**基于MetaBCI的人工智能睡眠分期工具**

**队伍名称： 迈动数康 队伍单位：杭州迈动数康科技有限公司**

张行健1\*, 武晓龙2, 何戈文1, 赵康卿3, 尚珽钰1

1 杭州迈动数康科技有限公司

2 巴斯大学

3 北京大学第六医院

\* 主要联系人. Tel.: (+86) 15858205121; email: xingjian.zhang@mindmatrixes.com

***摘要***—睡眠是生命中最重要的基本过程之一，然而，世界上有大量人群因长期睡眠质量不佳而导致严重的健康问题。为了有效解决该问题，需要通过睡眠分期来监测睡眠质量。本项目旨在基于MetaBCI平台开发高效的睡眠分期算法，在临床睡眠研究、个人健康监测和睡眠质量评估等场景为用户提供精确的睡眠分期信息，帮助识别睡眠障碍和优化睡眠质量。预期在MetaBCI的Brainda中新增对睡眠分期多任务的支持和新增单通道和多通道的算法模型等。技术路径包括基于生理信号的算法开发，含单通道及多通道数据处理、模型训练与测试，以及结果展示等。项目预期应用效果是使新用户能够使用MetaBCI平台便捷地开发出基于单通道或多通道数据的人工智能睡眠分期工具，从而提高睡眠分期的准确性。

### ***关键词*: *睡眠分期*、*多任务兼容、多通道兼容、人工智能、便捷高效***

### 1 项目背景

MetaBCI是一款针对BCI的开源平台，它可以帮助使用者完成快速获取数据，快速解码数据等一系列BCI实验所需操作，保证使用者可以便捷的完成BCI实验并且搭建算法模型。

睡眠分期是一类常见的BCI实验。睡眠是人类生命中最重要的基本过程之一，睡眠的质量关乎着人类的身体健康，心理健康，免疫系统等多方面的健康问题。但是，许多人正面临睡眠障碍。通过分析睡眠分期是监控睡眠质量的重要方法，睡眠分期指的是睡眠过程中不同的阶段，包括清醒期（W）、浅睡眠（N1）、中度睡眠（N2）、深睡眠（N3）和快速眼动睡眠（REM）。

专业的睡眠分期通常在医院或者睡眠实验室中，记录受试者的脑电（EEG）、眼电（EOG）、肌电（EMG）、心电（ECG）等通道的数据，睡眠专家根据多导睡眠图（PSG）按照标准的睡眠分期指南（如 AASM）对给时间段进行标注分期。这一过程繁琐，需要大量的通道的数据，且需要消耗大量人力进行标注。基于深度学习的睡眠分期可以大大减少人力耗费且准确高效。

本项目的睡眠分期算法是基于Emadeldeen Eldele 提出了一种名为AttnSleep 的基于注意力的新型深度学习架构。我们对模型的细节做出了修改以使用于MetaBCI的架构中，并且增加了多通道脑电信号的适配。

### 2 设计架构

### 2.1 分层架构

我们将睡眠分期这一任务拆解为多个功能模块，并根据其功能的不同将这些模块添加到对应的大类中区。本项目的分层架构包括：

1. UI层：提供给用户一个可交互的UI界面，可以实现数据可视化和分析可视化。
2. 逻辑推理层：提供核心算法，包括模型搭建和训练，结果评估和预测。
3. 数据获取层：提供数据获取，预处理和存储功能。
4. 运行层：根据需求运行项目。

### 2.2 模块化设计

分层架构中的每一层有都具有多种功能，细节如下：

首先是数据获取模块，包含三步，1.获取原始数据集可以是本地加载或者在线下载，2.数据集预处理，合并标签和数据，存储到本地3.读取npz数据集。我们针对每一种不同的数据集都封装成了模块。

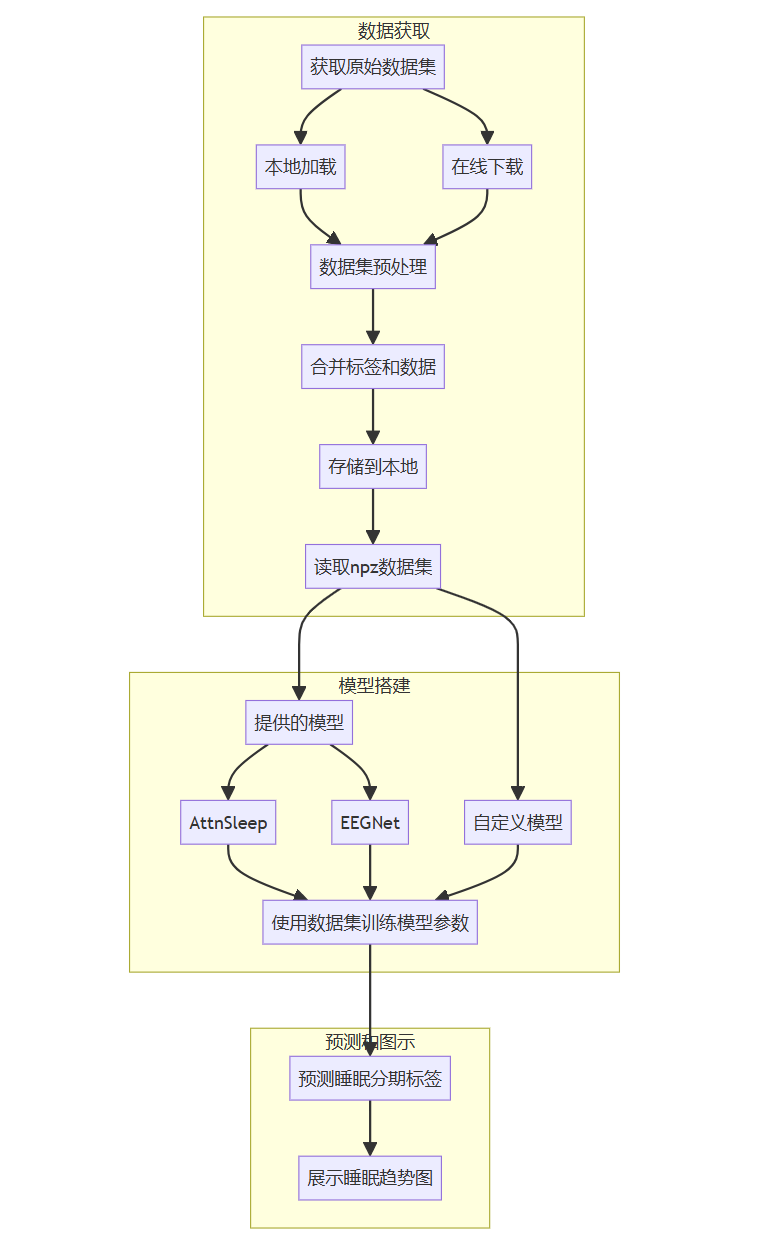
其次，搭建模型模块，可以选用提供的attnsleep和eegnet，或者自定义模型。使用数据集训练模型参数并保存（可以使用交叉验证法）。

最后，结果展示模块，训练好的模型进行预测并展示睡眠趋势图。每一个步骤封装为类或者方法，以增加便利性和可读性。

### 3 技术路径

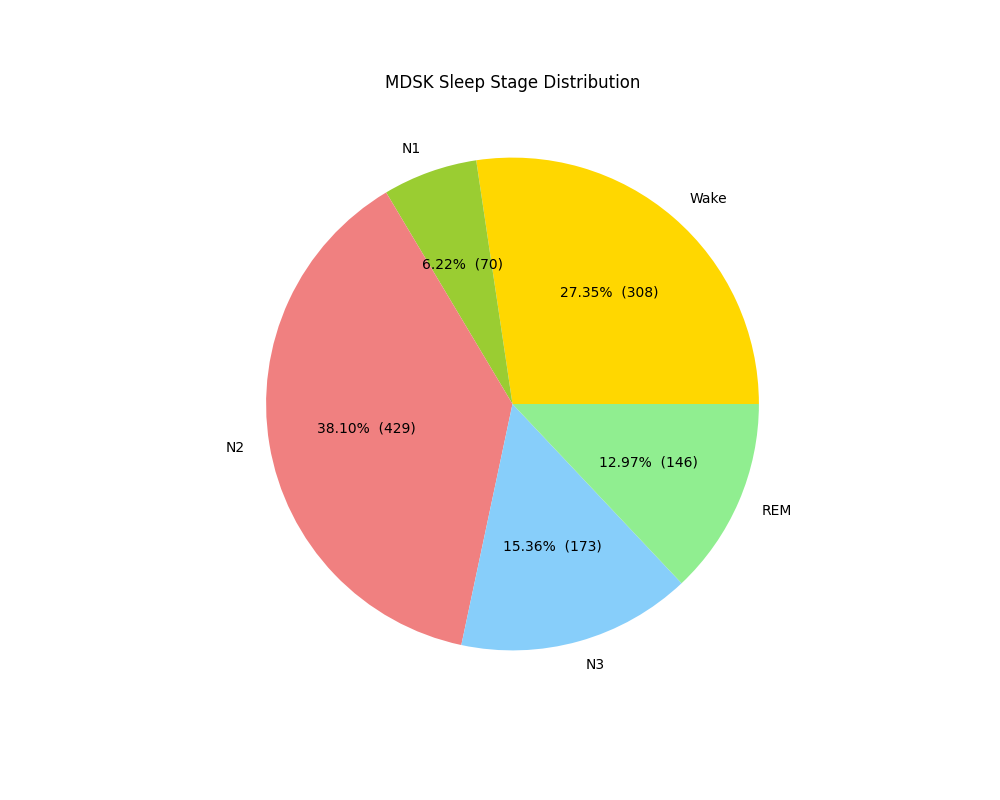
首先我们翻阅了大量文献来支持人工智能睡眠分期的技术可行性。下面列举该项目的部分具体技术路径：

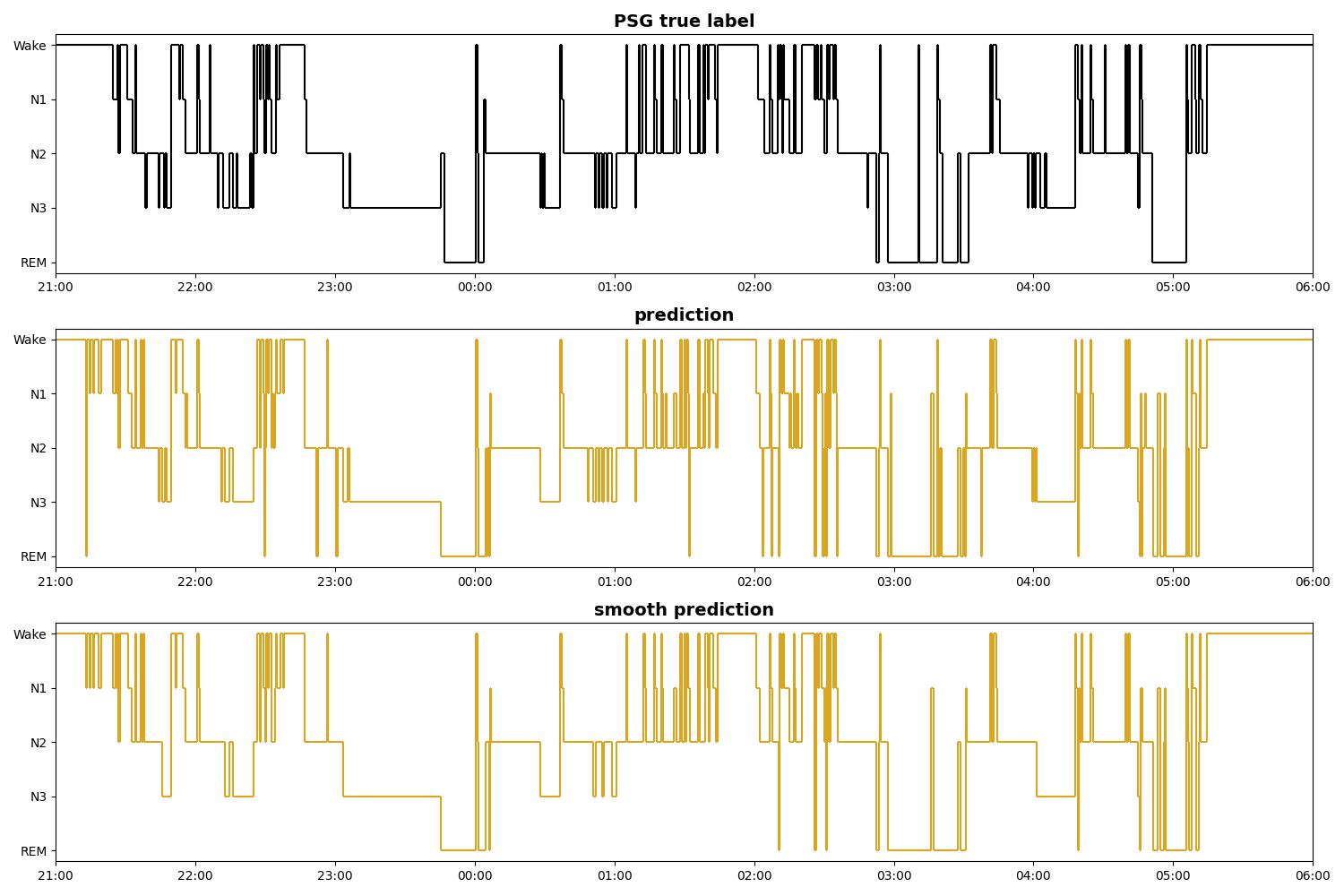
1. 原始数据集获取，其中原始数据的格式都为edf，标签的格式则不一。Sleep\_telemetry和SleepCassette数据集有公开下载方式，我们将下载方式直接封装进数据集方法中。对于大部分非公开的数据集我们提供了网盘下载方式可供自行下载。
2. 数据预处理：从原始数据源中获取EEG数据和标签。确保数据和标签的时间戳对齐。读取并处理标签数据，计算其在EEG数据中的位置。去除不需要的数据，处理数据长度不匹配的情况。数据分割和保存：将数据分割为多个30秒的epochs。保存处理后的数据和标签为.npz文件，以便进一步使用。
3. 模型搭建。使用Attnsleep模型，关键技术路径有：多分辨率卷积神经网络 (MRCNN)：处理输入数据，提取多尺度的特征信息。自适应特征重标定 (AFR)：优化特征表示，提升模型对重要特征的关注。时间上下文编码器 (TCE)：结合多头注意力 (MHA) 捕捉时间依赖关系，提高时间序列数据的处理能力。
4. 对于双通道数据，我们将双通道数据从输入中分离出来，以适配各自的 MRCNN 处理。然后分别对 EEG 和 EOG 通道的数据进行 MRCNN 特征提取，以捕捉各自的特征。接着将处理后的 EEG 和 EOG 特征拼接在一起，以形成一个综合特征表示。最后通过全连接层进行最终的分类预测。
5. 三通道数据的合并方式与双通道数据基本类似。
6. 构建了一个工厂类AttnSleep ，根据输入的参数（通道数量和类别数量）来实例化并返回相应通道的AttnSleep 模型，实现对整个模型的封装。
7. 构建了一个SkorchNet\_sleep类，该类与base中的SkorchNet类似，目的是为了设计模型训练的参数。使用装饰器模式将SkorchNet\_sleep类应用于AttnSleep，来达到模型训练和预测的目的
8. 构建了两个图示方法，通过输入预测的标签来画出睡眠分期各阶段占比饼图和睡眠分期趋势图（在应用效果中给出展示）。

流程图如下：

### 4 应用效果

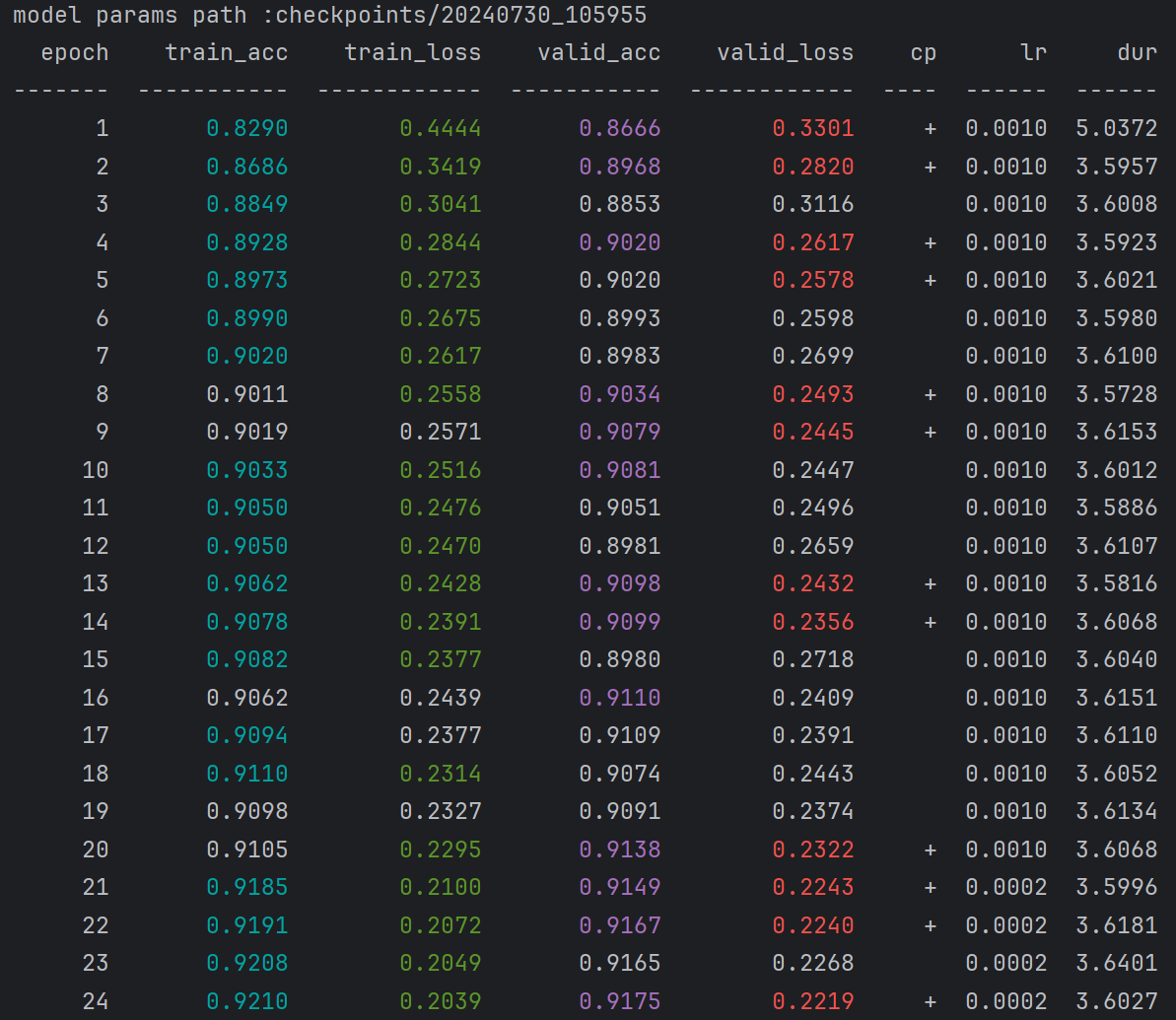
### 4.1 睡眠分期结果图示





### 4.2 模型训练效果图

训练结果保存的地址在同路径下，文件名称会根据时间自动输出。



**代码功能明细表**

1. 【**必填】代码开源仓库网址（Github 或 Gitee）**：

https://github.com/Shangtingyu/MetaBCI

1. 【**选填】项目视频网址**：
2. 【**必填】新增功能点说明**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 子平台 | 功能点描述 | 代码路径 | 类与函数 |
| 1 | Brainda | 新增睡眠分期数据 | metabci/brainda/datasets/sleep\_edf.py | 1.class Sleep\_EDF  2.class Sleep\_EDF\_data  3.eeg\_filter() |
| 2 | Brainda | 新增睡眠AttnSleep分期算法 | metabci/brainda/algorithms/deep\_learning/AttnSleep.py | 1.class AttnSleep  2.class SkorchNet\_sleep |
| 3 | demo\_sleep | 新增eegnet实现睡眠分期任务 | demo\_sleep/eegnet-run.py | 1.main() |
| 4 | demo\_sleep | 新增交叉验证功能 | demo\_sleep/cross\_validation.py | 1.cross\_train\_model() |
| 5 | demo\_sleep | AttnSleep实现睡眠分期任务 | demo\_sleep/Attnsleep\_run.py | 1.main() |
| 6 | demo\_sleep | 新增保存预测结果功能 | demo\_sleep/predict.py | 1.save\_res\_pre() |
| 7 | demo\_sleep | 新增预测结果功能平滑处理功能 | demo\_sleep/smooth.py | 1.smooth() |
| 8 | demo\_sleep | 新增分期结果图示化功能 | demo\_sleep/show.py | 1. plotAnalyze() 2. plotTime() |
| 9 |  |  |  |  |

1. 【**必填】修复功能点说明**

[说明：填写对MetaBCI进行修复、增强、优化的功能点，及该功能点在项目仓库中的代码路径、相关类和函数。（灰色字体为示例）]

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 子平台 | 功能点描述 | 代码路径 | 类与函数 |
| 1 | Brainda | 对base类抽象方法完善 | metabci/brainda/datasets/base.py | 1.save\_processed\_data()  2.get\_processed\_data() |
| 2 | Brainda | 完善fit方法，使其能传入指定训练集和测试集 | metabci/brainda/algorithms/deep\_learning/base | 1.fit()  2.get\_split\_datasets() |
| 3 | Brainda | 补充1d平均池化功能 | metabci/brainda/algorithms/deep\_learning/base | 1.AvgPool1dWithConv |
| 4 | Brainda | 随着数据集的大小自动调整学习率和权重衰减 | metabci/brainda/algorithms/deep\_learning/Attnsleep.py | 1.class NeuralNetClass  2.adjust\_lr（）  3.adjust\_weight\_decay（） |
|  |  |  |  |  |