

# 基于半监督双流自注意力对抗图对比学习的跨被试脑电情感识别

## 摘要

脑电图 (EEG) 作为一种客观的情感识别工具, 具有广泛的应用前景。然而, 标注数据的稀缺性仍然是该领域的一大挑战, 限制了基于 EEG 的情感识别技术的广泛应用。本实验尝试了一种半监督的双流自注意对抗图对比学习框架, 旨在解决跨被试 EEG 情感识别中的标注数据稀缺问题。该框架包含两个并行的流结构, 用于提取非结构化和结构化 EEG 特征。在非结构化流中, 采用半监督的多领域自适应方法, 以减少标注源领域、未标注源领域和未知目标领域之间的分布差异。结构化流则提出了一种图对比学习方法, 从多个 EEG 通道中提取有效的基于图的特征表示, 且采用半监督的方式进行学习。此外, 提出了一种自注意力融合模块, 用于特征融合、样本选择和情感识别, 能够突出与情感相关的 EEG 特征, 并选择那些距离目标领域较近的标注源领域样本。本实验在 SEED-IV 数据集上进行, 并采用半监督的跨被试留一被试法交叉验证作为评估方法。通过与现有方法的比较, 实验结果显示, 所复现的 DS-AGC 模型在不同不完全标注条件下性能稳定, 在标注数据极端稀缺的条件下, 准确率也能到达 50% 以上, 从而证明了该框架在处理跨个体 EEG 情感识别中的标注数据稀缺问题上的有效性。

**关键词:** EEG 信号; 情绪识别; 半监督; 图对比学习

## 1 引言

情感计算 (Affective Computing) 是一个跨学科领域, 它结合了计算机科学、认知科学和神经科学, 旨在通过分析人的生理和行为信号来识别和模拟人的情感状态。脑电图 (EEG) 作为一种直接测量大脑活动的技术, 在情感识别领域中扮演着重要角色。随着深度学习技术的发展, 基于 EEG 的情感识别研究取得了显著进展, 但仍面临着一些挑战, 尤其是在数据标注方面。现有的研究主要依赖于监督学习方法, 这些方法需要大量的高质量标注数据进行模型训练。然而, 获取这些数据既费时又昂贵, 且过程复杂。

为了减轻标注数据的负担, 研究者开始探索半监督学习方法, 这种方法结合少量标注数据和大量未标注数据来构建模型。这种方法不仅可以提高模型的泛化能力, 还可以减轻标注工作的负担。然而, 在半监督 EEG 情感识别中, 一个关键挑战是开发能够有效利用标注和未标注数据以改善模型学习的算法。此外, 以往的方法在特征提取时忽略了不同 EEG 通道之间的复杂关系, 导致模型缺乏丰富的结构信息。

主要有以下两个研究问题:

- (1) 如何有效利用未标记数据

在基于 EEG 的情感识别领域，如何有效地利用未标注数据是一个重要的研究问题。尽管已有研究尝试使用半监督学习方法，但它们在构建稳健的域分类器方面存在限制，无法有效区分标注的源域、未标注的源域和未知的目标域。这种局限性导致模型在适应未标注源域时表现不佳，尤其是在源数据主要由未标注样本组成的情况下。因此，如何设计算法以充分利用未标注数据，提高模型性能，成为一个待解决的问题。

## (2) 如何整合 EEG 信号的结构信息

另一个研究问题是如何整合 EEG 信号的结构信息。以往的研究通常从孤立的 EEG 通道提取特征，忽略了不同通道之间的复杂特征表示，导致模型缺乏丰富的结构信息。为了捕捉 EEG 信号中复杂的连接模式和拓扑关系，需要探索新的方法来整合 EEG 信号的结构信息，以提高情感识别的准确性和鲁棒性。

## 2 相关工作

近年来，脑电图 (EEG) 作为一种直接反映大脑活动的信号，广泛应用于情感识别研究中。随着情感计算和脑-机接口技术 (aBCI) 的发展，基于 EEG 的情感识别引起了越来越多的关注，特别是在利用 EEG 信号识别个体情感状态方面。国内外的研究不断推动着这一领域的发展，尤其是在提高情感识别的准确性和跨被试泛化能力方面，取得了显著进展。

### 2.1 基于 EEG 的情感识别

在基于 EEG 的情感识别研究初期，许多研究者尝试从 EEG 信号中提取与情感相关的特征，并使用传统的机器学习方法进行情感分类。例如，Duan 等人提取了情感相关的 EEG 特征，并采用支持向量机 (SVM) 进行情感分类，这为后续的研究奠定了基础 [1]。随着深度学习技术的快速发展，越来越多的基于神经网络的模型被提出以增强 EEG 情感识别的准确性。比如，Zhang 等人提出了一种级联并行卷积递归神经网络，能够有效地学习判别性 EEG 特征 [2]；Song 等人则提出了一种动态图卷积神经网络，能够动态地捕捉不同 EEG 通道之间的内在关系 [3]；Niu 等人结合脑网络分析和通道空间注意力机制，提出了一种新型的深度残差神经网络 [4]。这些研究为 EEG 情感识别提供了多种深度学习框架，并在被试内情感识别任务中取得了较好的性能。

然而，由于不同被试之间 EEG 信号的个体差异，现有模型在跨被试情感识别任务中表现较差。这是因为在跨被试任务中，训练数据和测试数据来自不同的被试，导致模型在不同被试之间的泛化能力显著下降 [5] [6] [7]。

### 2.2 跨被试 EEG 情感识别

为了提升跨被试情感识别模型的泛化能力，研究者们提出了迁移学习 (Transfer Learning) 方法，通过对不同被试之间的特征分布进行协调，减小个体差异的影响。例如，Zheng 等人通过迁移分量分析 (TCA) 和传导参数迁移 (TPT) 方法，成功地提高了跨被试情感识别的性能，从 56.73% 提升至 76.31% [5]。近年来，深度迁移学习方法被提出，以进一步提升跨被试情感识别任务的性能。例如，Jin 等人引入了深度迁移学习框架与对抗神经网络 (DANN)，将情感识别准确率从 76.31% 提高至 79.19% [8]。基于 Jin 等人的工作，Li 等人提出了一种领域自适应方法，旨在最小化分布偏移，使情感识别模型能够更好地迁移到不同个体 [9]。此外，

He 等人将对抗判别器和时间卷积网络 (TCN) 相结合, 进一步增强了 EEG 情感识别任务中的分布匹配 [10]。

尽管深度迁移学习方法在跨被试情感识别中取得了良好的性能, 但这些方法通常需要大量的标注数据进行训练, 获取足够的高质量标注数据是一个既费时又昂贵的过程。在这一背景下, 半监督学习框架的提出显得尤为重要, 它能够有效地利用少量标注数据和大量未标注数据, 从而减少对标注数据的依赖, 并提高跨被试情感识别模型的性能。

## 2.3 对比学习

近年来, 对比学习 (Contrastive Learning) 作为一种有效的特征学习方法, 已在多个领域得到广泛应用, 如计算机视觉 (CV) [11]、自然语言处理 (NLP) [12] 和生物信息学 [13] [14] 等。在 EEG 情感识别领域, 对比学习的潜力也逐渐被研究者们发掘。例如, Mohsenvand 等人使用 SimCLR 框架, 通过学习增强后的 EEG 样本之间的相似性, 提升了模型提取判别性特征的能力 [15]。Shen 等人提出了 CLISA 模型, 旨在最大化经历相同情绪刺激的不同被试之间的特征表示相似性, 从而改善情感识别的性能 [16]。这些研究表明, 对比学习能够有效提升基于 EEG 的情感识别效果。

尽管如此, 目前基于对比学习的 EEG 建模仍面临一些挑战, 尤其是在如何有效地将 EEG 信号中的结构化信息整合进对比学习框架中。EEG 信号具有天然的空间和时间依赖性, 不同脑电极和脑区之间存在复杂的连接模式, 这些信息对于理解大脑的动态活动和情感处理至关重要。图对比学习 (GCL) 作为一种能够整合数据结构信息的有效方法, 为对比学习在 EEG 分析中的应用提供了新的思路 [17]。通过将 GCL 与 EEG 分析结合, 有望深入揭示大脑功能的复杂性, 尤其是不同 EEG 通道之间的交互关系对情感识别的贡献。

## 3 本文方法

### 3.1 本文方法概述

如图 1 所示, DS-AGC 模型由三个主要部分组成: 非结构流、结构流和自注意力融合模块, 旨在有效提取和融合脑电图 (EEG) 中的非结构化和结构化特征, 以实现情绪识别。首先, 非结构化流通过从标记源域 S、未标记源域 U 和目标域 T 中提取 EEG 特征, 利用梯度反转层 (GRL) 实现领域不变的特征学习, 确保不同域间提取的特征不可区分。其次, 结构化流通过构建图卷积网络 (GCN) 对 EEG 信号的空间特征进行建模, 并通过数据增强生成正样本, 确保结构化特征在正样本之间保持一致性。最后, 自注意力融合模块将两种流的特征进行融合, 并通过多头自注意力机制突出情绪相关的判别性特征, 抑制不相关信息。融合后的特征通过训练分类器进行优化, 以实现情绪识别。通过这种结构, DS-AGC 模型能够联合提取域不变的非结构化特征与一致性的结构化特征, 从而提高情绪识别的准确性和鲁棒性。

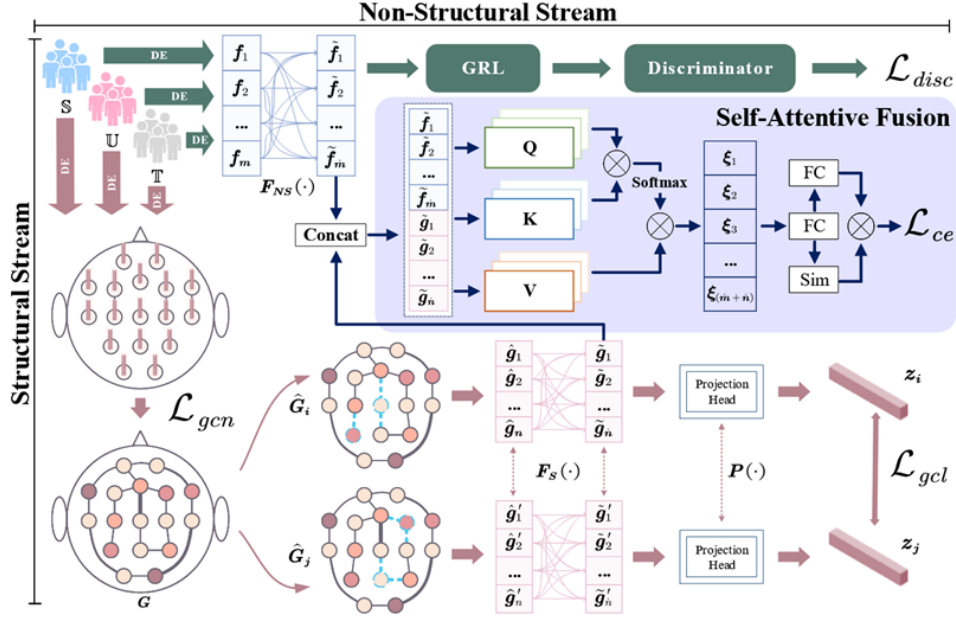


图 1. DS-AGC 的概览图。S、U 和 T 指的是标记的源域、未标记的源域和未知的目标域 [18]

### 3.2 非结构化流

传统的基于 EEG 信号的监督域适应方法主要依赖对抗神经网络 (DANN)，通过对源域和目标域之间的特征分布进行对齐来实现有效的迁移学习。虽然这种方法能够在一定程度上减小源域与目标域之间的分布差异，但在半监督学习中，将标记源数据和未标记源数据作为一个统一的域进行训练，增加了适应过程的复杂性，并可能对下游任务的性能产生负面影响。为了解决这一问题，DS-AGC 模型采用了一种新的半监督多域适应方法，它能够在标记源域 S、未标记源域 U 和目标域 T 之间对特征表示进行对齐，从而减小这些域之间的特征分布差异，并进一步提高模型的泛化能力。

#### 3.2.1 多域适应方法

在 DS-AGC 模型中，定义了三个不同的域，具体如下：标记源域 S：该域包含标记样本及其对应的情绪标签，表示为  $D_s = \{X_s, Y_s\} = \{(x_{s_i}, y_{s_i})\}_{i=1}^{N_s}$ ，其中  $x_{s_i}$  是第  $i$  个样本的 EEG 数据， $y_{s_i}$  是该样本的情绪标签， $N_s$  为标记样本的数量。未标记源域 U：包含未标记样本，没有情绪标签，表示为  $D_u = \{X_u\} = \{(x_{u_i})\}_{i=1}^{N_u}$ 。目标域 T：包含来自一个未见过的领域的样本，目标是对其进行情绪分类，表示为  $D_t = \{X_t\} = \{(x_{t_i})\}_{i=1}^{N_t}$ 。

在训练过程中，标记源域提供了情绪标签，而未标记源域和目标域没有标签。模型的目标是在未标记的源域和目标域上进行学习，使得这些数据能帮助提升目标域的分类性能。为了解决源域和目标域之间的分布差异，还引入了以下目标误差的上界公式来描述源域和目标域之间的关系： $\epsilon_T(h) \leq \delta + \epsilon_S(h) + d_H(D_S, D_T)$ ，其中， $\delta$  表示源域和目标域之间标签函数的差异，通常假设其较小； $\epsilon_S(h)$  表示源域的分类误差； $d_H(D_S, D_T)$  表示源域和目标域之间的 H-散度，用于度量两者分布的差异。

在半监督学习中，由于标记数据有限，准确估计源域和目标域之间的 H-散度以及源误差变得非常困难。为了解决这一问题，该模型提出将标记源域 S 和未标记源域 U 结合起来，形

成一个混合源域  $S^* = \{S, U\}$ ，通过加权方式来估算源误差和 H-散度。这种混合源域的引入能有效减少标记数据不足带来的影响，从而帮助模型更好地适应目标域的数据。

### 3.2.2 特征对齐与生成模型

为了对齐标记源域、未标记源域和目标域之间的特征表示，引入了多域对抗神经网络 (MDANN)。该网络通过对抗训练来学习跨域的共享特征表示，使得模型在不同域之间能够共享有意义的特征表示，从而提高在目标域上的性能。

首先，从 EEG 信号中提取差分熵 (DE) 特征，并将这些特征展平为一维特征向量。然后，这些特征被输入到特征提取器  $F_{NS}(\cdot)$  中，该提取器的任务是生成更具区分性的特征表示。提取后的特征会进入到一个判别器  $d(\cdot)$ ，该判别器用于判断输入的样本数据来自于哪个域（标记源域、未标记源域或目标域）。判别器的输出被用于优化特征对齐过程，确保不同域的特征分布趋于一致。

### 3.2.3 判别器与对抗训练

判别器的核心任务是判断输入的特征来自哪个域。为了实现这一目标，模型通过对抗训练来优化判别器，使其能够有效区分来自不同域的特征。通过对抗训练，特征提取器和判别器共同优化，确保特征在不同域间的分布差异最小化。

具体来说，判别器的损失函数  $L_{disc}$  被定义为：

$$L_{disc}(\theta_f, \theta_d) = - \sum_{x_i} l(x_i) \log d(F_{NS}(x_i)) \quad (1)$$

其中， $l(x_i)$  是样本  $x_i$  的域标签（表示其属于标记源域、未标记源域或目标域）， $F_{NS}(x_i)$  是从输入样本中提取的非结构特征。通过优化这个损失函数，模型能够最小化不同域之间的特征分布差异，从而使特征对齐更加精确。判别器通过最大化分类误差来逼迫特征提取器学习到更加共享的跨域特征表示。经过这种对抗训练后，特征提取器能够生成在不同域之间都具有高泛化能力的特征，使得模型在目标域的表现得到了显著提升。

## 3.3 结构化流

在 EEG 信号的情绪识别任务中，除了提取非结构特征外，还需要关注 EEG 信号的结构信息，特别是不同 EEG 通道之间的相互关系。为了更好地捕捉这种结构信息，该模型采用图卷积网络 (GCN) 来建模 EEG 通道之间的依赖关系，通过提取结构特征，帮助更深入地理解 EEG 信号中的复杂交互与依赖性。这一方法不仅能够提升情绪识别的准确度，还能提高模型的泛化能力。

### 3.3.1 图卷积网络 (GCN)

为了捕捉 EEG 信号中的结构信息，本文首先通过构建图的方式表示 EEG 信号的各个通道及其之间的关系。图中，节点代表 EEG 通道，边则表示不同 EEG 通道之间的依赖关系。具体而言，我们定义了一个无向图  $G = (V_G, E_G, A_G)$ ，其中： $V_G$  表示图的节点集合，大小为  $|V_G| = N_G$ ，即 EEG 信号中通道的数量； $E_G$  表示图的边集合，表示通道之间的连接关系；



$A_G$  表示图的邻接矩阵，表示节点之间的相似度或关系。对于每个样本数据  $x_i$ ，输入特征为  $\Psi_i = [\psi_i^1, \psi_i^2, \dots, \psi_i^{N_G}]^T \in \mathbb{R}^{N_G \times C_{de}}$ ，其中， $\psi_i^k \in \mathbb{R}^{C_{de}}$  表示第  $k$  个通道的差分熵 (DE) 特征。

通过图卷积网络 (GCN) 来进一步聚合邻居信息，以提取空间特征，并通过图结构来学习通道之间的相互关系。不同于传统的 GCN 使用固定邻接矩阵 (如  $k$  近邻图)，提出使用动态邻接矩阵  $A_G$  来表示不同通道之间的关系。具体地，邻接矩阵的每个元素  $A_G^{jk}$  由该公式定义：

$$A_G^{jk} = \frac{\exp(-\text{ReLU}(w^T \|\psi_i^j - \psi_i^k\|))}{\sum_{k=1}^{N_G} \exp(-\text{ReLU}(w^T \|\psi_i^j - \psi_i^k\|))} \quad (2)$$

其中， $\psi_i^j$  和  $\psi_i^k$  分别是第  $j$  和第  $k$  个通道的 DE 特征， $w$  是学习的权重向量。我们使用 ReLU 激活函数来确保特征的非线性表达，从而增强模型对复杂模式的学习能力。GCN 的目标是通过最小化以下损失函数，来优化邻接矩阵，同时保持图的稀疏性：

$$L_{gcn} = \lambda \sum_{j,k=1}^{N_G} \|\psi_i^j - \psi_i^k\|^2 A_G^{jk} + \|A_G\|_F^2 \quad (3)$$

其中，第一项量化了节点之间的相似性，确保相似的节点之间的连接权重较大，异构节点之间的连接权重较小；第二项是 Frobenius 范数，作为稀疏性约束项，鼓励学习到的邻接矩阵更加稀疏。这里， $\lambda \geq 0$  是正则化参数，用来调节图学习与稀疏性之间的权衡。在学习到邻接矩阵后，节点表示可以通过 Chebyshev 多项式展开来进一步提取，这可以通过该公式实现： $G = \sum_{\phi=0}^{\Phi-1} \theta_\phi T_\phi(L)x_i$ ，其中， $T$  是图拉普拉斯矩阵  $L$  的 Chebyshev 多项式， $\theta$  是 Chebyshev 系数向量， $x_i$  是输入样本数据。通过 Chebyshev 多项式，GCN 能够捕捉图中不同阶层 (邻接节点之间的关系) 的信息，从而为节点提供更加丰富、全面的表示。

### 3.3.2 图对比学习 (GCL)

为了进一步增强特征表示的鲁棒性，本文引入了图对比学习 (GCL)，通过对比学习的方式，使得相似的样本在表示空间中靠得更近，而不相似的样本则被拉远。图对比学习不仅有助于提升模型的鲁棒性，还能够改善模型在不同数据增强或扰动下的表现。在上一节中学到的图  $G$  的基础上，本文生成了两个增强图  $\hat{G}_i$  和  $\hat{G}_j$  作为正样本。具体做法是，随机丢弃图中的  $\zeta\%$  节点，类似于数据增强中的 dropout 操作。然后，我们将增强图中的节点特征展平成一维特征向量，分别记为  $\{\hat{g}_1, \dots, \hat{g}_n\}$  和  $\{\hat{g}'_1, \dots, \hat{g}'_n\}$ 。然后，通过投影网络进一步降低特征维度，得到投影后的特征表示  $z_i$  和  $z_j$ 。为了确保由同一输入生成的两幅增强图的特征表示一致性，本文定义了图对比学习的损失函数  $L_{gcl}$ ，其形式为归一化的温度缩放交叉熵损失：

$$L_{gcl} = -\log \frac{\exp(\text{Sim}(z_i, z_j)/\tau)}{\sum_{k=1, k \neq i}^B \exp(\text{Sim}(z_i, z_k)/\tau)} \quad (4)$$

其中， $\text{Sim}(z_i, z_j)$  表示余弦相似度， $T$  是温度参数，用于调整特征学习的性能。该损失函数通过最大化正样本  $z_i$  和  $z_j$  之间的相似性，同时最小化负样本  $z_i$  和其他样本  $z_k$  之间的相似性，进一步提高了特征的可区分性。

### 3.4 自注意力融合模块

为了有效地突出关键特征并赋予与目标数据更接近的源数据更高的权重，该模型引入了自注意力融合机制，从而获得更具信息量的特征表示。此外，为了确保提取的特征能够有效区分情绪类别，还在模型的学习过程中加入了监督分类部分。

#### 3.4.1 信息特征融合

在本实验中，首先通过  $\mathcal{F}_{NS}(\cdot)$  提取的非结构化特征和通过  $\mathcal{F}_S(\cdot)$  提取的结构化特征被拼接成一个新的特征表示，记作  $\{\tilde{f}_1, \dots, \tilde{f}_m, \tilde{g}_1, \dots, \tilde{g}_n\}$ 。为了使特征更加突出情绪相关信息，减少无关特征的影响，本文引入了多头自注意力机制。该机制的主要作用是根据输入数据和任务的具体需求，自动调整关注的重点，通过加权不同特征来突出那些最为重要的情绪相关特征。

具体而言，从拼接后的特征表示中，通过线性变换生成三个矩阵  $Q$ 、 $K$  和  $V$ ，这些矩阵分别表示查询、键和值。接着，计算注意力权重，这里使用的是标准的缩放点积注意力机制，公式如下：

$$\text{ATT}(Q, K, V) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d}}\right) \quad (5)$$

在此基础上，进一步扩展到多头机制。每个矩阵被划分为  $H$  个子空间，分别记作  $Q = \{Q_1, \dots, Q_H\}$ ,  $K = \{K_1, \dots, K_H\}$ ,  $V = \{V_1, \dots, V_H\}$ ，然后在每个子空间内分别计算注意力权重： $A_h = \text{ATT}(Q_h, K_h, V_h)$ 。最终，所有  $H$  个表示被拼接在一起，得到最终的多头注意力输出： $\text{MHA}(X) = \text{Concat}(A_1, \dots, A_H)$ ，该输出用于情绪分类任务，增强了特征表示的情绪区分能力。

#### 3.4.2 信息样本选择

在模型训练过程中，除了传统的监督学习方法外，本文还采用了一种基于自注意力机制的样本加权策略，以提高情绪识别的效果。通过加权已标记源数据的重要性，本文能够更好地聚焦在对情绪识别最有帮助的数据上。具体地，首先通过多头自注意力机制得到的特征表示  $\text{MHA}(\cdot)$  被输入到一个全连接层  $\phi(\cdot)$  中，生成每个源数据的表示  $\mathcal{R}(X_s)$  和每个目标数据的表示  $\mathcal{R}(X_t)$ ，公式如下：

其中， $B$  表示批次大小。接着，对于每个已标记源数据  $r_{sb}$ ，计算其与所有未知目标数据的余弦相似度。相似度计算公式为：

$$\text{Sim}(r_{sb}) = \frac{1}{B} \sum_{k=1}^B \frac{r_{sb} \cdot r_{tk}}{\|r_{sb}\| \|r_{tk}\|} \quad (6)$$

通过计算所有源数据与目标数据的相似度，可以得出每个源数据对目标数据的贡献程度。然后，采用  $\text{Softmax}$  函数对这些相似度进行归一化处理，得到归一化后的权重。最后，利用这些权重调整多类交叉熵损失函数中的样本贡献，公式为：

$$L_{ce} = -\frac{1}{B} \sum_{b=1}^B \sum_{c=1}^C y_{cb} \log(\text{Sim}(r_{sb}) \cdot \hat{y}_{cb}) \quad (7)$$

其中,  $y_{cb}$  和  $\hat{y}_{cb}$  分别表示第  $b$  个已标记源数据的实际情绪标签和预测情绪标签,  $C$  是情绪类别的总数。通过这种方式, 模型能够更好地聚焦于最具信息的样本, 从而提高情绪识别的准确性和鲁棒性。

## 4 复现细节

### 4.1 与已有开源代码对比

原论文有开源代码, 代码可在<https://github.com/Vesan-yws/DS-AGC>下载。在复现过程中, 遇到的第一个问题是数据预处理的问题, 开源代码中缺失数据预处理部分, 所以本文首先补充了数据预处理的代码。其次, 原论文在网络优化中采用的是 RMSprop 优化器, 本文将其替换成 AdamW 优化器, RMSprop 和 Adam 的 `weight_decay` 参数实际上会影响梯度更新过程的动量计算, 而 AdamW 将权重衰减与动量分离, 独立地施加重权衰减, 从而更符合理论上的权重正则化, 并且 RMSprop 在处理复杂数据 (如 EEG 数据的高维时间序列特征) 时可能导致收敛速度较慢。AdamW 的自适应学习率机制能够加速训练, 尤其是在早期阶段。

为提升模型性能并降低计算复杂度, 提出了以下改进: 一是增加 Batch Normalization 和 Dropout 层以稳定梯度流动并提高泛化能力; 二是引入动态学习率调整策略 (StepLR), 优化训练过程的收敛效果; 三是将激活函数由 ReLU 替换为 Swish, 以改善梯度流动并提升深度模型的表现。这些修改针对模型的复杂性和数据稀缺性问题, 能够在保持架构完整性的同时提高训练效率和泛化性能。

### 4.2 实验环境搭建

本实验基于 Python 编程语言和 PyTorch 深度学习框架实现, 运行环境为支持 GPU 加速的设备, 确保训练效率。实验使用了 SEED-IV 数据集, 采用跨被试 (Cross-Subject) 实验设计, 每次选择一个被试的数据作为目标域, 其余被试数据划分为标注源域和未标注源域。数据加载函数 `get_dataset_IV` 根据数据集类型对数据进行预处理和划分。实验中, 随机种子通过 `setup_seed` 函数固定, 保证结果的可复现性。训练过程中, 使用 RMSprop 优化器, 超参数字典设置批量大小 (48)、初始学习率 ( $1e-3$ )、权重衰减 ( $1e-5$ ) 和训练轮数 (100 轮)。实验结果包括各目标被试的最佳测试准确率和训练过程中的准确率曲线, 并通过 NumPy 保存为 .numpy 文件, 便于后续分析和比较。实验在 CUDA 环境中运行, 使用的显卡为 NVIDIA A6000, 设备通过参数 `DEVICE: 'cuda:0'` 指定。

### 4.3 使用说明

运行本实验需要确保实验环境配置完整, 包括安装 Python 3.8 及以上版本、PyTorch 深度学习框架 (建议版本为 1.10 或更高)、NumPy、SciPy 等依赖库, 并在支持 GPU 加速的设备上运行 (推荐 NVIDIA A6000 或性能相近的显卡)。实验使用 SEED-IV 数据集, 该数据集来源于上海交通大学情感与脑科学实验室, 包含多被试的脑电信号数据。数据集需按照实验代码的要求组织路径, 运行前需确保已正确下载并存放至指定文件夹中, 下载路径为<https://bcmi.sjtu.edu.cn/~seed/index.html>。数据加载过程中, 数据会通过函数 `get_dataset_IV` 进行预处理, 包括特征提取 (如频域特征)、归一化操作以及数据划分 (标注源域、未标注源



域和目标域)。实验的主要配置参数包括批量大小 (48)、初始学习率 ( $1e-3$ )、训练轮数 (100) 等, 可通过代码中的 `parameter` 和 `net_params` 字典进行修改。实验通过调用 `main()` 函数执行, 结果将以 .`numpy` 文件形式保存于指定路径下, 用于后续分析和可视化。运行时建议固定随机种子 (通过 `setup_seed` 函数) 以保证结果的可复现性。

#### 4.4 创新点

首先, 补充了原代码中缺失的数据预处理部分, 通过规范化的数据处理流程, 确保实验数据的完整性和一致性。其次, 在优化器方面, 替换了原论文中使用的 RMSprop 优化器, 采用更适合高维时间序列特征训练的 AdamW 优化器, 该优化器通过独立施加权重衰减, 显著提升了训练的稳定性 and 早期收敛速度。此外, 在模型结构上, 本文增加了 Batch Normalization 和 Dropout 层, 以提升模型的梯度稳定性和抗过拟合能力; 通过动态学习率调整策略 (StepLR), 进一步优化了训练后期的收敛效率; 同时, 将激活函数由 ReLU 替换为 Swish, 以优化特征表达能力。

### 5 实验结果分析

#### 5.1 留一受试者交叉验证方法

如图 2 所示, 本实验采用了留一受试者交叉验证 (Leave-One-Subject-Out Cross-Validation, LOSO-CV) 实验协议, 以全面评估模型的性能与泛化能力。针对 SEED-IV 数据集 (包含 15 名受试者), 每次选取 14 名受试者作为源域, 剩余 1 名受试者作为目标域进行模型测试。为了确保实验的公平性与结果的全面性, 每名受试者都有机会作为目标域参与测试, 因此共进行了 15 轮模型训练与测试。最终的分类结果通过这 15 轮实验的平均准确率和标准差来报告, 以全面反映模型在不同实验条件下的稳定性和性能。

在源域的构建中, 将 14 名受试者进一步划分为未标记的源域  $U$  和标记的源域  $S$ 。具体而言, 假设将第 1 名受试者分配为目标域  $T$ , 那么从第 2 名到第  $2 + N - 1$  名受试者被划分为未标记的源域  $U$ , 而剩余的  $14 - N$  名受试者则被划分为标记的源域  $S$ 。为了评估模型在不同标签稀缺情况下的性能, 实验通过调整  $N$  的取值 (从 1 到 13), 系统性地研究了未标记数据比例对模型性能的影响。通过这种划分方式, 实验既能模拟实际应用中未标记数据占比不同的情况, 又能有效测试模型在标签稀缺场景下的鲁棒性。

在模型的训练阶段, 未标记的源域  $U$  并未从一开始就参与训练, 而是仅使用标记的源域  $S$  和目标域  $T$  进行前  $E_t$  次迭代的模型训练。此时, 域判别器  $d(\cdot)$  的输出层被设计为包含两个神经元, 执行标记源域  $S$  与目标域  $T$  之间的二元分类任务。通过这种方式, 模型能够初步学习到标记源域和目标域之间的分布差异, 建立基本的领域适配能力。当训练达到一定的稳定性后, 未标记的源域  $U$  被引入到模型训练中, 域判别器  $d(\cdot)$  的输出层扩展为三个神经元, 从而能够执行标记源域  $S$ 、未标记源域  $U$  和目标域  $T$  之间的三元分类任务。这种训练策略逐步引入未标记源域数据, 确保了模型适配过程的稳定性和训练的渐进性。

需要特别注意的是, 在整个训练过程中, 分类器的交叉熵损失  $L_{ce}$  仅在标记的源域  $S$  的数据上进行计算, 未标记的源域  $U$  和目标域  $T$  的数据仅用于领域判别任务。通过这种训练方式, 模型能够在充分利用未标记数据的同时, 最大程度保证分类器的性能不受到目标域和

未标记源域噪声的影响。这一训练过程设计有效平衡了标记数据的稀缺性与未标记数据的丰富性，在有限标注条件下实现了对目标域情感识别的准确性提升。

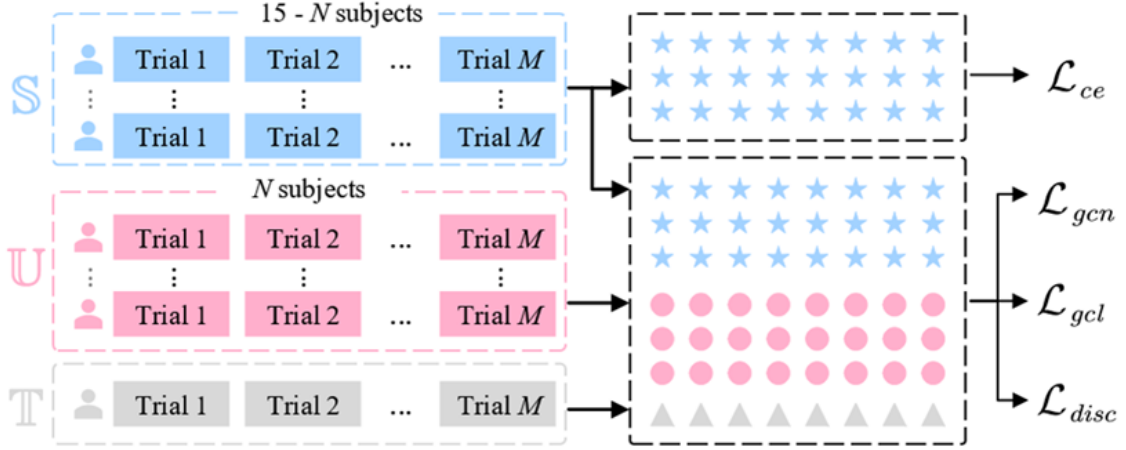


图 2. 标签不完整的跨受试者留一受试者出交叉验证实验方案 [18]

## 5.2 标签不完整的情感识别

图 3 是原论文中所呈现的实验结果，而表 1 则展示了本文在复现过程中获得的实验结果。通过对比可以发现，本文成功复现了原论文中的实验结论，实验数据表明，DS-AGC 模型在未标记源域数量不断增加的情况下，依然能够保持情绪识别准确率在 50% 以上，充分展现了其在复杂场景中的稳健性和高效性。这一发现与现有文献中的研究结果高度一致，进一步证明了 DS-AGC 模型在情绪识别任务中的卓越性能。特别是在 SEED-IV 数据集上的实验中，DS-AGC 模型的表现超越了众多领先的深度学习模型，包括但不限于 DAN、DANN 和 DDC 等。这表明 DS-AGC 模型在情绪识别领域具有显著的技术优势。

在极端标签稀缺的实验条件下（即未标记源域数量  $N = 13$ ），可用的标记数据资源极其有限，然而 DS-AGC 模型依然取得了 52.5% 的情绪识别准确率。这一成绩在同类研究中显得尤为突出，说明 DS-AGC 模型能够在标记数据稀缺的情况下充分利用未标记数据和目标域的结构信息，以实现优异的分类性能。相比之下，许多传统的领域适配模型在标记数据不足时容易出现性能急剧下降的问题，而 DS-AGC 模型的稳健性明显更强。

实验结果还表明，DS-AGC 模型在不同  $N$  值的设置下均表现出极高的平均性能，显著优于现有的主流情绪识别方法。具体来说，随着未标记源域数据的增加，模型能够通过域适配机制逐步学习到源域和目标域的共同特征分布，从而确保分类准确率的平稳提升。这种能力在标记源域数据稀缺的情况下尤为重要，DS-AGC 模型通过对比学习和注意力机制的引入，有效缓解了数据标注不足带来的性能瓶颈。

Methods	$N = 1$	$N = 2$	$N = 3$	$N = 4$	$N = 5$	$N = 6$	$N = 7$
SVM* [60]	47.47±13.59	47.52±13.48	45.35±13.16	43.28±10.00	45.51±12.74	44.69±14.10	48.61±15.16
TCA* [61]	43.58±10.74	44.15±10.73	43.19±08.76	42.24±11.14	43.71±10.85	40.69±08.88	41.51±09.06
SA* [62]	40.82±14.33	38.87±09.93	40.15±10.61	40.34±13.42	33.07±08.48	35.24±11.01	36.19±14.92
KPCA* [63]	41.41±08.02	41.05±07.72	41.18±08.85	38.53±10.62	39.20±08.80	38.31±10.38	41.25±11.21
RF* [64]	53.65±14.12	51.64±14.80	49.97±14.29	45.94±13.38	50.28±13.56	46.33±14.64	47.88±15.83
AdaBoost* [65]	54.97±15.55	51.82±11.41	52.21±11.82	50.96±11.89	54.40±15.43	51.76±14.46	53.05±13.27
CORAL* [66]	50.88±15.00	46.39±13.46	45.17±13.29	46.74±13.13	47.86±14.55	47.11±15.74	47.75±15.33
GFK* [67]	41.62±09.07	42.54±06.90	42.24±06.44	40.93±06.14	40.73±07.71	40.79±07.87	42.55±09.49
KNN* [68]	37.67±09.33	40.04±05.55	39.50±08.22	39.57±06.00	37.01±07.66	36.72±06.06	38.73±06.73
DAN* [69]	57.83±07.73	59.35±08.58	57.47±09.01	53.59±09.25	55.22±09.82	54.15±08.26	51.94±09.97
DANN* [70]	57.65±09.73	56.89±07.99	56.81±09.11	54.95±10.85	56.14±10.54	55.87±09.76	56.72±11.18
DCORAL* [69]	58.71±08.36	56.81±07.80	58.43±08.54	54.57±09.35	55.18±09.81	54.60±10.53	53.19±11.17
DDC* [70]	55.90±08.07	54.77±08.56	53.40±09.45	51.73±08.39	50.61±10.33	51.35±09.02	51.02±10.92
PARSE* [71]	66.57±10.81	66.19±09.71	66.38±10.34	64.28±10.13	63.68±09.40	61.12±10.74	60.87±12.27
MixMatch* [4]	61.98±08.85	62.58±09.42	61.60±07.71	60.72±09.11	61.42±07.94	59.55±08.63	57.85±08.44
AdaMatch* [6]	65.22±09.32	64.26±09.35	65.79±09.31	62.23±09.91	63.42±10.25	62.14±09.99	61.12±10.26
FlexMatch* [71]	62.77±08.93	62.75±08.83	62.68±09.06	61.13±09.15	61.06±09.41	60.29±08.75	59.85±09.37
SoftMatch* [72]	62.46±09.14	62.57±08.87	62.51±09.15	61.14±09.41	60.74±09.71	60.94±08.74	59.93±09.50
DS-AGC	65.79±09.55 (-0.78)	66.00±07.93 (-0.19)	65.30±06.99 (-1.08)	63.15±08.30 (-1.13)	63.24±10.15 (-0.44)	62.40±09.65 (+0.26)	61.21±10.35 (+0.09)

Methods	$N = 8$	$N = 9$	$N = 10$	$N = 11$	$N = 12$	$N = 13$	Average
SVM* [60]	47.19±15.04	46.93±14.06	46.60±11.47	46.18±11.23	44.36±16.42	42.07±11.60	45.83±13.50
TCA* [61]	42.45±09.05	41.31±09.23	42.64±10.68	37.97±10.84	39.30±12.12	40.06±10.75	41.75±10.45
SA* [62]	34.28±13.13	39.69±13.06	34.85±14.87	43.07±13.75	38.52±12.78	32.82±08.54	37.53±12.84
KPCA* [63]	38.22±11.35	38.48±11.19	40.27±14.15	34.49±10.61	36.80±11.32	37.12±13.88	38.95±11.00
RF* [64]	44.52±11.25	45.74±16.01	47.21±14.39	45.34±13.38	46.63±13.10	40.36±12.84	47.35±14.44
AdaBoost* [65]	51.48±11.56	50.79±14.56	51.36±13.33	50.66±15.36	48.16±12.26	43.33±12.09	51.15±13.72
CORAL* [66]	48.59±13.99	46.92±15.07	47.66±12.11	44.66±12.47	44.14±14.35	43.83±10.56	46.75±14.01
GFK* [67]	42.63±08.23	43.07±08.46	40.59±11.63	37.95±10.90	43.60±10.64	44.08±12.55	41.79±09.29
KNN* [68]	38.24±08.36	39.47±08.97	36.75±13.21	36.15±12.60	42.32±10.79	42.18±12.26	38.80±09.47
DAN* [69]	54.49±08.19	53.34±10.22	51.66±10.48	51.33±12.00	51.70±10.89	49.34±12.07	53.95±10.21
DANN* [70]	53.68±12.30	55.28±09.85	51.73±11.13	51.91±10.80	50.84±13.06	47.85±12.62	54.33±11.14
DCORAL* [69]	53.85±09.01	53.11±10.69	52.17±09.21	49.39±09.84	48.48±09.85	49.57±11.02	53.70±10.17
DDC* [70]	50.01±09.46	52.63±10.11	53.10±12.24	48.76±09.45	50.72±12.97	46.91±10.91	51.61±10.35
PARSE* [71]	60.49±11.57	60.61±10.32	60.61±12.18	55.97±09.16	55.49±10.66	52.50±08.37	61.13±11.31
MixMatch* [4]	59.02±08.43	57.65±07.72	54.94±08.76	53.53±06.19	53.00±10.59	47.55±06.82	57.80±09.44
AdaMatch* [6]	60.42±10.90	59.77±11.17	59.11±10.63	55.61±10.72	55.14±11.86	52.54±08.26	60.52±10.91
FlexMatch* [71]	59.25±08.53	59.83±08.81	57.49±11.27	55.84±09.54	55.27±09.95	53.86±08.69	59.39±09.70
SoftMatch* [72]	59.01±08.61	59.54±08.76	57.90±10.54	56.19±09.51	55.32±10.06	53.83±08.89	59.39±09.71
DS-AGC	62.17±10.89 (+1.68)	62.54±09.60 (+1.93)	59.55±10.53 (-1.06)	58.75±11.33 (+2.56)	57.09±08.52 (+1.60)	50.00±09.31 (-3.86)	61.32±10.41 (+0.19)

图 3. 原论文 [18] 在 SEED-IV 数据集上的实验结果

表 1. 标签条件不完整的 SEED-IV 数据集上的情绪识别结果

未标记源域数量	N=1	N=2	N=3	N=4	N=5	N=6	N=7	N=8	N=9	N=10	N=11	N=12	N=13
subject_1	0.663	0.691	0.736	0.646	0.711	0.585	0.634	0.647	0.609	0.709	0.703	0.591	0.567
subject_2	0.625	0.641	0.711	0.718	0.697	0.698	0.638	0.599	0.656	0.694	0.628	0.590	0.524
subject_3	0.679	0.657	0.514	0.628	0.512	0.543	0.509	0.491	0.524	0.471	0.572	0.593	0.483
subject_4	0.440	0.451	0.436	0.531	0.472	0.527	0.481	0.479	0.436	0.430	0.541	0.432	0.384
subject_5	0.527	0.533	0.570	0.523	0.482	0.466	0.548	0.468	0.500	0.433	0.445	0.479	0.485
subject_6	0.746	0.772	0.777	0.760	0.766	0.723	0.796	0.719	0.675	0.672	0.702	0.699	0.544
subject_7	0.726	0.697	0.661	0.660	0.657	0.627	0.640	0.636	0.642	0.630	0.493	0.625	0.518
subject_8	0.763	0.777	0.839	0.833	0.832	0.748	0.770	0.763	0.775	0.716	0.839	0.807	0.674
subject_9	0.598	0.541	0.561	0.563	0.688	0.649	0.618	0.600	0.578	0.522	0.481	0.580	0.545
subject_10	0.706	0.671	0.658	0.700	0.696	0.697	0.622	0.602	0.576	0.583	0.540	0.571	0.574
subject_11	0.636	0.597	0.725	0.647	0.682	0.676	0.673	0.676	0.721	0.549	0.533	0.448	0.592
subject_12	0.686	0.732	0.671	0.758	0.705	0.728	0.719	0.648	0.648	0.679	0.634	0.503	0.525
subject_13	0.671	0.670	0.539	0.586	0.618	0.575	0.597	0.598	0.555	0.443	0.540	0.470	0.436
subject_14	0.634	0.713	0.635	0.542	0.656	0.716	0.598	0.646	0.652	0.646	0.665	0.720	0.639
subject_15	0.605	0.529	0.565	0.533	0.538	0.513	0.521	0.554	0.506	0.489	0.469	0.350	0.360
平均值	0.647	0.645	0.640	0.642	0.647	0.631	0.624	0.608	0.604	0.578	0.586	0.564	0.523

### 5.3 消融实验

本实验还进行了消融研究以评估 DS-AGC 模型中各个组件对性能的具体贡献，见表 2。通过逐一移除模型中的不同组件，能够量化每个组件对整体性能的影响。以下是消融实验的详细结果，这些结果均在 SEED-IV 数据集上通过交叉验证获得：

移除判别器：当模型中不包含判别器时，其性能从 64.54% 下降至 64.03%。这一结果表明，判别器在捕捉域间分布差异方面起到了重要作用，尽管其对整体性能的影响相对较小。

移除对比学习：对比学习的缺失导致模型性能从 64.54% 下降至 62.98%。这说明对比学习在提升特征区分能力方面具有显著效果，对模型性能有正面影响。

移除注意力特征融合：没有使用注意力特征融合的模型性能显著下降，从 64.54% 降至 60.34%。这强调了注意力机制在特征融合过程中的关键作用，它有助于模型更有效地整合不同来源的信息。

移除注意力样本选择：当模型不采用注意力样本选择时，性能从 64.54% 轻微下降至 63.89%。尽管这一变化不如其他组件移除时显著，但它仍然表明样本选择过程对模型性能有积极贡献。

完整模型(DS-AGC): 包含所有组件的 DS-AGC 模型在 SEED-IV 数据集上达到了 64.54% 的性能。这一结果证明了模型中每个组件的整合对于实现最佳性能至关重要。

表 2. SEED-IV 数据集不同方法的情绪识别结果比较

Methods	SEED-IV (%)
without discriminator	64.03
without contrastive learning	62.98
without attentive feature fusion	60.34
without attentive sample selection	63.89
<b>DS-AGC</b>	<b>64.54</b>

## 6 总结与展望

本文聚焦于利用半监督学习方法解决 EEG 情感识别中标注数据稀缺的问题，复现了一个融合非结构化流、结构化流和自注意力特征融合模块的 DS-AGC 模型。通过非结构化流实现多域适应，结构化流结合图卷积和图对比学习建模 EEG 通道关系，自注意力模块进一步提升特征融合效果。在 SEED-IV 数据集上的实验表明，DS-AGC 模型在情感识别任务中取得了优异的性能，尤其在标签稀缺的情况下表现出较强的鲁棒性。

当前实现存在数据来源单一、未标注数据利用不足以及模型复杂度较高等问题。实验仅基于 SEED-IV 数据集，没有验证模型在其他数据集或多模态场景中的适用性；未标注数据的利用主要依赖对抗训练和对比学习，未充分挖掘其潜在信息；同时，模型复杂度较高，难以应用于资源受限环境。未来研究可通过引入多模态数据进行特征融合、结合生成模型或一致性正则化挖掘未标注数据潜力，并优化模型结构以降低计算成本，同时结合时间序列建模增强对 EEG 信号动态变化的捕捉能力。

## 参考文献

- [1] Ruo-Nan Duan, Jia-Yi Zhu, and Bao-Liang Lu. Differential entropy feature for eeg-based emotion classification. In *2013 6th international IEEE/EMBS conference on neural engineering (NER)*, pages 81–84. IEEE, 2013.
- [2] Dalin Zhang, Lina Yao, Xiang Zhang, Sen Wang, Weitong Chen, Robert Boots, and Boualem Benatallah. Cascade and parallel convolutional recurrent neural networks on eeg-based intention recognition for brain computer interface. In *Proceedings of the aaai conference on artificial intelligence*, volume 32, 2018.
- [3] Tengfei Song, Wenming Zheng, Peng Song, and Zhen Cui. Eeg emotion recognition using dynamical graph convolutional neural networks. *IEEE Transactions on Affective Computing*, 11(3):532–541, 2018.
- [4] Weixin Niu, Chao Ma, Xinlin Sun, Mengyu Li, and Zhongke Gao. A brain network analysis-based double way deep neural network for emotion recognition. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 31:917–925, 2023.
- [5] Wei-Long Zheng and Bao-Liang Lu. Personalizing eeg-based affective models with transfer learning. In *Proceedings of the twenty-fifth international joint conference on artificial intelligence*, pages 2732–2738, 2016.
- [6] Wojciech Samek, Frank C Meinecke, and Klaus-Robert Müller. Transferring subspaces between subjects in brain–computer interfacing. *IEEE Transactions on Biomedical Engineering*, 60(8):2289–2298, 2013.
- [7] Hiroshi Morioka, Atsunori Kanemura, Jun-ichiro Hirayama, Manabu Shikauchi, Takeshi Ogawa, Shigeyuki Ikeda, Motoaki Kawanabe, and Shin Ishii. Learning a common dictionary for subject-transfer decoding with resting calibration. *NeuroImage*, 111:167–178, 2015.
- [8] Yi-Ming Jin, Yu-Dong Luo, Wei-Long Zheng, and Bao-Liang Lu. Eeg-based emotion recognition using domain adaptation network. In *2017 international conference on orange technologies (ICOT)*, pages 222–225. IEEE, 2017.
- [9] Jinpeng Li, Shuang Qiu, Changde Du, Yixin Wang, and Huiguang He. Domain adaptation for eeg emotion recognition based on latent representation similarity. *IEEE Transactions on Cognitive and Developmental Systems*, 12(2):344–353, 2019.
- [10] Zhipeng He, Yongshi Zhong, and Jiahui Pan. An adversarial discriminative temporal convolutional network for eeg-based cross-domain emotion recognition. *Computers in biology and medicine*, 141:105048, 2022.



- [11] Ming Chen, Zhewei Wei, Zengfeng Huang, Bolin Ding, and Yaliang Li. Simple and deep graph convolutional networks. In *International conference on machine learning*, pages 1725–1735. PMLR, 2020.
- [12] Jacob Devlin. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.
- [13] Pengyong Li, Jun Wang, Yixuan Qiao, Hao Chen, Yihuan Yu, Xiaojun Yao, Peng Gao, Guotong Xie, and Sen Song. An effective self-supervised framework for learning expressive molecular global representations to drug discovery. *Briefings in Bioinformatics*, 22(6):bbab109, 2021.
- [14] Xianggen Liu, Yunan Luo, Pengyong Li, Sen Song, and Jian Peng. Deep geometric representations for modeling effects of mutations on protein-protein binding affinity. *PLoS computational biology*, 17(8):e1009284, 2021.
- [15] Mostafa Neo Mohsenvand, Mohammad Rasool Izadi, and Pattie Maes. Contrastive representation learning for electroencephalogram classification. In *Machine Learning for Health*, pages 238–253. PMLR, 2020.
- [16] Xinke Shen, Xianggen Liu, Xin Hu, Dan Zhang, and Sen Song. Contrastive learning of subject-invariant eeg representations for cross-subject emotion recognition. *IEEE Transactions on Affective Computing*, 14(3):2496–2511, 2022.
- [17] Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, and Yang Shen. Graph contrastive learning with augmentations. *Advances in neural information processing systems*, 33:5812–5823, 2020.
- [18] Weishan Ye, Zhiguo Zhang, Fei Teng, Min Zhang, Jianhong Wang, Dong Ni, Fali Li, Peng Xu, and Zhen Liang. Semi-supervised dual-stream self-attentive adversarial graph contrastive learning for cross-subject eeg-based emotion recognition. *IEEE Transactions on Affective Computing*, 2024.