6}$$

$\boldsymbol{\theta}_{vis}$ 的优化仅与损失相对于 $\boldsymbol{v}_i$的梯度有关，而 $\boldsymbol{\theta}_{text}$ 的优化则与损失相对于 $\boldsymbol{t}_i$ 和 $\hat{\boldsymbol{t}}_i$的梯度有关。

梯度优化仅通过 $\boldsymbol{\theta}_{vis}$ 和 $\boldsymbol{\theta}_{text}$的变化影响特征嵌入。但我们首先指出超球面嵌入存在的问题。因此，我们假设优化可以直接影响嵌入位置，而不考虑由 $\boldsymbol{\theta}_{vis}$ 和 $\boldsymbol{\theta}_{text}$引起的梯度效应。我们推导出损失相对于 $\boldsymbol{v}_i$、 $\hat{\boldsymbol{t}}_i$ 和 $\boldsymbol{t}_i$的梯度：

$$\frac{\partial \mathcal{L}_{\text{T-HN}}(\boldsymbol{v}_i)}{\partial \boldsymbol{v}_i} = \hat{\boldsymbol{t}}_i - \boldsymbol{t}_i, \frac{\partial \mathcal{L}_{\text{T-HN}}(\boldsymbol{v}_i)}{\partial \hat{\boldsymbol{t}}_i} = \boldsymbol{v}_i, \frac{\partial \mathcal{L}_{\text{T-HN}}(\boldsymbol{v}_i)}{\partial \boldsymbol{t}_i} = -\boldsymbol{v}_i. \tag{7}$$

损失梯度方向与 $\boldsymbol{v}_i$、$\hat{\boldsymbol{t}}_i$ 和 $\boldsymbol{t}_i$的方向有关。梯度可以分解为两个分量。其中一个超球面切向方向分量



（a）损失　　（b）Δs

Fi图 5. VSE（MLP）在 Flick　　r30K 上的训练性能。

有助于模型优化，例如图4b中绿色箭头所示的梯度分量。另一个分量垂直于超球面的切向方向。该分量会将特征推离或拉向超球面，而经过 L_2 归一化后，它对模型优化不起作用。我们分别用 $\boldsymbol{g}_{\text{T-HN}}(\boldsymbol{v}_i)$、 $\boldsymbol{g}_{\text{T-HN}}(\hat{\boldsymbol{t}}_i)$ 和 $\boldsymbol{g}_{\text{T-HN}}(\boldsymbol{t}_i)$表示有助于模型优化的 $\boldsymbol{v}_i$、 $\boldsymbol{t}_i$ 和 $\boldsymbol{t}_i$ 的梯度分量。注意，由于神经网络使用梯度下降进行优化，$\boldsymbol{g}_{\text{T-HN}}(\boldsymbol{v}_i)$、 $\boldsymbol{g}_{\text{T-HN}}(\hat{\boldsymbol{t}}_i)$和 $\boldsymbol{g}_{\text{T-HN}}(\boldsymbol{t}_i)$的方向与 $\frac{\partial \mathcal{L}_{\text{T-HN}}(\boldsymbol{v}_i)}{\partial \boldsymbol{v}_i}$ 、 $\frac{\partial \mathcal{L}_{\text{T-HN}}(\boldsymbol{v}_i)}{\partial \hat{\boldsymbol{t}}_i}$ 和 $\frac{\partial \mathcal{L}_{\text{T-HN}}(\boldsymbol{v}_i)}{\partial \boldsymbol{t}_i}$ 的方向相反。我们用 $\beta(\cdot,\cdot)$ 表示两个特征向量之间的夹角，如图4a所示。 $\boldsymbol{g}_{\text{T-HN}}(\boldsymbol{v}_i)$、 $\boldsymbol{g}_{\text{T-HN}}(\hat{\boldsymbol{t}}_i)$和 $\boldsymbol{g}_{\text{T-HN}}(\boldsymbol{t}_i)$的模长可表示为：

$$
\begin{aligned}
\left\|\boldsymbol{g}_{\text{T-HN}}(\boldsymbol{v}_i)\right\| &= \left\|\frac{\partial \mathcal{L}_{\text{T-HN}}(\boldsymbol{v}_i)}{\partial \boldsymbol{v}_i}\right\| \cdot \sin\beta(\boldsymbol{v}_i, \hat{\boldsymbol{t}}_i - \boldsymbol{t}_i) \\
&= \left\|\hat{\boldsymbol{t}}_i - \boldsymbol{t}_i\right\| \cdot \left(\boldsymbol{v}_i^\top \hat{\boldsymbol{t}}_i - \boldsymbol{v}_i^\top \boldsymbol{t}_i\right), \\
\left\|\boldsymbol{g}_{\text{T-HN}}(\hat{\boldsymbol{t}}_i)\right\| &= \left\|\frac{\partial \mathcal{L}_{\text{T-HN}}(\boldsymbol{v}_i)}{\partial \hat{\boldsymbol{t}}_i}\right\| \cdot \sin\beta(\boldsymbol{v}_i, \hat{\boldsymbol{t}}_i) \\
&= \|\boldsymbol{v}_i\| \cdot \boldsymbol{v}_i^\top \hat{\boldsymbol{t}}_i = \boldsymbol{v}_i^\top \hat{\boldsymbol{t}}_i, \\
\left\|\boldsymbol{g}_{\text{T-HN}}(\boldsymbol{t}_i)\right\| &= \left\|\frac{\partial \mathcal{L}_{\text{T-HN}}(\boldsymbol{v}_i)}{\partial \boldsymbol{t}_i}\right\| \cdot \sin\beta(\boldsymbol{v}_i, \boldsymbol{t}_i) \\
&= \|-\boldsymbol{v}_i\| \cdot \boldsymbol{v}_i^\top \boldsymbol{t}_i = \boldsymbol{v}_i^\top \boldsymbol{t}_i.
\end{aligned}
\tag{8}
$$

我们观察到， $\boldsymbol{g}_{\text{T-HN}}(\boldsymbol{v}_i)$、 $\boldsymbol{g}_{\text{T-HN}}(\hat{\boldsymbol{t}}_i)$ 和$\boldsymbol{g}_{\text{T-HN}}(\boldsymbol{t}_i)$ 的模与正样本对 $\boldsymbol{v}_i^\top \boldsymbol{t}_i$ 以及负样本对 $\boldsymbol{v}_i^\top \hat{\boldsymbol{t}}_i$ 的相似度密切相关。

我们定义 $\Delta s = \boldsymbol{v}_i^\top \hat{\boldsymbol{t}}_i - \boldsymbol{v}_i^\top \boldsymbol{t}_i$。我们统计了 VSE（FC）训练过程中的 Δs，如图3d所示。在训练初期（训练轮次 0–7），Δs 趋近于 0。当$\Delta s \to 0$时， $\|\boldsymbol{g}_{\text{T-HN}}(\boldsymbol{v}_i)\| \to 0$。$\boldsymbol{\theta}_{vis}$ 的优化仅与 $\boldsymbol{v}_i$的梯度有关。因此，图像编码器的优化过程中存在梯度消失。另一方面， $\boldsymbol{\theta}_{text}$ 的优化与 $\boldsymbol{t}_i$和 $\boldsymbol{t}_i$有关。当 $\Delta s \to 0$时， $\boldsymbol{g}_{\text{T-HN}}(\hat{\boldsymbol{t}}_i)$ 与 $\boldsymbol{g}_{\text{T-HN}}(\boldsymbol{t}_i)$ 大小相等、方向相反，如图4c所示。优化 $\boldsymbol{\theta}_{text}$时，这两个梯度会相互抵消。因此，文本编码器的优化过程中也存在梯度消失。我们的梯度分析和实验结果得出了梯度消失发生条件的一致结论：

评估任务 →		图像到文本			文本到图像			RSUM
模型 ↓	损失 ↓	R@1	R@5	R@10	R@1	R@5	R@10	
VSE（FC）	$\mathcal{L}_{\text{Triplet}}$	64.5	89.5	95.0	48.4	77.4	86.0	460.7
VSE（FC）	$\mathcal{L}_{\text{T-HN}}$	76.0	92.5	97.0	53.9	81.2	88.2	488.8
VSE（FC）	$\mathcal{L}_{\text{T-SelHN}}$	77.1	93.6	97.1	56.3	82.8	89.2	**496.1**
VSE（MLP）	$\mathcal{L}_{\text{Triplet}}$	59.1	86.0	92.8	48.4	77.2	85.9	449.4
VSE（MLP）	$\mathcal{L}_{\text{T-HN}}$	44.0	70.0	79.1	31.9	61.6	72.9	359.4
VSE（MLP）	$\mathcal{L}_{\text{T-SelHN}}$	75.5		96.9	55.9		89.3	**492.8**
RVSE（MLP）	$\mathcal{L}_{\text{Triplet}}$	64.8	89.3	95.0	49.9	78.5	87.1	464.6
RVSE（MLP）	$\mathcal{L}_{\text{T-HN}}$	72.4	92.7	96.8	54.1	80.9	87.7	484.6
RVSE（MLP）		77.4		97.3	56.8		89.7	**498.6**

表 1. 在 Flickr30K 测试集上使用不同损失函数训练的三个图文匹配模型的匹配性能。

当 $\Delta s \to 0$时，图文匹配模型的训练会受到梯度消失的影响。

其他模型上的实验结果也得出了相同的结论。VSE（MLP）模型的训练性能如图所示2c。使用 $\mathcal{L}_{\text{T-HN}}$ 进行训练会遭遇严重的梯度消失， Δs 始终趋近于 0，且 $\mathcal{L}_{\text{T-HN}}$ 下降非常缓慢。由于 VSE（MLP）模型的层数多于 VSE（FC），已有研究表明，随着层数增加，使用梯度下降法优化神经网络时更容易发生梯度消失 [14]。表 1 展示了使用不同损失函数训练的若干模型的匹配性能。由于梯度消失严重，使用 $\mathcal{L}_{\text{T-HN}}$ 训练的 VSE（MLP）模型匹配性能大幅下降。这表明，梯度消失会使模型无法充分发挥其表征能力，从而对模型的匹配性能产生负面影响。

3.3. 选择性困难负样本挖掘

$\mathcal{L}_{\text{Triplet}}$ 和 $\mathcal{L}_{\text{T-HN}}$ 各有优缺点。通过困难负样本挖掘， $\mathcal{L}_{\text{T-HN}}$能够取得比 $\mathcal{L}_{\text{Triplet}}$ 更好的匹配性能。与 $\mathcal{L}_{\text{T-HN}}$相比， $\mathcal{L}_{\text{Triplet}}$ 在训练过程中不会出现梯度消失。为了充分发挥困难负样本挖掘的优势并缓解模型训练中的梯度消失，我们提出了一种选择性困难负样本挖掘（SelHN）策略。SelHN会根据梯度消失情况决定是否挖掘困难负样本。以 V_i 为锚点，采用 SelHN 的三元组损失形式如下：

$$
\mathcal{L}_{\text{T-SelHN}}(V_i) = \begin{cases} \mathcal{L}_{\text{T-HN}}(V_i), & \Delta s(V_i) > \epsilon, \\ \mathcal{L}_{\text{Triplet}}(V_i), & \text{otherwise}, \end{cases} \tag{9}
$$

其中

$$
\Delta s(V_i) = \left| s(V_i, \hat{T}_i) - s(V_i, T_i) \right|, \tag{10}
$$

$$
\mathcal{L}_{\text{T-HN}}(V_i) = [s(V_i, \hat{T}_i) - s(V_i, T_i) + \lambda]_+, \tag{11}
$$

$$
\mathcal{L}_{\text{Triplet}}(V_i) = \frac{1}{B} \sum_{j=1, i \neq j}^{B} [s(V_i, T_j) - s(V_i, T_i) + \lambda]_+, \tag{12}
$$

ϵ 是用于判断损失是否应当挖掘困难负样本的阈值。 ϵ 取一个较小的正数。在我们的实验中， ϵ 取值为 0.01。当$\Delta s(V_i) > \epsilon$时，模型训练不易受到梯度消失的影响，并且通过困难负样本挖掘可以提升模型的判别能力。当$\Delta s(V_i) < \epsilon$时，使用 $\mathcal{L}_{\text{Triplet}}(V_i)$ 对模型进行优化，以增大 $\Delta s(V_i)$。如图3 和图5所示， $\mathcal{L}_{\text{T-SelHN}}$展现出优异的训练性能。以 $\mathcal{L}_{\text{T-SelHN}}$ 作为优化目标时，模型训练不会受到梯度消失的影响，并且能够比 $\mathcal{L}_{\text{Triplet}}$ 和 $\mathcal{L}_{\text{T-HN}}$更快地达到最佳匹配性能。

3.4. 残差视觉语义嵌入

SelHN能够缓解模型训练中的梯度消失，并充分利用困难负样本挖掘。表1 显示， $\mathcal{L}_{\text{T-SelHN}}$ 的匹配性能显著优于 $\mathcal{L}_{\text{Triplet}}$ 和 $\mathcal{L}_{\text{T-HN}}$ ，无论是在 VSE（FC）还是 VSE（MLP）上都是如此。然而，我们观察到，层数更多的 VSE（MLP）整体性能不如更简单的 VSE（FC）模型。这是因为随着模型层数增加，梯度在反向传播过程中更容易出现梯度消失。残差连接是缓解梯度消失的有效方法。表1 显示，采用 $\mathcal{L}_{\text{T-SelHN}}$ 进行 RVSE（MLP）训练可获得最佳性能。

为获得更好的匹配性能，我们构建了一个结合 SelHN 的残差视觉语义嵌入模型，记为 RVSE++。RVSE++ 的图像编码器采用图2c所示的 RVSE（MLP）结构。文本编码器采用更具代表性的 BERT 模型 [16] 来提取词级特征 $\{\boldsymbol{w}_i^n\}_{n=1}^N$。随后，我们使用一个全连接层将 $\{\boldsymbol{w}_i^n\}_{n=1}^N$ 的维度映射到 d ，并使用 GPO 将文本特征集合聚合为文本嵌入 $\boldsymbol{t}_i$。最后，我们使用余弦相似度计算图像与文本之间的匹配分数。RVSE++网络结构简单、参数量少且推理速度快。得益于我们的网络设计和 SelHN策略，RVSE++ 的表征能力得以充分发挥。后续实验从多个方面验证了 RVSE++的优势。

4. 实验

4.1. 数据集与实验设置

数据集。我们在两个基准数据集上评估 SelHN 和 RVSE++：Flickr30K [41] 和 MS-COCO [21]。Flickr30K 包含 31,000 张图像，每张图像都标注了 5 个句子。我们使用 1,000 张图像作为验证集，1,000 张作为测试集，其余 29,000 张用于训练。MS-COCO 包含 123,287 张图像，每张图像都关联 5 个句子。我们使用 113,287 张图像进行训练，5,000 张作为验证集，另有 5,000 张作为测试集。我们在 1,000 张测试图像上报告结果（平均值——

评估任务 →	图像到文本			文本到图像			RSUM
方法 ↓	R@1	R@5	R@10	R@1	R@5	R@10	
SCAN$_{(ECCV'18)}$ [17]	67.4	90.3	95.8	48.6	77.7	85.2	465.0
+ SelHN	73.4	↑	96.6	53.5	↑	87.9	**486.2**
BFAN$_{(MM'19)}$ [22]	68.1	91.4	95.9	50.8	78.4	85.8	470.4
+ SelHN	75.3	↑	97.1	55.2	↑	87.3	**489.1**
SGRAF$_{(AAAI'21)}$ [8]	77.8	94.1	97.4	58.5	83.0	88.8	499.6
+SelHN	80.0	95.1	98.2	59.8	↑	89.5	**507.1**

表2. 在 Flickr30K 测试集上将 SelHN 应用于现有图文匹配模型的匹配性能。

（按 5 折划分）以及 MS-COCO 的全部 5,000 张测试图像。

评估指标。在 Flickr30K 和 MS-COCO 上进行评估时，我们遵循 [9], ，采用 Recall@K (R@K)，并以 $K = \{1, 5, 10\}$ 作为该任务的评估指标。R@K 表示模型在前 K 个结果中返回正确项目的查询所占比例。我们遵循 [5] ，使用 RSUM 作为图文匹配模型整体性能的平均指标。RSUM 定义为文本到图像匹配和图像到文本匹配在 $K = \{1, 5, 10\}$ 处的召回率指标之和。

实现细节。 我们的所有实验均使用 PyTorch 在 NVIDIA GeForce RTX 3090 图形处理器上进行。对于 SelHN，两种模型和两个数据集的超参数均设为 $\epsilon = 0.01$ 和 $\lambda = 0.2$ 。我们的 RVSE++使用 AdamW 训练 30 个训练轮次，两个数据集的批大小均为 128。模型的初始学习率在前 15 个训练轮次设为 0.0005，随后在最后 15 个训练轮次衰减 10 倍。更多实现细节和实验结果见附录。

4.2. 对现有图文匹配模型的改进

为了证明我们的 SelHN 优于现有的图文匹配模型，我们仅通过替换损失函数，在 SCAN [17],BFAN [22], 和 SGRAF [8] 上开展了实验。SCAN 和 BFAN 属于交叉注意力方法，而 SGRAF 属于 VSE 与交叉注意力结合方法。我们之前的实验和梯度分析主要基于视觉语义嵌入模型。本实验表明，我们得出的结论可以扩展到其他类型的模型。表2展示了这些模型在 Flickr30K 数据集上的改进情况。将损失函数替换为我们的 SelHN 后，三种基线方法的性能均得到提升。与原始模型相比，在 SCAN 和 BFAN 上使用 SelHN 的模型在 RSUM 指标上分别显著提升了 21.2% 和 18.7%。在更先进的 SGRAF 上，应用 SelHN 后，RSUM 也提升了 7.5%。我们的 SelHN 策略可以即插即用地应用于现有的图文匹配模型。实验结果表明，应用 SelHN 能够充分发挥模型的表征能力，并实现更好的匹配性能。

评估任务 →	图像到文本			文本到图像			RSUM	图像到文本			文本到图像			RSUM
损失 ↓	R@1	R@5	R@10	R@1	R@5	R@10		R@1	R@5	R@10	R@1	R@5	R@10	
$\mathcal{L}_{\text{Triplet}}$	64.5	89.5	95.0	48.4	77.4	86.0	460.7	41.4	72.5	83.3	30.6	60.0	73.0	360.7
$\mathcal{L}_{\text{T-HN}}$	76.0	92.5	97.0	53.9	81.2	88.2	488.8	54.3	82.3	90.2	37.3	67.6	79.0	410.7
SHN	74.0	93.4	97.4	55.3	82.2	88.8	491.0	50.4	79.8	88.7	35.8	66.1	78.2	399.1
SCT	75.8	93.6	97.1	55.3	82.0	88.6	492.3	54.6	82.4	90.2	36.5	67.1	78.9	409.7
$\mathcal{L}_{\text{T-SelHN}}$	77.1	93.6	97.1	56.3	82.8	89.2	**496.1**	54.6	82.7	90.5	38.0	68.5	80.1	**414.4**

表3. 与其他改进训练行为的方法在 Flickr30K 测试集和 MS-COCO 5K测试集上的性能比较。

评估任务 →		图像到文本			文本到图像		RSUM
模型 ↓	损失 ↓	R@1	R@5	R@10	R@1	R@5	R@10	
BFAN	$\mathcal{L}_{\text{T-HN}}$	68.1	91.4	95.9	50.8	78.4	85.8	470.4
BFAN	SSP	71.3	92.6	96.2	52.5	79.5	86.6	478.7
BFAN	AOQ	73.2	94.5	97.0	54.0	80.3	87.7	486.7
BFAN	Meta-SPN	72.5	93.2	96.7	53.3	80.2	87.2	483.1
BFAN	$\mathcal{L}_{\text{T-SelHN}}$	75.3	[illegible]	97.1	55.2	[illegible]	87.3	**489.1**
SGRAF	$\mathcal{L}_{\text{T-HN}}$	77.8	94.1	97.4	58.5	83.0	88.8	499.6
SGRAF	NCR	77.3	94.0	97.5	59.6	84.4	89.9	502.7
SGRAF		80.0	[illegible]	98.2	59.8	[illegible]	89.5	**507.1**

表4。在 Flickr30K 测试集上为图文匹配提出的其他损失函数的性能比较。

4.3. 与其他损失函数的比较

我们的 SelHN 专注于缓解图文匹配模型训练中的梯度消失。因此，我们将 SelHN 与其他改善训练行为的方法进行比较：

• **SHN**：半难负样本挖掘 [30] 不会优化困难负样本，只挖掘半难负样本 $s_n < s_p$ 用于优化。

• **SCT**：选择性对比三元组损失 [39] 使用对比损失优化困难负样本，并使用三元组损失优化其余样本。

SelHN 与上述损失在 VSE（FC）模型上的实验结果如表3所示。与这些损失函数相比，SelHN 提升了大多数评估指标。SHN 和 SCT 改善了训练行为，但它们放弃了对由困难负样本组成的三元组进行优化。困难负样本对于学习判别模型至关重要。SelHN 在不太可能发生梯度消失时挖掘困难负样本。因此，SelHN 不仅改善了训练行为，还充分发挥了困难负样本挖掘的作用。

另一方面，已有若干用于 ITM 的损失函数，因此我们将 SelHN 与这些损失进行比较：

• **SSP**：自相似多项式损失 [36] 是一种加权三元组损失，分别为正样本对和负样本对定义了权重函数。

• **Meta-SPN**：元自步网络 [35] 自动为图文匹配学习加权方案。

• **AOQ**：自适应离线五元组损失 [6] 从整个训练集挖掘离线负样本。

评估任务 →		图像到文本			文本到图像		RSUM
方法 ↓		R@1	R@5	R@10	R@1	R@5	R@10	
SCAN$_{(ECCV'18)}$ [17]		67.4	90.3	95.8	48.6	77.7	85.2	465.0
CAMP$_{(CVPR'19)}$ [34]		68.1	89.7	95.2	51.5	77.1	85.3	466.9
BFAN$_{(MM'19)}$ [22]		68.1	91.4	95.9	50.8	78.4	85.8	470.4
VSRN$_{(ICCV'19)}$ [18]		71.3	90.6	96.0	54.7	81.8	88.2	482.6
CVSE$_{(ECCV'20)}$ [32]		73.5	92.1	95.8	52.9	80.4	87.8	482.5
IMRAM$_{(CVPR'20)}$ [4]		74.1	93.0	96.6	53.9	79.4	87.2	484.2
GSMN$_{(CVPR'20)}$ [23]		76.4	94.3	97.3	57.4	82.3	89.0	496.8
SGRAF$_{(AAAI'21)}$ [8]		77.8	94.1	97.4	58.5	83.0	88.8	499.6
视觉语义嵌入∞ $_{(CVPR'21)}$ [5]		81.7	95.4	97.6	61.4	85.9	91.5	513.5
CMCAN$_{(AAAI'22)}$ [42]		79.5	95.6	97.6	60.9	84.3	89.9	507.8
NAAF$_{(CVPR'22)}$ [43]		81.9	96.1	98.3	61.0	85.3	90.6	513.2
RVSE++		83.6	[illegible]	98.6	64.3	[illegible]	93.0	**524.2**

表5。在 Flickr30K 测试集上与最先进的图文匹配方法的性能比较。

• **NCR**：噪声对应校正器 [15] 专注于利用噪声对应关系进行图文匹配学习。

为进行公平比较，我们采用上述各种损失函数论文中相同的模型，但将损失函数替换为我们的 SelHN。SelHN 与其他损失在 Flickr30K 上的实验结果如表 4 所示。与其他损失函数相比，SelHN 提升了大多数评估指标。SelHN 无需像 SSP 那样引入许多超参数来分配权重，也无需像 Meta-SPN 那样训练额外的网络来分配权重。与 AOQ 相比，SelHN 只需进行在线困难负样本挖掘，不会增加复杂度。

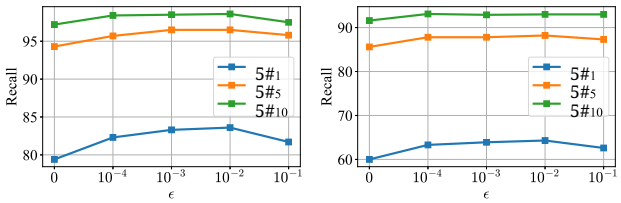
4.4. 与最先进方法的比较 **hods**

我们将所提出的 RVSE++ 与两个基准数据集上的最先进方法进行了比较。为确保比较公平，我们比较的所有方法均使用 BUTD 特征 [1]。表 5 展示了 RVSE++在 Flickr30K 上的定量结果。我们的 RVSE++在所有评估指标上都显著优于这些方法。具体而言，与近期最先进的方法 NAAF 相比，RVSE++ 在 RSUM 上取得了 11% 的相对提升。

MS-COCO 1K 测试集上的实验结果如表6所示。可以看出，我们的 RVSE++ 在大多数评估指标上均优于这些方法。与 NAAF 相比，RVSE++的 RSUM 相对提升了 5.2%。实验结果表明

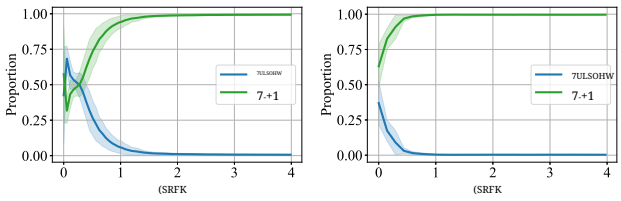
评估任务 →	图像到文本			文本到图像			RSUM
方法 ↓	R@1	R@5	R@10	R@1	R@5	R@10	
SCAN$_{(ECCV'18)}$ [17]	72.7	94.8	98.4	58.8	88.4	94.8	507.9
CAMP$_{(CVPR'19)}$ [34]	72.3	94.8	98.3	58.5	87.9	95.0	506.8
BFAN$_{(MM'19)}$ [22]	74.9	95.2	98.3	59.4	88.4	94.5	510.7
VSRN$_{(ICCV'19)}$ [18]	76.2	94.8	98.2	62.8	89.7	95.1	516.8
CVSE$_{(ECCV'20)}$ [32]	74.8	95.1	98.3	59.9	89.4	95.2	512.7
IMRAM$_{(CVPR'20)}$ [4]	76.7	95.6	98.5	61.7	89.1	95.0	516.6
GSMN$_{(CVPR'20)}$ [23]	78.4	96.4	98.6	63.3	90.1	95.7	522.5
SGRAF$_{(AAAI'21)}$ [8]	79.6	96.2	98.5	63.2	90.7	96.1	524.3
视觉语义嵌入∞ $_{(CVPR'21)}$ [5]	79.7	96.4	98.9	64.8	91.4	96.3	527.5
CMCAN$_{(AAAI'22)}$ [42]	81.2	96.8	98.7	65.4	91.0	96.2	529.3
NAAF$_{(CVPR'22)}$ [43]	80.5	96.5	98.8	64.1	90.7	96.5	527.2
RVSE++	81.6		98.8	66.6		96.6	**532.4**

表6. 在 MS-COCO 1K 测试集上与最先进的图文匹配方法进行性能比较。



(a) 图像到文本　　(b) 文本到图像

图6. 使用 RVSE（MLP）模型在 Flickr30K 测试集上评估不同参数配置的效果。



(a) Flickr30K　　(b) MS-COCO

图7. 训练过程中损失函数的选择。纵轴表示 $\mathcal{L}_{\text{Triplet}}$ 和 $\mathcal{L}_{\text{T-HN}}$ 占 $\mathcal{L}_{\text{T-SelHN}}$的比例。

我们的 RVSE++具有有效性。RVSE++模型的残差设计以及 SelHN 策略对梯度反向传播的保障，使 RVSE++模型能够充分发挥模型的表征能力。

4.5. 消融研究

我们的 SelHN 中有一个可调节的超参数阈值 ϵ。我们使用 RVSE++ 模型，在 Flickr30K 上测试了几种参数设置。 $\epsilon = 0$表示模型使用 $\mathcal{L}_{\text{T-HN}}$进行训练。如图 6所示，当 ϵ 处于 10^{-4} 至 10^{-1}的范围内时，模型的匹配性能优于 $\epsilon = 0$。总体而言，图文匹配模型的匹配性能不会随 ϵ发生显著变化。这表明我们提出的 SelHN 对超参数并不敏感，且便于应用于各种图文匹配模型。具体而言，在图 6a中，

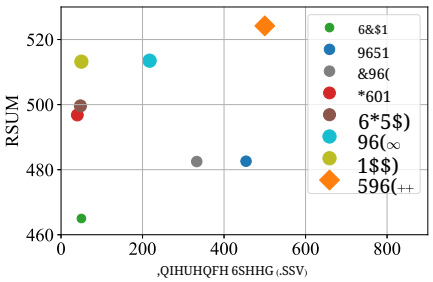


图8. Flickr30K 测试集上的匹配性能与速度对比。Kpps 表示每秒计算的图文对匹配分数数量。

当 ϵ 的值为 10^{-1}时，匹配性能会略有下降。因为当 ϵ 较大时，SelHN 会在优化过程中丢弃更多困难负样本，从而导致模型的匹配性能下降。

我们的 SelHN 根据梯度消失条件决定是否挖掘困难负样本。图7展示了 RVSE 中损失函数的选择++ 模型训练。在训练初期，由于此时容易发生梯度消失，部分三元组不会挖掘困难负样本。使用三元组损失训练模型以提高 Δs后，大多数三元组会挖掘困难负样本，从而使模型能够学习到更具判别性的表示。

4.6. 效率分析

在实际应用场景中，匹配速度同样十分重要。因此，我们在 Flickr30K 上同时报告匹配性能和速度，以进行更全面的比较。我们将 RVSE++ 与几种当前最先进的图文匹配方法进行了比较。需要注意的是，这些方法的匹配速度是在相同实验设置下重新实现其开源代码后测得的。如图 8所示，可以看出，我们的 RVSE++在匹配速度和匹配性能两方面均领先于其他方法。因此，我们的 RVSE++在有效性和效率两个方面都优于这些方法。

5. 结论

本文针对图文匹配模型训练过程中的梯度消失问题提出了一种解决方案。具体而言，我们推导了训练过程中梯度消失的条件。为缓解梯度消失，我们提出了 SelHN 策略，根据该条件决定是否挖掘困难负样本。SelHN 可以即插即用地应用于现有图文匹配模型，从而改善其训练行为。为进一步确保梯度反向传播，我们构建了 RVSE++模型，其性能大幅超越最先进的方法。未来，我们计划探索该方法在更多视觉-语言跨模态任务上的训练性能。

本附录提供了正文中省略的实现细节和实验结果。

A. 实现细节

我们的所有实验均使用 PyTorch 在 NVIDIA GeForce RTX 3090 图形处理器上进行。对于 SelHN，超参数设置为 $\epsilon = 0.01$ 和 $\lambda = 0.2$，适用于两个数据集。VSE++ （全连接层）、VSE（MLP）和 RVSE（MLP）使用 AdamW [24] 训练 20 个训练轮次，两个数据集的批大小均为 128。嵌入维度 d 设为 1,024。模型的学习率设为 0.0005。RVSE++ 使用 AdamW 训练 30 个训练轮次，两个数据集的批大小均为 128。嵌入维度 d 设为 1,024。模型的初始学习率在前 15 个训练轮次设为 0.0005，随后在最后 15 个训练轮次衰减 10 倍。

VSE（MLP）、RVSE（MLP）和 RVSE++ 中使用的双层多层感知机（MLP）结构如图 9所示。该 MLP 是一种瓶颈结构，由两个批量归一化（BN）层和两个全连接（FC）层组成。MLP 的输入和输出维度均为嵌入维度 d。MLP 的隐藏层维度为 $d/2$。

B. 实验结果

B.1. 对现有图文匹配模型的改进

为证明我们的 SelHN 优于现有图文匹配模型，我们仅替换损失函数，分别在 SCAN [17],BFAN [22], 和 SGRAF [8] 上开展实验。SCAN 和 BFAN 属于交叉注意力方法，SGRAF 属于 VSE 与 CA 结合方法。此前的实验和梯度分析主要基于 VSE 模型。本实验表明，我们得出的结论也适用于其他类型的模型。表 7 展示了现有图文匹配模型在 Flickr30K 数据集上的改进情况。表 7报告了单模型和集成模型的匹配性能。将损失函数替换为我们的 SelHN 后，三种基线方法的性能均有所提升。无论是单模型还是集成模型，应用我们的 SelHN 都能提升模型的匹配性能。

表 8和表 9 展示了现有图文匹配模型在 MS-COCO1K 和 5K 测试集上的改进情况。应用我们的 SelHN 后，BFAN 和 SGRAF 在大多数评估指标上都有所提升。我们的 SelHN 策略可以即插即用于现有图文匹配模型。实验结果表明，应用 SelHN 能够充分发挥模型的表征能力，并取得更好的匹配性能。

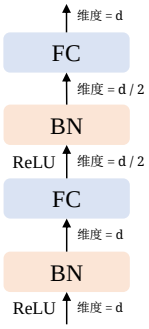


图 9. 多层感知机的网络结构。

评估任务 →	图像到文本			文本到图像			RSUM
方法 ↓	R@1	R@5	R@10	R@1	R@5	R@10	
SCAN 图像到文本平均值 [17]	67.9	89.0	94.4	43.9	74.2	82.8	452.2
+ $\mathcal{L}_{\text{T-SelHN}}$	66.3		95.2	45.5		83.7	**456.0**
SCAN 文本到图像平均值 [17]	61.8	87.5	93.7	45.8	74.4	83.0	446.2
+ $\mathcal{L}_{\text{T-SelHN}}$	70.3			53.3			**477.5**
SCAN$_{(ECCV'18)}$ [17]	67.4	90.3	95.8	48.6	77.7	85.2	465.0
+ SelHN	73.4		96.6				**486.2**
BFAN（相等） [22]	64.5	89.7	-	48.8	77.3	-	-
+ $\mathcal{L}_{\text{T-SelHN}}$	71.5		96.0			85.7	**477.8**
BFAN（概率） [22]	65.5	89.4	-	47.9	77.6	-	-
+ $\mathcal{L}_{\text{T-SelHN}}$	71.8		95.9	52.5		85.6	**476.8**
BFAN$_{(MM'19)}$ [22]	68.1	91.4	95.9	50.8	78.4	85.8	470.4
+ SelHN	75.3		97.1	55.2		87.3	**489.1**
SAF [8]	73.7	93.3	96.3	56.1	81.5	88.0	488.9
+ $\mathcal{L}_{\text{T-SelHN}}$	76.0		97.1	57.9	83.5	89.2	**498.5**
SGR [8]	75.2	93.3	96.6	56.2	81.0	86.5	488.8
+ $\mathcal{L}_{\text{T-SelHN}}$	77.8		98.0	58.2		87.2	**498.0**
SGRAF$_{(AAAI'21)}$ [8]	77.8	94.1	97.4	58.5	83.0	88.8	499.6
+ $\mathcal{L}_{\text{T-SelHN}}$	80.0						**507.1**

表7. 将 SelHN 应用于 Flickr30K 上现有图文匹配模型的匹配性能。

评估任务 →	图像到文本			文本到图像			RSUM
方法 ↓	R@1	R@5	R@10	R@1	R@5	R@10	
BFAN$_{(MM'19)}$ [22]	74.9	95.2	98.3	59.4	88.4	94.5	510.7
+ $\mathcal{L}_{\text{T-SelHN}}$	76.2		98.4	60.7		94.4	**513.4**
SGRAF$_{(AAAI'21)}$ [8]	79.6	96.2	98.5	63.2	90.7	96.1	524.3
+ $\mathcal{L}_{\text{T-SelHN}}$	79.9						**526.9**

表 8. 在 MS-COCO 1K 测试集上，将 SelHN 应用于现有图文匹配模型的匹配性能。

B.2. 与其他损失函数的比较

我们的 SelHN 专注于缓解图文匹配模型训练中的梯度消失。因此，我们将 SelHN 与其他能够改善训练行为的方法进行比较。

半难负样本挖掘（SHN） [30]不会优化困难负样本，而只挖掘带有 $s_n < s_p$ 的半难负样本进行优化。应用于图文匹配的 SHN 三元组损失包含图像到文本

评估任务 →	图像到文本			文本到图像			RSUM
方法 ↓	R@1	R@5	R@10	R@1	R@5	R@10	
BFAN(MM'19) [22]	52.9	82.8	90.6	38.3	67.8	79.3	411.7
+ $\mathcal{L}_{\text{T-SelHN}}$	55.6		90.6	39.3	68.0	78.6	**414.3**
SGRAF(AAAI'21) [8]	57.8	-	91.6	41.9	-	81.3	-
+ $\mathcal{L}_{\text{T-SelHN}}$	59.8		92.4	43.8		82.7	**436.8**

表 9. 在 MS-COCO 5K测试集上，将 SelHN 应用于现有图文匹配模型的匹配性能。

评估任务 →	图像到文本			文本到图像			RSUM
损失 ↓	R@1	R@5	R@10	R@1	R@5	R@10	
三元组	64.5	89.5	95.0	48.4	77.4	86.0	460.7
HN	76.0	92.5	97.0	53.9	81.2	88.2	488.8
SHN	74.0	93.4	97.4	55.3	82.2	88.8	491.0
SCT	75.8	93.6	97.1	55.3	82.0	88.6	492.3
$\mathcal{L}_{\text{T-SelHN}}$	77.1	93.6	97.1	56.3	82.8	89.2	**496.1**

表10. 与其他能够改善 MS-COCO1K 测试集上训练行为的方法进行性能比较。

和文本到图像两个方向，可表示为：

$$\mathcal{L}_{\text{T-SHN}} = \sum_{i=1}^{B} \Big([s(V_i, \hat{T}_i) - s(V_i, T_i) + \lambda]_+ + [s(\hat{V}_i, T_i) - s(V_i, T_i) + \lambda]_+ \Big), \tag{13}$$

其中

$$\begin{aligned} \hat{T}_i &= \arg \max_{j=1, s(V_i, T_j) < s(V_i, T_i)}^{B} s(V_i, T_j) \\ \hat{V}_i &= \arg \max_{j=1, s(V_j, T_i) < s(V_i, T_i)}^{B} s(V_j, T_i), \end{aligned} \tag{14}$$

是半难负样本。

选择性对比三元组损失（SCT） [39]使用对比损失优化困难负样本，并使用三元组损失优化其余样本。以 V_i为锚点，SCT 的形式如下：

$$\mathcal{L}_{\text{SCT}}(V_i) = \begin{cases} \mathcal{L}_{\text{T-HN}}(V_i), & s(V_i, \hat{T}_i) < s(V_i, T_i), \\ s(V_i, \hat{T}_i), & \text{otherwise}, \end{cases} \tag{15}$$

其中

$$\mathcal{L}_{\text{T-HN}}(V_i) = [s(V_i, \hat{T}_i) - s(V_i, T_i) + \lambda]_+. \tag{16}$$

在 MS-COCO 1K 测试集上，SelHN 与上述损失函数在 VSE（FC）模型上的实验结果如表10所示。与这些损失函数相比，SelHN 提升了大多数评估指标。SHN 和 SCT 改善了训练行为，但它们放弃了对由困难负样本构成的三元组进行优化。困难负样本对于学习具有判别能力的模型至关重要。SelHN 会在不太可能发生梯度消失时挖掘困难负样本。因此，SelHN 不仅改善了训练行为，还充分发挥了困难负样本挖掘的作用。

评估任务 →	图像到文本			文本到图像			RSUM
方法 ↓	R@1	R@5	R@10	R@1	R@5	R@10	
SCAN(ECCV'18) [17]	50.4	82.2	90.0	38.6	69.3	80.4	410.9
BFAN(MM'19) [22]	52.9	82.8	90.6	38.3	67.8	79.3	411.7
VSRN(ICCV'19) [18]	53.0	81.1	89.4	40.5	70.6	81.1	415.7
IMRAM(CVPR'20) [4]	53.7	83.2	91.0	39.7	69.1	79.8	416.5
SGRAF(AAAI'21) [8]	57.8	-	91.6	41.9	-	81.3	-
视觉语义嵌入∞ (CVPR'21) [5]	58.3	85.3	92.3	42.4	72.7	83.2	434.3
CMCAN(AAAI'22) [42]	61.5	-	92.9	44.0	-	82.6	-
NAAF(CVPR'22) [43]	58.9	85.2	92.0	42.5	70.9	81.4	430.9
RVSE++	60.6		92.8	44.5		84.5	**443.4**

表11. 在 MS-COCO 5K测试集上与最先进的图文匹配方法的性能比较。

B.3. 与最先进的方法比较

MS-COCO 5K测试集上的实验结果如表11所示。可以看到，我们的 RVSE++在大多数评估指标上优于这些方法。与 NAAF 相比，RVSE++在所有评估指标上均表现更优，RSUM 相对提升了 12.5%。实验结果证明了我们的 RVSE++的有效性。RVSE++ 模型的残差设计，以及 SelHN 策略对梯度反向传播的保障，使 RVSE++ 模型能够充分发挥其表征能力。

参考文献

[1]Peter Anderson、Xiaodong He、Chris Buehler、Damien Teney、Mark Johnson、Stephen Gould 和 Lei Zhang。用于图像描述和视觉问答的自底向上和自顶向下注意力。发表于 CVPR，2018年。3，7[2] Dzmitry Bahdanau、Kyung Hyun Cho 和 Yoshua Bengio。通过联合学习对齐与翻译实现神经机器翻译。发表于 ICLR，2015年。3[3] Paola Cascante-Bonilla、Hui Wu、Letao Wang、Rogerio S Feris 和 Vicente Ordonez。SimVQA：探索用于视觉问答的模拟环境。发表于 CVPR，2022年。2[4] Hui Chen、Guiguang Ding、Xudong Liu、Zijia Lin、Ji Liu 和 Jungong Han。IMRAM：利用循环注意力记忆进行跨模态图文检索的迭代匹配。发表于 CVPR，2020年。1，2，7，8，10[5] Jiacheng Chen、Hexiang Hu、Hao Wu、Yuning Jiang 和 Changhu Wang。学习视觉语义嵌入的最佳池化策略。发表于 CVPR，2021年。2，3，6，7，8，10[6] Tianlang Chen、Jiajun Deng 和 Jiebo Luo。用于图文匹配的自适应离线五元组损失。发表于 ECCV，2020年。2，7[7] Kyunghyun Cho、Bart van Merrienboer、C¸aglar G¨ulc¸ehre、Dzmitry Bahdanau、Fethi Bougares、Holger Schwenk 和 Yoshua Bengio。使用 RNN 编码器-解码器学习用于统计机器翻译的短语表示。发表于 EMNLP，2014年。2[8] Haiwen Diao、Ying Zhang、Lin Ma 和 Huchuan Lu。用于图文匹配的相似性推理与过滤。发表于 AAAI，2021年。1，2，4，6，7，8，9，10[9] Fartash Faghri、David J Fleet、Jamie Ryan Kiros 和 Sanja Fidler。VSE++：利用困难负样本改进视觉语义嵌入。发表于 BMVC，2018年。1，2，4，6[10] Andrea Frome、Greg S Corrado、Jon Shlens、Samy Bengio、Jeff Dean、Marc'Aurelio Ranzato 和 Tomas Mikolov。DeViSE：一种深度视觉语义嵌入模型。发表于 NeurIPS，2013年。2[11] Sonam Goenka、Zhaoheng Zheng、Ayush Jaiswal、Rakesh Chada、Yue Wu、Varsha Hedau 和 Pradeep Natarajan。FashionVLP：结合反馈进行时尚检索的视觉语言 Transformer。发表于 CVPR，2022年。2[12] Vipul Gupta、Zhuowan Li、Adam Kortylewski、Chenyu Zhang、Yingwei Li 和 Alan Yuille。SwapMix：诊断并正则化视觉问答中过度依赖视觉上下文的问题。发表于 CVPR，2022年。2[13] Raia Hadsell、Sumit Chopra 和 Yann LeCun。通过学习不变映射进行降维。发表于 CVPR，2006年。3[14] Kaiming He、Xiangyu Zhang、Shaoqing Ren 和 Jian Sun。用于图像识别的深度残差学习。发表于 CVPR，2016年。5[15] Zhenyu Huang、Guocheng Niu、Xiao Liu、Wenbiao Ding、Xinyan Xiao、Hua Wu 和 Xi Peng。利用噪声对应关系进行跨模态匹配学习。发表于 NeurIPS，2021年。7

[16] Jacob Devlin、Ming-Wei Chang、Kenton 和 Kristina Toutanova。Bert：用于语言理解的深度双向 Transformer 预训练。载于 NAACL，2019年。6[17] Kuang-Huei Lee、Xi Chen、Gang Hua、Houdong Hu 和 Xi-aodong He。用于图文匹配的堆叠交叉注意力。载于 ECCV，2018年。1、2、3、4、6、7、8、9、10[18] Kunpeng Li、Yulun Zhang、Kai Li、Yuanyuan Li 和 Yun Fu。用于图文匹配的视觉语义推理。载于 ICCV，2019 年。1 、 2 、 3 、 4 、 7 、 8 、 10[19] Yehao Li 、Yingwei Pan、Ting Yao 和 Tao Mei。理解并排序图像描述语义。载于 CVPR，2022年。2[20] Zheng Li、Caili Guo、Zerun Feng、Jenq-Neng Hwang 和 Xijun Xue。多视图视觉语义嵌入。载于 IJ-CAI，2022年。2[21] Tsung-Yi Lin、Michael Maire、Serge Belongie、James Hays、Pietro Perona、Deva Ramanan、Piotr Doll´ar 和 C Lawrence Zitnick。Microsoft COCO：上下文中的常见物体。载于 ECCV，2014年。6[22] Chunxiao Liu、Zhendong Mao、An-An Liu、Tianzhu Zhang、Bin Wang 和 Yongdong Zhang。聚焦注意力：用于图文匹配的双向焦点注意力网络。载于 ACM MM，2019年。6、7、8、9、10[23] Chunxiao Liu、Zhendong Mao、Tianzhu Zhang、Hongtao Xie、Bin Wang 和 Yongdong Zhang。用于图文匹配的图结构网络。载于 CVPR，2020年。1、2、7、8[24] Ilya Loshchilov 和 Frank Hutter。解耦权重衰减正则化。载于 ICLR，2019年。4、9[25] Haoyu Lu、Nanyi Fei、Yuqi Huo、Yizhao Gao、Zhiwu Lu 和 Ji-Rong Wen。COTS：用于跨模态检索的协作式双流视觉-语言预训练模型。载于 CVPR，2022年。2[26] Tomas Mikolov、Kai Chen、Greg Corrado 和 Jeffrey Dean。向量空间中词表示的高效估计。arXiv 预印本 arXiv:1301.3781，2013年。2[27] Hyun Oh Song、Yu Xiang、Stefanie Jegelka 和 Silvio Savarese。通过提升结构化特征嵌入进行深度度量学习。载于 CVPR，2016年。2、3[28] Leigang Qu、Meng Liu、Jianlong Wu、Zan Gao 和 Liqiang Nie。用于图像文本检索的动态模态交互建模。载于 SIGIR，2021年。2[29] Shaoqing Ren、Kaiming He、Ross Girshick 和 Jian Sun。Faster R-CNN：迈向实时目标检测的区域提议网络。NeurIPS，2015年。3[30] Florian Schroff、Dmitry Kalenichenko 和 James Philbin。FaceNet：用于人脸识别与聚类的统一嵌入。载于 CVPR，2015年。1、2、3、7、9[31] Paul Hongsuck Seo、Arsha Nagrani、Anurag Arnab 和 Cordelia Schmid。用于多模态视频描述的端到端生成式预训练。载于 CVPR，2022年。2[32] Haoran Wang、Ying Zhang、Zhong Ji、Yanwei Pang 和 Lin Ma。用于图文匹配的共识感知视觉语义嵌入。载于 ECCV，2020年。2、7、8[33] Sijin Wang、Ruiping Wang、Ziwei Yao、Shiguang Shan 和 Xilin Chen。用于关系感知图像文本检索的跨模态场景图匹配。载于 WACV，2020年。2

[34] Zihao Wang、Xihui Liu、Hongsheng Li、Lu S heng、Junjie Yan、Xiaogang Wang 和 Jing Shao。Camp：用于文本-图像检索的跨模态自适应消息传递。载于 CVPR，2019年。7、8[35] Jiwei Wei、Xing Xu、Zheng Wang 和 Guoqing Wang。用于跨模态匹配的元自步学习。载于 ACM MM，2021年。2、7[36] Jiwei Wei、Xing Xu、Yang Yang、Yanli Ji、Zheng Wang 和 Heng Tao Shen。用于跨模态匹配的通用加权度量学习。载于 CVPR，2020年。2、7[37] Jiwei Wei、Yang Yang、Xing Xu、Xiaofeng Zhu 和 Heng Tao Shen。用于跨模态检索的通用加权度量学习。TPAMI，2021年。2[38] Xing Xu、Tan Wang、Yang Yang、Lin Zuo、Fumin Shen 和 Heng Tao Shen。用于图文匹配的语义一致性交叉模态注意力。TNNLS，2020年。2[39] Hong Xuan、Abby Stylianou、Xiaotong Liu 和 Robert Pless。困难负样本虽难，但很有用。载于 ECCV，2020年。2、3、7、10[40] Shiyang Yan、Li Yu 和 Yuan Xie。用于图文匹配的离散-连续动作空间策略梯度注意力。载于 CVPR，2021年。2[41] Peter Young、Alice Lai、Micah Hodosh 和 Julia Hocken-maier。从图像描述到视觉指称：用于事件描述语义推理的新型相似度度量。TACL，2014年。2、6[42] Huatian Zhang、Zhendong Mao、Kun Zhang 和 Yongdong Zhang。展现你的信心：用于图文匹配的跨模态置信度感知网络。载于 AAAI，2022年。7、8、10[43] Kun Zhang、Zhendong Mao、Quan Wang 和 Yongdong Zhang。用于图像-文本匹配的负样本感知注意力框架。载于 CVPR，2022年。1、2、4、7、8、10[44] Qi Zhang、Zhen Lei、Zhaoxiang Zhang 和 Stan Z Li。用于图像文本检索的上下文感知注意力网络。载于 CVPR，2020年。2[45] Zhedong Zheng、Liang Zheng、Michael Garrett、Yi Yang、Mingliang Xu 和 Yi-Dong Shen。带实例损失的双路径卷积图像-文本嵌入。TOMM，2020年。2