

脑机图灵测试赛题说明

一、 题目介绍

图灵测试是伟大的计算机开拓者阿兰·图灵于 1950 年提出的一个关于判断机器是否能够思考的著名试验，测试某机器是否能表现出与人等价或无法区分的智能。人工智能技术急速发展，脑机接口技术与人工智能技术的融合也日益加深。很多脑机接口系统都使用了 AI 辅助决策，让脑机接口与 AI 共同执行任务，以提升工作效率。

但是，这也可能带来另一个问题，那就是，我们可能无法直接推断出一个指令究竟是来源于人类还是机器。例如，当全身无法活动的失能人群使用脑机接口打字时，外界可能无法判断输出的字符究竟是来自于使用者的真实意志，还是来自于 AI 系统的优化与扩展。那么，究竟有没有办法通过脑电信号来区分究竟是人在使用脑机接口系统发送指令，还是 AI 系统自动控制呢？这就提出本次比赛提出的脑机图灵测试问题。

二、 测试场景设计

以仿真环境下的足球机器人 2 VS 2 对抗为脑机图灵测试场景。如图 1 所示，场上双方各两台机器人，回合制射门比赛，一方进攻，一方防守。

每台机器人有左右两个运动自由度，可执行操作包括左移、右移、传球和射门。对于 1 号机器人和 2 号机器人，系统随机选择其中一台由人通过 BCI 参与控制，另一台自主运动。3 号机器人和 4 号机器人完全自主运动。

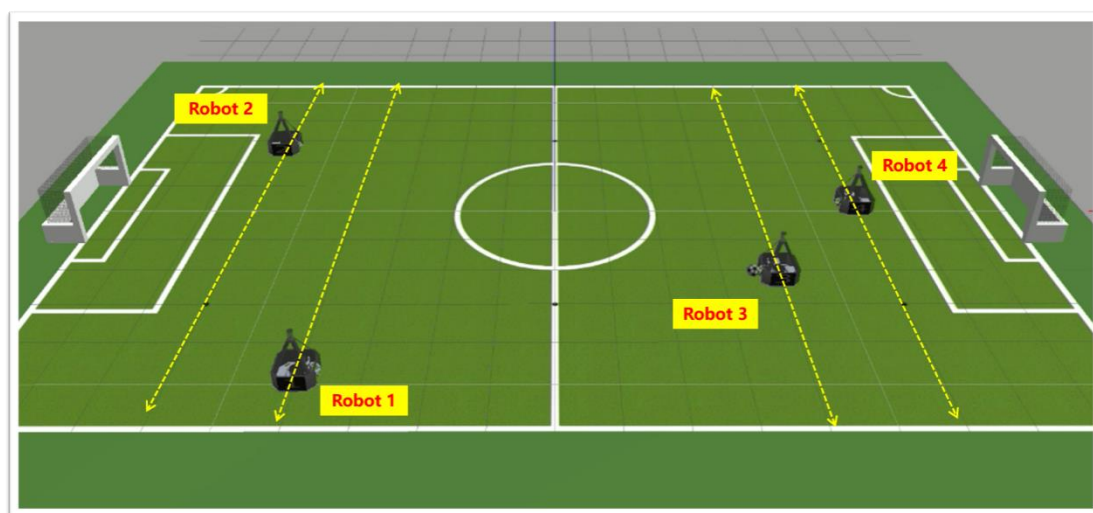


图 1 脑机图灵测试场景

三、 被试者参与方式

1. **MI-BCI:** 通过左/右手运动想象控制机器人运动方向，想象左手运动控制被控机器人左移，想象右手运动控制被控机器人右移。

2. **SSVEP-BCI**: 4 个选项，包括左移、右移、传球和射门，刺激频率分别为 15.4 Hz、15.5 Hz、15.6 Hz、15.7 Hz，初始相位均为 0，刺激亮度通过正弦调制实现（灰度范围：[-0.618,0.618]，1 对应 RGB (255, 255, 255)，-1 对应 RGB (0, 0, 0)）。

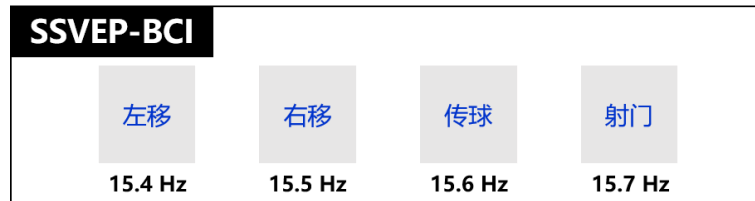


图 2 SSVEP-BCI 范式设计示意图

3. **Idle**: 被试者处于空闲状态，不参与机器人控制。

图 3 所示为脑机图灵测试 BCI 界面。

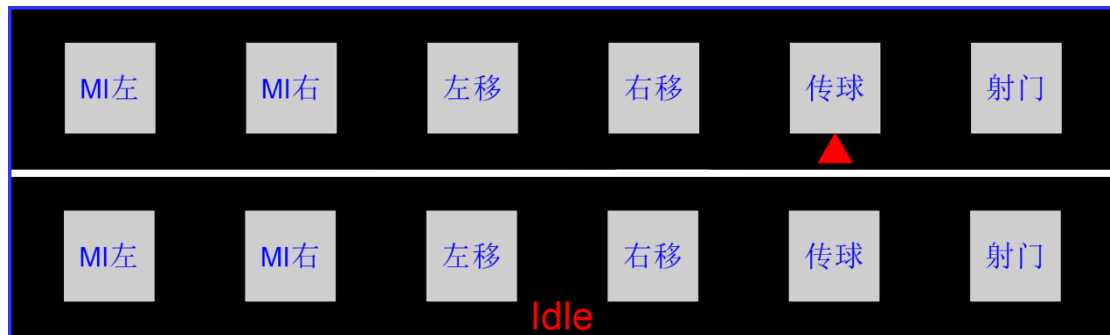


图 3a 任务提示界面

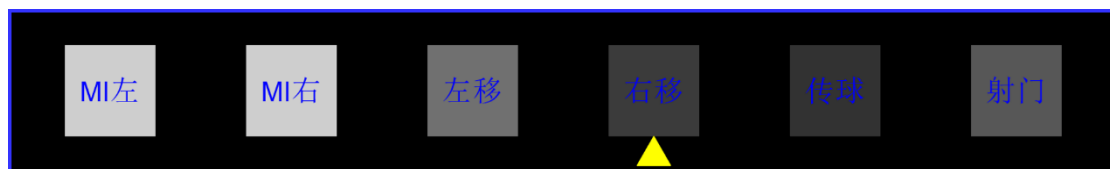


图 3b SSVEP 刺激闪烁阶段界面

图 3 脑机图灵测试 BCI 界面

四、 任务流程

每名被试需要完成 6 个 blocks，每个 block 由 3-5 个 rounds 组成，每个 round 包含 4-6 个 trials，实验流程如图 4 所示。

两个 block 之间，被试自主决定休息时长，block 内按流程连续执行。

一个 round 开始时，场上机器人位置随机初始化。

一个 trial 为一次完整的机器人行为控制，包括任务提示阶段(3 秒)，被试根据提示执行 BCI 任务阶段（5 秒），场上机器人根据指令动作阶段（5 秒）。

任务提示阶段：系统根据场上局势规划 1、2 号机器人动作并随机分配某一台机器人由被试通过 BCI 参与控制。根据机器人的行为生成相应的 BCI 任务，

提示被试执行。一种机器人行为可能对应的 BCI 任务见表 2。期间场上机器人保持静止；

被试执行 BCI 任务阶段：被试根据提示执行相应 BCI 任务，任务为 MI 则被试执行相应的左/右手运动想象，SSVEP 视觉刺激不闪烁；任务为 SSVEP 则 SSVEP 视觉刺激开始闪烁；任务为 Idle 则被试保持空闲状态。期间场上机器人保持静止；

机器人动作阶段：被试停止执行 BCI 任务，场上机器人按照规划执行动作。

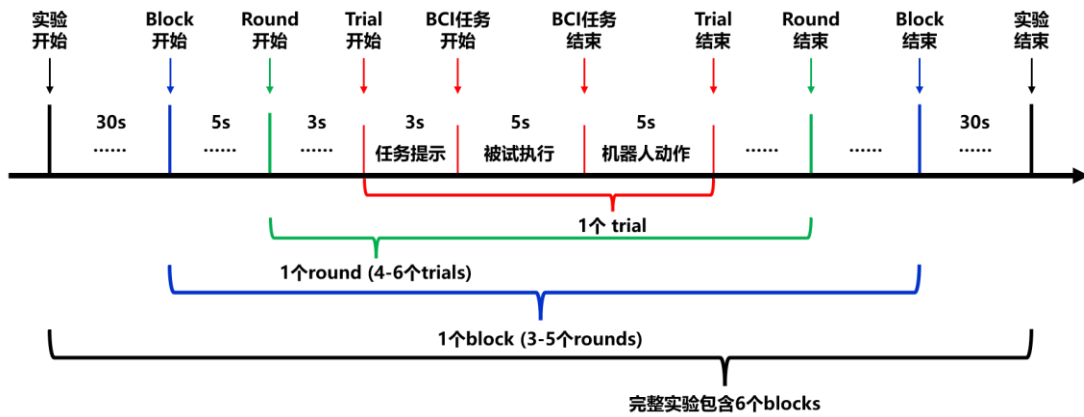


图 4 实验流程

五、 参赛队任务

赛题提供每名被试的 EEG 数据和相应 trigger，trigger 中包含每个 trial 中 1 号和 2 号机器人的动作标签。参赛队需要判断在**每个 trial 中**，被试执行了什么 BCI 任务，并报告对应的数值（数值 1-7，对应的 BCI 任务详见表 3）。

举例说明：

一个 trial 中，1 号机器人动作为左移，2 号机器人动作为射门。由表 2，被试执行的 BCI 任务可能有 MI-Left、SSVEP-Left、SSVEP-Shoot、Idle，参赛者应当报告该 4 类任务中的一种。

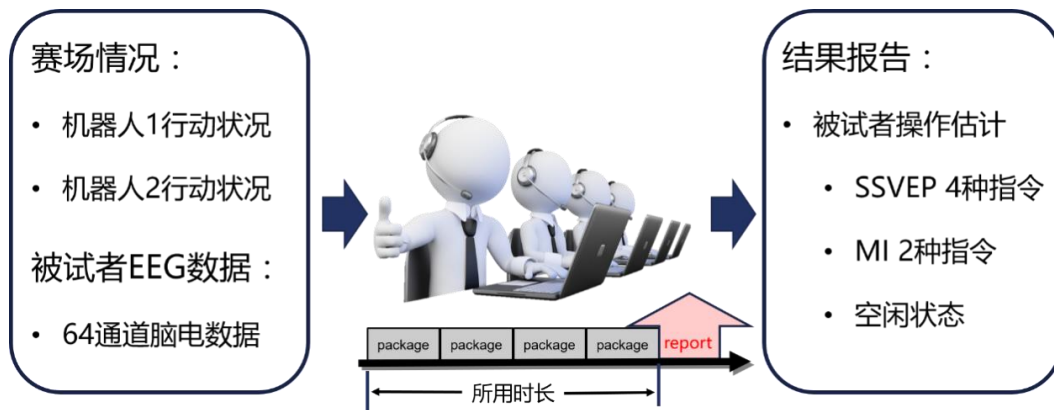


图 5 参赛队任务示意图

六、 Trigger 定义

Trigger 是一个单字节无符号整数 0-255 (uint8)。

Trigger 包含实验开始 250, 实验结束 251, block 开始 242, block 结束 243, trial 开始 240, trial 结束 241, (round 不标注 trigger), 详见表 1, 以及 BCI 任务与机器人状态信息, 详见表 2、表 3。其余为系统预留。

表 1 实验流程 trigger 定义

Trigger	定义	Trigger 二进制
250	实验开始	11111010
251	实验结束	11111011
242	Block 开始	11110010
243	Block 结束	11110011
240	Trial 开始	11110000
241	Trial 结束	11110001

当 trigger < 128 时, 即最高位为 0 时 (约定最高位为 bit7, 最低位为 bit0), bit6-bit3 标记机器人行为, bit2-bit0 标记被试执行的 BCI 任务, 数据发布给参赛队伍时, bit2-bit0 置为 0。

表 2 机器人行为 trigger 定义

Trigger bit6-bit3 (10 进制表示)	Trigger 二进制	机器人 1 行为	机器人 2 行为	被试可能执行的任务
1	00001###	左移	左移	Left-MI, Left-SSVEP, Idle
2	00010###	左移	右移	Left-MI, Right-MI, Left-SSVEP, Right-SSVEP
3	00011###	右移	左移	Left-MI, Right-MI, Left-SSVEP, Right-SSVEP
4	00100###	右移	右移	Right-MI, Right-SSVEP, Idle
5	00101###	传/接球	传/接球	Pass-SSVEP, Idle
6	00110###	射门	左移	Shoot-SSVEP, Left-SSVEP, Left-MI, Idle

7	00111###	射门	右移	Shoot-SSVEP, Right-SSVEP, Right-MI, Idle
8	01000###	左移	射门	Shoot-SSVEP, Left-SSVEP, Left-MI, Idle
9	01001###	右移	射门	Shoot-SSVEP, Right-SSVEP, Right-MI, Idle

表 3 被试 BCI 任务 Trigger 定义

Trigger bit2-bit0 (10 进制表示)	Trigger 二进制	被试的 BCI 任务
1	0####001	Left-MI
2	0####010	Right-MI
3	0####011	Left-SSVEP
4	0####100	Right-SSVEP
5	0####101	Pass-SSVEP
6	0####110	Shoot-SSVEP
7	0####111	Idle

七、 数据发布

实验数据使用博睿康 64 通道脑电采集设备采集,第 65 导联为 trigger 信息,原始采样率为 1000Hz,数据集中降采样到 250Hz,未做其他滤波处理。

表 4 导联序号-导联名称

导联序号	1	2	3	4	5	6	7	8
导联名称	Fpz	Fp1	Fp2	AF3	AF4	AF7	AF8	FZ
导联序号	9	10	11	12	13	14	15	16
导联名称	F1	F2	F3	F4	F5	F6	F7	F8
导联序号	17	18	19	20	21	22	23	24
导联名称	FCz	FC1	FC2	FC3	FC4	FC5	FC6	FT7
导联序号	25	26	27	28	29	30	31	32
导联名称	FT8	Cz	C1	C2	C3	C4	C5	C6
导联序号	33	34	35	36	37	38	39	40
导联名称	T7	T8	CP1	CP2	CP3	CP4	CP5	CP6
导联序号	41	42	43	44	45	46	47	48

导联名称	TP7	TP8	Pz	P3	P4	P5	P6	P7
导联序号	49	50	51	52	53	54	55	56
导联名称	P8	POz	PO3	PO4	PO5	PO6	PO7	PO8
导联序号	57	58	59	60	61	62	63	64
导联名称	Oz	O1	O2	ECG	HEOR	HEOL	VEOU	VEOL

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 40ms 的实验 EEG 数据(最后一个数据包长度可能小于 40ms)，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。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 数据的发送。而当所有实验数据发送完毕后，程序终止标记 finish_flag 将被置为 True。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。

八、 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。比赛系统通过记录算法报告识别结果的时间点来计算算法使用的数据长度。

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需小于 5 秒。从试次的 BCI 任务 trigger 信号起 (trigger < 240)，参赛算法最多采集 5 秒的 EEG 数据 (不含 5 秒)，就必须立刻给出反馈结果，否则视为误判。

3. 算法终止约束：

当接收到数据包中 end_flag =True 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。

九、 赛题框架

1. 参赛者用例

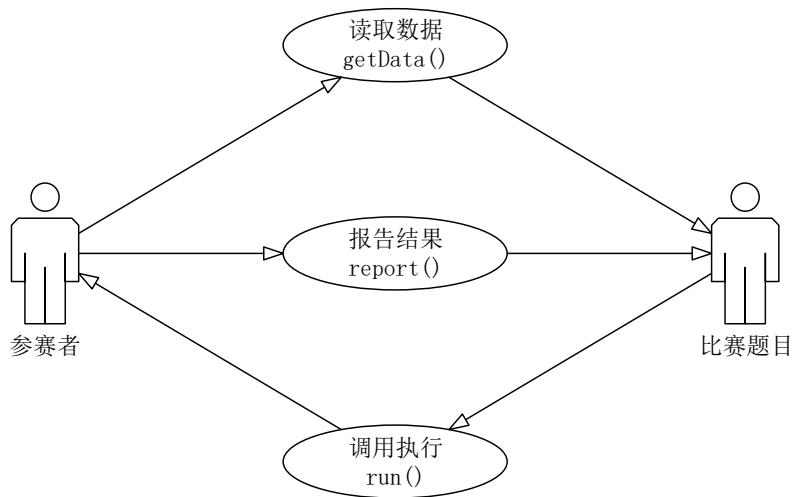


图 6 参赛者用例

2. 系统主体框架

系统主体框架如图 7 所示。

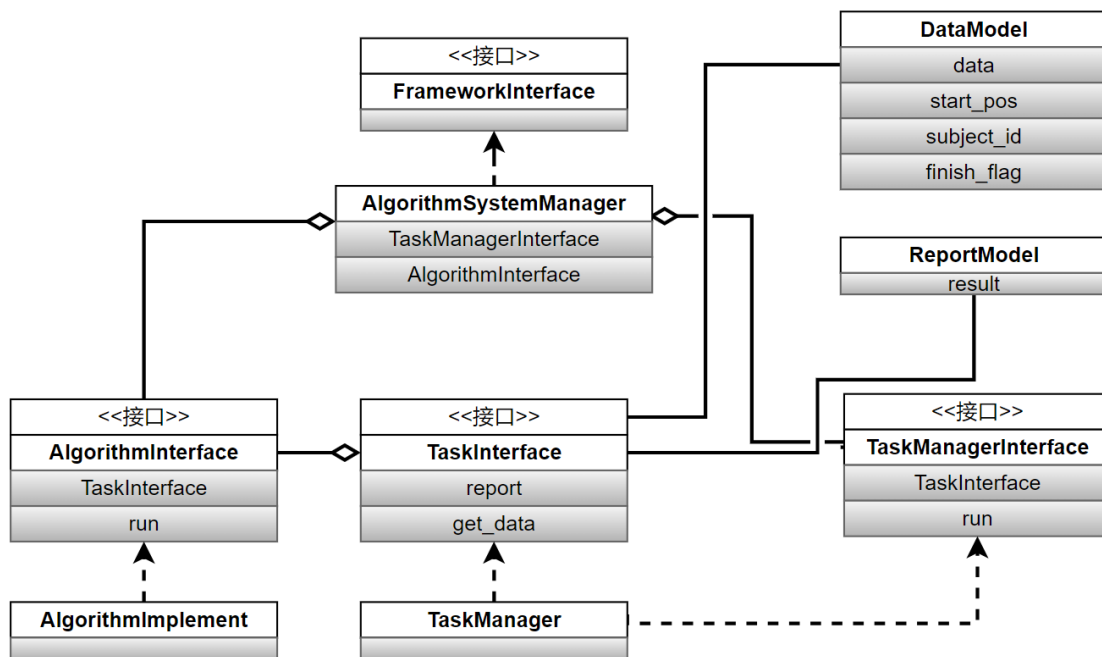


图 7 系统主体框架

(1) FrameworkInterface 框架接口

该接口主要负责赛题程序与外部执行系统的连接。该框架接口的实现类为 `AlgorithmSystemManager`，实现接口定义的所有函数。

(2) TaskInterface 任务接口

该接口是面向参赛者的赛题接口，主要负责题目与参赛算法之间的数据传递及结果报告。参赛者可以通过该接口获取比赛数据，并通过该接口报告识别结果。比赛题目需要根据参赛算法获取比赛数据的次数，以及报告结果的正确率综合给

出比赛评分。

(3) TaskManagerInterface 赛题接口

该接口主要负责实现赛题的数据填充、获取得分、清除数据以及清除报告结果，框架通过该接口实现对赛题的调用。

(4) AlgorithmInterface 算法接口

通过该接口比赛题目可以对参赛算法进行验证计算。**参赛者需要实现该接口。**在执行过程中，参赛算法需要通过 `TaskInterface` 接口获取数据，并且通过该接口报告结果。同时，参赛者需要控制算法的计算复杂度，否则当运行时间超过预定长度时，系统将自动终止该计算进程，所获成绩无效。

3. 数据模型

(1) DataModel 参赛者数据模型

- 1) `data`: float 类型矩阵，分段数据。例如包含有 64 导 EEG 数据+1 导 trigger 信号，在 250Hz 采样率下，以 40ms 对数据分段，则单次获取的 `data` 为 65*10 个点。
- 2) `start_pos`: int 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。
- 3) `subject_id`: int 类型标量，当前数据来源受试者序号。
- 4) `finish_flag`: bool 类型标量，测试结束标志。当参赛算法通过 `data_model.finish_flag` 获取数据包中该字段为 True 时，需要自行退出程序运行。

4. 参赛者相关接口函数

(1) TaskInterface

该接口由出题方负责实现，包括数据获取方法及结果反馈方法。在算法运行前，该接口的实现类会被注入参赛算法实现类中。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。出题方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) `def get_data(self):`
输入参数：无
输出参数：DataModel
实现功能：获取下一分段实验数据。
- 2) `def report(self, report_model):`
输入参数：report_model
输出参数：无
实现功能：反馈识别结果。

(2) AlgorithmInterface

参赛者需要将程序运行过程填入 run 函数中。在算法执行过程中，通过 TaskInterface 接口 get_data 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 get_data 获取的 DataModel 数据中 finish_flag 为 true 时，意味着数据处理完毕，该函数需要自行退出运行。

1) def run(self):

输入参数：无

输出参数：无

实现功能：算法分析过程。

5. 提交格式

本赛题程序使用 python 语言编写，需提交基于 python 3.8 版本的扩展名为.pyc 的文件。

6. 提交样例

参考配套代码。

参赛者可通过修改 Algorithm 文件夹中的代码完成算法，为了避免未知错误，请勿在主目录内添加文件夹。完成后重新打包程序（包含 AlgorithmImplement 文件夹和 config.toml）--> 分组 --> 具体分组 --> 计算单元 --> 定义计算单元 --> 上传程序包 --> 提交到比赛 --> 选择比赛 --> 部署 --> 完成比赛。

部署完成后在赛题的排行榜中查看比赛成绩；

需要注意的是，为防止参赛者修改代码框架作弊，保护评分程序会完全覆盖参赛者的代码（除了 AlgorithmImplement 目录和 config.toml）在提交到比赛 --> 部署时，启动的实际为评分程序 + 参赛者的 AlgorithmImplement 目录，其余运行配套代码均为服务器内置程序（包括 main.py 等文件，服务器内置评分程序与范例中程序框架基本相同，但包含评分功能和读取服务器比赛数据功能）。

7. 评分方式

本系统以正确结果数量与所用数据段长度之比作为评分标准，具体地：

- 1) 对于一个 block，依次检测其中的 trial，当正确报告的 trial 数达到该 block 内 trial 总数的 80%（向上取整）时即停止检测。不符合此要求，该 block 得分为 0；
- 2) 一个 block 内实时评分：从该 block 的第一个 trial 到当前第 i 个 trial T_{bi} ，记 i 个 trials 中正确检测结果数为 N_{bi} ， i 个 trials 所用的数据段长度和为 $L_{bi} = \text{sum}(L_{bi})$ ，则当前评分 $S_{bi} = K \cdot \frac{N_{bi}}{L_{bi}}$ （ $K = 200$ 为评分系数）。
- 3) 一个 block 的最终评分为： $S_b = K \cdot \frac{N_b}{L_b}$ （ $K = 200$ ）， $N_b = \text{ceil}(0.8 \cdot N_{bt})$ ， N_{bt} 为该 block 所包含的 trial 总数， L_b 为该 block 检测停止时所用的数据段累计长度。

- 4) 单个 trial 所用数据段长度为: 从 BCI 任务 trigger 开始到反馈方法被调用时所获取的 EEG 数据长度;
- 5) 最终成绩以所有 block 的平均成绩作为评分依据。

8. 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机器人行为 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。

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。

9. 结果反馈异常处理

(1) 报告次数

在一个试次时间内, 参赛者只能也必须报告一次结果。

(2) 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 5 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。

(3) 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。

十、 训练数据

赛题提供 MI 训练数据, 训练数据的数据规范、trigger 定义与测试数据一致, 采样率为 250Hz, 每名被试 10 个 blocks。

一个 block 包括 20 个 trials, 每个 trial 由提示阶段 (3s) 和想象运动任务阶段 (5s), 因此一个 trial 中包含一个 5s left-MI / right-MI 数据样本。

两个 trial 之间间隔 5s, 被试在该间隔期间以及 3s 提示阶段均保持空闲状态, 因此参赛者可以从该 8s 空闲状态中截取数据作为 Idle 训练样本。

训练实验流程:

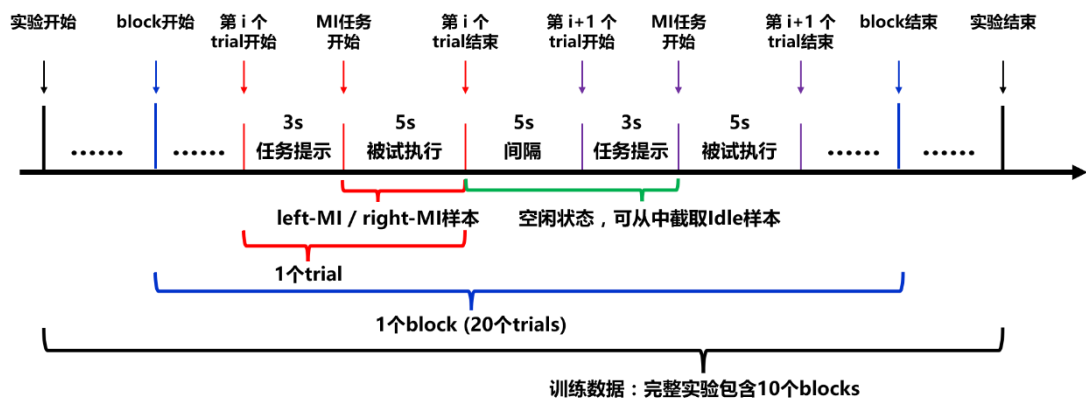


图 8 训练实验流程