TGCA-PVT：用于表情包情感识别的主题引导上下文感知金字塔视觉变换器

摘要

在线聊天已成为我们日常互动的一个重要方面，表情包成为比纯文本更生动地传达情感的流行工具。传统的图像情感识别侧重于全局特征，而表情包情感识别需要结合全局和局部特征，以及文本等附加模式。为了解决这个问题，本文引入了主题 ID 引导变压器方法，以方便对表情包进行更细致的分析。考虑到每个表情包都会有一个主题，并且相同主题的表情包将具有相同的对象，因此我们引入一个主题 ID，并将具有相同主题 ID 的表情包视为主题上下文。我们的方法包含新颖的主题引导上下文感知模块和主题引导注意力机制，能够从共享相同主题 ID 的表情包中提取全面的主题上下文特征，从而显着提高情绪识别的准确性。此外，我们集成了频率线性注意模块，以利用频域信息更好地捕获表情包的对象信息，并集成了局部增强的重新注意机制，以改进局部特征提取。同时复现了分层交叉熵损失函数，在 SER30K 数据集上进行实验分析。

关键词：表情包情绪识别；多模态学习；情绪分析

1 引言

随着网络和社交软件的普及，人们开始经常进行线上聊天。在线聊天过程中，人们除了使用基本的文字，还经常使用表情包这种信息丰富的图像来进行更好的表达。表情包作为图片和文字的有效载体 [5]，在聊天过程中往往起到一图胜千言的作用，能够有效反映用户的情绪，更好的了解表情包的情绪有助于我们对聊天的情感基调有更好的理解。因此表情包情感识别旨在分析表情包中的情感信息。表情包情感识别的进步可能推动其他研究领域，如仇恨检测 [9]、对话情感分析 [12]，也将有利于多个实际应用场景，如推荐系统 [4]。

图像情感识别作为计算机视觉中的一项技术，近年来有许多方法提出，从早期的传统方法到现在基于深度学习的方法，从手工设计各种视觉特征到使用通用网络架构再到注意力机制的引入，基于图像的情感识别方法取得了令人鼓舞的结果。然而与普通图像相比，表情包具有多种视觉元素，如动画人物、面部表情、文字贴图等，有效融合这些元素并正确理解是一项艰难的任务，因为表情包情感识别信息复杂且缺乏足够的数据，因此对表情包进行情感识别的研究较少。Liu 等人推出了第一个大规模表情包情感识别数据集 SER30K [13]，为表情包情感识别研究提供了基础和便利。

表情包作为图像的一个分支，除了本身具有全局和局部特征，它可能还会有文本信息。同时，表情包具有主题属性，主题可以找到同一主题下的表情包，进行全局特征提取，从而设

定情感基调；另一方面，同一主题下的表情包具有不同的情感标签，这取决于它们的细节不同，也就是它们的局部特征。因此，充分探索全局和局部特征并有效结合对于表情包情感识别来说至关重要。

在此背景下，本文将提出了主题引导上下文感知网络来捕获表情包的全局和局部特征，并和文本信息特征进行融合预测情感。每张表情包都具有主题 ID，将具有相同主题 ID 的表情包的特征视为表情包的上下文信息。本文设计了线性注意模块（FLA-Module）以增强表情包的对象特征，主题引导上下文感知模块（TGCA-Module）用于捕获表情包的上下文信息作为全局特征，局部增强再注意（LERA-Module）作为增强图像局部特征的模块，并将上述三种模块引入到预训练的金字塔视觉变换器（PVT）模型中，最后将在特征融合模块引入主题引导注意（TG-Attention）来根据主题 ID 增强全局特征。该模型成为 TGCA-PVT，在大规模表情包情感识别数据集 SER30K 上进行了大量的实验去证明该模型的有效性和可解释性。

2 相关工作

2.1 图像情感识别

图像情感识别旨在分析图像中的情感信息，以便更好地理解图像如何引起观众的情绪状态。图像情感识别方法主要分为两种实现方式，早期的传统方法是使用基于心理学和艺术理论设计的特征来代表图像的情感内容 [14]。另一种是基于深度学习的方法，因为端到端的卷积神经网络（CNN）在计算机视觉任务的成功应用，研究人员开始尝试将 CNN 应用于图像情感识别中 [19]，最初由于情感标签的人工标注昂贵，规模较小，早期大多数的方法是结合从大规模通用数据集学到的网络权重，在该任务上微调来进行情感预测 [1]。Rao 等人 [11] 提出了一个可以学习图像多级深层表示的网络结构（MldrNet），所提出的网络通过结合图像语义、图像美学和低层视觉特征多层表示来预测图像情感。但早期大多数基于 CNN 的图像情感分类方法是从整个图像中提取整体特征，忽略了可以利用局部区域来进行情感预测。后来研究人员开始考虑图像的局部信息 [8]，Sun 等人 [17] 使用现成的对象工具生成候选对象，进行筛选的候选对象与神经网络连接以发现情感区域，与整张图像特征结合后产生最终预测。Yang 等人 [16] 提出了三分支的网络结构，从图像中选择特定的情感刺激（即颜色、物体、面部），从不同的刺激中提取不同的情感特征。这些方法虽然能有效地捕捉图像的情感特征，但由于表情包的特殊性，并不完全适用于表情包的情感识别。

2.2 表情包和情感

在线聊天中，表情包与表情符号相比更具表现力，它包括多样化的动画、多个对象和文本，它可以在聊天中帮助人们表达情感和信息。由于其表达能力，表情包在情感强度、积极性和亲密性方面更具优势 [7]，但它们也有缺点，即情感误解 [2]。由于表情包必须作为单独的消息发送，因此对表情包的情感误解通常比表情符号和表情符号更常见。由此可见，实现表情包的准确情绪识别是一项非常具有挑战性的任务。早期对表情包的研究主要在推荐系统，结合上下文信息与表情包进行分析。由于缺乏足够的数据，专门对表情包进行情感识别的研究较少。Liu 等人推出第一个大规模表情包情感识别数据集 SER30K [13]，并搭建了网络 LORA，通过捕获图像的全局和局部特征，融合表情包中的文本内容预测情绪。

3 本文方法

我们使用预训练的 PVT 模型和 Bert 模型作为骨干网络来设计 TGCA-PVT。针对表情符号情感识别的特点，引入了多个模块，进一步提高情感识别的准确率。本文提出的 TGCA-PVT 的整个框架如图 1所示。

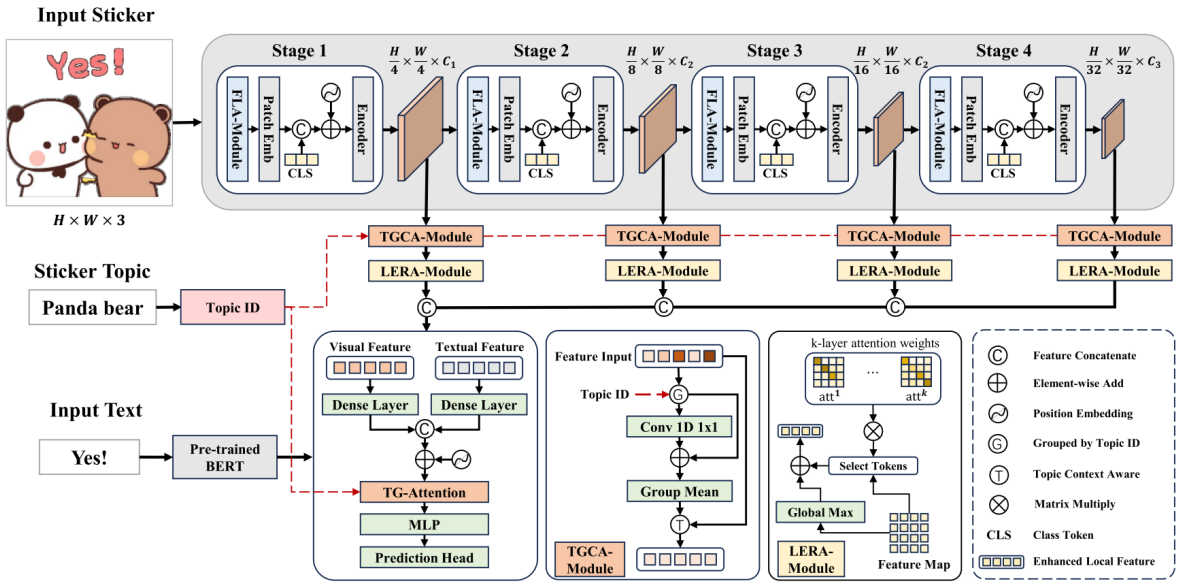

图 1. 提出的 TGCA-PVT 模型概述

3.1 主干网络

考虑到 PVT 模型在捕获图像的全局关系信息方面的出色性能 [15]，本文采用包含四个阶段的预训练 PVT 模型作为骨干视觉编码器。首先，给定一个贴纸输入 $X \in \mathbb{R}^{3\times H\times W}$，其中 H 和 W 代表贴纸的高度和宽度，PVT 将首先使用 Conv2d 层对其进行投影，然后将其展平为补丁特征序列 $X_{\text{patch}} \in \mathbb{R}^{N\times C}$，其中 $N = \frac{HW}{P^2}$，P 表示补丁大小，这个过程也称为补丁嵌入。在 PVT 的每个阶段开始前，都会引入 FLA 模块（将在 3.2 节进行介绍），以在将表情包馈送到补丁嵌入之前增强表情包的频率特征。然后，我们在补丁特征之前连接一个额外的 CLS 标记 $X_{CLS} \in \mathbb{R}^{1\times C}$，以更好地捕获全局和局部补丁。更重要的是，我们还添加了位置嵌入 X_{pos} 来处理输入补丁标记的位置不可知问题，然后再将其馈送到 PVT 模型的编码器。因此，每级编码器的输入可以表示为公式 1

$$X_{in}^{l} = Concat(X_{CLS}^{l}, X_{patch}^{l}) + X_{pos}^{l}, X_{in}^{l} \in \mathbb{R}^{(N+1)\times C} \tag{1}$$

其中 X_{in}^{l} 表示第 l 阶段编码器的输入，$Concat$ 表示串联操作。PVT 模型的编码器在前三个阶段使用空间缩减注意力（SRA）来代替多头注意力（MHA）。与 MHA 类似，使用两个线性投影将 X_{in}^{l} 获取到查询、键和值嵌入中，公式 2至公式 5.

$$Q^{l} = W_{q}^{l} X_{in}^{l} + b_{q}^{l} \tag{2}$$

$$KV^{l} = W_{kv}^{l} X_{in}^{l} + b_{kv}^{l} \tag{3}$$

$$KV^{nl} = Reshape(KV^{l}) \tag{4}$$

$$K^l = KV'^l[0], V^l = KV'^l[1], \tag{5}$$

其中 $Q^l \in \mathbb{R}^{(N^l+1)\times C^l}$ 是查询嵌入，键嵌入和值嵌入是通过线性投影获得的，因此 $KV^l \in \mathbb{R}^{(N^l+1)\times(C^l\times 2)}$。然后将 KV^l 重塑为 $KV'^l \in \mathbb{R}^2 \times (N^l+1) \times C^l$ 。最后将其分离为 $K^l, V^l \in \mathbb{R}^{(N^l+1)\times C^l}$。$C^l$ 代表第 l 阶段的隐藏维度。空间缩减计算如公式 6所示：

$$SR(x) = LN(Reshape(X', S_l)W^S), \tag{6}$$

其中 X^l 是从 X_{in} 中去除 X_{CLS} 得到的空间特征。LN 表示层归一化，S_l 表示第 l 阶段的缩减率，W^s 是可学习的线性变换。SRA 的整个过程如公式 7和公式 8所示：

$$SRA(Q, K, V) = Concat(head_1, ..., head_{N_l})W^O \tag{7}$$

$$head_i = Attention(QW_i^Q, SR(K)W_i^K, SR(V)W_i^V) \tag{8}$$

其中 $head_i$ 表示第 i 个自注意力头，N_l 是第 l 阶段的自注意力头编号，$W^O \in \mathbb{R}^{C^l\times C^l}, W_i^Q, W_i^K, W_i^V \in \mathbb{R}^{C^l\times d_{head}}$，$d_{head}$ 是每个注意力头的维度。然后通过线性变换 W_{global} 来投影 CLS 令牌，以更好地捕获全局特征 x_g^l，然后将其与 SR 之后的 X 连接起来。注意头操作如公式 9所示：

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_{head}}})V \tag{9}$$

最后，第 l 阶段的输出特征 X_{out}^l 被馈送到 TGCAModule 和 LERAModule。如果表情包输入有文本信息，我们采用预训练的 BERT 模型来获得上下文化的单词表示 X_t。

3.2 FLA 模块

受 Frequency MLP [18] 的启发，我们设计了一个频率线性注意力模块 (FLA-module)，捕获频域信息以增强对象特征，这可以帮助区分对象与背景 [20]。所提出的 FLA 模块如图 2所示。给定输入 X，我们首先使用更快的快速傅立叶变换 (FFT) 来获取频率分量 $X_{fre} = x_r + jx_i$，这是一个复数特征。根据复数乘法，我们引入可学习权重 $W = W_r + jW_i$ 和 $B = B_r + jB_i$ 进行频率线性计算，如如公式 10所示：

$$h_r + jh_i = \sigma(x_rW_r - x_iw_i + B_r) + j\sigma(x_rW_i + x_iw_r + B_i) \tag{10}$$

然后，我们引入挤压和激励模块（SE 模块）[6] 来重新校准通道特征响应，如公式 11所示：

$$h_r' + jh_i' = SE(h_r) + jSE(h_i), \tag{11}$$

其中 SE 代表 SE 块。此外，我们添加原始频率信息，然后将它们叠加以获得复数作为输出特征。最后，我们使用快速傅里叶逆变换（IFFT）来获得输入 X 的频率增强特征 y。整个过程如公式 12所示：

$$y = IFFT(y_r + jy_i) = IFFT((x_r + h_r') + j(x_ih_i')) \tag{12}$$

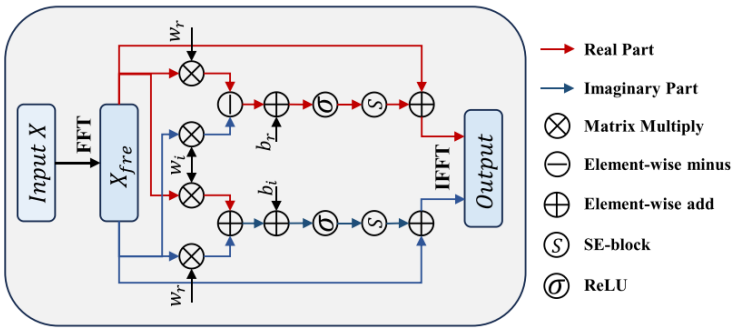

图 2. 提出的 FLA 模块的框架

3.3 TGCA 模块

为了更好地捕获主题信息，我们首先为每个表情包分配一个主题 ID，给定一批表情包输入 X^l_{out}，我们首先按主题 ID 对它们进行分组，然后内核大小为 1 的 Conv1d 层用于捕获同一组特征的通道相关性跳跃连接用于确保同一组中的特征之间的特异性。然后对于每个组，我们采用所有表情包的平均值作为组上下文特征，每个表情包将获得相应的上下文特征，并将它们叠加作为上下文感知特征。然后我们还利用 W_{global} 来获取全局上下文 x^l_{cg}。

利用编码器中 CLS 令牌和上下文感知特征的两个全局特征，我们进一步设计了一个主题上下文感知层，以受到注意力机制的启发来融合它们。这样，全局特征就可以有效地学习到具有相同主题 ID 的表情包的上下文特征。然后我们将每个阶段的全局特征堆叠为 X^{ll}_g 来表示表情包的全局表示。整个流程操作如公式 13和公式 14所示：

$$X^l_c = X^l_{out} + Conv1d(X^l_{out}) \tag{13}$$

$$X^{ll}_g = X^l_g + Softmax(X^l_g(X^l_{cg})^T)X^l_{cg} \tag{14}$$

3.4 LERA 模块

为了更好地捕捉区域信息和表情包情感之间的关系，提出了一种局部增强的重新注意模块（LERA 模块）。由于表情和姿势等不同的局部信息具有不同的尺度，因此在 PVT 编码器的每个阶段都使用 LERA 模块来开发多尺度特征。在每个编码器的注意力机制中，我们还可以获得补丁之间的注意力权重分布，如公式 15所示：

$$a_l = Softmax(\frac{QK^T}{\sqrt{d_{head}}}) \tag{15}$$

其中 a_l 是第 l 阶段对应的注意力权重。考虑到每个编码器都有 K_l 个注意力层，我们将每个注意力层中的注意力权重视为公式 16：

$$a^n_l = [a^{n1}_l, a^{n2}_l, ..., a^{n3}_l], i \in 1, 2, ..., K_l \tag{16}$$

其中 a^n_l 是第 l 阶段中的第 i 个注意力头。然后将同一阶段每个头的注意力权重相乘，就可以得到最终的注意力权重，如公式 17所示：

$$a^{final}_l = \prod_{l=1}^{L} a_l = \prod_{l=1}^{L} [a^1_l, a^2_l, ..., a^N_l], \tag{17}$$

其中 N 是每个注意力层中注意力头的数量。然后我们引入一个选择超参数 α 来选择具有最高注意力权重的补丁特征 h_{local}。同时，我们引入了补丁特征在补丁维度上的最大值来微调根据注意力机制选择的重要局部信息。我们首先使用全局最大池化来获得最大特征 h_{max}，然后将其扩展到与 h_{local} 相同的维度，并获得最终的局部特征，如公式 18所示：

$$h_{local}^{\prime l} = h_{local} + Softmax(h_{local}(h_{max})^T)h_{max}, \tag{18}$$

其中 $h_{local}^{\prime l}$ 表示第 l 阶段的最终局部特征。最后，我们使用线性变换来保持每个阶段的最终局部特征为相同的维度，并将它们堆叠为表情包的局部表示，设置为 X_{local}。此外，利用我们之前获得的全局表示 X_{global}，我们将 X_{local} 连接到 X_{global} 作为最终的视觉表示。

3.5 预测模块

对于给定的视觉特征和文本特征，我们首先使用两个线性变换将它们投影到同一维度。然后，我们将文本特征与视觉特征连接起来，并为其添加位置嵌入。我们设计了一个主题引导注意力可以更好地从具有相同主题 ID 的表情包中捕获特征，如图 3所示。通过线性变换获得 Q、K、V 后，我们利用 TCA 模块从 Q 中获取主题令牌并然后将其作为查询和 K, V 提供给 Softmax Attention 以获得主题特征。此外，我们使用 Q 作为查询，主题令牌作为键，主题特征作为值来进行 Softmax Attention。这样，我们可以更好地融合具有相同主题 ID 的表情包的表情包表示和上下文特征。最后，我们利用 MLP 和线性预测头来获得情感识别结果。

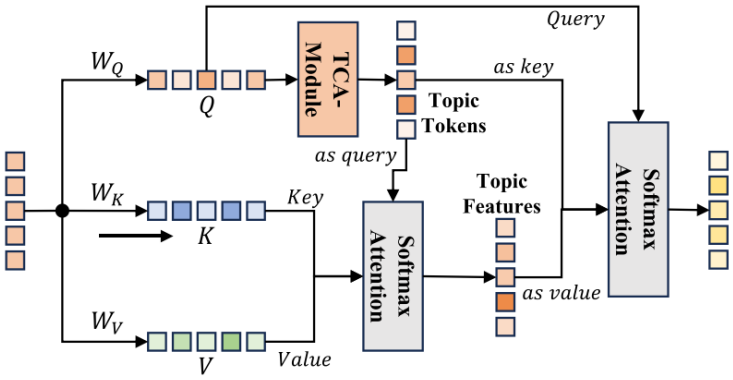

图 3. 提出的主题引导注意力框架

4 复现细节

4.1 与已有开源代码对比

本文参考了这篇文章的上一个工作，即 LORA 模型的开源代码https://github.com/nku-shengzheliu/SER30K，并在此基础上实现了 FLA 模块、TGCA 模块、LERA 模块和 TG Attention，搭建成为本文的 TGCA-PVT。

4.2 分层交叉熵损失函数

观察到该任务中的需要识别的情感有 7 类（即愤怒、厌恶、恐惧、快乐、中性、悲伤和惊讶），可以区分为三大类（即积极、消极和中性），参考 Yang 等人 [16] 提出的分层交叉熵

损失函数，将分层交叉熵损失函数用于该模型中。

在传统的分类任务中，交叉熵损失函数用来评估当前模型训练输出的概率分布与真实分布的差异情况。它刻画的是实际输出（概率）与期望输出（概率）的距离，也就是交叉熵的值越小，两个概率分布就越接近，则模型的训练效果越好，它经常用于分类问题中。但是交叉熵的前提是类别在空间上分离且彼此不相关。但根据心理学理论 [20]，情绪存在着极性的特点，如积极和消极，相同极性的情绪之间存在一定的联系，相反极性的情绪之间的距离更大。

传统的交叉熵损失计算中，只有预测正确的样本和预测错误的样本，但因为情绪具有极性的特点，本文将预测错误的样本分为简单的错误样本和难的错误样本。简单的错误样本的情况是情绪分类错误，但极性正确，如正确的情绪应该是兴奋，预测的情绪是娱乐，但它们都是积极的情绪；难的错误样本的情况是情绪分类错误，极性也预测错误，如正确的情绪应该是兴奋，预测的情绪是恐惧，前者的情绪极性是积极，后者的情绪极性为消极。难的错误样本中预测值和正确值的距离更大，因此应当对其增加更多的惩罚。具体来说，就是在情感标签上实现传统交叉熵损失的基础上提出了辅助极性损失，在区别简单的错误样本和难的错误样本的同时也对应增加不同程度的惩罚。我对分层交叉熵损失函数进行复现以满足本文任务的需求，分层交叉熵损失函数如图 4所示。

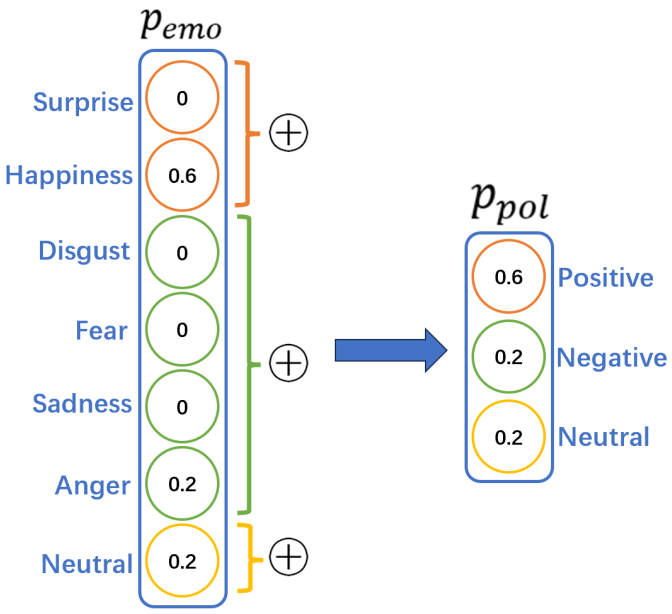

图 4. 分层交叉熵损失函数

4.3 实验环境设置

4.3.1 数据集

SER30K 数据集包含 30,739 个表情包，其中包括收集的 1,887 个表情包主题。每个表情包由三个注释者用情感标签注释，并且属于一个共同主题。同一主题内的表情包具有相似的主题特征。情感标签有 7 类（即愤怒、厌恶、恐惧、快乐、中性、悲伤和惊讶），里面有 5,886 张表情包标注有文本信息。表1提供了 SER30K 数据集不同情感标签中包含的样本的更多详细信息。

表 1. SER30K 数据集的详细信息

SER30K	Samples	Samples with text
Anger	2750	439
Disgust	211	17
Fear	826	58
Happiness	11255	1965
Neutral	10815	2832
Sadness	3359	346
Surprise	1523	229
Total	30739	5886

4.3.2 实现细节

本文以 7:1:2 的比例将 SER30K 数据集随机分为训练集、验证集和测试集。基于 Pytorch 框架 [10] 实现了方法。本工作中的所有实验均在 NVIDIA GTX 3090 上进行。对于文本输入，预训练的 Bert 模型获得的特征的最大序列设置为 30，特征维度为 768。对于预训练的 PVT 编码器对于视觉特征，我们采用在 ImageNet1k [3] 上预训练的 PVT-small [15]。所提出的 TGCA-PVT 使用学习率为 $10e^{-4}$ 的 SGD 算法进行优化。输入模型的表情包尺寸为 448x448 批量大小设置为 16，epoch 设置为 50。

5 实验结果分析

SER30K 数据集测试集的实验结果如表2所示。其中 TGCA-PVT 表示的是复现论文的 TGCA-PVT，TGCA-PVT+Loss 表示的是将损失函数改为分层交叉熵损失函数的 TGCA-PVT，原本的模型损失函数使用的是 LabelSmoothingCrossEntropy。可以看出使用了分层交叉熵损失函数的模型正确率与原本的模型相差不大，同时通过观察训练过程，发现使用了分层交叉熵损失函数的模型存在着较为严重的过拟合。可能的原因是 SER30K 数据集有类别不平衡的问题，LabelSmoothingCrossEntropy 通过在标签上施加平滑，使得训练过程中的目标标签不那么“严格”，这有助于防止过拟合。它通过减少训练时模型对某个单一标签的过度依赖来增强模型的泛化能力。我复现的分层交叉熵损失函数强制模型明确区分不同的情感类别（正面、负面、中性）。这种更严格的分类方式可能导致模型对训练数据过拟合，尤其是在数据集不平衡的情况下，某些类别的样本数远高于其他类别，分层交叉熵损失可能会导致模型过度关注那些频繁出现的类别，从而出现过拟合。

TGCA-PVT 和 TGCA-PVT+ 分层交叉熵损失函数的混淆矩阵如图 5和图 6所示。可以看出换成分层交叉熵损失函数后具体类别分类上仍无改进，可能的原因是数据集存在类别不平衡，模型会倾向于预测样本较多的类别，从而导致分类性能的下降，尤其是在细分类别上，分层交叉熵损失和情绪极性损失的联合优化可能导致优化目标不明确，使得模型在训练过程中出现过拟合或训练不稳定。

表 2. SER30K 数据集实验结果。每个情绪类别的精度以及总体分类精度。表中的所有值均以百分比表示。

Model	Precision on each emotion category							Accuracy
	Surprise	Happiness	Disgust	Fear	Sadness	Anger	Neutral	
TGCA-PVT	33.44	80.32	4.76	30.91	39.43	38.73	65.05	62.59
TGCA-PVT+Loss	29.84	80.85	7.14	27.27	40.62	35.82	63.99	62.02

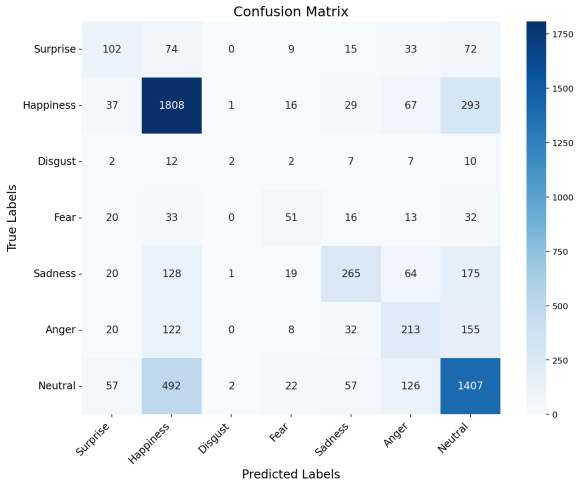

图 5. TGCA-PVT 在 SER30K 数据集上的混淆矩阵

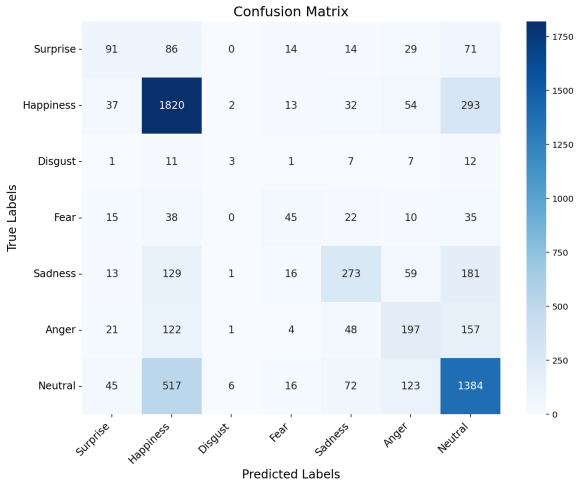

图 6. TGCA-PVT+ 分层交叉熵损失函数在 SER30K 数据集上的混淆矩阵

6 总结与展望

本文实现了主题引导上下文感知网络 TGCA-PVT，引入主题 ID 的概念来帮助模型学习同一主题表情包的共同主题特征，并基于预训练视觉编码器 PVT-small 和文本编码器 Bert 设计了几个模块来提高性能。

由于表情包情感识别类似于图像情感识别，因此在复现过程中我尝试使用分层交叉熵损失函数，但在训练过程中发现存在严重的过拟合现象，可能的原因是 SER30K 存在着较为严重的类别不平衡现象。在之后的研究中，将对数据集有更全面的认知，有助于增进对该任务的理解，并在此基础上对损失函数和网络结构做进一步的调整。

参考文献

[1] Víctor Campos, Amaia Salvador, Xavier Giró i Nieto, and Brendan Jou. Diving deep into sentiment: Understanding fine-tuned cnns for visual sentiment prediction. *Proceedings of the 1st International Workshop on Affect & Sentiment in Multimedia*, 2015.

[2] Yoonjeong Cha, Jongwon Kim, Sangkeun Park, Mun Yong Yi, and Uichin Lee. Complex and ambiguous: Understanding sticker misinterpretations in instant messaging. 2(CSCW), November 2018.

[3] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, K. Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[4] Shen Gao, Xiuying Chen, Li Liu, Dongyan Zhao, and Rui Yan. Learning to respond with your favorite stickers: A framework of unifying multi-modality and user preference in multi-turn dialog. *ACM Trans. Inf. Syst.*, 39(2), February 2021.

[5] Susan C. Herring and Ashley R. Dainas. "nice picture comment!" graphicons in facebook comment threads. In *Hawaii International Conference on System Sciences*, 2017.

[6] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7132–7141, 2018.

[7] Joon Young Lee, Nahi Hong, Soomin Kim, JongHwan Oh, and Joonhwan Lee. Smiley face: why we use emoticon stickers in mobile messaging. *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct*, 2016.

[8] Bing Li, Weihua Xiong, Weiming Hu, and Xinmiao Ding. Context-aware affective images classification based on bilayer sparse representation. *Proceedings of the 20th ACM international conference on Multimedia*, 2012.

[9] Shreyash Mishra, S Suryavardan, Parth Patwa, Megha Chakraborty, Anku Rani, Aishwarya N. Reganti, Aman Chadha, Amitava Das, Amit P. Sheth, Manoj Kumar Chinnakotla, Asif Ekbal, and Srijan Kumar. Memotion 3: Dataset on sentiment and emotion analysis of codemixed hindi-english memes. *ArXiv*, abs/2303.09892, 2023.

[10] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. *ArXiv*, abs/1912.01703, 2019.

[11] Tianrong Rao, Min Xu, and Dong Xu. Learning multi-level deep representations for image emotion classification. *Neural Processing Letters*, 51:2043 – 2061, 2016.

[12] Ying Tang and Khe Foon Timothy Hew. Emoticon, emoji, and sticker use in computer-mediated communication: A review of theories and research findings. *International Journal of Communication*, 13:27, 2019.

[13] Sheng tong Liu, Xin Zhang, and Jufeng Yang. Ser30k: A large-scale dataset for sticker emotion recognition. *Proceedings of the 30th ACM International Conference on Multimedia*, 2022.

[14] Wei-Ning Wang, Yinglin Yu, and Jian chao Zhang. Image emotional classification: static vs. dynamic. *2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No.04CH37583)*, 7:6407–6411 vol.7, 2004.

[15] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 548–558, 2021.

[16] Jingyuan Yang, Jie Li, Xiumei Wang, Yuxuan Ding, and Xinbo Gao. Stimuli-aware visual emotion analysis. *IEEE Transactions on Image Processing*, 30:7432–7445, 2021.

[17] Jufeng Yang, Dongyu She, Ming Sun, Ming-Ming Cheng, Paul L. Rosin, and Liang Wang. Visual sentiment prediction based on automatic discovery of affective regions. *IEEE Transactions on Multimedia*, 20:2513–2525, 2018.

[18] Kun Yi, Qi Zhang, Wei Fan, Shoujin Wang, Pengyang Wang, Hui He, Defu Lian, Ning An, Longbin Cao, and Zhendong Niu. Frequency-domain mlps are more effective learners in time series forecasting. *ArXiv*, abs/2311.06184, 2023.

[19] Quanzeng You, Jiebo Luo, Hailin Jin, and Jianchao Yang. Robust image sentiment analysis using progressively trained and domain transferred deep networks. *ArXiv*, abs/1509.06041, 2015.

[20] Yijie Zhong, Bo Li, Lv Tang, Senyun Kuang, Shuang Wu, and Shouhong Ding. Detecting camouflaged object in frequency domain. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4494–4503, 2022.