# 绪论

（讲抑郁症+机器学习在抑郁症诊断中有多么有用）

抑郁症，作为一种高发的心理障碍，近年来已在全球范围内引起广泛关注。根据世界卫生组织（WHO）的统计，全球每年有超过2.8亿人受到抑郁症的影响，国内抑郁症患病率为 6. 9%[2.4]且其发病率程逐年上升趋势[1]。抑郁症不仅会严重影响个体的情绪和行为，还可能导致认知障碍、自残和自杀等极端行为，给社会和经济带来沉重的负担[2][3]。

抑郁症是导致全球精神和身体残疾的主要原因,据估计,到 2030年,抑郁症可能成为第一致残疾病[2.2]。 有研究发现, 抑郁症终身患病率可达18. 1%[2.3]

MD很难诊断：与大多数身体疾病相比，常见的症状，如悲观、自卑的情绪，更主观，因此更难发现。因此，大约33%的抑郁症患者在临床诊断程序中没有被识别出来，只有不到40%的抑郁症患者得到了适当的治疗s29]。幸运的是，抑郁症在某些情况下是可以治疗的，例如早期诊断s24，34]，但让大多数人都能获得如此大规模的诊断将大大增加心理学家的工作压力，这表明了辅助工具对MD端到端自动估计的重要性。

讳疾忌医的情况在青少年中格外明显：超过四成(41.8%)的家长在孩子主动求助3次及以上才带其就医，62.8%的家长认为孩子存在中等程度及其以上的就诊延误，在青少年期患抑郁症对其人生的影响又格外巨大：诊断情绪障碍的儿童青少年中曾经有过请假经历的占比82.3%，有过休学经历的占53.85%，平均休学次数为1.71次。第一次休学的平均年龄为13.74岁，主要集中在14岁。

社交媒体和抑郁的关系：野生vlog

随着智能化技术的发展，越来越多的研究着眼于使用机器学习的方式对抑郁症患者的有关数据进行分析，以达到对其做出更早更准确的诊断的目的。如今人们越来越倾向于在社交媒体上分享自己的生活状态，由此也产生了不少基于用户的社交媒体数据来预测其抑郁症倾向的研究，但截止到目前，多数研究都是基于Reddit[t52，71，59]和Twitter[t63，10，40]等社交媒体网站的文本中检测抑郁症，这可以归因于注释的相对容易性（自我提及[t86，88]，某些子版块的参与[t53，32]）以及在相关上下文中（即作为一个线程）可以检索到的大量可用文本数据。此外也有一些仅限于图像的抑郁症检测研究[56，30，11，24，14，83]。然而，许多社交媒体网站（YouTube、TikTok、Instagram）专注于用户生成的视频内容，目前，从野生视频中检测抑郁症在很大程度上仍未得到探索。视频模式为抑郁症检测带来了一系列额外的挑战，因为它的噪音明显更大，视频长度变化很大（从几秒钟的短片到半小时的视频博客，在这些视频博客中，人直接与相机交谈），面部和身体并不总是可见的，背景变化很快，音频质量很低，尤其是在业余视频中。上传的视频大多脱离上下文，是独立的，而不是社交媒体帖子中的互动讨论。在这种情况下，任务是处理视频中受试者行为的微妙抑郁线索。

非语言线索：由于目前仍未出现中文的抑郁症标注vlog数据集，使用语言线索对抑郁症进行预测会出现在中文环境中泛化困难的问题；此外无论是实验室环境中的抑郁症分类数据集还是野生vlog数据集，研究者在对视频进行抑郁症标注时，一般是基于视频主体的自我报告，因此基于语言线索的视频抑郁症预测项目的高准确率，很大程度上来自于模型检测到了视频主体明确提到了“抑郁症”、“自我伤害”等字眼，这种类似于“过拟合”的预测模式，不易于模型在野生视频中的泛化与早期抑郁倾向的检测，于本项目的目标存在分歧。

本项目提出的模型：tcn-transformer(TCN-T)，本项目使用了五种预训练的模型进行基于视频的模态提取

本项目使用到的五种模态：音频，面部肌肉运动，面部标志，头部姿势，眼睛注视

本研究的旨在通过用户在社交媒体中发布的野生vlog视频，仅使用非语言线索，来对其的抑郁水平进行预测，以达到方便快捷的进行抑郁症的早期识别和评估的目标

相比传统工具，该系统能够更全面地捕捉用户的心理特征，显著提高抑郁症预测的准确性与鲁棒性，为心理健康领域的研究提供新的理论依据与实践路径。从社会价值来看，抑郁症的早期识别对其干预的及时性与治疗的实施都至关重要，本课题开发的智能化评估系统，通过简单易用的界面和高效精准的评估能力，降低了心理健康服务的门槛，特别是为偏远地区及心理服务资源匮乏的人群抑郁症诊疗提供了可行的辅助工具。综合上述，本系统的应用能帮助用户尽早发现心理健康问题，进行及时的干预和治疗，减少因抑郁症导致的社会和经济损失，同时推动心理健康服务的智能化和普惠化发展。

# 文献综述

**单模态->多模态（[4]）**

目前,主流的抑郁症检测方法主要可以分为3类:基于视频的检测、基于音频的检测和基于多模态的检测。抑郁症患者常常表现出面部表情的减少或呆滞,他们的面部表情可能缺乏生动度和情感表达。研究者通过机器学习分析面部特征在辅助诊断抑郁症方面取得了比较大的进展[4.2]。例如,孙浩浩等[4.3]基于人脸图像的全局和局部特征,构建了一种融合通道层注意力机制的多支路卷积网络模型。音频作为传达情感的媒介,抑郁症患者和非抑郁症患者之间的言语模式存在明显的差异[4.4]。MA等[5]提出了DepAudioNet深度模型,结合卷积神经网络(CNN)和长短期记忆(LSTM),用于编码声道中的抑郁症相关特征,从而提供更全面的音频表示,取得了较好的检测效果。这些深度学习架构在提取有意义的音频或视频特征以及提高抑郁症检测的效果方面发挥着重要作用。然而,仅依赖音频或者视频特征可能会丢失测试对象的其他动态信息,从而限制了抑郁症检测的性能。

除了上述单模态方法,通过多模态信息融合方法整合多种模态在提高抑郁症检测性能方面也显示出不错的效果。HE等[4.8]通过特征层融合将提取的音频和视频特征串联成一个高维特征向量,并使用支持向量回归(SVR)进行抑郁症预测。然而,这种方法容易产生高维特征表示,从而导致维度灾难。YANG等[4.9]将获得的音频视频结果和文本结果进行决策融合,以获得最终的抑郁症检测结果。但是,决策层融合单独考虑不同模态,无法捕捉它们之间的内在关系。更多的研究者通过模型层融合考虑模态之间的关系。NIU等[4.10]采用多模态注意力特征融合方法整合音频模态和视频模态。谷明轩等[4.11]结合了脑电信号和音频特征提出了基于全连接神经网络的多模态特征融合模型。但是,这些模型层融合方法在模态之间的交互性方面仍存在不足。近年来,Transformer[4.12]技术引起了广泛关注,Transformer模型中的编码器和解码器组件利用多头自注意机制捕捉输入序列数据的长距离上下文信息。ILIAS等[4.13]提出了一种将语言之外的信息融入基于Transformer的模型,用于社交媒体中抑郁症和压力检测,这一方法展现出良好的应用前景。

**语言线索->非语言线索**

写已有的语言线索的研究（多），已有的非语言线索的研究(少),然后说我们这个项目应该选用后者而不是选用前者

**特征提取的优越性**

1.一些文献说明，为什么我们要先模态提取而不是做仅仅基于像素的

2.一些文献说明，为什么我们选这四种模态（五种）是最有用的

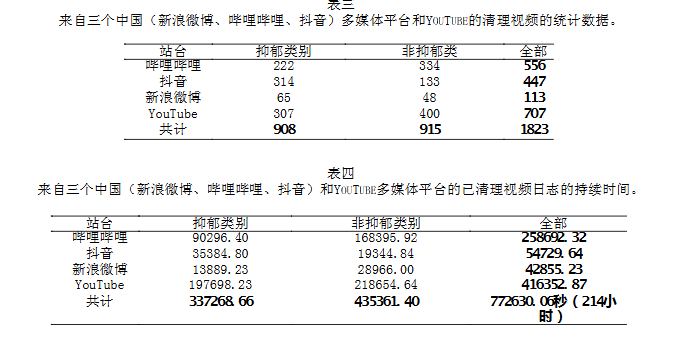
**三种层面的融合方式的比较(我在哪看到过来着？)**

此项目选用了后两种模型融合方法

# 研究方法

## 讲数据集的数据缺失

共有1823个样本，其中6个样本为空白缺失样本，即共有1817个可用于训练的样本，去除缺失样本后，抑郁有：908-1=907，非抑郁有：915-5=910，类别较为平衡，对空缺模态进行补0处理；抑郁是正样本，是1；非抑郁是负样本，是0。



## 特征提取

在本研究中，为全面捕捉个体面部表情及头部运动特征，构建了基于多模态数据的特征表示体系。主要提取的特征包括面部动作单元（Action Units, AUs）、头部姿态（Head Pose）、面部关键点（Landmarks）以及眼动（Eye Gaze）信息。

AUs

面部动作单元（Action Units, AUs）是由心理学家Ekman和Friesen提出的，用于描述面部肌肉运动的基本构成单元。它们构成了面部动作编码系统（Facial Action Coding System, FACS）的核心，通过对面部各部位肌肉群活动的精细划分和定量分析，实现了对面部表情的系统描述。每个动作单元对应于特定肌肉或肌群的收缩或放松，例如眉毛的提升、眼睑的闭合或嘴角的拉伸等。通过组合不同的AUs及其激活强度，可以准确地反映出个体的情感状态、意图及其他非语言交流信息。AUs已广泛应用于情感识别、心理学研究、人机交互以及计算机视觉等领域，成为解析和理解面部表情的重要工具。本项目选用了与抑郁情绪表达相关的17个AU，其含义如下表所示。

|  |  |
| --- | --- |
| 编号 | 对应面部动作及含义 |
| AU01 | 内侧眉毛上提：眉毛内侧上抬，常见于惊讶、困惑等表情。 |
| AU02 | 外侧眉毛上提：眉毛外侧上抬，多与惊讶或好奇等情绪相关。 |
| AU04 | 皱眉：眉毛向内及向下收缩，通常出现在愤怒、专注或困惑状态。 |
| AU05 | 上眼睑提拉：上眼睑上抬，可能伴随眼部紧张或情绪激动。 |
| AU06 | 提颊：眼部周围皮肤上提，导致面颊隆起，常见于微笑时的面部表情。 |
| AU07 | 眯眼 |
| AU09 | 鼻皱：鼻梁区域出现皱纹，通常反映出厌恶、困惑或不适的情绪。 |
| AU10 | 上唇提升：上唇向上抬起，常用于表达轻蔑、不满或怀疑。 |
| AU12 | 嘴角上扬：嘴角向上拉，是微笑或愉悦表情的典型标志。 |
| AU14 | 嘴角挤压：嘴角轻微收缩或挤压，反映出真实微笑中的细微变化。 |
| AU15 | 嘴角下拉：嘴角向下拉，通常与悲伤、失落或消极情绪相关。 |
| AU17 | 下巴上提：下巴区域上提，可能反映出紧张、决心或咬牙动作。 |
| AU20 | 嘴唇拉伸：嘴唇向两侧扩展，常见于紧张、恐惧或不适时的面部反应。 |
| AU23 | 嘴唇紧闭：嘴部肌肉紧缩，表现为情绪抑制、不满或集中注意力。 |
| AU25 | 嘴唇分开：嘴巴轻微张开，常见于放松、惊讶或自然呼吸状态。 |
| AU26 | 下巴下垂：下巴部分放松或下垂，可能伴随疲惫、放松或轻微惊讶。 |
| AU45 | 眨眼 |

Head Pose(6维，头部平移三维，头部旋转三维)

头部姿态特征主要用于描述被试在三维空间中的位置与方向。本文采用六个参数来表示头部运动信息，其中包括头部在空间中的平移量（三个维度：x、y、z）以及头部旋转的角度（偏航、俯仰和横滚）。

Landmarks在

通过68个面部关键点（landmarks）来描述人脸各个局部区域的几何形状和位置分布，是一种被广泛使用的标准表示方式，这些关键点通过对面部主要结构（如下颌、眉毛、鼻子、眼睛及嘴巴）的精细定位，实现了对人脸几何形态的精确描述。



Eye Gaze

眼动信息主要反映个体视觉注意力的分布及注视点变化。通过跟踪眼球运动，提取出眼球注视方向及轨迹数据，这些数据在揭示个体的认知状态与情感反应中具有重要意义。本项目使用一个8维向量来描述视频主体的眼动信息，具体含义如表（）所示，其中gaze\_angle\_x、gaze\_angle\_y是基于头部方向归一化后计算的整体注视方向。

|  |  |
| --- | --- |
| **特征** | **含义** |
| gaze\_0\_x, gaze\_0\_y, gaze\_0\_z | 左眼的三维注视方向向量 |
| gaze\_1\_x, gaze\_1\_y, gaze\_1\_z | 右眼的三维注视方向向量 |
| gaze\_angle\_x | 水平方向的注视角度 |
| gaze\_angle\_y | 垂直方向的注视角度 |

## 数据处理：

主要是我如何处理视频和音频数据使之缩小数据量并对齐的

**1. 读取数据：**

* 程序首先通过 os.listdir 获取指定路径 trainPath 下的所有文件。然后它只选择 .csv 格式的文件进行处理。
* 使用 pandas.read\_csv() 读取每个 CSV 文件，并将其转换为一个 numpy 数组 data。

**2. 处理帧数据：**

* 调用 validFrame() 函数对读取的帧数据进行处理。具体来说：
  + 遍历所有帧，找到最后一个有效帧（即第5列为1的帧），并将其保存为 validFrame。
  + 再次遍历所有帧，对于无效帧（第5列为0），将该帧从第6列开始的数据替换为最近有效帧的数据。
  + 处理完成后，返回更新后的帧数据。

**3. 数据抽样与填充：**

* 调用 split() 函数对数据进行抽样和填充。该函数执行以下操作：
  + 选择数据的前 5490 行，并将其传递给 chouzhen() 函数，进行抽样处理，每隔6行取一次数据。
  + 如果抽样后的数据行数少于 915 行，则使用零值填充，直到行数达到 915 行。
  + 最后返回处理后的数据。

**4. 删除不需要的列：**

* 使用 numpy.delete() 删除数据中的前 5 列（即列索引 0 到 4），这些列包含不需要的特征。

**5. 提取不同区域的特征：**

* **gaze 特征**：提取数据的前 6 列作为 gaze 特征。
  + 创建一个与 gaze 形状相同的全零数组，并将其与 gaze 水平拼接，得到更宽的 gaze 特征。
* **pose 特征**：提取数据中第 288 到 293 列作为 pose 特征。
* **features 特征**：提取数据中第 294 到 429 列作为 features 特征。
* **au 特征**：提取数据中第 430 到 447 列作为 au 特征。

**6. 拼接特征：**

* 将提取出的 gaze、pose、features 和 au 特征水平拼接，形成最终的特征矩阵 feature。

**7. 检查 NaN 值：**

* 对拼接后的 feature 矩阵进行检查，确保其中没有 NaN 值。若存在 NaN 值，则打印文件名进行提示。

## 模型架构

###### 图1.本项目提出的方法的总体架构

本项目的目标时通过给定一个发表在常见社交媒体上的vlog数据集，训练一个模型来估计视频中的主体是否患有抑郁症。在这种设定下，抑郁症检测相当于视频级别的分类。然而，如果没有不切实际的大量标注数据，直接从像素进行高级心理疾病的分类是不可行的。因此，在我们的工作中，我们仅考虑了高级别的非语言线索，并丢弃了任何文本信息以消除对话主题的偏差[t80]。大多数抑郁症数据集的收集方式是基于直接提及的诊断[t86,88]，通过仅使用非语言线索，模型的性能更能反映其在现实环境中的表现。非语言线索是使用最先进的预训练模型[t54,70,15,81,90,89,改]提取的，随后进行处理以进行分类。图1提供了我们从视频中进行抑郁症分类的流程的高级概述。

正式地来讲，给定一个带有抑有症标签的视频数据集 ，其中是视频，是其对应的标签，目标是找到模型的最优参，以最小化整个数据集上的平均交叉熵损失：



其中，表示交叉熵损失函数。

### MDDformer 的结构

为了有效检测视听线索中的判别模式，提出了 MDDformer。在输入 MDDformer 之前，定义音频特征为  ，视频特征为  。其中， $D\_{a}$ 表示音频的维度，  表示视频的维度，  表示序列的长度。最初，Transformer 架构［L43］被提出用于建模自然语言处理（NLP）任务中的关系，其由编码器－解码器结构组成。编码器和解码器均由多个相同的层组成，每层包含两个主要子模块，即多头自注意力机制和逐位置前馈网络。在我们的任务中，为了融合音频和视频分支的模式，提出了 MDDformer。

### 输入特征映射

输入音频特征  通过三个线性变换映射为三个矩阵，即键  ，查询  和值  ：



其中， , 和表示线性变换的权重。

### 特征生成与融合

将  与  相乘生成特征  ，将  与  相乘生成特征  。然后，将  和  拼接生成特征图  ：

### 自注意力机制

音频分支的自注意力可以表示为：其中， $d\_{k}$ 是 $F\_{a v}$ 矩阵的维度。

### 多头注意力与特征融合

将每个头的输出拼接并重塑，与特征  相加，生成融合特征  ：其中，  是多头自注意力的头数，  和  是权重矩阵。前馈网络与归一化在拼接操作之后，将  输入前馈网络，并采用加法和归一化操作生成特征  ：其中，Norm 表示归一化操作，FFN 表示前馈网络。

### 多层感知机

使用多层感知机（Multilayer Perceptron，MLP）进行最后的抑郁症分数预测。该过程可以表示为\begin{array}{c}\qquad Z=X\_{v}^{\prime} \oplus X\_{a}^{\prime} \oplus X\_{r}^{\prime} \in R^{3 k} \\\text { Attention }(Z)=\operatorname{softmax}\left(\frac{\boldsymbol{Q}\_{m} \boldsymbol{K}\_{m}^{\mathrm{T}}}{\sqrt{d\_{k}}}\right) \boldsymbol{V}\_{m} \in R^{3 k} \\y=M L P(Z)\end{array}其中， y 是最后的抑郁症得分。

## 结果

### 实验设置

我们使用 PyTorch 深度学习工具包实现并训练了 MDDformer 模型。为了验证 MDDformer 的有效性，采用了 10 折交叉验证。Adam 优化器的参数设置为  和  ，批量大小为 4 。初始训练学习率为 0.00001 ，随后通过余弦退火学习率衰减（CosineAnnealingLR）进行更新。为了克服过拟合问题，在线性层中采用了 0.2 的 dropout(Dropout 会在每次前向传播时，按某个预设的概率 p 随机“丢弃”网络中的一部分神经元（即让它们的激活值变为零）)，总训练轮数设置为 300 。我们的架构在四块 NVIDIA Tesla V100－DGX（32GB）上进行评估。

### 学习率更新

初始学习率为  ，通过余弦退火学习率衰减更新：其中，  是最小学习率，  是当前训练步数，  是总训练步数。（这个调度器会随着 epoch 的进行，使得学习率从初始学习率逐渐衰减到 0。具体的学习率值会根据当前 epoch 和总的 epoch 数量来计算，并按照余弦函数变化，达到平滑衰减的效果。）

### 评估指标

通常，二分类问题的分类性能主要通过五个指标进行评估：准确率（Accuracy）、精确率（Precision）、召回率（Recall，也称为灵敏度）、特异性（Specificity）和 F1 分数（F1-score）。其中，真正例（True Positive, TP）表示正类样本被正确预测为正类；真负例（True Negative, TN）、假正例（False Positive, FP）和假负例（False Negative, FN）分别表示负类样本被正确预测为负类、负类样本被错误预测为正类以及正类样本被错误预测为负类。相关公式如下：

1．准确率（Accuracy）



2．精确率（Precision）



3．召回率（Recall）



4．F1 分数（F1－score）



### 基线方法的性能

为了进一步验证 MDDformer 的性能，我们采用了多种机器学习和深度学习架构作为基线方法，包括 KNN、SVM、LR（逻辑回归）、RF（随机森林）、Xception、ViT（视觉 Transformer）、BiLSTM 和 SEResnet。为了公平比较，采用了加权准确率、精确率、召回率和 F1 分数作为评估指标。如表 V 所示，MDDformer 在所有评估指标上均取得了最佳性能。术语“add”和“concat”分别表示加法和拼接操作。在我们的任务中，模型按准确率升序排列。从表 V 中可以看出，准确率值分别为 58.34%、64.66%、64.88%、69.23%、70.99%、71.16%、71.38%、72.31%、72.54%、72.59%、72.92%、73.03% 和 76.88%。

# web系统架构

# 结论