

无训练稳态视觉诱发电位脑-机接口比赛系统说明

1. 项目简介

实际应用中，通常对脑-机接口系统的易用性具有较高要求。某些应用场景下，难以获取足够的个体数据进行脑-机接口系统预先训练，因此需要研究在无训练数据条件下的脑-机接口分析方法。

本项目基于稳态视觉诱发电位脑机接口数据，测试不同算法在缺乏预训练数据条件下的识别性能。并根据多名受试者测试数据中的信息传输速率(Information transfer rate, ITR)确定比赛成绩。

2. 任务介绍

2.1 任务目标

本项目中，参赛者需要编写稳态视觉诱发电位脑-机接口识别算法，在无训练数据条件下对包含有稳态视觉诱发响应的 EEG 信号进行识别分类，最终判定目标 EEG 信号中包含有哪一类刺激所诱发的稳态视觉诱发响应。

比赛系统提供一套标准程序接口，由参赛者编写参赛算法并给出具体实现。在参赛算法通过数据读取方法获取数据，并通过反馈方法报告识别结果。使用有效数据越短，识别准确率越高，则最终成绩越高。

系统主要包括三个接口方法：

- 1) 算法执行方法：AlgorithmInterface.run()
- 2) 数据读取方法：ProblemInterface.getData()
- 3) 反馈方法：ProblemInterface.report()

详细说明参见第 7 节

2.2 数据简介

2.2.1 实验范式简介



刺激范式包含 40 个目标，刺激频率分别 8~15.8Hz，间隔 0.2Hz。

实验数据以 block 为单位，每个 block 中均为连续采集 EEG 数据，其中包含数量不定、次序随机的刺激试次。单一 block 中，每种刺激试次的出现次数也不固定。单个试次持续 3.5-4.5s，开始 0.5s 为目标提示阶段，屏幕中会出现一个红块提示目标位置并持续 0.5s。之后 3-4s 为刺激阶段，刺激过程中，40 个目标同时闪烁，每个目标的亮度按照其预定频率呈正弦变化，受试被要求严格注视提示的目标，从而在其 EEG 信号中产生稳态视觉诱发响应。每一个试次中，trigger 被记录在刺激阶段起始位置。

2.2.2 实验数据简介

实验数据为降采样到 250Hz 的 EEG 数据，但并未经过其他滤波处理。在检测过程中，参赛者可根据自身算法选择合适的预处理滤波。

数据流采用模拟在线方式提供。每调用一次数据读取

方法，可获得一个新数据包，数据包中包含一定长度的实验 EEG 数据，以及在该数据包记录过程中收到的 trigger 信息。

在同一 block 中，数据包按照时间顺序依次发送。当一个 block 中数据发送完毕后，最后一个数据包中将 block 终止标记 blockEndFlag 将被置为 1。

若测试数据中包含多组 block 数据，则一组 block 数据发送完毕后，数据读取方法被再次调用时，将会开始下一组 block 数据的 EEG，同时数据包内 blockID 也会随之更新。参赛算法可以根据 blockID 的变化确定数据包是否在同一个 block 之中。

而当所有实验数据发送完毕后，程序终止标记 finishedFlag 将被置为 1。参赛算法检测到 finishedFlag 为 1 后，需要自行结束 run() 方法执行。

需要指出，由于实验数据来自真实 EEG 信号，每个 block 中最后一个数据包的长度可能不是一个定值，在算法开发过程中请特别注意。

数据包具体格式参见 7.4 节

2.2.3 数据导联编号表

通道序号	1	2	3	4	5	6	7	8	9	10
导联名称	FP1	FPZ	FP2	AF3	AF4	F7	F5	F3	F1	FZ
通道序号	11	12	13	14	15	16	17	18	19	20
导联名称	F2	F4	F6	F8	FT7	FC5	FC3	FC1	FCZ	FC2
通道序号	21	22	23	24	25	26	27	28	29	30
导联名称	FC4	FC6	FT8	T7	C5	C3	C1	Cz	C2	C4

通道序号	31	32	33	34	35	36	37	38	39	40
导联名称	C6	T8	M1	TP7	CP5	CP3	CP1	CPz	CP2	CP4
通道序号	41	42	43	44	45	46	47	48	49	50
导联名称	CP6	TP8	M2	P7	P5	P3	P1	Pz	P2	P4
通道序号	51	52	53	54	55	56	57	58	59	60
导联名称	P6	P8	P07	P05	P03	P0z	P04	P06	P08	CB1
通道序号	61	62	63	64						
导联名称	01	0z	02	CB2						

2.2.4 反馈结果、目标频率、初始相位对照表

结果编号	1	2	3	4	5	6	7	8	9	10
目标频率 (Hz)	8.6	8.8	9.0	9.2	9.4	9.6	9.8	10	10.2	10.4
初始相位	0	0.5π	π	1.5π	0	0.5π	π	1.5π	0	0.5π
结果编号	11	12	13	14	15	16	17	18	19	20
目标频率 (Hz)	10.6	10.8	11.0	11.2	11.4	11.6	11.8	12.0	12.2	12.4
初始相位	π	1.5π	0	0.5π	π	1.5π	0	0.5π	π	1.5π
结果编号	21	22	23	24	25	26	27	28	29	30
目标频率 (Hz)	12.6	12.8	13.0	13.2	13.4	13.6	13.8	14.0	14.2	14.4
初始相位	0	0.5π	π	1.5π	0	0.5π	π	1.5π	0	0.5π
结果编号	31	32	33	34	35	36	37	38	39	40

目标频率 (Hz)	14.6	14.8	15.0	15.2	15.4	15.6	15.8	8.0	8.2	8.4
初始相位	π	1.5π	0	0.5π	π	1.5π	0	0.5π	π	1.5π

2.3 算法规范

参赛算法调用数据读取方法获取脑电数据。数据读取方法被调用一次，比赛系统会返回一个新数据包，参赛算法可以对新数据包进行缓存并处理。当算法认为接收到的数据足以满足判决条件时，需要调用反馈方法向比赛系统报告识别结果。比赛系统根据数据读取方法的调用次数计算出算法使用有效数据长度，并结合反馈正确率，综合计算出模拟信息传输速率。

参赛算法需要同时满足以下几个约束条件：

1) 试次起止约束：

在对单一试次数据的检测识别过程中，参赛算法需要在接收到该试次 trigger 之后开始检测，并且最迟在接收到下一 trigger 时进行反馈报告。否则，报告结果将被错判为后续试次的识别结果。

2) 单试次最大数据长度约束

本项目对于单一试次最长检测时间设置为 3 秒。从试次的 trigger 信号起，参赛算法最多采集 3 秒的 EEG 数据，就必须立刻给出反馈结果，否则该试次识别结果将被视为无效。

3) 算法终止约束

当接收到数据包中 Endflag = 1 时，意味着所有实验数据均已发送完毕，参赛算法需要停止处理并自行退出。

3. 评分准则

3.1 评分指标

本系统以模拟信息传输速率作为评分标准

$$\text{ITR} = \frac{60}{T} \left[\log_2 Q + P \log_2 P + (1 - P) \log_2 \left(\frac{1 - P}{Q - 1} \right) \right] \quad (1)$$

其中， Q 表示目标个数， P 表示识别正确率， T 表示平均试次时长。ITR的单位是 bits/min。

特别需要指出，本系统中 ITR 是按照理想 ITR 进行计算，即平均试次时间不包含模拟休息时长。为了避免歧义，当识别正确率低于机会水平 ($1/Q$) 时，最终评分将被置零。

3.2 评分细则

3.2.1 性能评估方法

参赛算法通过数据读取方法获取新数据包。当所得数据包内含有 trigger 信号时，评分系统将自动开始记录算法识别过程中所使用 EEG 信号的长度，直至反馈方法被调用。从 trigger 开始到反馈方法被调用时所获取的 EEG 数据长度将作为该试次的模拟试次时长。而平均准确率将根据算法反馈结果与真实刺激的一致性进行计算。

需要特别指出，在本比赛项目中每一个包含 trigger 的数据包，其依然被视为是前一试次的数据。而新试次数据是从包含 trigger 数据包的下一个数据包开始计算。因此参赛算法不可在获取到包含 trigger 信号的数据包时立刻反馈，而最早需获取到下一数据包后才可反馈。

3.2.2 结果反馈异常处理

1) 重复多次报告

在一个试次时间内，参赛算法多次反馈结果，则仅按照第一次反馈的时间及结果进行记录。

2) 结果未反馈

若在一个试次时间内，参赛算法未反馈结果，则该试次时长将被记录为 3 秒，同时判决结果将被记录为误判。

3) 结果反馈超时

结果反馈时，从当试次 trigger 开始计算参赛算法已经获取了超过 3 秒的 EEG 数据，则判决结果将被记录为误判，同时该试次时长将按照从 trigger 开始时刻到结果反馈时刻的数据长度进行计算。

4) 算法执行超时

为满足脑-机接口系统实时处理需求，本项目同时对参赛算法的计算复杂度有一定要求。本比赛项目将会根据比赛数据量大小确定一个计算时间。若算法复杂度过高导致系统运行超时，则该算法比赛成绩将被视为无效。

4. MATLAB 青年组比赛流程

1) MATLAB 青年组比赛分为两轮。赛前，参赛选手将通过提供的训练数据和比赛框架自行测试，并保证代码可运行。运行通畅后，各赛队需将最终代码在报名截止日期前（7 月 24 日）通过邮件的方式发送到邮箱 BCIRobotContest@163.com，并备注标题为“参赛选手唯一注册编号+选手姓名+联系方式+学校”。（未在规定时间内且正确备注邮件标题的赛队将视为未提交。）

2) 第一轮比赛将通过各参赛选手提交的代码，大赛该组别负责人将使用该赛题抽签数据 B 对提交的代码进行测试，得出排名，筛选排名前 50 位选手进入决赛，并反馈代码测试问题。

3) 总决赛期间，各参赛选手现场提交代码，根据代码和抽签数据 C 进行计分排名。

4) 赛题框架和测试数据下载方式如下：

链接：[https://pan.baidu.com/s/1gC8_-](https://pan.baidu.com/s/1gC8_-K4GBf2R7HgYcnrqoA)

K4GBf2R7HgYcnrqoA

提取码：07n4

5. MATLAB青年组报名要求

- 1) 报名时间：2022年6月1日-7月24日
- 2) 报名 MATLAB 青年组的参赛选手以个人为单位报名该赛题，并使用正版 MATLAB 软件。
- 3) 报名该赛题的参赛选手需为 2000 年 8 月 31 日以后出生的学生。
- 4) 若参赛选手没有 MATLAB 软件，可通过说明最后所列方式进行免费申请，使用时间为申请后至比赛当年结束。

6. MATLAB青年组奖项设置

奖项设置（以下为税前金额）：

- 一等奖一名：奖金 12000 元
- 二等奖两名：奖金 7000 元
- 三等奖三名：奖金 3000 元
- 单项鼓励奖若干：奖状

评选结果、奖金及奖状将由组委会公布并颁发。

MATLAB 青年组参赛赛队可在比赛准备及比赛期间免费申请软件使用权限，说明如下：

申请免费 MATLAB 软件许可：参赛选手所在院校如已购买 MATLAB 校园版（Campus-wide License）授权，可直接使用校园版软件参赛；如无校园版，建议申请使用 MATLAB 青年组专用的免费软件许可：该申请链接将在您报名预审通过后提供。

技术支持及资源

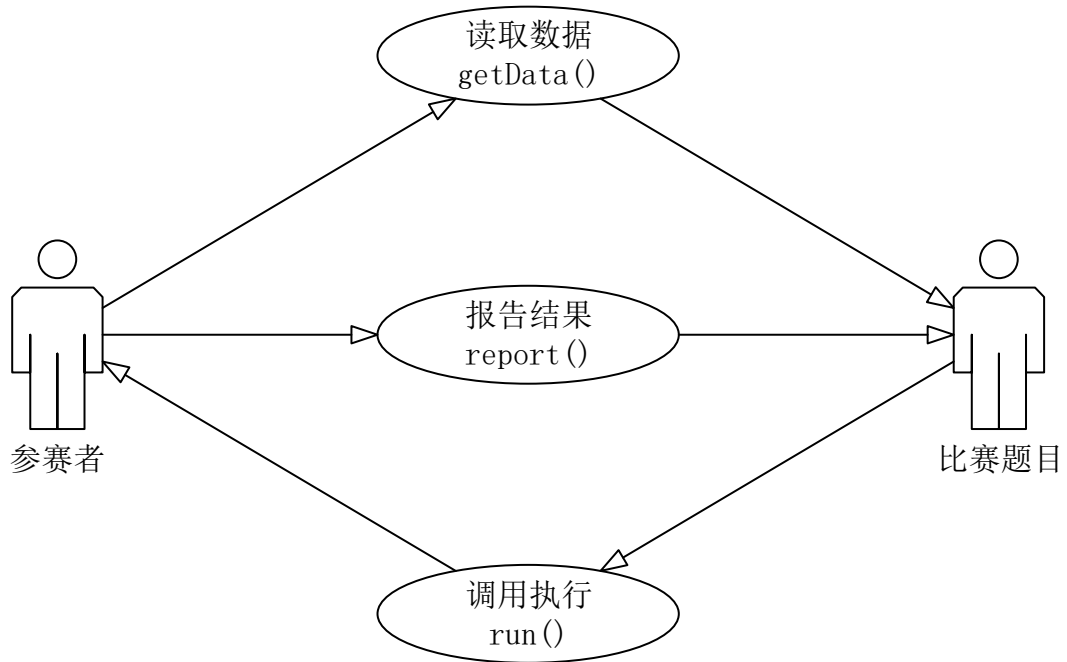
- [10+门免费在线交互式 MATLAB/Simulink 入门课程](#)（覆盖深度学习/机器学习/信号处理等）
- [小迈步系列 MATLAB/Simulink 直播课程](#)
- [MATLAB 中国](#)（哔哩哔哩官方账号）
- [MATLAB 官方微信公众账号](#)

- [MATLAB 中文论坛](#)（微信小程序搜索“ilovematlab”）
- [官网在线培训视频库](#)

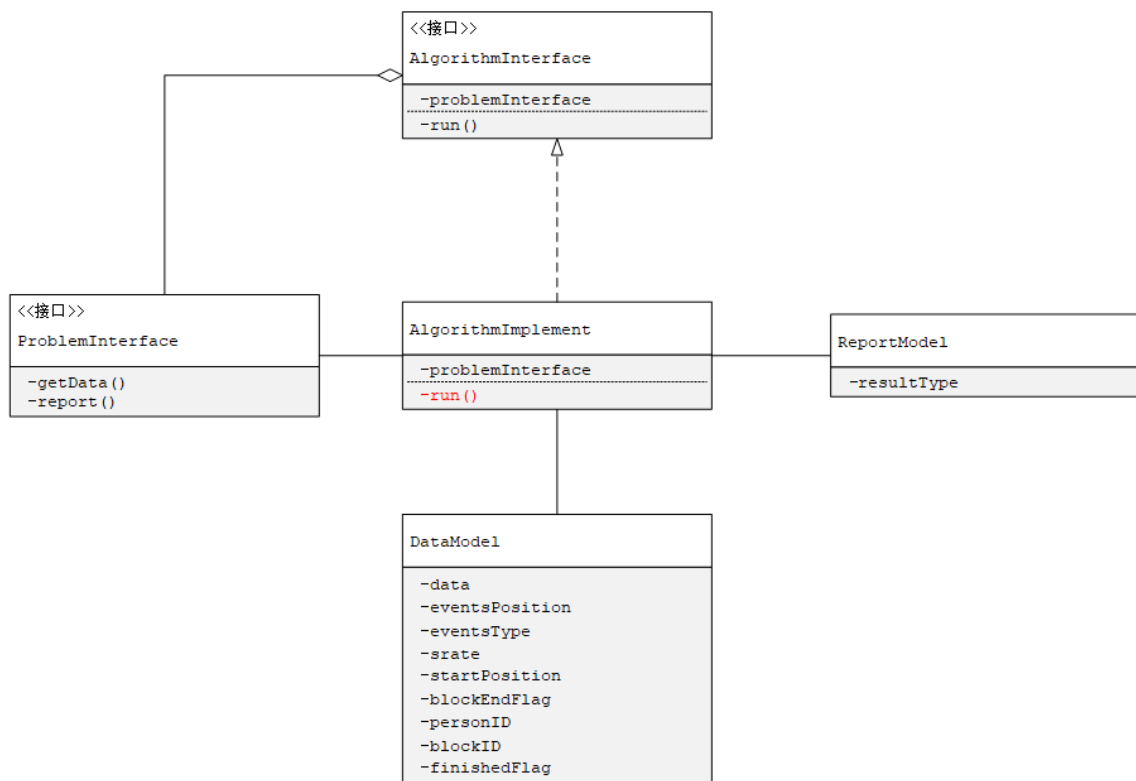
如果您对我们的竞赛支持有任何疑问，请发邮件至：
BCIRobotContest@163.com

7. 程序接口

7.1 参赛者用例



7.2 系统结构



7.3 接口函数

(1) ProblemInterface

比赛题目接口。包括数据获取方法及结果反馈方法。算法执行过程中，可以调用该接口获取数据，并通过结果反馈方法报告识别结果。比赛系统根据数据获取方法的被调用次数，及结果反馈的正确性进行综合评分。

1) `function` dataModel = getData(obj)

输入参数：无

输出参数：DataModel 类型对象

实现功能：获取下一分段实验数据。

2) `function` report(obj,reportModel)

输入参数：ReportModel 类型对象

输出参数：无

实现功能：反馈识别结果。

(2) AlgorithmInterface

参赛算法接口。参赛者需要实现该接口，并将程序运行过程填入 run() 函数中。在算法执行过程中，通过 ProblemInterface 接口 getData() 方法获取 DataModel 类型数据，并通过 report() 方法返回 ReportModel 类型结果。当通过 getData() 获取的 DataModel 数据中 finishedFlag 为 true 时，意味着数据处理完毕，该函数需要自行退出运行。

1) `function` run(obj)

输入参数：无

输出参数：无

实现功能：算法分析过程。

7.4 数据模型

(1) DataModel 参赛者数据模型

- 1) data:double 类型矩阵，分段数据，例如包含有 9 导 EEG 数据在 250Hz 采样率下，以 40ms 对数据

分段，则单次获取的 data 为 9*10 个点。

- 2) eventsPosition: double 类型数组，当前分段数据中事件位置。该数值为相对于该 block 数据起始时刻的索引位置
- 3) eventsType: double 类型标量，当前分段数据中事件类型
- 4) srates: double 类型标量，数据采样率
- 5) startPosition: double 类型标量，当前分段数据起始时刻相对于该 block 数据起始时刻的索引位置
- 6) blockEndFlag: boolean 类型标量，当前 block 数据是否结束标志
- 7) personID: double 类型标量，当前数据来源受试者序号(有训练预留，在无训练条件下该值始终不变。)
- 8) blockID: double 类型标量，当前数据来源 block 序号(有训练预留，在无训练条件下该值始终不变。)
- 9) finishedFlag: boolean 类型标量，测试结束标志。

(2) ReportModel 结果反馈模型

resultType: double 类型标量，参赛者返回的识别结果编号

8. 开发建议

8.1 本地调试建议流程

- 1) 下载比赛网站上公开的本地调试框架以及公开调试数据。
- 2) 将公开调试数据依照原有文件结构放置于\TestData 目录下。标准结构为\TestData\受试者\数据内容。

- 3) 编写算法代码，实现 `AlgorithmInterface` 接口，并以 `run()` 函数作为入口函数。将所有参赛代码放置于 `\AlgorithmImplement` 目录下。
- 4) 在算法实现过程中，根据参赛手册要求，通过 `problemInterface.getData()` 获取数据，通过 `problemInterface.report()` 报告结果。
- 5) 当检测到 `finishedFlag` 后结束 `run()` 函数运行。
- 6) 根据本地框架的报错信息进行调试。并将调试通过的代码 (`\AlgorithmImplement` 目录下所有内容) 打包上传到线上比赛系统中。
- 7) 上传的代码包括 `AlgorithmInterface` 的实现类，及其他与算法相关的辅助类。

8.2 开发示例

参见示例代码。