EFiMAS: EVIDENCE-GUIDED TASK-CONDITIONED EEG–FNIRS FUSION

Anonymous authors
Paper under double-blind review

ABSTRACT

While EEG and fNIRS provide complementary views of neural processes, the cross-modal relationships of interest vary depending on the task and available measurement information. Existing methods typically rely on predefined computational structures, limiting their ability to adapt fusion design choices to the current task and recorded data. To address this, we investigate EEG–fNIRS fusion as a task-conditioned problem of program construction and revision. We introduce EFiMAS, an evidence-guided multi-agent system. Using dual-index Retrieval-Augmented Generation (RAG), role-specific agents in EFiMAS retrieve neuroscience evidence to propose and independently verify a modeling relationship, while utilizing method evidence to construct and dynamically revise the executable pathway based on observed outputs. Empirically, evaluations on ten public EEG–fNIRS tasks, along with comprehensive component ablations and execution trace analyses, demonstrate that EFiMAS effectively synthesizes task-specific programs and outperforms both dedicated fusion networks and foundation models.

1 INTRODUCTION

Electroencephalography (EEG) and functional near-infrared spectroscopy (fNIRS) are noninvasive techniques for studying brain activity. EEG records rapid electrical activity from neuronal populations (Buzsáki et al., 2012), whereas fNIRS reflects slower hemodynamic responses through changes in cortical oxygenated and deoxygenated hemoglobin (Scholkmann et al., 2014). By capturing complementary physiological signals, these modalities enhance the recognition of motor, cognitive, and emotional states. (Si et al., 2025; Li, 2026; Shin et al., 2018) Consequently, effective EEG–fNIRS fusion remains a central objective in neural analysis.

In EEG–fNIRS fusion, the cross-modal relationships of primary interest vary across tasks (Hillman, 2014; Tachtsidis & Scholkmann, 2016). Recognition of motor state often emphasizes spatial lateralization, linking localized EEG μ or β band activity with corresponding hemodynamic responses. Conversely, cognitive conflict recognition typically prioritizes the temporal synchronization between rapid frontocentral electrical events and subsequent hemodynamic shifts. Since a critical relationship in one task may offer minimal discriminative value in another, effective fusion must prioritize task-specific interactions. Furthermore, translating these theoretical priors into discriminative features requires empirical analysis and iterative testing. Figure 1 summarizes this design space across temporal alignment, EEG electrical activity, fNIRS responses, and spatial organization.

Existing fusion architectures target different subsets of these relational perspectives. Spatial guidance and adaptive temporal alignment methods organize spatiotemporal features (Wang et al., 2026), while attentional mechanisms map cross-modal dependencies (Li, 2026). Although these methods successfully model specific EEG–fNIRS relationships and support data-driven interactions, their core computational structures and fusion stages are fixed during architecture design. A model optimized for one cross-modal interaction may structurally limit others. When empirical analysis reveals that predefined pathways fail to capture target relationships, these fixed architectures lack flexibility. Reconfiguring representations and interaction schemes thus relies heavily on manual redesign and experimental iteration.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

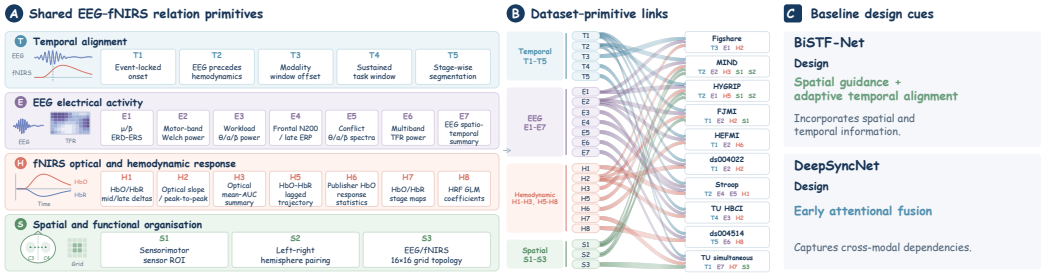

Figure 1: Shared EEG–fNIRS relation primitives motivate dataset-specific fusion, with specific relation primitives further detailed in Appendix A.3.

This highlights a core modeling requirement: given a specific task and empirical measurements, the adoption of any fusion strategy must be explicitly justified and structurally adaptable. Addressing this requires ensuring that the fusion choices themselves are intrinsically evidence-based, implementable, and verifiable. This process involves three iterative questions: **What to select?** (identifying scientifically grounded and data-supported relationships), **How to implement?** (defining exact representations and mathematical operations), and **Is it appropriate?** (evaluating execution traces to confirm predictive advantages). When implementations deviate from expectations or lack empirical support, specific operations or the overall fusion strategy must be redefined. Figure 1 maps this choice space.

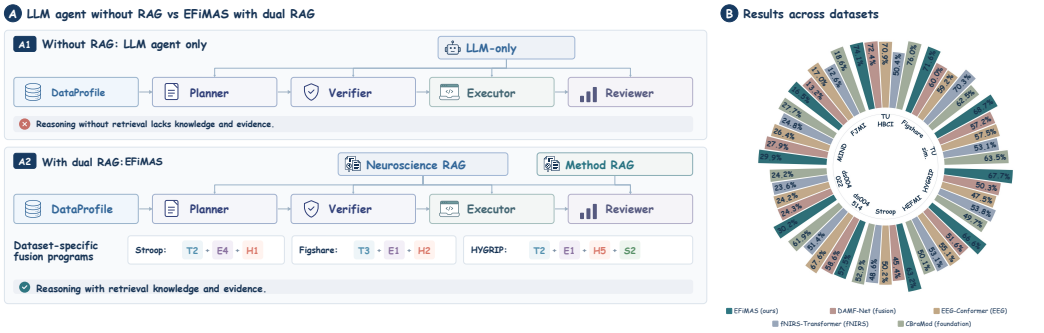

Figure 2: Dual-RAG evidence guides EFiMAS fusion programs across datasets.

To fulfill this requirement, we propose EFiMAS, an evidence-guided multi-agent system that constructs and iteratively revises task-conditioned EEG–fNIRS fusion pathways (Figure 2). To support dynamic reasoning—from assessing relationship validity to diagnosing execution constraints—EFiMAS maintains separate retrieval indexes for neuroscience literature and methodological knowledge. Using role-specific Retrieval-Augmented Generation (RAG), the system formulates targeted queries based on the current fusion scheme, verification feedback, and execution traces. Through this mechanism, external knowledge is integrated directly with empirical data to actively drive and correct fusion decisions.

Within EFiMAS, the **Planner** uses task profiles and evidence to propose a fusion strategy. The **Verifier** independently retrieves evidence to validate this proposal's scientific rationale and data applicability. The **Executor** uses methodological evidence and tools to translate the strategy into an executable pathway, applying local corrections. Finally, the **Reviewer** evaluates execution traces and validation results to accept the program, route errors back to the Executor, or instruct the Planner to reconsider foundational choices. The finalized program is then fixed for held-out evaluation.

We evaluate **EFiMAS** on ten public EEG–fNIRS tasks against dedicated fusion networks, unimodal models, and EEG foundation models. EFiMAS achieves the highest balanced accuracy on seven tasks. On identical splits, its task-specific programs outperform a universal fusion pathway on nine tasks. Ablations and execution trace analyses on three tasks document how retrieved evidence and execution feedback adjust representations, operations, and pathways. These experiments validate the predictive advantage of task-conditioned fusion and show how EFiMAS refines architectural choices using multimodal evidence.

2 PRELIMINARY

2.1 PROBLEM DEFINITION AND FUSION PROGRAM

Each EEG–fNIRS fusion task τ comprises a dataset, a prediction target, and an evaluation protocol. Let X^E and X^N denote the EEG and fNIRS inputs from the dataset associated with task τ. A fixed profiling procedure maps these data, released metadata $\mathcal{M}$, and the predefined protocol Π to a DataProfile D:

$$D = \mathsf{Profile}\big(\mathrm{X}^E, \mathrm{X}^N, \mathcal{M}, \Pi\big)\,. \tag{1}$$

The profile summarizes the task, available signals, sampling rates, channels, cross-modal timing, events, signal quality, and evaluation protocol.

Let G be the fusion program for task τ. It takes X^E and X^N as data inputs and applies the following processing steps in sequence, where each step may contain an operation sequence: first, A^E and A^N process the two modalities to produce different signal representations; next, cross-modal fusion F makes the two representations interact; finally, classifier C maps the fused representation to predictions. We write

$$G(\mathrm{X}^E, \mathrm{X}^N) = C\big(F\big(A^E(\mathrm{X}^E), A^N(\mathrm{X}^N)\big)\big)\,. \tag{2}$$

EFiMAS constructs a fusion program G for each task τ and uses it to produce predictions. We evaluate the program's accuracy under the task's predefined protocol.

2.2 RETRIEVAL-AUGMENTED GENERATION

Retrieval-augmented generation (RAG) combines evidence retrieval with language-model generation. Given a knowledge base $\mathcal{K}$ and a query $q^{(k)}$ formed from context $c^{(k)}$, it retrieves evidence

$$E^{(k)} = \mathsf{Retrieve}\Big(q^{(k)}; \mathcal{K}\Big)\,, \tag{3}$$

and generates

$$y^{(k)} = \mathsf{Generate}\Big(c^{(k)}, q^{(k)}, E^{(k)}\Big)\,. \tag{4}$$

The query may describe the task, current state, target decision, and constraints.

EFiMAS independently indexes a **method knowledge base with 983 records** and a **neuroscience knowledge base with 232 records**. A tool library $\mathcal{T}$ contains **227 executable entries**, including **28 commonly used tool combinations**. Each entry describes its function, input and output requirements, applicable conditions, and usage. For a tool query $q^{(k)}_{\mathrm{tool}}$,

$$\mathcal{T}^{(k)} = \mathsf{SearchTools}\Big(\mathcal{T}, q^{(k)}_{\mathrm{tool}}\Big)\,. \tag{5}$$

Retrieved tool descriptions guide tool selection, composition, or adjustment, extending RAG to executable resources. These resources provide scientific and computational knowledge without held-out labels, predictions, or other held-out target information. Appendices K and L describe their construction and provide examples; Appendix H.3 reports retrieval settings and sources.

2.3 MULTI-AGENT SYSTEMS

A multi-agent system (MAS) contains agents that make decisions and interact by passing intermediate results and feedback. EFiMAS uses four roles: the **Planner** proposes task-specific hypotheses and plans; the **Verifier** checks their basis and feasibility; the **Executor** converts an approved plan into a program; and the **Reviewer** examines a complete attempt and returns a decision, summary, and revision advice. Their agent set is

$$\mathcal{A} = \{\mathrm{Planner}, \mathrm{Verifier}, \mathrm{Executor}, \mathrm{Reviewer}\}\,. \tag{6}$$

At interaction round k, agent $\mathrm{Agent}_r \in \mathcal{A}$ combines context $c_r^{(k)}$ and evidence $E_r^{(k)}$ to produce a structured result in the role-specific output set $\mathcal{Z}_r$:

$$z_r^{(k)} = \mathrm{Agent}_r\Big(c_r^{(k)}, E_r^{(k)}\Big) \in \mathcal{Z}_r. \tag{7}$$

Contexts may contain task descriptions, intermediate results, observations, and feedback. Outputs are P for the **Planner**, $\widehat{P}$ or b for the **Verifier**, G for the **Executor**, and (d, s_R, b_R) for the **Reviewer**. Prompts require each agent to return structured files that can be parsed and passed between agents. Each role maintains separate prompts, queries, evidence packets, and decision states, although one model may serve all roles. The role-specific prompts and structured output contracts are provided in Appendix C.

3 METHOD

3.1 OVERVIEW

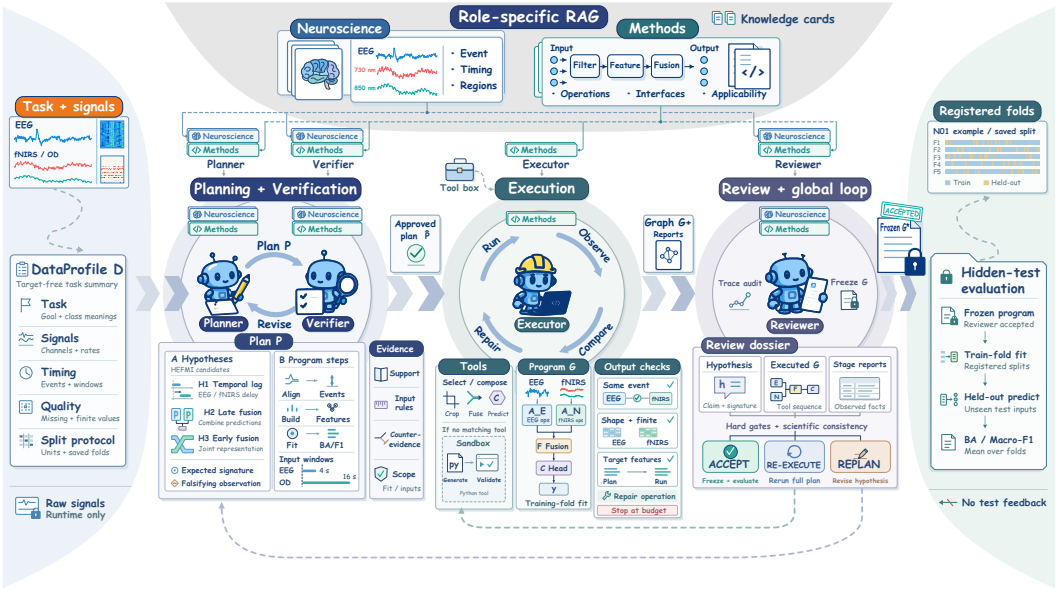

Figure 3: EFiMAS overview. Task-conditioned evidence guides planning, execution, review, and hidden-test evaluation.

As shown in Figure 3, a fixed procedure maps the released data, metadata, and protocol for task τ to the DataProfile D_τ defined in Section 2.1. Once the task is fixed, we write this profile as D below. The workflow has three steps:

(1) Fusion planning and verification with diverse evidence. The **Planner** uses D and retrieved evidence to propose a fusion hypothesis and plan P. The **Verifier** independently retrieves evidence and inspects tool descriptions to assess the hypothesis, data suitability, and feasibility of the planned operations, returning feedback or an approved plan $\widehat{P}$ with operations and expected outputs.

(2) Hypothesis-associated execution loop. The **Executor** uses method evidence and tool search to turn $\widehat{P}$ into G, checking outputs and revising local operations before passing the trajectory and validation results to the **Reviewer**.

(3) Independent review and decision-making. The **Reviewer** independently retrieves evidence to assess the attempt, accepts G, or returns feedback for re-execution or replanning. The retry budget is finite; if no attempt is accepted within the budget, the task is declared unsuccessful and no held-out result is produced. An accepted program is frozen as $G^\star$ and evaluated once on the held-out set.

The workflow above can be represented as follows:

$$\tau \xrightarrow{\text{fixed procedure}} D_\tau \xrightarrow[\text{RAG}_{\text{planner}}]{\text{Planner}} P \xrightarrow[\text{RAG}_{\text{verifier}}]{\text{Verifier}} \widehat{P} \xrightarrow[\text{RAG}_{\text{executor}}]{\text{Executor}} G \xrightarrow[\text{RAG}_{\text{reviewer}}]{\text{Reviewer}} G^\star \xrightarrow{\text{if accepted: hidden test}} \text{BA}_{\text{hidden}}. \quad (8)$$

3.2 Fusion Planning and Verification with Diverse Evidence

Intuition. To select a relationship and fusion strategy suitable for each task, the **Planner** reasons over task information and supporting evidence to propose a suitable fusion relationship and corresponding plan, while the **Verifier** examines its basis and feasibility.

Fusion Planning and Verification. The **Planner** receives D and combines retrieved evidence with controlled data inspection when needed to propose a fusion hypothesis and plan. Let $E_P^{(k)}$ and $S^{(k)}$ denote its evidence and inspection result at planning round k. The initial plan is

$$P^{(1)} = \text{Planner}\Big(D, E_P^{(1)}, S^{(1)}\Big). \tag{9}$$

The **Verifier** independently retrieves both evidence types to check the scientific basis of the hypothesis and whether the available data support the proposed analysis. It consults tool library $\mathcal{T}$ to assess whether the planned operations can be implemented, then returns an approved plan or feedback:

$$\text{Verifier}\Big(P^{(k)}, E_V^{(k)}, \mathcal{T}\Big) \longrightarrow \widehat{P} \quad \text{or} \quad b^{(k)}. \tag{10}$$

The **Verifier** returns $\widehat{P}$ when the hypothesis is supported, the data satisfy the conditions, and the operations are implementable; otherwise, $b^{(k)}$ identifies the required revision. The **Planner** combines feedback $b^{(k)}$, the previous reasoning record $R^{(k)}$, the existing plan, and updated retrieval and inspection results to produce the next plan:

$$P^{(k+1)} = \text{Planner}\Big(D, E_P^{(k+1)}, S^{(k+1)}, b^{(k)}, R^{(k)}, P^{(k)}\Big). \tag{11}$$

This loop refines the fusion hypothesis and planned operations. The approved plan is

$$\widehat{P} = \Big(\widehat{H}, \widehat{P}^{A^E}, \widehat{P}^{A^N}, \widehat{P}^{F}, \widehat{P}^{C}\Big), \tag{12}$$

where $\widehat{H}$ is the approved hypothesis, and the four components specify operations and expected outputs for EEG, fNIRS, fusion, and prediction. They guide construction of A^E, A^N, F, and C in G.

Diverse Evidence Support. The profile D is passed to the **Planner** as text, statistical summaries, and representative metadata. If the information is insufficient, the **Planner** can request fixed tools to inspect specific data properties under controlled access and return $S^{(k)}$. Raw signal sequences do not enter its context. Appendix H.2 gives the complete fields and a TU simultaneous example.

The **Planner** forms queries from D and, in later rounds, the previous plan and feedback. Each query states the issue to resolve, retrieval perspective, and relevant fusion stage. For query $q_P^{(k)}$ and indexes $\mathcal{K}_N, \mathcal{K}_M$, the evidence packet is

$$E_P^{(k)} = \Big(\mathsf{Retrieve}\Big(q_P^{(k)}; \mathcal{K}_N\Big), \mathsf{Retrieve}\Big(q_P^{(k)}; \mathcal{K}_M\Big)\Big). \tag{13}$$

The **Verifier** constructs an independent query $q_V^{(k)}$ from the hypothesis, its proposed mechanism, expected observations, falsification criteria, and planned operations:

$$E_V^{(k)} = \Big(\mathsf{Retrieve}\Big(q_V^{(k)}; \mathcal{K}_N\Big), \mathsf{Retrieve}\Big(q_V^{(k)}; \mathcal{K}_M\Big)\Big). \tag{14}$$

The **Verifier** also queries $\mathcal{T}$ to inspect tool functions, expected effects, applicable conditions, and input–output formats. It can inspect tool descriptions but cannot request tool execution. Missing capabilities and incompatible interfaces are reported to the **Planner**.

Note. The two indexes are queried separately and retain source labels. Independent retrieval budgets keep the two evidence types balanced and prevent record counts from making one dominate.

3.3 HYPOTHESIS-ASSOCIATED EXECUTION LOOP

Intuition. To implement the fusion plan and capture the selected fusion relation, the execution stage progressively constructs G through a local loop, allowing the **Executor** to revise it based on observed and expected results. At each step, method evidence and candidate tools are retrieved separately to provide operation-specific execution support.

Evidence-Guided Tool Selection and Program Construction. The **Executor** implements $\widehat{P}$ as fusion program G by connecting tools into a tool graph, with an ordered tool sequence for each program component. Method queries combine the hypothesis, mechanism, stage objective, and expected output; tool queries specify the required capability, available input, and target output. For operation i,

$$E_M^{(i)} = \mathsf{Retrieve}\Big(q_{\text{exec}}^{(i)}; \mathcal{K}_M\Big), \qquad \mathcal{T}^{(i)} = \mathsf{SearchTools}\Big(\mathcal{T}, q_{\text{tool}}^{(i)}\Big). \tag{15}$$

The **Executor** selects a tool or tool sequence T_i for the operation and connects its inputs to preceding outputs. If no suitable tool is available, it adjusts an existing tool or generates and validates a custom Python tool in a restricted sandbox (Appendix H.6). All tools restrict fitting and training-label access to the corresponding training fold.

Expected-Output Checks and Local Loop. Let x_i be the input, O_i the observed output, and $\widehat{O}_i$ the expected output. The **Executor** checks whether O_i satisfies the planned requirements, such as tensor shape, target features, or cross-modal alignment:

$$O_i = \mathsf{Execute}(T_i, x_i), \qquad d_i = \text{Executor}\Big(\widehat{P}, \widehat{O}_i, O_i\Big) \in \{\text{continue}, \text{revise}\}. \tag{16}$$

The **Executor** repeats the current operation until its output meets the expected requirements or the budget is exhausted. It can revise parameters or code, or select another tool. Outputs that meet the requirements feed subsequent operations, advancing construction of G. After execution, it passes G, stage reports, and research-validation results to the **Reviewer**.

3.4 INDEPENDENT REVIEW AND DECISION-MAKING

Intuition. After each operation produces the expected output, the system must determine whether the full trajectory forms a consistent, correct, and effective fusion chain from planning to execution. The **Reviewer** combines independent retrieval evidence with the plan, execution, and results to produce a decision and revision advice.

Independent Retrieval and Review Rules. The **Reviewer** forms an independent query from the fusion hypothesis, its mechanism, expected outputs, and stage checks. It adds terms related to scientific evidence and limitations to retrieve literature that helps explain the observed results. Let $[a]$ denote attempt a, distinct from planning round (k); its evidence packet is

$$E_R^{[a]} = \Big(\mathsf{Retrieve}\Big(q_R^{[a]}; \mathcal{K}_N\Big), \mathsf{Retrieve}\Big(q_R^{[a]}; \mathcal{K}_M\Big)\Big). \tag{17}$$

Following its prompt rules (Appendix C) and retrieved evidence, the **Reviewer** checks **hypothesis–evidence alignment**: whether evidence supports the hypothesis; **plan–execution consistency**: whether the tool path and outputs match the plan; and **conclusion–result consistency**: whether results support the conclusions. It then considers whether the retrieved evidence suggests improvements to G.

Note. The acceptance contract and chance-relative significance rule are specified in Appendix C; the Reviewer audit and its agreement with human experts are reported in Appendix B.3.1.

Review Decisions. Let $\mathcal{O}^{[a]}$ contain the stage reports and research-validation results. The **Reviewer** returns decision $d^{[a]}$, summary $s_R^{[a]}$, and advice $b_R^{[a]}$:

$$\Big(d^{[a]}, s_R^{[a]}, b_R^{[a]}\Big) = \text{Reviewer}\Big(\widehat{P}^{[a]}, G^{[a]}, \mathcal{O}^{[a]}, E_R^{[a]}\Big). \tag{18}$$

The decisions are **accept**, **re-execute**, and **replan**. The summary records the plan, differences between observed and expected outputs, and unresolved issues; the advice identifies hypotheses or operations to change. If the hypothesis or strategy needs revision, the **Planner** combines the previous plan with the **Reviewer**'s summary of the plan and results, revision advice, and updated evidence and inspection results:

$$P^{[a+1]} = \text{Planner}\left(D, E_P^{[a+1]}, S^{[a+1]}, P^{[a]}, s_R^{[a]}, b_R^{[a]}\right). \tag{19}$$

The updated plan passes through the verification loop in Section 3.2. If the plan remains applicable and only its implementation needs correction, the **Reviewer** retains $\widehat{P}$ and requests re-execution:

$$G^{[a+1]} = \text{Executor}\left(\widehat{P}^{[a]}, s_R^{[a]}, b_R^{[a]}\right). \tag{20}$$

Validation BA is used to compare candidate programs across attempts; final acceptance is determined by the **Reviewer**. An accepted program is frozen as $G^\star$ and evaluated once on the held-out set. If the retry budget is exhausted without acceptance, the task is declared unsuccessful and no held-out evaluation is performed. Held-out results never enter the feedback loops. Algorithm 1 in Appendix H.1 summarizes the workflow.

4 Experiments

4.1 Experimental Setup

Datasets. We evaluate ten released EEG–fNIRS tasks: ds004022 (Lee et al.), MIND (Feng et al., 2026b), Stroop (Chen et al., 2023), ds004514 (Rybár et al., 2025), TU HBCI (Technical University of Berlin, 2017), Figshare (Buccino, 2015), TU simultaneous (Shin et al., 2018), HEFMI (Shi et al., 2025), HYGRIP (Ortega et al., 2020), and FJMI (Yi et al., 2025). The ten tasks use dataset-specific cross-subject, within-subject, subject–session-grouped, and release-defined protocols. All methods within a dataset share its registered folds, and main-table results use seeds 0, 1, and 2. Specific fold assignments, preprocessing, and held-out evaluation units are provided in Appendix A.

Baselines. We compare EFiMAS with fusion baselines DAMF-Net (Feng et al., 2026a), DeepSyncNet (Li, 2026), and BiSTF-Net (Wang et al., 2026); unimodal EEG/fNIRS decoders EEGConformer (Song et al., 2023), BIOT (Yang et al., 2023), fNIRS-Transformer (Wang et al., 2022b), fNIRSNet (Wang et al., 2023), and BenchNIRS CNN (Benerradi et al., 2023); and EEG foundation models LaBraM (Jiang et al., 2024), CBraMod (Wang et al., 2025), LUNA (D"oner et al., 2025), CodeBrain (Ma et al., 2026), and STEEGFormer (Yang et al., 2026). Baseline descriptions are provided in Appendix A.4.

4.2 Implementation Details

Model settings. Each foundation-model baseline is evaluated in up to three settings: *Train* learns all weights from scratch, *Pretrained FT* updates a released checkpoint, and *Frozen probe* fixes the backbone and trains a linear head. We denote these settings by *tr*, *ft*, and *fr*. Additional baseline and preprocessing details are provided in Appendix A.4 and Appendix A.

EFiMAS protocol. As defined in Section 3, EFiMAS turns the DataProfile D_τ into an initial fusion plan, independently verifies that plan with evidence, executes it as a program, and freezes it only after Reviewer acceptance. We use DeepSeek V4 Pro for Planning, Verification, Execution, and Review. The research budget allows up to ten complete planning-to-review attempts. Appendix B.3.2, Figure 16, reports post-hoc scores for all ten attempts; these diagnostic scores are not exposed to the agents or used to select the frozen program.

Evaluation. We evaluate each selected model or program once on the held-out set of every registered fold. Balanced accuracy (BA) is the primary metric for imbalanced tasks, and macro-F1 is also reported; definitions are given in Appendix A.5. Test labels remain hidden from Planning, Verification, Execution, and Review, while runtime tools may use training labels for fitting. BA (hidden BA) and macro-F1 are returned only after Reviewer acceptance freezes the program.

4.3 Main Results

Table 1 reports the complete ten-task comparison; ranking colors, foundation-model settings, and unavailable entries are defined in the caption.

Method	ds004022		MIND		Stroop		ds004514		TU HBCI	
	BA	Macro-F1	BA	Macro-F1	BA	Macro-F1	BA	Macro-F1	BA	Macro-F1
End-to-end models										
DAMF-Net (2026)	0.2433	0.1999	0.2786	0.1882	0.4537	0.4329	0.5862	0.5862	0.7235	0.7203
DeepSyncNet (2026)	0.2524	0.1478	0.2628	0.2326	0.5263	**0.5180**	0.4767	0.4359	0.6818	0.6791
BiSTF-Net (2026)	0.2595	0.1278	0.2683	0.2267	0.5000	0.3333	0.4894	0.4195	0.6923	0.6910
EEG-Conformer (2023)	0.2425	0.1647	0.2640	0.2582	0.5025	0.4962	**0.6762**	**0.6741**	0.7093	0.7091
BIOT (2023)	0.2542	0.1296	0.2581	0.1995	0.4919	0.4454	0.6047	0.5730	0.6331	0.6287
fNIRS-Transformer (2022)	0.2358	0.1848	0.2482	0.2123	0.4857	0.4693	0.5139	0.5048	0.5039	0.4932
fNIRSNet (2023)	0.2542	0.2286	0.2482	0.2092	0.4527	0.3932	0.5137	0.4840	0.4816	0.4239
BenchNIRS CNN (2023)	0.2608	0.1648	0.2480	0.2376	0.5025	0.3411	0.5398	0.5204	0.5000	0.3373
Foundation models										
LaBraM^{tr} (2024)	0.2533	0.1132	0.2537	0.1848	0.5081	0.4900	0.5711	0.5574	0.5860	0.5826
CBraMod^{tr} (2025)	0.2508	0.1347	0.2512	0.1627	0.5296	0.5090	0.6705	0.6649	0.7048	0.7042
CBraMod^{ft}	0.2417	0.1210	0.2767	**0.2622**	0.5289	0.4852	0.6191	0.6141	**0.7605**	**0.7602**
CBraMod^{fr}	0.2500	0.1000	0.2550	0.1608	0.5079	0.4072	0.5163	0.3818	0.5096	0.4918
LUNA^{tr} (2025)	0.2500	0.1000	0.2522	0.1264	0.4885	0.4340	0.5245	0.4361	0.5378	0.5296
LUNA^{ft}	0.2492	0.1642	0.2581	0.1706	0.4938	0.3644	0.5542	0.5203	0.5815	0.5737
LUNA^{fr}	0.2367	0.1506	0.2616	0.2295	0.5206	0.4635	0.5532	0.5234	0.5084	0.4254
CodeBrain^{tr} (2026)	0.2525	0.1239	0.2454	0.1708	0.5164	0.4939	0.5714	0.5576	0.6604	0.6571
CodeBrain^{ft}	0.2500	0.1265	0.2598	0.1570	0.5281	0.4725	0.5852	0.5725	0.7582	0.7580
CodeBrain^{fr}	0.2467	0.1253	0.2485	0.2323	0.4967	0.4186	0.5160	0.4385	0.4956	0.3976
STEEGFormer^{tr} (2026)	0.2492	0.1513	0.2520	0.1113	0.4982	0.4751	0.5653	0.5416	0.6786	0.6760
STEEGFormer^{ft}	0.2400	0.1321	0.2565	0.1628	0.5199	0.4702	0.5538	0.5319	0.5803	0.5142
STEEGFormer^{fr}	0.2450	0.1339	0.2500	0.1000	0.4977	0.4200	0.4999	0.3507	0.5311	0.5053
Agentic fusion synthesis										
EFiMAS	**0.3024**	**0.2388**	**0.2994**	0.2549	**0.6316**	0.5078	0.5750	0.4998	0.7411	0.6379

Method	Figshare		TU sim.		HEFMI		HYGRIP		FJMI	
	BA	Macro-F1	BA	Macro-F1	BA	Macro-F1	BA	Macro-F1	BA	Macro-F1
End-to-end models										
DAMF-Net (2026)	0.6000	0.6000	0.5718	0.5710	0.5160	0.5149	0.5026	0.3963	0.1318	0.0685
DeepSyncNet (2026)	0.5707	0.5338	0.5442	0.4319	0.5162	0.4080	0.4857	0.3506	0.1340	0.0659
BiSTF-Net (2026)	0.6347	0.6315	0.5160	0.3802	0.5271	0.4253	0.4964	0.3314	0.1378	0.0666
EEG-Conformer (2023)	0.5920	0.5829	0.5750	0.5470	0.5510	0.5065	0.4749	0.3996	0.1697	0.1604
BIOT (2023)	0.4867	0.4049	0.5872	0.5389	0.5075	0.4369	0.5087	0.3871	0.1358	0.1124
fNIRS-Transformer (2022)	0.7027	0.7018	0.5308	0.4939	0.5314	0.4814	0.5381	0.4265	0.1257	0.1057
fNIRSNet (2023)	0.6653	0.6624	0.5000	0.3333	0.5000	0.3333	0.5328	0.3999	0.1234	0.0913
BenchNIRS CNN (2023)	0.6787	0.6747	0.5212	0.4372	0.5330	0.5065	0.6242	**0.5240**	0.1309	0.0985
Foundation models										
LaBraM^{tr} (2024)	0.4907	0.3684	0.5442	0.4693	0.5150	0.4419	–†	–†	0.1234	0.1109
CBraMod^{tr} (2025)	0.5453	0.5283	0.5872	0.5337	0.5033	0.4384	0.5000	0.3662	0.1528	0.1455
CBraMod^{ft}	0.6253	0.6144	0.6353	0.5920	0.5013	0.4130	0.4967	0.3626	**0.1858**	**0.1708**
CBraMod^{fr}	0.5187	0.4493	0.4949	0.3471	0.4984	0.3465	0.5000	0.3333	0.1273	0.0434
LUNA^{tr} (2025)	0.5027	0.3686	0.5173	0.3970	0.5007	0.3709	–†	–†	0.1266	0.0644
LUNA^{ft}	0.5013	0.3968	0.5295	0.4313	0.5141	0.3911	–†	–†	0.1363	0.0901
LUNA^{fr}	0.5053	0.4499	0.5096	0.4180	0.4853	0.3980	–†	–†	0.1256	0.0938
CodeBrain^{tr} (2026)	0.5000	0.3333	0.5231	0.4301	0.5036	0.4124	0.5000	0.3638	0.1295	0.0811
CodeBrain^{ft}	0.5300	0.3967	0.5705	0.5054	0.5239	0.4430	0.4913	0.3835	0.1396	0.1226
CodeBrain^{fr}	0.5500	0.5445	0.4968	0.3905	0.4944	0.4008	0.5172	0.4900	0.1272	0.0995
STEEGFormer^{tr} (2026)	0.5320	0.4856	0.5449	0.5094	0.5085	0.4324	–†	–†	0.1271	0.0680
STEEGFormer^{ft}	0.5067	0.4307	0.5333	0.4560	0.4964	0.4089	–†	–†	0.1426	0.1051
STEEGFormer^{fr}	0.5013	0.4138	0.5103	0.4195	0.5059	0.4454	–†	–†	0.1256	0.0746
Agentic fusion synthesis										
EFiMAS	**0.7160**	**0.7124**	**0.6873**	**0.7602**	**0.6656**	**0.6597**	**0.6768**	0.5190	0.1653	0.1257

Table 1: Quantitative comparison on ten EEG–fNIRS tasks. Results are arranged in two five-dataset panels. Each entry reports the mean hidden balanced accuracy (BA) or macro-F1 over seeds 0, 1, and 2 under the dataset's registered evaluation folds. The best/second/third results in each numeric column are highlighted with yellow / orange / blue backgrounds, respectively. Superscripts *tr*/*ft*/*fr* denote train/pretrained fine-tuning/frozen probe. A dash denotes an unavailable metric; † marks HYGRIP entries with no official preprocessing.

Result analysis. EFiMAS achieves the highest BA on seven of the ten tasks. Among non-foundation end-to-end fusion and unimodal baselines, it ranks first on eight tasks; EEG-Conformer is higher only on ds004514 and FJMI. Among foundation models, EFiMAS leads on seven tasks; CBraMod is higher on ds004514 when trained from scratch and on TU HBCI and FJMI with pre-trained full fine-tuning, while CodeBrain full fine-tuning is also higher on TU HBCI. The matched replay in Appendix B.1.1 compares each selected program with one common fixed path on the same evaluation rows and splits.

4.4 Ablation Study

The ablations follow the protocol in Section 4.1 and change one evidence source or decision branch while keeping the remaining prompts and output contract fixed. We evaluate Figshare, TU HBCI, and TU simultaneous under the same ten-attempt budget. Table 2 reports the BA-selected attempt for each condition, using the latest attempt to break a BA tie and taking macro-F1 from that attempt; this rule is separate from the Reviewer-frozen program reported in the main table. Full EFiMAS is the main-table reference. Attempts are ordered research iterations rather than independent seeds, and fold-level BA and macro-F1 are averaged without fold-size weighting. Appendix B.3.2, Figure 10, shows all terminal outcomes and marks the selected attempts.

	Figshare		TU HBCI		TU sim.	
Condition	BA	Macro-F1	BA	Macro-F1	BA	Macro-F1
M-RAG	.5493	.5264	.6345	.6301	.5737	.4738
N-RAG	.6253	.6200	.6381	.6376	.6692	.6585
Merged	.6190	.5886	.6345	.6207	.6244	.6133
No-Ver.	.6333	.6241	.6238	.6199	.6610	.6526
No-Rev.	.6387	.6362	.6345	.5982	.6135	.5992
Full	**.7160**	**.7124**	**.7411**	**.6379**	**.6873**	**.7602**

Table 2: Component ablations. Unweighted fold-level BA and macro-F1; Full is the main-table reference.

The rows test *Dual-Evidence Retrieval*, *Independent Verification*, and *Stage Check and Program Revision*: M-RAG removes the neuroscience index, N-RAG removes the method index, Merged combines the two packets, No-Ver. removes the Verifier's independent check, and No-Rev. removes local re-execution or replanning after an output mismatch.

Evidence construction. Using either evidence index alone lowers BA on all three tasks. On Figshare, M-RAG performs worse than N-RAG. Merged also lowers BA and macro-F1 on all three tasks. Thus, access to both evidence types is not sufficient; keeping them separate by role also matters.

Verification and revision. Both No-Ver. and No-Rev. lower BA and macro-F1 on all three tasks. This consistent decline indicates that checking the plan before execution and correcting it after observed mismatches make complementary contributions.

Interpretation and audit trail. Together, these ablations show that performance depends on how evidence is separated, verified, and used to revise execution, in addition to evidence availability. Appendix G traces the first divergence between separate- and merged-evidence runs on TU simultaneous, while Appendix F gives the corresponding complete-system path and RAG decisions.The detailed discussion of task-specific path variation, matched-replay boundary cases, and study limitations is provided in Appendix E.

5 Related Work

Prior work includes unimodal EEG/fNIRS decoders, task-specific EEG–fNIRS fusion architectures, and agentic systems for brain-signal analysis (Song et al., 2023; Yang et al., 2023; Li, 2026; Si et al., 2025; Zhao et al., 2026; Zhou et al., 2026). These methods provide strong building blocks, but their analysis structures remain fixed after design. EFiMAS instead constructs, independently verifies, and revises a task-conditioned fusion program. The complete discussion is provided in Appendix D.

6 Conclusion

EFiMAS treats EEG–fNIRS fusion as a task-conditioned program-construction and revision problem. A DataProfile defines the signals and protocol. Role-specific evidence turns candidate relations into executable constraints. Observed outputs guide repair before Reviewer acceptance freezes the program. Across ten tasks, EFiMAS achieves the highest BA on seven and exceeds the common fixed program on nine. The selected paths range from spatial-prior fusion to delayed hemodynamic representations. The ds004514 result provides a useful boundary case: its compact fixed path remains stronger under the subject-dependent/within-subject five-fold formal replay. These results connect task evidence, executable choices, and held-out performance in one inspectable workflow.

ETHICS STATEMENT

This work performs secondary analysis of publicly released EEG–fNIRS datasets; we recruit no participants, collect no new measurements, and introduce no intervention. We use each dataset only within the access, citation, and usage conditions specified by its source repository. Our analysis does not request or expose direct personal identifiers, and the held-out labels and test outcomes remain outside the planning, retrieval, execution, and review loop. This separation reduces evaluation leakage but does not remove the possibility of bias inherited from the source cohorts, labels, recording protocols, or class imbalance. Reported performance should therefore be interpreted as benchmark results for the registered tasks and splits, rather than evidence of clinical validity or equal performance across populations.

The language-model agents can generate unsuitable hypotheses, tools, or interpretations. To limit these risks, EFiMAS uses role-specific verification, registered tool contracts, output checks, fold-local fitting, finite research budgets, and reviewer acceptance before held-out evaluation. Generated tools are subject to source, dependency, leakage, and resource checks, but these controls do not constitute a security or clinical certification. The system is intended for research assistance and reproducible method development, not diagnosis, triage, treatment, or other decisions affecting individuals.

REPRODUCIBILITY STATEMENT

We use public datasets with dataset identifiers and citations listed in the paper. The evaluation folds, preprocessing rules, seeds, metrics, baseline settings, evidence indexes, tool contracts, prompts, and execution traces are documented in the main text and appendices, including Appendices A, C, H, and F. The supplementary artifacts preserve the registered protocols and quantitative outputs needed to reproduce the reported comparisons. The anonymized implementation connection is available at https://anonymous.4open.science/r/efimas-anonymous-code-30E6; the code repository will be populated with the release implementation before publication.

AI USE STATEMENT

We used generative AI as an explicit component of the EFiMAS research workflow. In particular, the language-model agents proposed and refined EEG–fNIRS fusion hypotheses, helped design and implement dataset-specific fusion programs, supported experiment execution and review, and interpreted intermediate execution outputs. These uses are part of the method evaluated in this paper, rather than a substitute for the authors' scientific decisions. We also used generative AI to edit the manuscript for English readability. The authors reviewed and verified all AI-assisted text, code, experimental artifacts, claims, and citations, and take responsibility for the final content of this paper.

REFERENCES

Md Atik Ahamed, Mihir Parmar, Palash Goyal, Yiwen Song, Long T. Le, Qiang Cheng, Chun-Liang Li, Hamid Palangi, Jinsung Yoon, and Tomas Pfister. TFRBench: A reasoning benchmark for evaluating forecasting systems. *arXiv preprint arXiv:2604.05364*, 2026. doi: 10.48550/arXiv.2604.05364. URL https://arxiv.org/abs/2604.05364.

Johann Benerradi, Jeremie Clos, Aleksandra Landowska, Michel F. Valstar, and Max L. Wilson. Benchmarking framework for machine learning classification from fNIRS data. *Frontiers in Neuroergonomics*, 4, 2023. doi: 10.3389/fnrgo.2023.994969.

Andrea Biasiucci, Benedetta Franceschiello, and Micah M. Murray. Electroencephalography. *Current Biology*, 29(3):R80–R85, 2019. doi: 10.1016/j.cub.2018.11.052.

Alessio Buccino. EEG-fNIRS hybrid SMR BCI data s9–s15. Figshare dataset, 2015. URL https://figshare.com/articles/dataset/EEG_fNIRS_hybrid_SMR_BCI_data_S9_S15/1619641.

György Buzsáki, Costas A. Anastassiou, and Christof Koch. The origin of extracellular fields and currents: EEG, ECoG, LFP and spikes. *Nature Reviews Neuroscience*, 13:407–420, 2012. doi: 10.1038/nrn3241.

Danyang Chen, Jian Shi, Bo Tao, Xingwei Zhao, Zhixian Zhao, Shengjie Li, Yeguang Xu, Tao Ding, Peng Zhang, Qing Ye, Kuiyou Chen, Zhuojin Wu, Yingxin Tang, Wei Jiang, Kai Shu, Li Huang, Zheng You, Ping Zhang, and Zhouping Tang. A novel transfer learning-based hybrid EEG-fNIRS brain-computer interface for intracerebral hemorrhage rehabilitation. *Advanced Science*, 12(43): e05426, 2025a. doi: 10.1002/advs.202505426.

Zemeng Chen, Chenyang Gao, Ting Li, Xiang Ji, Shuyu Liu, and Ming Xiao. Open access dataset integrating EEG and fNIRS during stroop tasks. *Scientific Data*, 10:618, 2023. doi: 10.1038/s41597-023-02524-1.

Ziru Chen, Shijie Chen, Yuting Ning, Qianheng Zhang, Boshi Wang, Botao Yu, Yifei Li, Zeyi Liao, Chen Wei, Zitong Lu, Vishal Dey, Mingyi Xue, Frazier N. Baker, Benjamin Burns, Daniel Adu-Ampratwum, Xuhui Huang, Xia Ning, Song Gao, Yu Su, and Huan Sun. ScienceAgentBench: Toward rigorous assessment of language agents for data-driven scientific discovery. In *International Conference on Learning Representations*, 2025b. doi: 10.48550/arXiv.2410.05080. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/f12b4df26344f3be803c06b555252efe-Abstract-Conference.html.

Berkay D"oner, Thorir Mar Ingolfsson, Luca Benini, and Yawei Li. LUNA: Efficient and topology-agnostic foundation model for EEG signal analysis. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.48550/arXiv.2510.22257. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/66969a9e6bd7a26dfeccea7227178ca7-Abstract-Conference.html.

Lufeng Feng, Baomin Xu, Li Duan, Shifan Jia, Ziyu Jia, and Wei Ni. A new dual-attention multi-node fusion network for EEG-fNIRS motor imagery classification. *IEEE Journal of Biomedical and Health Informatics*, 2026a. doi: 10.1109/JBHI.2026.3706103.

Lufeng Feng, Baomin Xu, Haoran Zhang, Bihai Lin, Zuxuan Deng, Sidi Tao, Chenyu Liu, Shifan Jia, Li Duan, and Ziyu Jia. A multimodal fNIRS–EEG dataset for unilateral limb motor imagery. *Scientific Data*, 2026b. doi: 10.1038/s41597-026-07807-x.

Elizabeth M. C. Hillman. Coupling mechanism and significance of the BOLD signal: A status report. *Annual Review of Neuroscience*, 37:161–181, 2014. doi: 10.1146/annurev-neuro-071013-014111.

Wei-Bang Jiang, Li-Ming Zhao, and Bao-Liang Lu. Large brain model for learning generic representations with tremendous EEG data in BCI. In *International Conference on Learning Representations*, 2024. doi: 10.48550/arXiv.2405.18765. URL https://openreview.net/forum?id=QzTpTRVtrP.

James M. Kilner, Jérémie Mattout, Richard N. Henson, and Karl J. Friston. Hemodynamic correlates of EEG: A heuristic. *NeuroImage*, 28(1):280–286, 2005. doi: 10.1016/j.neuroimage.2005.06.008.

Patrick Tser Jern Kon, Qiuyi Ding, Jiachen Liu, Xinyi Zhu, Jingjia Peng, Jiarong Xing, Yibo Huang, Yiming Qiu, Jayanth Srinivasa, Myungjin Lee, Mosharaf Chowdhury, Matei Zaharia, and Ang Chen. EXP-Bench: Can ai conduct ai research experiments? In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/c411f5b2d9c55f1685e72db224ad8b0e-Abstract-Conference.html.

Youngchul Kwak, Woo-Jin Song, and Seong-Eun Kim. FGANet: fNIRS-guided attention network for hybrid EEG-fNIRS brain-computer interfaces. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 30:329–339, 2022. doi: 10.1109/TNSRE.2022.3149899.

Seho Lee, Hee Ra Jung, In-Nea Wang, Min-Kyung Jung, Hakseung Kim, and Dong-Joo Kim. Multimodal EEG and fNIRS biosignal acquisition during motor imagery tasks in patients with orthopedic impairment. OpenNeuro dataset ds004022.

Hao Li. DeepSyncNet: Deep synchronized fusion network for EEG-fNIRS multimodal brain-computer interfaces. *Biomedical Signal Processing and Control*, 113:109100, 2026. doi: 10.1016/j.bspc.2025.109100.

Haoxuan Li, Tianci Gao, Jianhe Li, Yang Fan, Runze Shi, Weiran Wang, Tianxiang Zhao, Zezhao Wu, Xiaoyang Jiang, Qihui Zhang, Jia Li, Xiao Xiao, Kai Du, Xiaoxuan Jia, Chao Xie, and Lu Mi. BrainPilot: Automating brain discovery with agentic research. *arXiv preprint arXiv:2607.15079*, 2026. doi: 10.48550/arXiv.2607.15079. URL https://arxiv.org/abs/2607.15079.

Rihui Li, Thomas Potter, Weitian Huang, and Yingchun Zhang. Enhancing performance of a hybrid EEG-fNIRS system using channel selection and early temporal features. *Frontiers in Human Neuroscience*, 11:462, 2017. doi: 10.3389/fnhum.2017.00462.

Mutian Liu, Banghua Yang, Lin Meng, Yonghuai Zhang, Shouwei Gao, Peng Zan, and Xinxing Xia. STA-Net: Spatial–temporal alignment network for hybrid EEG-fNIRS decoding. *Information Fusion*, 119:103023, 2025. doi: 10.1016/j.inffus.2025.103023.

Jingying Ma, Feng Wu, Qika Lin, Yucheng Xing, Chenyu Liu, Ziyu Jia, and Mengling Feng. CodeBrain: Bridging decoupled tokenizer and multi-scale architecture for EEG foundation model. In *International Conference on Learning Representations*, 2026. doi: 10.48550/arXiv.2506.09110. URL https://openreview.net/forum?id=msJgEkjwh5.

Vanesa Muñoz, Manuel Muñoz-Caracuel, Brenda Y. Angulo-Ruiz, and Carlos M. Gómez. Neurovascular coupling during auditory stimulation: Event-related potentials and fNIRS hemodynamic. *Brain Structure and Function*, 228(8):1943–1961, 2023. doi: 10.1007/s00429-023-02698-9.

Pablo Ortega, Tong Zhao, and A. Aldo Faisal. HYGRIP: Full-stack characterization of neurobehavioral signals (fNIRS, EEG, EMG, force, and breathing) during a bimanual grip force control task. *Frontiers in Neuroscience*, 14:919, 2020. doi: 10.3389/fnins.2020.00919.

Milan Rybár, Riccardo Poli, and Ian Daly. Simultaneous EEG and fNIRS recordings for semantic decoding of imagined animals and tools. *Scientific Data*, 2025. doi: 10.1038/s41597-025-04967-0.

Felix Scholkmann, Stefan Kleiser, Andreas Jaakko Metz, Raphael Zimmermann, Juan Mata Pavia, Ursula Wolf, and Martin Wolf. A review on continuous wave functional near-infrared spectroscopy and imaging instrumentation and methodology. *NeuroImage*, 85:6–27, 2014. doi: 10.1016/j.neuroimage.2013.05.004.

Jian Shi, Danyang Chen, Xingwei Zhao, Zhixian Zhao, Shengjie Li, Yeguang Xu, Tao Ding, Zheng Zhu, Peng Zhang, Qing Ye, Yingxin Tang, Ping Zhang, Bo Tao, and Zhouping Tang. HEFMI-ICH: A hybrid EEG-fNIRS motor imagery dataset for brain–computer interface in intracerebral hemorrhage. *Scientific Data*, 2025. doi: 10.1038/s41597-025-06100-7.

Jaeyoung Shin, Alexander von Lühmann, Do-Won Kim, Jan Mehnert, Han-Jeong Hwang, and Klaus-Robert Müller. Simultaneous acquisition of EEG and NIRS during cognitive tasks for an open access dataset. *Scientific Data*, 5:180003, 2018. doi: 10.1038/sdata.2018.3.

Xiaopeng Si, Shuai Zhang, Zhuobin Yang, Jiayue Yu, and Dong Ming. A bidirectional cross-modal transformer representation learning model for EEG-fNIRS multimodal affective BCI. *Expert Systems with Applications*, 266:126081, 2025. doi: 10.1016/j.eswa.2024.126081.

Yonghao Song, Qingqing Zheng, Bingchuan Liu, and Xiaorong Gao. EEG Conformer: Convolutional transformer for EEG decoding and visualization. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 31:710–719, 2023. doi: 10.1109/TNSRE.2022.3230250.

Ilias Tachtsidis and Felix Scholkmann. False positives and false negatives in functional near-infrared spectroscopy: Issues, challenges, and the way forward. *Neurophotonics*, 3(3):031405, 2016. doi: 10.1117/1.NPh.3.3.031405.

Technical University of Berlin. Open access dataset for simultaneous EEG and NIRS brain–computer interfaces. TU Berlin dataset release, 2017. URL https://doc.ml.tu-berlin.de/hBCI/.

Jiquan Wang, Sha Zhao, Zhiling Luo, Yangxuan Zhou, Haiteng Jiang, Shijian Li, Tao Li, and Gang Pan. CBramod: A criss-cross brain foundation model for EEG decoding. In *International Conference on Learning Representations*, 2025. doi: 10.48550/arXiv.2412.07236. URL https://openreview.net/forum?id=NPNUHgHF2w.

Yuqing Wang, Zhiqiang Yang, Hongfei Ji, Jie Li, Lingyu Liu, and Jie Zhuang. Cross-modal transfer learning from EEG to functional near-infrared spectroscopy for classification task in brain-computer interface system. *Frontiers in Psychology*, 13:833007, 2022a. doi: 10.3389/fpsyg.2022.833007.

Zenghui Wang, Jun Zhang, Xiaochu Zhang, Peng Chen, and Bing Wang. Transformer model for functional near-infrared spectroscopy classification. *IEEE Journal of Biomedical and Health Informatics*, 26(6):2559–2569, 2022b. doi: 10.1109/JBHI.2022.3140531.

Zenghui Wang, Jihong Fang, and Jun Zhang. Rethinking delayed hemodynamic responses for fNIRS classification. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 31:4528–4538, 2023. doi: 10.1109/TNSRE.2023.3330911.

Zirui Wang, Guanghao Huang, Zhuochao Chen, Xiaorui Liu, Yinhua Liu, and Keum-Shik Hong. Decoupled bidirectional spatio-temporal fusion network for hybrid EEG-fNIRS cognitive task classification. *Brain Sciences*, 16(2):241, 2026. doi: 10.3390/brainsci16020241.

Chaoqi Yang, M. Brandon Westover, and Jimeng Sun. BIOT: Biosignal transformer for cross-data learning in the wild. In *Advances in Neural Information Processing Systems*, volume 36, 2023. doi: 10.48550/arXiv.2305.10351. URL https://openreview.net/forum?id=c2LZyTyddi.

Liuyin Yang, Qiang Sun, Ang Li, and Marc M. Van Hulle. Are EEG foundation models worth it? comparative evaluation with traditional decoders in diverse BCI tasks. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=5Xwm8e6vbh.

Weibo Yi, Jiaming Chen, Dan Wang, Xinkang Hu, Meng Xu, Fangda Li, Shuhan Wu, and Jin Qian. A multi-modal dataset of electroencephalography and functional near-infrared spectroscopy recordings for motor imagery of multi-types of joints from unilateral upper limb. *Scientific Data*, 2025. doi: 10.1038/s41597-025-05286-0.

Takuro Zama and Sotaro Shimada. Simultaneous measurement of electroencephalography and near-infrared spectroscopy during voluntary motor preparation. *Scientific Reports*, 5:16438, 2015. doi: 10.1038/srep16438.

Hao Zhang, Yiqing Hu, Yang Li, Dongwei Li, Hanli Liu, Xiaoli Li, Yan Song, and Chenguang Zhao. Neurovascular coupling in the attention during visual working memory processes. *iScience*, 27 (4):109368, 2024. doi: 10.1016/j.isci.2024.109368.

Sha Zhao, Mingyi Peng, Haiteng Jiang, Tao Li, and Shijian Li. EEGAgent: A unified framework for automated EEG analysis using large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 18063–18071, 2026. doi: 10.1609/aaai.v40i21.38867.

Yangxuan Zhou, Sha Zhao, Jiquan Wang, Shijian Li, and Gang Pan. BrainAgent: A large language model-driven multi-agent framework for autonomous brain signal understanding. *arXiv preprint arXiv:2606.25400*, 2026. doi: 10.48550/arXiv.2606.25400. URL https://arxiv.org/abs/2606.25400.

Yu Zhu, Runkai Zhao, Zhimin Zhou, Jinyu Cai, Zhouheng Yao, Chutian Zhang, Peiyuan Li, Xiaoxing Zhang, Qihao Zheng, Jiamin Wu, Mianxin Liu, Chi Zhang, Bin Min, Lei Bai, Chengyu Li, Jingpeng Wu, and Chunfeng Song. EasyBCI Agent: Towards universal neural data preprocessing for brain-computer interfaces. *arXiv preprint arXiv:2607.29007*, 2026. doi: 10.48550/arXiv.2607.29007. URL https://arxiv.org/abs/2607.29007.

APPENDIX

Contents

A Datasets, Evaluation Protocols, and Baselines 16
- A.1 Datasets . . . 16
- A.2 Evaluation protocols . . . 17
- A.3 Scientific basis of EEG–fNIRS relations . . . 17
- A.4 Baseline models . . . 19
- A.5 Evaluation metrics . . . 20

B Supplementary Quantitative Evaluation 20
- B.1 The Necessity of Task Conditioning . . . 20
- B.2 Evidence-Driven Program Synthesis . . . 24
- B.3 Execution Reliability and Mechanism Robustness . . . 29

C Prompt Protocol 38

D Extended Related Work 40

E Discussion and Limitations 41

F Case Analysis of a Selected EFiMAS Path 41
- F.1 Selected path . . . 42
- F.2 RAG decisions for the same path . . . 42
- F.3 Cross-task summary . . . 43
- F.4 Task-conditioned hypothesis formation and revision . . . 43

G Qualitative Component-Ablation Trace 46

H Implementation Details 47
- H.1 Workflow Pseudocode . . . 47
- H.2 DataProfile example . . . 47
- H.3 Dual-evidence retrieval details . . . 48
- H.4 Registered tool examples . . . 48
- H.5 Stage output checks . . . 49
- H.6 Generated-tool sandbox . . . 49
- H.7 Correspondence between method and implementation . . . 50

I Qualitative Execution Trace 51

J Robustness Across Language Models 51
- J.1 Experimental setup . . . 51
- J.2 Aggregate performance across models . . . 52
- J.3 Task-level sensitivity and trajectory quality . . . 52

K Dual RAG Knowledge Bases 53
- K.1 Knowledge-base construction and quality control . . . 53
- K.2 Example neuroscience evidence card . . . 55
- K.3 Example method evidence card . . . 56

L Tool Library and Tool-Card Contract 56
- L.1 Typical tool source . . . 56
- L.2 Example usage instruction supplied to the model . . . 57

M Three Feedback Loops in an Accepted Trajectory 57
- M.1 From a released dataset to a Planner hypothesis . . . 58
- M.2 Planner–Verifier loop . . . 60
- M.3 Execution/ReAct loop . . . 61
- M.4 Review and global feedback . . . 62

N Analysis of the Selected MIND Program 63

O Selected Paths Versus Existing Algorithms 64

A Datasets, Evaluation Protocols, and Baselines

This section details the datasets, evaluation protocols, and baselines used in the main comparisons. We describe each dataset and baseline individually to make the signal representation, task definition, and required input adaptation explicit.

A.1 Datasets

- **OpenNeuro ds004022** (Lee et al.) is a four-class Reach/Grasp/Lift/Twist motor task. We start from the released BIDS EEG and MATLAB MCOS fNIRS recordings, pair the modalities at the event boundary, and materialize EEG $[0, 6]$ s and fNIRS $[0, 10]$ s windows at their native sampling rates. Boundary padding is recorded rather than silently dropping an event.
- **MIND** (Feng et al., 2026b) is a four-class unilateral-movement task. Its public input already consists of event-level epochs: 64-channel EEG sampled at 250 Hz and 102-channel HbO/HbR fNIRS sampled at 10 Hz. We preserve these released epochs and their class order instead of treating them as continuous raw intensity streams.
- **Stroop** (Chen et al., 2023) contrasts neutral and incongruent cognitive-control trials. Published EEG epochs are paired with the corresponding MATLAB HbO/HbR blocks for 19 participants. Channel semantics are resolved before axis adaptation, and the published EEG duration and 100-Hz fNIRS representation are retained.
- **OpenNeuro ds004514** (Rybár et al., 2025) is a binary animal-versus-tool semantic-imagery task. The release provides BDF EEG, SNIRF fNIRS, and event annotations. We use EEG $[-1, 4]$ s at 200 Hz and fNIRS $[-2, 14]$ s at 10 Hz, allowing the electrical response and delayed hemodynamic response to retain different physical windows. The fixed-program replay aligns 24 trials per class for each of 12 subjects with the selected feature package. Its fixed branch follows the author's cue contract: a 0–0.6-s EEG cue representation at 128 Hz and a delayed fNIRS summary at 7.8125 Hz. The two branches share the same 576 trial identities, labels, and five within-subject folds; representation choice remains part of the program.
- **TU HBCI** (Technical University of Berlin, 2017) contrasts mental arithmetic with rest. The release contains continuous 32-channel EEG at 200 Hz and 36 optical paths at each of 760 and 850 nm sampled at 10 Hz. Across 29 participants and 87 recordings, 1,740 eligible events are materialized with a guided $[0, 10]$ s window and explicit boundary padding.
- **Figshare sensorimotor recordings** (Buccino, 2015) contain synchronized continuous EEG and two-wavelength fNIRS for left- and right-hand motor execution. The evaluated release has 15 participants, 21 EEG channels, 34 optical paths per wavelength, and 750 eligible task events. We materialize EEG $[0, 6]$ s and fNIRS $[0, 12]$ s windows at their native rates.
- **TU simultaneous** (Shin et al., 2018) is a binary word-generation task with simultaneous EEG and fNIRS from 26 participants over three sessions. EEG is represented at 200 Hz and the published HbO/HbR signals at 10 Hz. Trial identity pairs the modalities while the native time axes are retained within the guided $[0, 12]$ s raw event-analysis boundary. The selected derived representation is a downstream input: its release-specific event epoch and lag grid are used for feature construction, after which it slices three-second modality segments (600 EEG samples and 30 fNIRS samples) and constructs lagged fNIRS views before forming the spatial-prior token. Thus the three-second tensors do not replace the raw $[0, 12]$ s boundary.
- **HEFMI** (Shi et al., 2025) is a binary motor-imagery task in an intracerebral-hemorrhage cohort. We read the released EEG/NIRS/HCX representation, preserve the stated oxy-, deoxy-, and total-hemoglobin semantics, and use a guided $[-5, 23]$ s window. Delayed HbO/HbR/tHb views are therefore interpreted from the release manifest rather than treated as interchangeable channels.
- **HYGRIP** (Ortega et al., 2020) is a motor-execution task released as HDF5 EEG and oxy/deoxy fNIRS streams. The evaluated windows are EEG $[-2, 21]$ s and fNIRS $[0, 21]$ s at 200/10 Hz. Class definitions follow the released task annotations, and modality-specific windows preserve the expected neurovascular delay.
- **FJMI** (Yi et al., 2025) is an eight-class multi-joint motor-imagery task. CNT EEG and NIRX fNIRS are aligned by the released event codes; the evaluated representations use EEG $[0, 4]$ s and fNIRS $[3, 7]$ s at 200/10 Hz. The eight-class order is fixed before fitting.

A.2 Evaluation protocols

We use four protocol terms throughout the paper. *Cross-subject* means that complete participants are held out. *Within-subject* means that blocks or trials are held out within each participant. *Subject–session grouped* keeps every participant–session unit atomic. *Release-defined* uses the official split supplied with the dataset. The task-specific settings are:

- **ds004022, MIND, Figshare, HYGRIP, and FJMI:** five cross-subject, subject-grouped folds. The held-out unit is one or more complete participants, and no trial from a held-out participant contributes to fitting or model selection.
- **ds004514:** within-subject five-fold evaluation. Each participant contributes the same five-fold structure, trials are held out within that participant, and all preprocessing, feature construction, representation selection, and prediction-head fitting are performed using the participant's training trials only. The selected and fixed branches use the same 576 trial identities and the same fold assignments.
- **Stroop:** exhaustive paired leave-one-block-out evaluation within each of the 19 participants. A held-out unit is one paired task block from the same participant.
- **TU HBCI:** the release-defined official Dataset B five-fold split. Release units remain atomic and one official fold is held out at a time.
- **TU simultaneous:** three subject–session-grouped folds over 78 participant–session units. A complete participant–session unit is unavailable during fitting when it belongs to the held-out fold.
- **HEFMI:** five within-subject folds for each participant. The held-out unit consists of trials from the same participant, while every fitted transformation is estimated only from that participant's training trials.

The data-access boundary is otherwise identical across tasks. Splits are fixed before fitting. Preprocessing parameters, feature transformations, fusion parameters, feature selection, normalization, and prediction heads are fitted using the training partition only. After the executable program is frozen, the evaluator accesses each held-out unit once. Held-out labels, predictions, and metrics are never returned to Planning, Verification, Execution, or Review. These task-specific splits preserve each registered comparison; all main-table comparisons are therefore made within a dataset, not by averaging incompatible protocols across datasets.

A.3 Scientific basis of EEG–fNIRS relations

EEG–fNIRS fusion can exploit temporal alignment, electrical activity, hemodynamic responses, and spatial organization. These four relation families structure the choices illustrated in Figure 1. Table 3 connects each family to its neurophysiological or methodological basis. EFiMAS uses this evidence, together with the available signals and event metadata, to formulate task-specific fusion hypotheses.

Relation family	Scientific interpretation	Representative support
Temporal alignment (T1–T5)	Event locking, the earlier electrical response, modality-specific windows, and stage-wise segmentation. The family encodes that EEG and fNIRS are paired by event but need not share the same physical time window.	Neurovascular timing and delayed hemodynamics (Hillman, 2014; Kilner et al., 2005; Muñoz et al., 2023; Wang et al., 2026)
EEG electrical activity (E1–E7)	ERP, oscillatory power, ERD/ERS, time–frequency representations, and EEG spatiotemporal summaries. These atoms describe electrical activity without treating it as a hemodynamic or anatomical measurement.	EEG field and spectral analyses, including motor-band features (Buzsáki et al., 2012; Biasiucci et al., 2019; Li et al., 2017)
fNIRS optical/hemodynamic response (H1–H8)	HbO/HbR/tHb changes, optical-density summaries, delayed trajectories, AUC/slope/peak statistics, and HRF/GLM coefficients. The family keeps optical and hemoglobin quantities tied to the release semantics.	fNIRS methodology, hemodynamic inference, and delayed-response modeling (Scholkmann et al., 2014; Tachtsidis & Scholkmann, 2016; Kilner et al., 2005; Wang et al., 2023)
Spatial and functional organization (S1–S3)	Sensorimotor ROIs, left–right pairing, and common EEG/fNIRS topology. These atoms specify where a relation is measured, rather than adding a new temporal or signal-amplitude claim.	Motor preparation, fNIRS-guided spatial attention, and multimodal spatial alignment (Zama & Shimada, 2015; Kwak et al., 2022; Liu et al., 2025; Shin et al., 2018)

Table 3: Scientific basis of the four EEG–fNIRS relation families in Figure 1. T, E, H, and S denote temporal alignment, EEG electrical activity, fNIRS hemodynamic responses, and spatial organization, respectively.

Table 4 summarizes how these relations combine for individual tasks. Each combination links the task's available measurements to a candidate representation and fusion mechanism.

Dataset	Relation combination	Mechanistic interpretation and source
Figshare	T3 + E1 + H2	Different EEG/fNIRS event windows, motor-band EEG power, and optical-density slope/peak features (Buccino, 2015; Li et al., 2017).
MIND	T2 + E2 + H3 + S1 + S2	Fast EEG spectral features are paired with slower fNIRS channel summaries and sensorimotor spatial pairings in an asymmetric interaction (Feng et al., 2026b; Muñoz et al., 2023).
HYGRIP	T2 + E1 + H5 + S1 + S2	Motor-band lateralization and sensorimotor ROIs are paired with delayed HbO–HbR trajectories and left–right channel differences (Ortega et al., 2020; Zama & Shimada, 2015).
FJMI	T1 + E2 + H2 + S1	Public event codes align the trial, after which log-Welch EEG bands, hemodynamic slopes, and sensorimotor ROIs are fused (Yi et al., 2025).
HEFMI	T1 + E2 + H6	Event-locked EEG bands are combined with release-defined HbO/oxy statistics (Shi et al., 2025).
ds004022	T1 + E2 + H2	Event-aligned motor EEG and fNIRS reference features are combined without changing the release event identity (Lee et al.).
Stroop	T2 + E4 + E5 + H1	Early frontocentral ERP/spectral features are paired with later HbO/HbR response and quality measures (Chen et al., 2023; Kilner et al., 2005).
TU HBCI	T4 + E3 + H2	Workload EEG bands are combined with wavelength-specific optical-density time features (Technical University of Berlin, 2017).
ds004514	T5 + E6 + H8	Concept/stage EEG time–frequency views are paired with fNIRS HRF/GLM coefficients (Rybár et al., 2025).
TU simultaneous	T1 + E7 + H7 + S3	Event-paired EEG/fNIRS stage and wavelength contrasts are summarized with a shared grid spatial topology (Shin et al., 2018).

Table 4: Task-specific combinations of EEG–fNIRS relations and their scientific interpretations. These combinations guide hypothesis formation before the executable path is selected.

The literature supplies the physiological interpretation and conditions of use of each relation. The available measurements and training-fold requirements then determine how it can be implemented, and held-out evaluation measures the predictive performance of the resulting program. This connection between evidence, data, and execution explains how EFiMAS constructs different fusion paths for different tasks.

A.4 BASELINE MODELS

A.4.1 EEG–FNIRS FUSION BASELINES

- **DAMF-Net** (Feng et al., 2026a) is a dual-attention, multi-node fusion network for EEG–fNIRS motor-imagery classification. Its two modality inputs are adapted separately to the released multimodal interface before the attention and fusion blocks are trained within each dataset fold.
- **DeepSyncNet** (Li, 2026) is a deep synchronized fusion network that learns a joint EEG–fNIRS representation for multimodal BCI decoding. We preserve its early synchronized-fusion mechanism while adapting each dataset's paired windows to the required channel–time input.
- **BiSTF-Net** (Wang et al., 2026) performs decoupled, bidirectional spatio-temporal fusion. Its adapter supplies paired EEG and fNIRS tensors while retaining the model's spatial guidance and temporal alignment operations.

A.4.2 SINGLE-MODALITY BASELINES

- **EEG-Conformer** (Song et al., 2023) combines convolutional local feature extraction with Transformer self-attention for EEG decoding. Dataset EEG windows are mapped to its released channel–time interface.
- **BIOT** (Yang et al., 2023) is a Transformer for cross-dataset biosignal representation learning. We provide only the EEG stream and use the released EEG tokenization and task-head interface.
- **fNIRS-Transformer** (Wang et al., 2022b) applies a Transformer to channel–time fNIRS signals. The dataset fNIRS window is adapted to this interface without introducing EEG information.
- **fNIRSNet** (Wang et al., 2023) explicitly models delayed hemodynamic responses for fNIRS classification. It receives the dataset's fNIRS representation and the task-specific physical window defined above.
- **BenchNIRS CNN** (Benerradi et al., 2023) is the convolutional fNIRS classifier provided by the BenchNIRS benchmarking framework. We retain its convolutional input contract and train it on the fNIRS stream only.

A.4.3 EEG FOUNDATION-MODEL BASELINES

- **LaBraM** (Jiang et al., 2024) learns generic EEG representations from large-scale BCI corpora. The evaluated EEG input is mapped to its released interface. We evaluate the task-trained setting reported in the main table.
- **CBraMod** (Wang et al., 2025) is a criss-cross EEG foundation model that captures spatial and temporal dependencies. We report training from initialization, official pretraining followed by full fine-tuning, and an official-pretrained frozen linear probe.
- **LUNA** (D"oner et al., 2025) is an efficient, topology-agnostic EEG foundation model. The same three initialization and update settings are evaluated after mapping each dataset's EEG channels to its released input contract.
- **CodeBrain** (Ma et al., 2026) combines a decoupled time–frequency tokenizer with a multi-scale EEGSSM encoder. EEG is resampled to 200 Hz and divided into 200-sample patches, following the released input contract. The official backbone is retained across variable channel counts and window lengths; channel–time tokens are globally averaged before the task head. We evaluate training from initialization, official pretraining followed by full fine-tuning, and an official-pretrained frozen linear probe.
- **STEEGFormer** (Yang et al., 2026) is an EEG representation model evaluated across diverse BCI tasks. We use the same task input adapter and the same training, full-fine-tuning, and frozen-probe definitions as for the other foundation models.

For every baseline, adaptation changes only the representation required by the model interface. The task head and any fitted normalization are estimated inside the training partition of each fold. For foundation models, *Train* initializes the architecture without a pretrained checkpoint, *Pretrained FT* starts from the released checkpoint and updates all weights, and *Frozen probe* trains only a linear task head over a fixed released backbone.

All baseline results are averaged over seeds 0, 1, and 2 under each dataset's registered evaluation folds. The *Train*, *Pretrained FT*, and *Frozen probe* settings of each foundation model use the same folds and the same three seeds, keeping their comparison focused on initialization and update regime.

A.5 Evaluation metrics

For a task with K classes, let k index a class, and let TP_k, FP_k, and FN_k denote its true-positive, false-positive, and false-negative counts. Balanced accuracy is the macro-average of class recall,

$$\mathrm{BA} = \frac{1}{K}\sum_{k=1}^{K}\frac{TP_k}{TP_k + FN_k}, \tag{21}$$

and macro-F1 is

$$\text{Macro-F1} = \frac{1}{K}\sum_{k=1}^{K}\frac{2TP_k}{2TP_k + FP_k + FN_k}. \tag{22}$$

Both metrics are computed by the held-out evaluator after the Reviewer-accepted program or baseline model has been frozen.

B Supplementary Quantitative Evaluation

Beyond the aggregate results in Table 1, we study how *EFiMAS* constructs task-specific programs, how retrieved evidence changes those programs, and how reliably the resulting graphs execute under replay and input perturbations. The supplementary evaluation is organized into three parts:

- **The necessity of task conditioning:** We first compare the selected programs with a matched fixed program, then ask whether a mechanism selected for one task transfers after its input contract is adapted, and finally quantify diversity among the executed paths (Sections B.1.1–B.1.5).
- **Evidence-driven program synthesis:** We next examine how counterfactual *DataProfiles*, the two retrieval branches, and the available context change the resulting executable program (Sections B.2.1–B.2.5).
- **Execution reliability and robustness:** The last sections quantify ReAct repair dynamics, Reviewer logic, exact graph replay, runtime contracts, and behavior under held-out modality perturbations (Sections B.3–B.3.4).

All comparisons retain the dataset-specific evaluation protocols stated in Section 4.1; no private test outcome is returned to Planning, Verification, Execution, or Review. Reviewer acceptance directly freezes the executable program for held-out evaluation. In particular, every ds004514 result in this supplement uses the within-subject five-fold evaluation defined in Appendix A.2.

B.1 The Necessity of Task Conditioning

B.1.1 Comparison with a sample- and split-matched fixed-program control

The main tables establish the task-level ordering against the complete baseline set. Table 5 adds uncertainty estimates for the selected EFiMAS programs and records the executable structure associated with each result. Confidence intervals are bootstrapped at the subject level when subject-resolved predictions are available and otherwise at the seed or held-out-fold level shown in the last column. These units describe the selected-program intervals in this table. The matched replay below reports its paired test units separately. The programs span quality-gated interaction, asymmetric fusion, spatial priors, delayed hemodynamic selection, and fold-local linear models; the selected solution is therefore not one fixed multimodal block repeated across tasks.

Dataset	Selected executable program	BA	95% CI	CI resampling unit
ds004022	Event windows → reference features → gated residual fusion → logistic head	.3024	[.2845,.3286]	7 subjects
MIND	Asymmetric EEG–fNIRS encoding → fNIRS gate → multiplicative interaction → logistic head	.2994	[.2792,.3206]	30 subjects
Stroop	Paired conflict blocks → ERP/frequency and HbO/HbR responses → quality-gated interaction	.6316	[.5658,.6974]	19 subjects
ds004514	Stage-aware semantic views → task-view fusion → ExtraTrees	.5750	[.5271,.6283]	12 subjects
TU HBCI	Event windows → workload OD features → fold-safe assembly → linear probe	.7411	[.7353,.7466]	3 seeds
Figshare	Motor-lateralization pair → fold-local feature concatenation → linear probe	.7160	[.6653,.7613]	15 subjects
TU simultaneous	Coordinate-aware spatial-prior token → Agent-prior MLP head	.6873	[.6856,.6889]	3 seeds
HEFMI	Delayed HbO/HbR/tHb representation → fold-local selection → fNIRS linear head	.6656	[.6368,.6965]	37 subjects
HYGRIP	Mu/beta lateralization → lagged hemodynamic trajectory alignment	.6768	[.6082,.7407]	14 subjects
FJMI	Event preprocessing → log-Welch and hemodynamic slopes → shrinkage LDA	.1653	[.1560,.1751]	5 held-out folds

Table 5: Selected EFiMAS programs and uncertainty estimates. The BA values are task-level results, and each confidence interval uses the independent resampling unit specified in the final column.

To separate task-conditioned program construction from a generic fixed alternative, we replayed each selected program and a common fixed program on the same evaluation rows and splits. The fixed path applies identity summaries, a deterministic six-dimensional tanh projection (registered as `concat_mlp`), and a fold-fitted ridge linear probe. Figure 4 reports the paired differences, bootstrap intervals, and two-sided sign-flip results. Repeated folds, sessions, and seeds are averaged within the independent subject or seed unit before inference. Paired inference is restricted to the selected and fixed programs, which share identical evaluation rows and splits. For ds004514, "identical splits" refers to the five within-subject folds for each of the 12 participants.

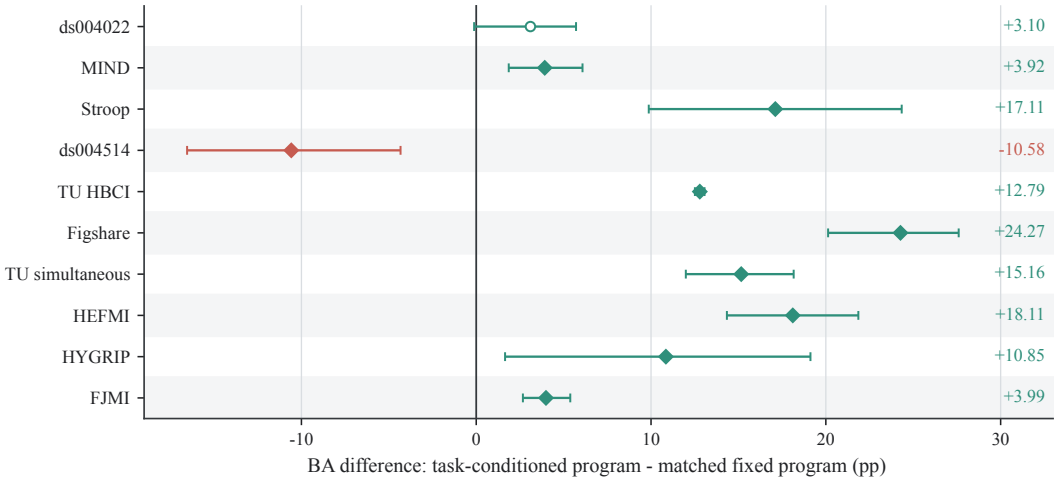

Figure 4: **Task-conditioned programs versus a sample- and split-matched fixed program.** Points show the paired difference in balanced accuracy (BA), in percentage points, on identical evaluation rows and splits; horizontal lines are paired 95% bootstrap intervals. For ds004514, the matched split uses within-subject five-fold evaluation. The vertical line marks equal performance. The open marker is not significant at $p < .05$ under the two-sided sign-flip test.

The selected program is better than the fixed program on nine of ten tasks, with eight positive differences reaching $p < .05$. The fixed program leads on ds004514 by .1058 BA under within-subject five-fold evaluation. This comparison aligns 576 trial identities across 12 subjects, averages five paired folds within each subject, and then applies the exact sign-flip test to the 12 subject differences. The selected branch uses 44 stage-aware features with ExtraTrees. The fixed branch uses the author cue contract, a delayed fNIRS summary, the deterministic projection, and the ridge probe. Its compact representation is better suited to this dataset. Across the ten-dataset comparison, EFiMAS

has the highest BA on seven tasks, while supervised EEG models lead ds004514, TU HBCI, and FJMI.

B.1.2 Comparison with the strongest main-table baseline

The matched fixed-program control above asks whether task-conditioned programs improve over one deliberately generic alternative. Here we give a direct, descriptive comparison with the highest-BA baseline listed for each task in Table 1. The comparator therefore varies by task; ΔBA is EFiMAS BA minus the corresponding baseline BA.

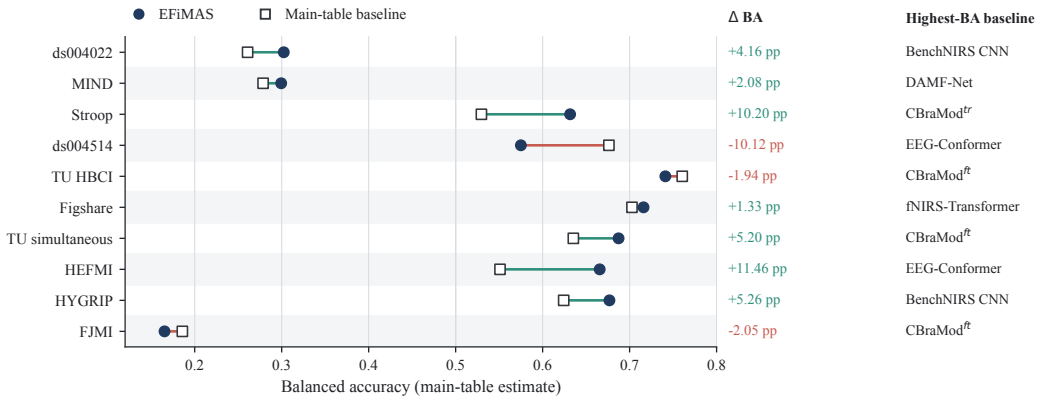

Figure 5: **Direct comparison with the highest-BA main-table baseline.** Each row connects the EFiMAS BA with the highest available baseline BA from Table 1; the baseline name is shown at right. The colored annotation is the direct EFiMAS minus baseline difference in percentage points; positive values favor EFiMAS and negative values favor the baseline.

The selected program has the larger main-table BA on seven tasks, while the highest main-table baseline is higher on three. The unweighted mean direct difference is +.0256 BA. The comparison identifies complementary regimes; the ds004514 contrast is evaluated under the subject-dependent/within-subject five-fold evaluation described in Appendix A.2: task-conditioned synthesis is strongest on HEFMI, Stroop, and TU simultaneous, whereas specialized baselines retain an advantage on ds004514, TU HBCI, and FJMI. These are descriptive main-table contrasts rather than pooled or paired participant-level effects.

B.1.3 Matched unimodal and fixed-fusion controls

We next isolate a simpler question that is obscured when architectures and splits differ: whether fixed feature concatenation is uniformly preferable to using EEG or fNIRS alone. Within every task, all three inputs share the same folds, training budget, test boundary, and downstream learner. Out-of-fold predictions are reconstructed before calculating BA. The paired contrast is fixed concatenation minus the stronger of the two unimodal controls, with the subject as the resampling unit and Holm correction over tasks.

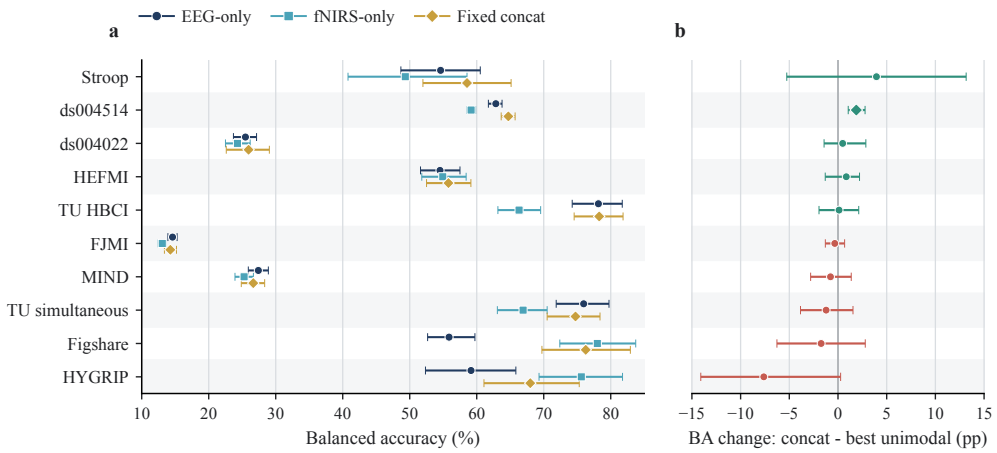

Figure 6: **Protocol-matched modality controls on ten tasks.** The figure reports EEG-only, fNIRS-only, and fixed-concatenation BA under identical task-specific splits. The paired panel compares concatenation with the stronger unimodal control. A diamond marks the comparison that remains significant after Holm correction.

Fixed concatenation has the highest pooled BA on five tasks, EEG on three, and fNIRS on two. Its mean difference from the stronger unimodal control is -.0044 BA (median -.0011), and only ds004514 remains significantly positive after correction. The corresponding average ranks are 1.50, 2.00, and 2.50 for concatenation, EEG, and fNIRS, respectively. The heterogeneous task-wise pattern motivates the interactions, gates, alignments, and unimodal fallbacks available to the task-conditioned system instead of a common concatenation rule.

HEFMI is the concrete asymmetric case in this comparison: its selected fNIRS-only path remains consistent with the formal two-input program by retaining the fNIRS branch while assigning no predictive role to EEG. It is therefore an extreme asymmetric-fusion case, not a change to the program definition or a requirement that both modalities contribute to every task.

B.1.4 Cross-task transfer after input adaptation

A direct graph-transfer test found that 96 of 99 cross-task transfers were blocked by incompatible sampling, channel, event, or shape contracts. We therefore introduced only target-side format adapters for sampling rate, channel summaries, and tensor dimensions while preserving each source graph's temporal, spectral, quality-gating, asymmetric-interaction, lag, or spatial mechanism. No target-task retrieval, replanning, graph alteration, or tool replacement was allowed. Every adapted program used the same class-balanced logistic head fitted inside the target training fold.

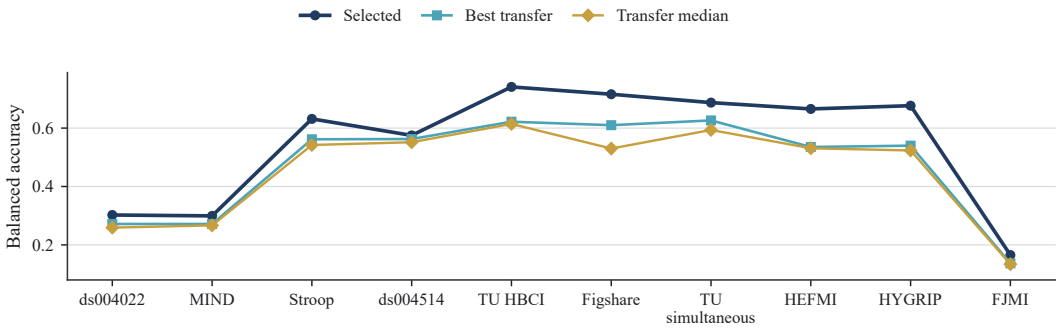

Figure 7: Cross-task transfer after input adaptation. The selected program outperforms both the median and strongest of nine adapted source-task mechanisms for every target. The three series share one BA axis; lines connect task columns only as visual guides.

Format adaptation raises executability from 3.03% to 100%, enabling a direct predictive comparison of the transferred mechanisms. All 90 adapted transfers remain below the program selected for the target task, with mean and median gaps of .0911 and .0906 BA, respectively. Input compatibility alone is thus insufficient for competitive transfer; the representation and fusion logic must also respond to the target task (Figure 7).

B.1.5 Diversity among executed paths

We quantify whether task-conditioned search produces structurally distinct programs. We sampled five unique completed paths from each of five task-specific run pools using a fixed seed. Tool aliases were normalized before computing set distance $1 - J$ and order distance $1 - \mathrm{LCS}/\max(n_i, n_j)$ over all ten within-task pairs.

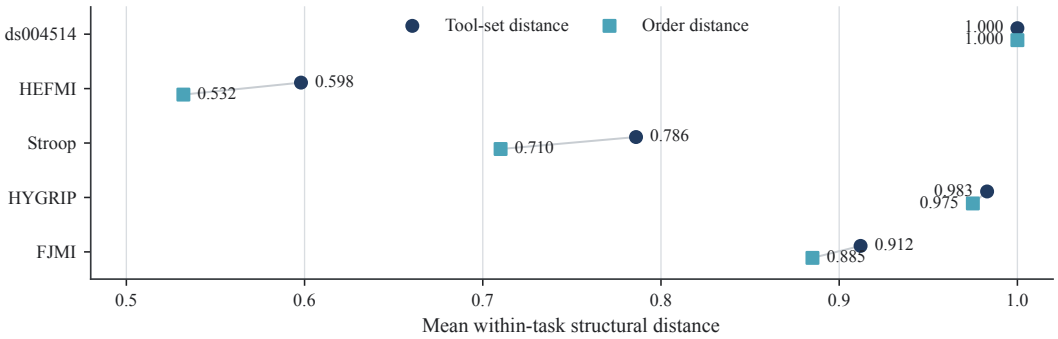

Figure 8: Diversity among executed paths. Markers show mean tool-set and operation-order distances over the ten within-task pairs formed by five sampled executable paths. The horizontal axis begins at .48 to distinguish the high-distance values; point labels report the exact mean distances.

The sampled paths cover unimodal models, feature concatenation, delayed coupling, quality gating, score fusion, and nonlinear interaction. High set and order distances in all five tasks show that the observed search space does not collapse to one sequence with different names (Figure 8). This diversity motivates output-guided revision during program construction, followed by held-out evaluation of the accepted programs.

B.2 Evidence-Driven Program Synthesis

B.2.1 Responses to counterfactual DataProfiles

We next test whether the decision process actually responds to task descriptions. For Figshare, TU HBCI, and TU simultaneous, Planning and Verification were run with the complete profile, a masked profile, and a profile whose task semantics were swapped. This intervention stops before downstream training, isolating the effect of profile information on the plan. Table 6 summarizes the resulting representation, fusion stage, head, and Verifier decision.

Task	Profile	Representation	Fusion / alignment	Head	Decision
Figshare	Full	Raw EEG + HbO	4-s shift, early 1D convolution	Linear softmax	Approve
Figshare	Masked	Alpha power + HbO	Separate 0–2/4–8-s windows, cross-attention	2-layer MLP	Approve
Figshare	Swapped	C3/C4 alpha + motor HbO	Early/late windows, bidirectional attention	2-layer MLP	Reject
TU HBCI	Full	Theta/alpha + HbO/HbR	0–10-s window, late concatenation	2-layer MLP	Revise
TU HBCI	Masked	Theta/alpha/beta + HbO/HbR	+2-s fNIRS, Pearson interaction	Linear softmax	Revise
TU HBCI	Swapped	Alpha log-power + HbO AUC	Session offset, late concatenation	Linear SVM	Approve
TU simultaneous	Full	Sliding EEG power + HbO/HbR	Temporal embedding, regularized tCCA	L2 logistic	Revise
TU simultaneous	Masked	Relative bandpower + HbO/HbR	Separate event windows, early concatenation	LDA/ridge	Approve
TU simultaneous	Swapped	Motor CSP + HbO/HbR	Fixed 2-s shift, regularized CCA	Linear LDA/SVM	Reject

Table 6: Planner–Verifier responses to complete, masked, and task-swapped DataProfiles. "Revise" denotes a candidate returned for another revision.

Across six executable fields–representation, alignment, quality handling, fusion operator, stage, and head–the masked and swapped plans differ from their complete-profile counterparts in 30 of 36 fields (mean normalized distance .8333). The Verifier changes the Planner's preferred candidate in 3 of 8 comparable exchanges. The two rejected swapped profiles show that task semantics can invalidate an otherwise plausible physiological plan, while the approved TU HBCI swapped case shows that this sensitivity is not perfect. This profile-level evidence complements Figure 7: the planner changes its program when the task description changes, and frozen programs lose accuracy when that adaptation is suppressed.

B.2.2 Downstream consequences of DataProfile interventions

The planning-only intervention in Table 6 establishes that profile fields change proposed programs. We extend that test through execution and held-out evaluation on Figshare, Stroop, and HEFMI. For each task, the original inputs, labels, runtime, folds, model endpoint, and single-attempt budget are fixed. The *masked* intervention removes descriptive profile fields, while *swapped* supplies another task's descriptive semantics but retains the target dataset identity and evaluation contract. All nine task–profile cells complete with a finite BA.

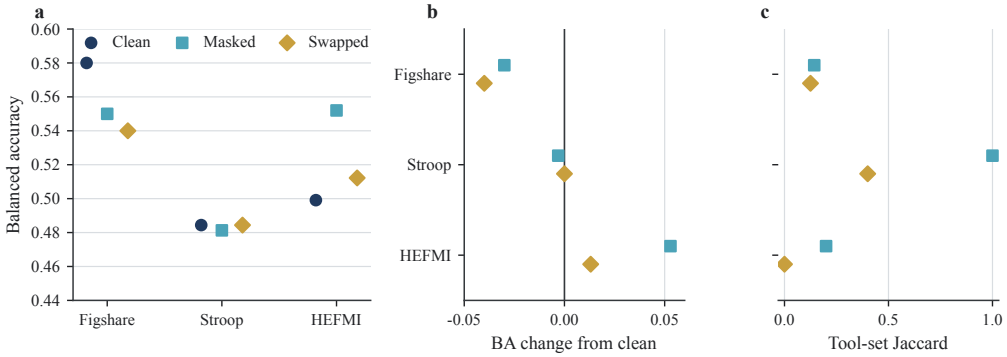

Figure 9: **DataProfile interventions propagated to executable programs and held-out BA.** Panels compare clean, masked, and task-swapped profiles, the resulting BA change, and tool-set overlap with the clean program. A low Jaccard index indicates that the intervention changed the tools actually executed, not only the natural-language rationale.

Five of the six clean–intervention comparisons alter the executable tool set. The associated BA response is task dependent: both interventions reduce Figshare BA, Stroop is nearly invariant despite

a structural change under the swapped profile, and both HEFMI interventions exceed this particular clean run. The controlled intervention establishes the DataProfile as an input to executable program selection: profile changes first alter hypotheses and contracts and can then propagate to different programs and predictions. The direction of the predictive change remains task dependent.

B.2.3 Ablating retrieval indices and revision roles

Table 2 quantifies the predictive effect of removing the neuroscience index, the method index, separate evidence packets, Verification, or output-guided revision. Table 7 then examines whether those performance differences are accompanied by changes in the implemented algorithm.

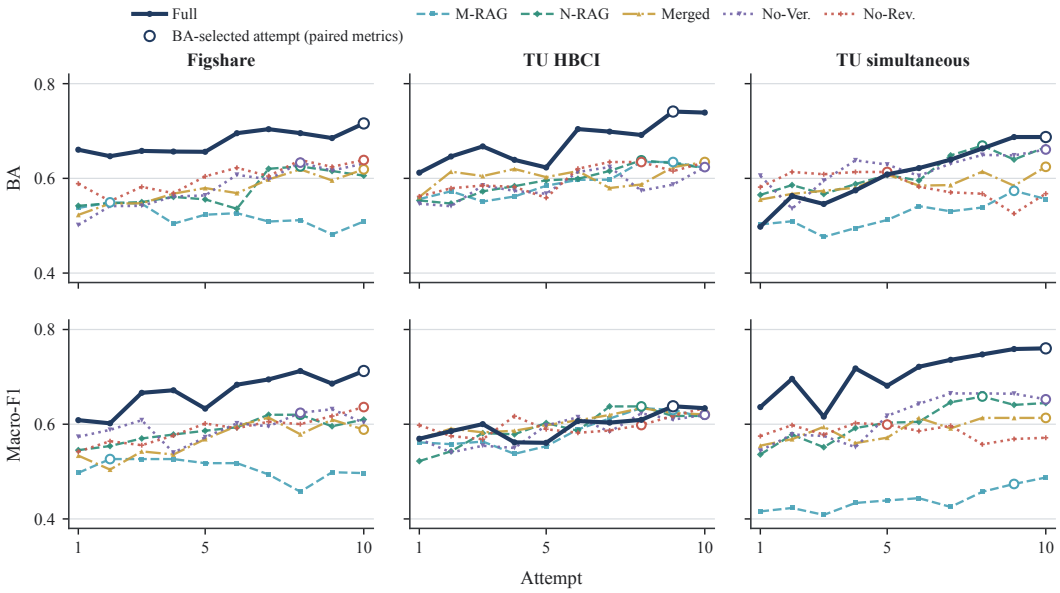

Figure 10: **Attempt-level terminal outcomes for the complete system and all component ablations.** Columns correspond to the three ablation tasks and rows to balanced accuracy (BA) and macro-F1. Each panel places Full EFiMAS and all five treatments under the same ten-attempt budget. Full is the complete-system reference; M-RAG and N-RAG retain only the method and neuroscience retrieval branches, respectively. Merged combines the evidence packets, No-Ver. removes independent Verification, and No-Rev. removes output-guided revision. Attempts are ordered research iterations rather than independent seeds. Open markers identify the BA-selected attempt used in the main ablation table; the same attempt is marked in both the BA and macro-F1 rows. A BA tie uses the latest tied attempt.

Figure 10 exposes the complete sequence of terminal outcomes rather than only one summary value per condition. The shared axes make the between-condition separation directly comparable within each task, while the nonmonotone paths show how the selected paired result relates to the earlier research attempts. The reported BA and macro-F1 correspond to the same BA-selected attempt.

Task	Evidence condition	Executed program after event materialization	BA
Figshare	Full	OD sensorimotor features → fold-local feature concatenation → logistic	.7160
Figshare	Method only	OD sensorimotor features → standardization → linear probe	.5493
Figshare	Neuroscience only	OD sensorimotor features → trial concatenation → linear probe	.6253
Figshare	Merged packet	Program generated from merged neuroscience and method evidence	.6190
Figshare	No Verification	Reference features → fold fusion → logistic	.6333
Figshare	No revision	OD Hilbert ERD/ERS → fold finalize → linear probe	.6387
TU HBCI	Full	OD workload features → fold fusion → logistic	.7411
TU HBCI	Method only	OD workload features → fold finalize → logistic	.6345
TU HBCI	Neuroscience only	OD workload features → fold finalize → linear probe	.6381
TU HBCI	Merged packet	OD workload features → fold finalize → logistic	.6345
TU HBCI	No Verification	Reference features → fold finalize → logistic	.6238
TU HBCI	No revision	OD workload features → fold finalize → logistic	.6345
TU simultaneous	Full	Spatial-prior token → prior-token head	.6873
TU simultaneous	Method only	Native-clock adapter → OOF probability stacking	.5737
TU simultaneous	Neuroscience only	HbO/HbR preprocessing → reliability interaction	.6692
TU simultaneous	Merged packet	HbO/HbR preprocessing → reference features → linear probe	.6244
TU simultaneous	No Verification	HbO/HbR preprocessing → fold fusion → block-token head	.6610
TU simultaneous	No revision	Physical window → prior token → prior-token head	.6135

Table 7: Representative executable programs and balanced accuracy for the component treatments in Table 2.

Removing neuroscience retrieval replaces task-specific physiological structure with more generic standardization, native-clock, or stacking routes. Removing method retrieval preserves a physiological representation more often, but changes how it is converted into a classifier. Merging the packets produces a completed but less effective program on all three tasks. The two indices therefore make different contributions: one narrows the scientific relation to test, while the other maps it to an executable operator and head. Their separation is also useful to Verification because a mechanism claim and an implementation claim can be checked independently.

The TU simultaneous no-Verification condition also produces a different program, with both BA and macro-F1 below the complete-system reference. Across the matrix, retrieval and revision components change both the executable path and its predictive outcome.

B.2.4 Controlled interventions on retrieved evidence

The preceding trace comparison shows how evidence conditions map to different programs. To test the effect under a common execution boundary, we ran a six-arm intervention on Figshare, TU HBCI, and TU simultaneous. The model, task protocol, global budget, evaluator, and downstream boundary are fixed; only the evidence supplied to the agents changes. The arms retain both RAG branches (Full), remove neuroscience evidence, remove method evidence, merge the two packets, shuffle the retrieved evidence, or exclude target-dataset evidence. All 18 cells complete and produce finite BA, so every contrast compares executed programs under the corresponding evidence condition.

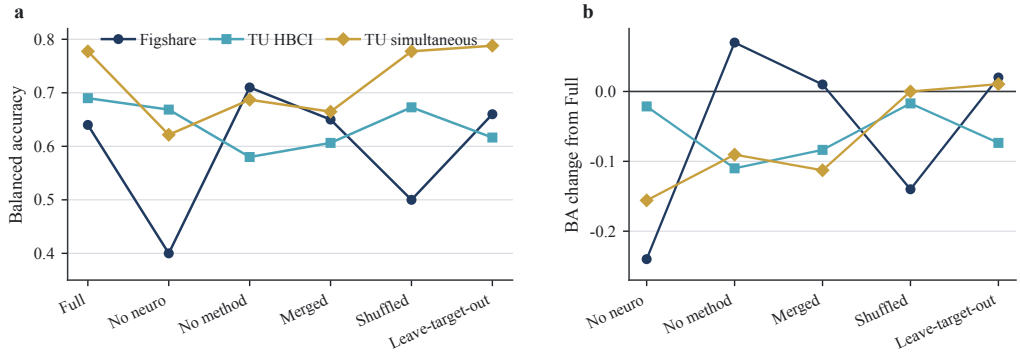

Figure 11: **Controlled RAG interventions.** Panels show the six BA outcomes and their within-task contrasts to full dual RAG. All 18 task–condition cells produce executable results.

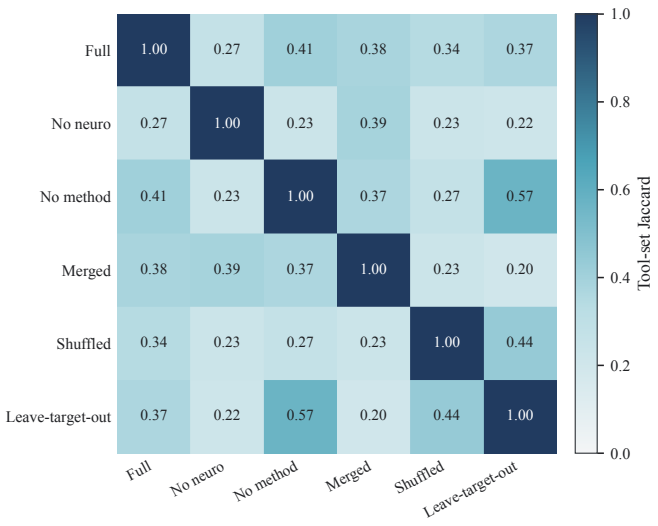

Figure 12: Mean tool-set Jaccard similarity between the executable programs selected under each evidence condition. Similarity to Full ranges from .273 to .406 across the ablated arms, showing that evidence interventions change the implemented program as well as its score.

Removing neuroscience evidence reduces BA on all three tasks, by .2400, .0214, and .1559, respectively. Its mean contrast is -.1391 and its leave-one-task-out direction is stable in all three subsets. With only three task-level pairs, the smallest attainable two-sided exact sign-flip p value is .25. Other interventions are less uniform: removing method evidence improves Figshare but reduces both TU tasks, and target-excluded evidence is close to Full on average while reducing TU HBCI by .0736. Together with the low path similarities in Figure 12, these results show that domain evidence is an active input to program construction. Neuroscience evidence is the most directionally stable component in this three-task intervention, while the value of evidence organization and target specificity remains task dependent.

B.2.5 Context efficiency and decision completeness

The preceding analyses concern scientific decisions. We also quantify the amount of context required to expose the data and tool space to the model. Table 8 compares retrieval with complete-catalog exposure and a bounded DataProfile with direct sequence serialization. Character-to-token conversions are estimates when provider token counts are unavailable.

Comparison	Bounded condition	Full exposure	Reduction / retrieval result
Tool retrieval, mean prompt tokens	15,023.5	483,906.5	96.90% reduction
Offline tool needs	top-6 retrieval	complete catalog	hit@6=.857, MRR@6=.452
Serialized data context, characters	8,484	334,788,403	99.9975% reduction
Planner prompt, characters	62,196	–	measured prompt
Verifier prompt, characters	211,210	–	measured prompt

Table 8: Context-size and retrieval measurements. Retrieval metrics quantify relevant-tool coverage and ranking within the top six results.

The complete catalog contains 227 tools and 1,784,322 characters. In four task-specific retrieval evaluations, candidate payloads contain 2–50,494 characters, and the majority of needs with explicit tool truth are retrieved within the top six. The four downstream blueprint drafts are nevertheless noncompilable. This separation is consequential: retrieval reduces context and can recover a relevant operator, but contract checks and iterative repair are still required to convert that operator into a valid program.

We also measure field completeness across eight sampled decision sequences. All eight contain a hypothesis, expected signature, Planner intent, and either a Verifier finding or a revision; six contain an explicit stage handoff, and five contain a finite BA result. These coverage counts complement the examples in Appendices F, G, and I by quantifying how consistently the corresponding decision fields are retained.

B.3 Execution Reliability and Mechanism Robustness

We next examine execution reliability at three resolutions: role-chain variants on one dataset, repair chains across completed executions, and a balanced test of Reviewer decisions. MIND is reported through the selected path used by the main comparison: an asymmetric EEG–fNIRS encoder, an fNIRS-conditioned gate, a multiplicative interaction, and a LogisticRegression head. The main table aggregates this path over seeds 0, 1, and 2; its BA is .2994. Held-out predictive outcomes are returned only after the graph is frozen and are not used to alter the selected program.

B.3.1 Repair dynamics and Reviewer logic

Table 9 compares completed Figshare executions with different active roles and feedback links using identical prepared inputs. Every setting produces evaluator-ready predictions; calls and tokens quantify the interaction cost associated with each completed execution.

Role-chain setting	BA	Macro-F1	Calls	Tokens	Attempt BA(s)
Direct, RAG, no loop	.6200	.6194	4	118,510	.6200
Direct, RAG, with loop	.5000	.4949	6	153,338	.5000
Full, one attempt	.7000	.6956	27	464,853	.7000
Planner + Executor feedback (run 1)	.7100	.6991	8	191,916	.7100
Planner + Verifier	.7600	.7596	13	241,164	.7600
Direct, no RAG, no loop	.6300	.6255	4	108,405	.6300
Planner + Executor feedback (run 2)	.5200	.5130	12	284,563	.5200
Full, three attempts	.7100	.7076	81	1,575,843	.7100/.5000/.6700

Table 9: Completed role-chain variants on Figshare. Calls and token totals quantify the interaction cost of each variant alongside its predictive performance.

The best single completed row is Planner + Verifier at .7600 BA, but the two Planner + Executor runs differ by .1900 and the three-attempt run is nonmonotone. These outcomes motivate a trace-level question: when an execution error is observed, does a later ReAct revision actually repair it?

Repair dynamics during execution The analysis covers 18 unique runs over five tasks and 219 numbered ReAct rounds. An initial failure chain begins with a tool execution error or an unmet input, output, artifact, shape, fit-scope, or schema prerequisite. Repeated messages with the same run, attempt, stage, and root cause are counted once. A chain is marked repaired only when a later round in the same attempt and stage records `advance` or `stage_completed`; provider failures and BA review decisions are excluded from the denominator.

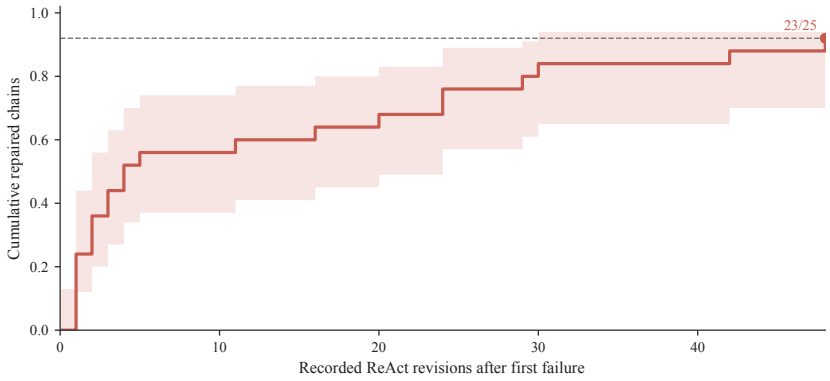

Figure 13: Cumulative repair of 25 observed tool or contract failure chains as a function of ReAct revisions after the first failure. The band is a run-cluster bootstrap interval; the dashed line marks the final repair rate.

Twenty-three of 25 chains are repaired, for a rate of 92.0% (Wilson 95% CI [75.0%,97.8%]; run-cluster bootstrap CI [72.7%,100%]). The median repaired chain requires four revisions. Six chains are repaired within one revision, 13 within four, and 23 within 48; the two unresolved chains remain in the denominator. Figure 13 therefore shows both the capacity for recovery and the sometimes substantial interaction cost of that recovery.

Reviewer discrimination of consistent and invalid conclusions Starting from 20 valid Reviewer contexts, we construct one matched invalid version of each context. Each pair contains the original logically consistent context and one version with exactly one controlled contradiction: an inconsistent metric, a mismatched hypothesis identifier, nonexistent evidence, an unexecuted tool path, or an unsupported causal/superiority conclusion. The resulting 40 cases are balanced between consistent and invalid conclusions. The numerical gate tests whether BA is significantly above the task's chance level rather than applying one BA threshold across heterogeneous tasks.

True class	Predicted consistent	Predicted invalid
Consistent	17	3
Invalid	3	17

Metric	Estimate	Pair-cluster 95% CI
Accuracy	.850	[.750,.950]
Balanced accuracy	.850	[.750,.950]
Invalid-class F1	.850	[.737,.950]
MCC	.700	[.503,.900]

Table 10: Reviewer discrimination on 20 consistent and 20 controlled-invalid trajectory contexts. Confidence intervals cluster the bootstrap by the 20 source-trajectory pairs.

The Reviewer correctly classifies 34 of 40 contexts, with three errors in each direction. The balanced errors and .850 balanced accuracy show that the Reviewer distinguishes logical validity from execution completion and scalar performance alone.

To quantify this distinction with an independent domain judgment, two EEG–fNIRS experts rated the same 40 contexts without access to the constructed labels or the model output. Their logical judgments agreed on 37 of 40 cases (raw agreement .925, Cohen's $\kappa = .850$, pair-cluster 95% CI [.700,1.000]). The Reviewer reached balanced accuracy .850 against the first expert and .781 against the second; its MCC values were .700 and .556, respectively. Thus the automated audit agrees with independent scientific judgment at the level of the central logical decision.

The paired expert ratings also quantify what changes when a logically consistent context is converted into its controlled-invalid counterpart. Table 11 reports the mean difference over the 20 source-

trajectory pairs. The largest separations occur in hypothesis–evidence alignment and overall research quality, followed by input–output compatibility and falsifiability. These dimensions are the parts of the trajectory that the Reviewer is required to inspect.

Expert dimension	Consistent	Invalid	Difference	95% CI	p_{Holm}
Evidence relevance	2.975	2.500	+.475	[.100,.950]	.0910
Physiological plausibility	3.550	3.450	+.100	[.025,.200]	.0910
Hypothesis–evidence alignment	3.250	1.700	+1.550	[1.350,1.750]	$< .001$
Input–output compatibility	4.425	3.525	+.900	[.350,1.550]	.0318
Falsifiability	3.450	3.050	+.400	[.250,.575]	.0060
Overall research quality	3.250	2.000	+1.250	[1.100,1.400]	$< .001$

Table 11: Independent expert ratings of 20 consistent and 20 controlled-invalid Reviewer contexts. Scores use a 1–5 scale. Differences are paired within the 20 source-trajectory pairs; confidence intervals use pair-cluster bootstrap and p values are Holm-corrected across the six dimensions.

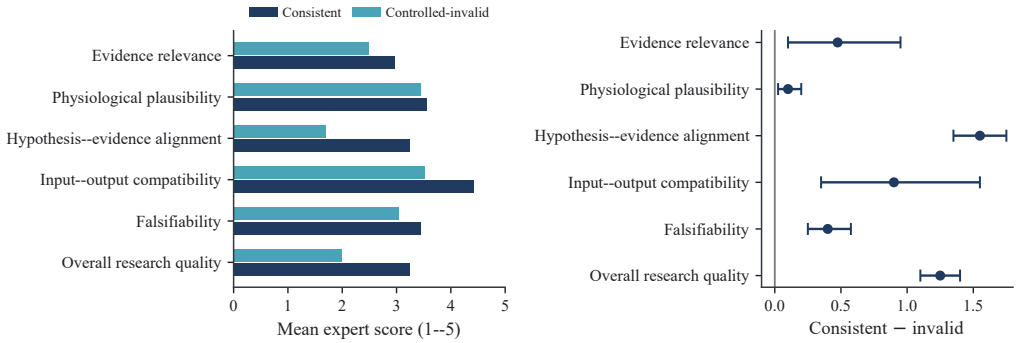

Figure 14: Expert assessment of Reviewer contexts. The left panel shows mean ratings for logically consistent and controlled-invalid contexts; the right panel shows the paired difference and its 95% confidence interval.

The separate expert audit of 40 trajectory fragments provides a second measure of the process evidence used in Appendices F.4, G, and M. Table 12 reports all six assessed dimensions. The highest mean scores are obtained for physiological plausibility (3.470), falsifiability (3.417), and stage-handoff quality (3.146). Overall research quality reaches 2.837 with a source-cluster 95% CI of [2.671,2.986]. Agreement is strongest for evidence relevance and input–output compatibility, while the stage-handoff judgments also show substantial consistency between the two experts.

Expert dimension	n	Mean	Source-cluster 95% CI	ICC(2,2)
Evidence relevance	8	2.688	[2.000,3.357]	.971
Physiological plausibility	33	3.470	[3.393,3.607]	.479
Input–output compatibility	24	2.854	[2.500,3.119]	.754
Stage-handoff quality	24	3.146	[2.833,3.407]	.753
Falsifiability	24	3.417	[3.267,3.611]	.589
Overall research quality	40	2.837	[2.671,2.986]	.649

Table 12: Independent expert assessment of trajectory fragments sampled from planning, execution, and feedback records. Ratings use a 1–5 scale. The number of applicable fragments varies by dimension; confidence intervals are clustered by source trajectory.

Three-seed evaluation of the selected MIND program MIND is a four-class task evaluated with the same group-wise five-fold protocol for seeds 0, 1, and 2. The selected program has BA 0.299444, rounded to .2994 in the main table. The generated candidate is frozen before evaluation and follows the asymmetric EEG–fNIRS path listed below.

Field	Setting	Executed path	BA	Macro-F1
Program	Selected MIND path	EEG/fNIRS asymmetric encoding → fNIRS gate → multiplicative interaction → LogisticRegression	.2994	.2549
Seeds	0, 1, 2	The same frozen path is evaluated under the registered group-wise five-fold protocol for each seed.	–	–

Table 13: Selected MIND program used in the main comparison. The program and evaluation protocol are fixed across seeds 0, 1, and 2; BA and macro-F1 are averaged over the three seeds.

The selected MIND program is the same path evaluated in Table 1 and in the executable-trace table: train-fold standardization, asymmetric EEG and fNIRS encoders, an fNIRS-conditioned EEG gate, an element-wise interaction, additive projection, and a LogisticRegression probe. Evaluation repeats this frozen program under the same protocol for each of the three seeds.

B.3.2 Budget–performance envelopes across tasks and model backbones

The MIND result above evaluates a fixed program over seeds 0, 1, and 2, with BA .2994. We quantify the complementary budget question on Figshare and HEFMI with DeepSeek Flash (`deepseek-flash`) and DeepSeek V4 Pro (`deepseek-v4-pro`), abbreviated Flash and Pro, respectively. We use three independent planning repetitions per dataset–model pair. Budgets one, two, and three are nested prefixes of the same three-attempt trajectory, not independent reruns. We report both the best BA available within each prefix (the budget envelope) and the latest-attempt BA. All 12 trajectories and all 36 attempts produce finite BA.

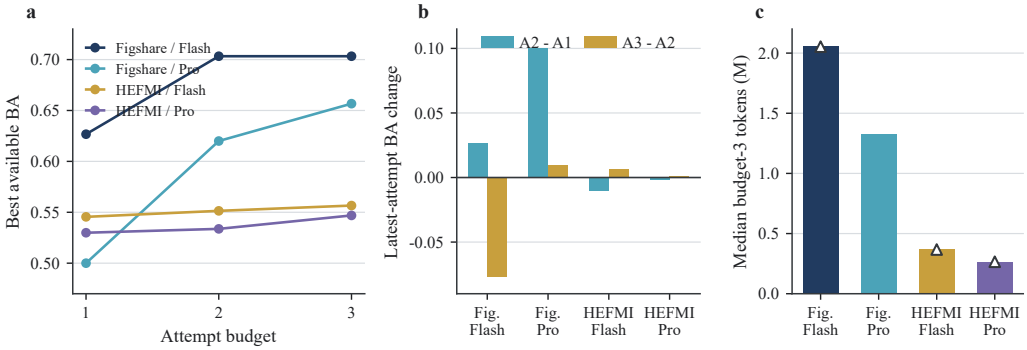

Figure 15: **Global-attempt budget curves.** Solid curves show the best BA available within each nested trajectory prefix; the accompanying panels expose the nonmonotone latest-attempt behavior and cumulative API cost. Each point aggregates three independent planning repetitions. Open triangles mark token totals that are lower bounds because some provider usage records are absent.

The best-available envelope is nondecreasing in all four settings, with much larger gains on Figshare than HEFMI. The latest-attempt sequence itself is nonmonotone: the mean changes from Attempts 1–2 and 2–3 are +.0267/-.0767 for Figshare–Flash, +.1000/+.0100 for Figshare–Pro, -.0106/+.0063 for HEFMI–Flash, and -.0021/+.0012 for HEFMI–Pro. Each transition has only three paired repetitions and all associated 95% intervals include zero. Additional attempts improve the best-available search envelope by creating more opportunities for search and repair, with gains that vary substantially across tasks and backbones.

Ten-attempt trajectories across the main tasks We additionally run the post-hoc evaluator on every research attempt from the ten-task study. Figure 16 reports the raw BA and macro-F1 sequences. These diagnostic scores are computed only for analysis; they are not returned to Planning, Verification, Execution, or Review and do not determine which program is frozen for the main table.

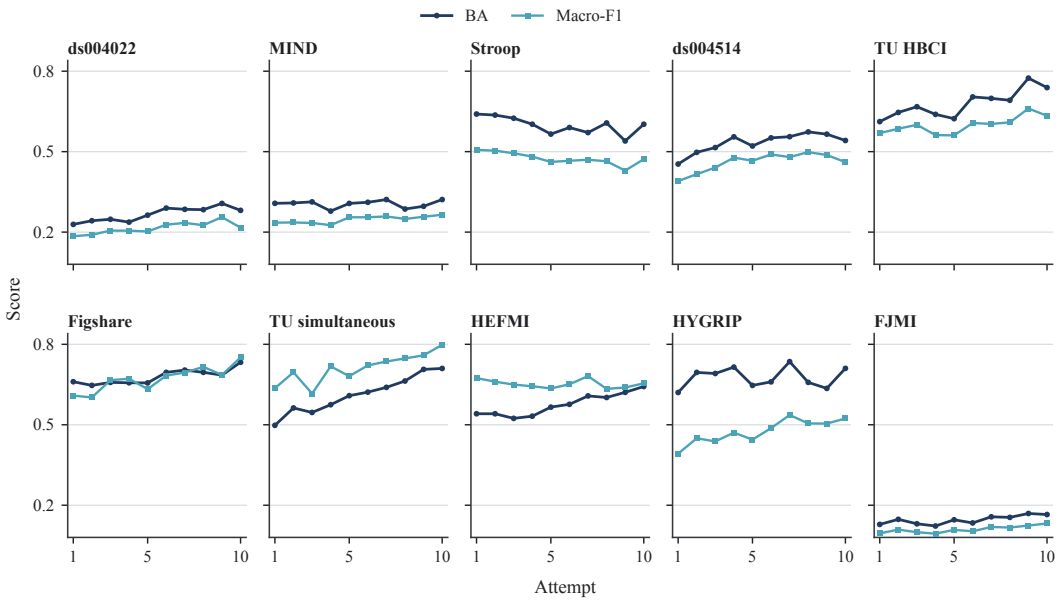

Figure 16: **Post-hoc performance trajectories over ten attempts.** Each panel shows balanced accuracy (BA) and macro-F1 for the ten research attempts of one task. Attempts are ordered research iterations rather than independent seeds, and the scores are withheld from the agent workflow. The main table reports the frozen program evaluated under the stated three-seed protocol.

The equal-weight mean across tasks rises from .4690 to .5447 BA and from .4292 to .4907 macro-F1 between Attempts 1 and 10. Attempt 10 exceeds Attempt 1 on BA for nine of ten tasks and on macro-F1 for eight of ten tasks. The trajectory remains nonmonotone: Attempt 10 is a within-sequence maximum on only four tasks for each metric. These values are descriptive because consecutive attempts belong to the same adaptive research sequence rather than independent replicates.

B.3.3 Strict replayability and runtime contract checks

Executable traces of the selected programs Tables 14 and 15 give the complete ordered computation for the selected program on every task. Composite registry entries are expanded into their constituent calls; operations inside a single registered or generated tool are shown in parentheses. Each trace starts from the dataset-adapter output specified in Appendix A, and all supervised transforms and heads are fitted only on the declared training fold. The ds004514 program uses within-subject five-fold evaluation; its held-out unit is a trial fold within a participant.

Task	Ordered executable tool trace	Main-table BA
ds004022	`joint_ds004022_gated_residual_upstream` (EEG linear projection → fNIRS linear projection → concatenation-conditioned sigmoid gate → EEG plus gated-fNIRS residual) → `joint_replay_logistic_probe`	.3024
MIND	`joint_mind_asymmetric_interaction_upstream` (train-fold modality standardization → asymmetric EEG and fNIRS encoders → fNIRS-conditioned EEG gate → element-wise EEG–fNIRS interaction → additive projection sum) → `joint_replay_logistic_probe`	.2994
Stroop	`block_conflict_eeg_erp_spectral_features` → `block_conflict_fnirs_response_quality_features` → `block_conflict_quality_gated_interaction` → fold-local class-balanced logistic regression	.6316
ds004514	`eeg_stage_band_features` → `fnirs_stage_features` → `stage_contrast_tokens` → fold-local `extra_trees`	.5750
TU HBCI	Event windows → workload OD features → fold-safe assembly → linear probe	.7411

Table 14: Selected executable programs for the first five tasks. Composite entry points are followed by the ordered calls and fixed operations that they execute.

Task	Ordered executable tool trace	Main-table BA
Figshare	`motor_lateralization_pair` (left–right EEG motor-ROI contrast → left–right fNIRS motor-ROI contrast → fold-local feature concatenation) → fold-local `linear_probe`	.7160
TU simultaneous	`official_prepared_topomap` → spatial-prior token construction (coordinate-grid interpolation → EEG temporal and spatial tokens → fNIRS early/middle/late and contrast tokens → cross-modal products and differences) → training-fold Agent-prior prediction (train-fold standardization → 128–96 MLP → two-class logits)	.6873
HEFMI	released aligned EPO trials → pre-event baseline subtraction for HbO, HbR, and tHb → delayed means over 2–5, 4–8, 6–12, and 8–16 s → train-fold `SelectKBest(f_classif, k=64)` → `StandardScaler` → class-balanced `LogisticRegression`	.6656
HYGRIP	released-stream event materialization → baseline-relative mu/beta ERD in 0–4, 4–12, and 12–21 s phases → EEG right–left trajectories → baseline-relative HbO/HbR means in 2–6, 6–12, and 12–21 s windows → hemodynamic right–left trajectories → lag-0/lag-1 normalized shape alignment and HbO–HbR direction → train-fold `StandardScaler` → inner-GroupKFold-selected class-balanced `LogisticRegression`	.6768
FJMI	`paired_continuous_event_preprocessor` → `event_log_welch_integrated_bandpower` → `event_hemodynamic_mean_normalized_slope` → `subject_unlabeled_feature_standardization(method=robust_iqr)` → `trial_feature_concatenation` → `fold_shrinkage_lda_classifier`	.1653

Table 15: Selected executable programs for the remaining five tasks. The subject-robust FJMI workflow is expanded into its constituent registered tools rather than displayed only as a bundle identifier.

On Figshare, `motor_lateralization_pair` constructs left–right EEG and fNIRS motor-ROI contrasts and concatenates the trial-level feature blocks. The fused representation then undergoes training-fold feature preparation and linear classification. TU simultaneous instead uses cross-modal products within its spatial-token representation, illustrating a different interaction between the modalities.

The resulting programs differ in fusion direction, temporal representation, interaction operator, and prediction head. This task-conditioned structure is preserved during replay: no retrieval, replanning, or tool replacement is performed once a program has been selected.

Replay of frozen programs For each task, the accepted frozen program is rerun from its declared input without agent calls, replanning, or graph replacement. A replay is considered numerically consistent when its BA differs from the corresponding original run by at most .02. For Figshare, this comparison uses the accepted program's registered five-fold, three-seed BA of .7160, matching the main table. The ds004514 replay uses its within-subject five-fold protocol.

Task	Original BA	Replay BA	$\lvert\Delta\mathrm{BA}\rvert$	Result
Figshare	.716000	.716000	.000000	Reproduced
Stroop	.631600	.631579	.000021	Reproduced
HYGRIP	.676829	.676829	.000000	Reproduced
FJMI	.165300	.165278	.000022	Reproduced
ds004022	.302381	.294048	.008333	Reproduced
MIND	.299444	.299167	.000278	Reproduced
ds004514	.575417	.575417	.000000	Reproduced
HEFMI	.665616	.665616	.000000	Reproduced
TU HBCI	.741108	.739171	.001937	Reproduced
TU simultaneous	.687269	.687269	.000000	Reproduced

Table 16: Replay of the ten selected tool programs. The ds004514 row uses the within-subject five-fold protocol. The mean absolute BA difference is .001059; all differences satisfy the prespecified .02 tolerance.

All ten programs complete and reproduce their original values within tolerance; all differ by less than .01 and the maximum difference is .008333 on ds004022. The accepted plans therefore compile into executable programs with reproducible predictions.

Pre-execution contract checks Replayability depends on rejecting calls whose required signals or sampling metadata are absent. We paired contract checks enabled and disabled on Figshare

and HEFMI under clean input and four invalid conditions per task: missing EEG, missing fNIRS, missing EEG rate, and missing fNIRS rate.

Runtime outcome	Count or value
Invalid cases blocked before execution	8/8
The same invalid cases producing tool errors without the check	8/8
Clean cases completing with the check	2/2
Invalid samples protected from erroneous execution	14,760/14,760
Mean contract-check time	.0205 ms
Median contract-check time	.0115 ms

Table 17: Paired evaluation of runtime precondition checks.

The check rejects every invalid call before tool execution and leaves both clean graphs executable. It therefore prevents a program from running with missing signals or sampling metadata.

B.3.4 Robustness under modality perturbations and quality gating

The replay tests establish behavior under the declared input. We next freeze the trained graph and model and perturb only held-out inputs, without retraining or graph replacement. The perturbations include channel removal, modality noise, and temporal displacement supported by each task's input representation. For ds004514, the diagnostic rows use the registered author-cue feature-boundary path under the same within-subject five-fold protocol; its clean BA corresponds to this robustness-test program.

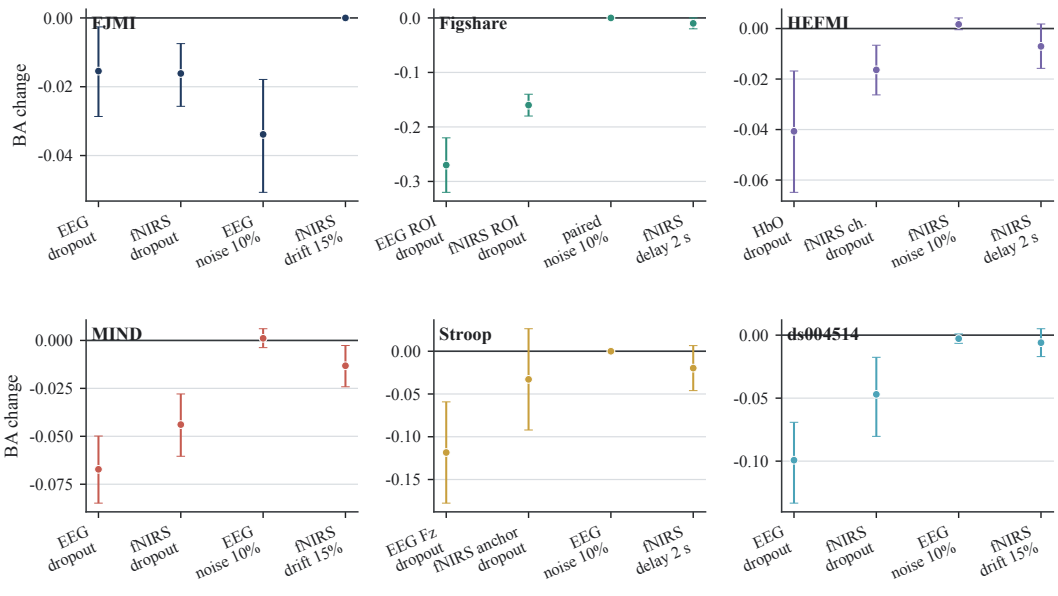

Figure 17: Change in held-out BA after test-side perturbations of six frozen paths. Points show the change from the corresponding clean path; bars are 95% subject-level intervals. Each panel uses a task-specific vertical range to reveal within-task differences, so vertical positions should not be compared directly across panels. Perturbation names follow the signal representation available for each task.

Across six tasks there are six clean and 24 perturbed evaluations. The mean perturbation effect is negative for every task: -.0164 on FJMI, -.1100 on Figshare, -.0156 on HEFMI, -.0308 on MIND, -.0428 on Stroop, and -.0388 on ds004514. Thirteen of the 24 perturbation intervals exclude zero. The largest drop is -.2700 on Figshare, while several noise or drift conditions are close to zero.

Figure 17 reveals task- and perturbation-specific sensitivity, including both substantial degradation and near-invariant conditions.

Conditional evidence for quality gating Stroop uses a quality-gated interaction, permitting a matched comparison with the same representation and an ungated Simple concat path. Both models are frozen while the test input is perturbed.

Test condition	Quality-gated BA	Ungated BA	Gated − ungated
Clean	.6316	.5855	+.0461
EEG Fz dropout	.5132	.5132	.0000
fNIRS anchor dropout	.5987	.5658	+.0329
10% EEG noise	.6316	.5921	+.0395
2-s fNIRS delay	.6118	.5658	+.0461

Table 18: Stroop quality gating under held-out modality perturbations. The clean gated-minus-ungated difference has a subject-level 95% CI of [0,.0987].

The gated path retains a positive advantage for fNIRS anchor dropout, EEG noise, and a 2-s delay, while removal of Fz eliminates the difference. We additionally examine whether the gate weight moves in the intended direction on ds004022 using a frozen quality-gated late mixture and a constant-weight counterpart.

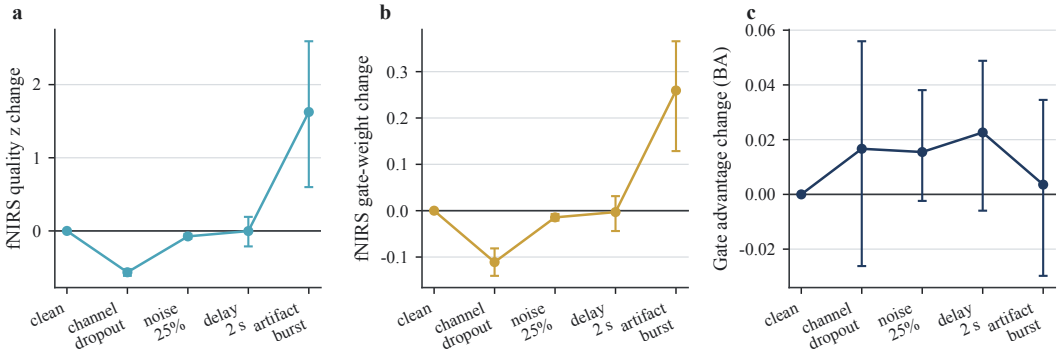

Figure 18: Response of the ds004022 quality gate. The panels show the change in the fNIRS quality statistic, the corresponding gate weight, and the change in gated-minus-constant BA relative to clean input. Bars are subject-level 95% intervals over seven subjects.

With 25% fNIRS channel dropout, the quality statistic falls by .5653 and the mean fNIRS weight falls by .1110; both intervals exclude zero. With 25% noise, the corresponding changes are -.0748 and -.0144. The gated-minus-constant BA advantage improves by .0167 and .0155 under these two perturbations, although the performance intervals cross zero. A 2-s delay changes the relative advantage by +.0226. An artifact burst exposes a failure mode of this particular quality proxy: it increases the quality statistic by 1.6259 and the fNIRS weight by .2594 rather than downweighting the corrupted modality. Together with Table 18, this supports quality gating as a conditional mechanism whose behavior depends on whether its quality statistic actually recognizes the perturbation.

Adaptive replanning under modality degradation The frozen-path experiments above measure sensitivity without changing the program. We now ask whether a label-free quality profile can trigger an appropriate task-conditioned response. Figshare, Stroop, and HEFMI are tested under EEG channel damage, partial fNIRS availability with pulse artifacts, and fNIRS delay, using DeepSeek V4 Pro (`deepseek-v4-pro`) for replanning. Perturbations affect held-out inputs only; training folds and test labels are unchanged. For every condition, the frozen clean path, a fixed concatenation control, and profile-aware replanning share subjects and folds. The full matrix contains 54 of 54 planned task–condition–strategy cells and 69,156 predictions.

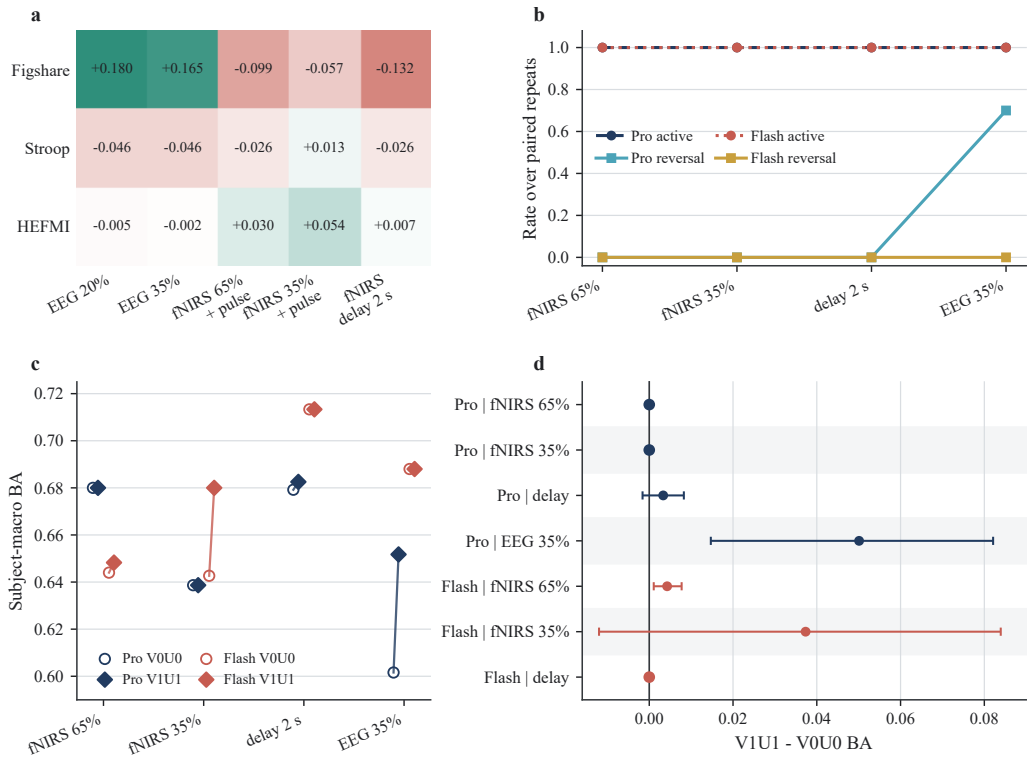

Figure 19: **Adaptive robustness and utility-constrained verification.** Panel A reports profile-aware replanning relative to the frozen path across three tasks and five perturbations. Panels B–D show a paired 2×2 intervention on Figshare in which Verification (V) and a training-only modality-utility constraint (U) are independently disabled or enabled.

Task	Held-out perturbation	Replan − frozen	95% CI	p_{Holm}	Interpretation
Figshare	20% EEG damage	+.180	[+.124,+.232]	.0066	Recovery
Figshare	35% EEG damage	+.165	[+.103,+.223]	.0143	Recovery
Figshare	2-s fNIRS delay	-.132	[-.192,-.072]	.0259	Mismatched response
HEFMI	35% fNIRS available + pulse	+.054	[+.032,+.079]	.0011	Recovery
HEFMI	65% fNIRS available + pulse	+.030	[+.013,+.046]	.0319	Recovery

Table 19: Prespecified perturbation contrasts that remain significant after Holm correction. The complete matrix, including null effects on Stroop and other conditions, is visualized in Figure 19.

Replanning substantially recovers the two Figshare EEG-damage conditions and two HEFMI fNIRS-corruption conditions, but it degrades Figshare under delayed fNIRS. The negative result is mechanistically informative: the one-shot plan responds to a perceived modality failure by discarding a stream that remains predictive for this task. We therefore test a 2×2 intervention on the same Figshare subjects: the *verification-off / utility-off* condition disables both Verification and a training-only modality-utility constraint, while the *verification-on / utility-on* condition enables both. We evaluate this intervention separately with DeepSeek V4 Pro and DeepSeek Flash (`deepseek-flash`), abbreviated Pro and Flash, respectively. Each model-specific analysis retains 15 subjects, five folds, 10 paired planning repetitions per condition, and 120,000 predictions; subject observations are never pooled across model backbones.

For the Pro backbone under 35% EEG damage, the unconstrained plan hard-drops a modality in 70% of paired repetitions. Enabling both safeguards reverses all seven such decisions and raises subject-macro BA from .6016 to .6517 ($\Delta = .0501$; 90% CI [.0147,.0821]); the prespecified non-inferiority test against a -.020 margin gives one-sided $p = .0036$. The Flash backbone does not make the same hard-drop decision in this negative-control condition. Its corresponding safeguarded contrasts for

65% and 35% fNIRS availability are +.0043 and +.0373 BA, but neither remains significant after Holm correction. Task- and damage-matched replanning therefore recovers substantial degradation in several conditions, while the utility constraint repairs the erroneous modality drop made by the Pro backbone. The remaining null and negative contrasts identify when the one-shot response is unnecessary or mismatched to the perturbation.

C Prompt Protocol

This appendix summarizes the role-specific prompts described in Sections 3.2–3.4. Each prompt specifies a role, a task, input context, an output format, and constraints. Task- and fold-specific values are supplied for the corresponding evaluation. Reviewer acceptance freezes the executable program for held-out evaluation.

Prompt cards

The prompts define the information available to each role and the structured decisions it can return. Retrieval, observation, and repair prompts support revisions within this shared protocol.

Prompt: Planner

You are a scientific planning agent. Plan one independent EEG–fNIRS autoresearch task from a target-free dataset brief and retrieved evidence.
Guidelines: 1. Generate one or more distinct, falsifiable hypotheses; do not pad the count. 2. Give every hypothesis its own complete early–middle–late plan, training protocol, decision points, risks, expected signatures, and falsifiers. 3. Request a bounded inspection only when the brief cannot resolve a scientific uncertainty. 4. Describe required capabilities rather than inventing registered tool IDs. 5. Mark a stage `skip` only when it is scientifically unnecessary; an all-skip hypothesis is invalid.
Input context: `task`; `dataset_research_brief`; retrieved `evidence`; inspections; planning memory; previous plan and feedback; retrieval, interaction, hypothesis-count, run-guidance, and ablation policies.
Output format: Return one JSON object with `next_action` equal to `inspect_data` or `submit_plan`. A submitted plan contains hypotheses, a recommended ID, rationale, evidence IDs, planning memory, and complete stage contracts: mode, inputs, operations, expected output, handoff, and failure checks.
IMPORTANT: Do not use raw signals, labels, predictions, fold assignments, split manifests, or observed metrics. Every hypothesis cites at least one ID copied from the supplied evidence packet.

Prompt: Planner Retrieval

You are a targeted literature-search designer. Design fresh searches that can change a hypothesis, repair a stage interface, identify an implementation pattern, or explain an observed failure.
Input context: `task`; dataset brief; previous plan; review feedback; verifier feedback.
Output format: Return one to eight `queries`, each with `query_id`, query, objective, perspective, and fusion stage, plus a `coverage_rationale`. At least one query must be non-profile and the set must address the observed failure.

Prompt: Verifier

You are an independent scientific and interface verifier. Audit every proposed hypothesis and select one executable candidate.
Guidelines: 1. Use each hypothesis's matching independent evidence packet and review every candidate before choosing one verified hypothesis. 2. Repair incomplete stage interfaces, metadata requirements, and fitting scopes. 3. Search for a missing capability when current and retained tool cards do not establish feasibility. 4. After two equivalent searches, compose registered tools, justify a generated tool, adopt an honest fallback, or reject the hypothesis. 5. Submit once feasibility or a blocking contradiction is established.
Input context: `task`; verification data context; proposed plan; independent evidence index and hypothesis-specific packets; tool-search results and retained tool cards; inspections; verification memory; policies.
Output format: For `next_action=submit_verification`, return one `hypothesis_review` for every proposed hypothesis, then `selected_hypothesis_id`, `approved`, findings, amendments, expectations, stage feasibility, contradictions, rationale, and planner feedback. For search or inspection, return only the request list and compact memory.
IMPORTANT: Never copy Planner or other-hypothesis evidence IDs into a review. Check numeric adjacency, representation, metadata, and fit scope. Do not inspect labels, predictions, or performance metrics.

Prompt: Verifier Candidate Choice

You are the independent candidate reviewer. Compare compact audit cards without replaying their full evidence packets.
Input context: `task`; verification data context; candidate verification cards.
Output format: Return one JSON object with `selected_hypothesis_id`, `approved`, candidate rationale, and planner feedback. Approve only a candidate whose independent audit has no blocking contradiction or infeasible interface.

Prompt: Executor

You are a local execution agent inside one labeled workspace. Convert the verified plan into an executable typed action batch.
Guidelines: 1. Start with a local plan and compare registered bundles with atomic compositions by prerequisites, coverage, outputs, and parameters. 2. Reuse applicable retrieved cards; do not repeat an equivalent search. 3. Search for the specific missing capability before generating code. 4. Execute an action, read its factual observation, then advance, revise, search, or hand off. 5. A terminal late stage hands off only after publishing finite logits or probabilities in runtime row order.
Input context: `task`; data profile; verified plan; current stage and react history; artifact catalog; tool cards and searches; stage evidence; workspace and handoff state; execution policies.
Output format: Return `turn_decision`; a local plan; optional tool or literature searches; a typed `action_batch`; generated-tool declarations; source inspections; and, for handoff, completed/unresolved objectives and exact artifact references.
IMPORTANT: Every action consumes explicit artifacts and publishes typed outputs. Generated tools require a prior targeted search and complete input/output contracts, fitting scope, source, dependencies, and entry point. Never infer fold membership or evaluator-owned metrics.

Prompt: Executor Observation

You are the same executor after an action. Read the factual observation, compare it with the verified expectation, and decide whether the workspace advances or revises.
Input context: the action; verified plan; factual execution observation; current artifacts, memory, and handoff state.
Output format: Return `decision`; `expectation_met`; observed results and evidence; unmet expectations; next-stage compatibility; rationale; and, when revising, a concrete revision instruction or replacement action batch. **IMPORTANT:** `expectation_met` is true exactly when `decision=advance`; failures, missing outputs, and unavailable dependencies require revision. Do not invent artifacts or facts.

Prompt: Executor Repair

You are repairing an invalid executor response, not executing a new action. Preserve the original decision and observed facts while replacing the invalid JSON.
Input context: current stage; validator error or deterministic contradiction; invalid action or judgment; current generated sources; factual observation; allowed evidence IDs; replacement contract.
Output format: Return one complete replacement action response or judgment under the original contract. Repeat complete generated-tool declarations when the rejected action used them. A judgment repair cannot alter the observation or execute another action.

Prompt: Reviewer

You are the attempt-level reviewer. Review the complete plan and execution stages, then choose `accept`, `replan`, or `reexecute`.
Guidelines: 1. Enforce hard gates. 2. Use `reexecute` only for a full early–middle–late rerun, not a stage rollback. 3. Use `replan` when the scientific assumption or plan must change. 4. Separate observed facts from scientific interpretation and make next-attempt feedback concrete.
Input context: task; verified plan; graph; stage reports; observed execution outputs; failed gates; prior feedback; review evidence; literature query; review policy. Held-out labels, predictions, and evaluation metrics are not provided.
Output format: Return action, rationale, feedback for the next attempt, failure summary, literature query, and a research report. Separate observed facts from interpretation; hard-gate failures cannot be overridden. An `accept` action freezes the current executable graph for private evaluation.

Acceptance contract. The Reviewer may return `accept` only when the executable program contains a complete tool graph, uses no held-out labels or other leaked information, produces valid evaluation metrics, and achieves research-validation balanced accuracy significantly above the task's chance level. For a K-class task, the chance level is $1/K$; significance is assessed with a subject-level, one-sided sign-flip test at $\alpha = 0.05$. This chance-relative rule replaces a fixed BA threshold across tasks. Held-out labels, predictions, and evaluation metrics remain unavailable to the Reviewer. The Reviewer audit in Appendix B.3.1 evaluates its discrimination of consistent and controlled-invalid contexts and its agreement with human experts.

D Extended Related Work

Unimodal brain-signal decoding. Single-modality decoders learn stronger representations from either EEG or fNIRS. For EEG, EEG-Conformer combines convolutional local feature extraction with Transformer self-attention, while BIOT learns transferable biosignal representations across datasets (Song et al., 2023; Yang et al., 2023). For fNIRS, fNIRS-Transformer models channel–time signals with a Transformer, fNIRSNet represents delayed hemodynamic responses, and BenchNIRS CNN provides a convolutional decoder (Wang et al., 2022b; 2023; Benerradi et al., 2023). EEG foundation models further aim to transfer representations across datasets and tasks: LaBraM learns

general representations from large BCI corpora, CBraMod captures spatial and temporal dependencies, LUNA is topology-agnostic, CodeBrain combines a decoupled time–frequency tokenizer with a multi-scale encoder, and STEEGFormer targets representation learning across BCI tasks (Jiang et al., 2024; Wang et al., 2025; D"oner et al., 2025; Ma et al., 2026; Yang et al., 2026). Despite stronger within-modality representations, these models decode EEG or fNIRS alone and therefore cannot use complementary information from a co-recorded, physiologically related but heterogeneous modality. EFiMAS instead treats EEG and fNIRS as two distinct but potentially related inputs and constructs a task-conditioned program that determines how their representations should interact.

EEG–fNIRS Fusion for Decoding. EEG–fNIRS fusion methods have evolved from feature concatenation to task-specific designs that fuse the two modalities at different stages. DeepSyncNet fuses EEG and fNIRS in an early feature space for mental arithmetic and motor imagery (Li, 2026). TSMMF combines early cross-modal interaction with late decision fusion for emotion recognition (Si et al., 2025). Other methods use neuroscience knowledge to guide the fusion design. BiSTF-Net uses spatial guidance and adaptive temporal alignment to account for the slower fNIRS response (Wang et al., 2026). FGANet uses fNIRS spatial information to guide EEG attention (Kwak et al., 2022), while STA-Net uses fNIRS to identify task-sensitive regions and weight EEG channels (Liu et al., 2025). Together, these studies show that the useful fusion direction and stage depend on the task and data. No single fixed fusion strategy can consistently perform best across different tasks. In contrast, EFiMAS uses task information and retrieved evidence to propose task-specific fusion hypotheses through language-model reasoning. It then constructs a separate fusion program for each task rather than reusing a fixed design.

Agentic systems for brain signal analysis. Agentic systems increasingly support brain-signal analysis through tool use and retrieval-augmented generation (RAG). EEG Agent plans and schedules tool use for automated EEG analysis (Zhao et al., 2026). BrainAgent uses a hierarchical multi-agent architecture to execute tools and connect experimental outputs with neuroscience knowledge (Zhou et al., 2026). EasyBCI uses a text-only data fingerprint and literature-grounded operator sequences to plan and execute neural-data preprocessing, while BrainPilot combines curated neuroscience knowledge and reusable methodology skills in an automated research workflow (Zhu et al., 2026; Li et al., 2026). These systems support knowledge-grounded planning, tool execution, and scientific analysis. EFiMAS focuses on task-specific EEG–fNIRS fusion: it proposes a fusion relation, independently verifies its rationale and feasibility, and uses execution observations and validation results to revise the fusion program. A knowledge base, a tool library, role-specific retrieval, and feedback loops support this workflow.

E Discussion and Limitations

The experiments show three recurring patterns. First, the selected paths vary in representation, cross-modal operation, timing, and prediction head. Second, retrieval and observed-output checks change the executable path before private evaluation. Third, each branch of *Dual-Evidence Retrieval*, *Independent Verification*, and *Stage Check and Program Revision* contributes to the three-task ablation results. The matched replay adds a fourth pattern: task-specific programs lead on nine tasks, while the compact fixed path fits ds004514 better under its subject-dependent/within-subject five-fold formal replay.

The five ablations are single treatments on three tasks. Each treatment lowers BA relative to the complete system in this matrix. Interactions among the components remain open for a larger factorial study. The unimodal controls in the supplement place the multimodal results alongside their EEG-only and fNIRS-only counterparts.

F Case Analysis of a Selected EFiMAS Path

This appendix follows the terms in Section 3 through one representative EFiMAS path. We use TU simultaneous because its released task contains paired whole-head EEG and fNIRS signals, and the selected path uses all three decisions in the fusion program: an event representation, a spatial-prior

fusion operation, and an Agent-prior prediction head. The example connects the executable path to the RAG decisions that produced it.

F.1 SELECTED PATH

The selected executable contract is

event-aligned input / coordinate-aware spatial-prior fusion / fold-local Agent-prior head / two-class output.

For the binary word-generation task in TU simultaneous (Shin et al., 2018), event identity pairs each EEG window with its HbO/HbR window while both modalities retain their native time axes. The raw event-analysis boundary is $[0, 12]$ s. The released preparation applies its event epoch and lag grid during feature construction, deriving three-second modality slices and lagged fNIRS views before token construction; these are derived inputs, not a replacement for the raw boundary. The fusion operation projects channel locations into a shared coordinate-aware representation and combines spatial, temporal, and cross-modal summaries into one label-free event vector. The Agent-prior head and its scaling are fitted within each training fold, and the final two-class outputs are emitted in held-out trial order. The released implementation specifies the exact tensor shapes and feature definitions associated with this contract.

F.2 RAG DECISIONS FOR THE SAME PATH

The RAG evidence is linked to the individual operations in the selected executable path. Its hierarchy is *task profile* → *dual-index planning* → *hypothesis-specific verification* → *method/tool execution* → *outcome-linked review*, as described in Sections 3.2–3.4. Table 20 shows the concrete evidence packet and the constraint it contributed to each path decision.

Path decision	RAG layer / index	Representative retrieved evidence	Executable constraint
Input representation	Planner dual search: neuroscience + method	Simultaneous acquisition and neurovascular timing (Shin et al., 2018; Hillman, 2014; Scholkmann et al., 2014; Kilner et al., 2005; Tachtsidis & Scholkmann, 2016); early hybrid feature construction (Li et al., 2017)	Pair by event identity; retain 200/10 Hz native rates and keep labels/folds out of the input artifact.
Spatial-prior fusion	Planner dual search, then Verifier packet: neuroscience + method	Task-specific EEG–NIRS temporal relation (Zama & Shimada, 2015; Muñoz et al., 2023); early/cross-modal fusion (Li, 2026; Si et al., 2025; Wang et al., 2022a); spatial guidance and fNIRS-to-EEG transfer (Wang et al., 2026; Kwak et al., 2022; Liu et al., 2025)	Preserve the coordinate-aware map and native time axes; emit one fixed-width, label-free token per event.
Agent-prior head	Executor method index, then Reviewer outcome packet	Agentic brain-signal analysis and hierarchical tool orchestration (Zhao et al., 2026; Zhou et al., 2026; Zhu et al., 2026; Li et al., 2026); outcome-linked scientific-agent evaluation (Ahamed et al., 2026; Chen et al., 2025b; Kon et al., 2026)	Fit scaling and the prediction head inside each training fold; accept only finite two-class outputs in held-out order.

Table 20: Evidence-to-decision trace for the selected path. Sources are shown at the RAG layer where they constrain each operation.

The same query is reused across the two planning indexes, while Verification adds the proposed hypothesis and searches for support, scope, and counterevidence. Execution narrows retrieval to

methods and registered tools; Review reopens the evidence packet when an observed output fails its expected contract. Through this role-specific sequence, retrieved papers become typed constraints on the selected representation, fusion operation, head, and output checks.

F.3 CROSS-TASK SUMMARY

Table 21 gives four representative EFiMAS paths. Each row places the selected path beside the RAG decision that supports it. The TU simultaneous example above expands this same path-to-evidence structure.

Dataset	Selected path	Key RAG decision
Figshare	**Input.** Sensorimotor EEG and fNIRS optical-density features. **Fusion.** Fold-local motor-lateralization feature concatenation. **Head.** Fold-fitted linear probe.	**Input.** Motor-task evidence made paired EEG/fNIRS features relevant. **Fusion.** Retrieval favored lateralized motor descriptors and aligned rows. **Head.** Verification retained a fold-local linear decision.
TU HBCI	**Input.** Event windows, EEG workload spectra, and released fNIRS optical-density features. **Fusion.** Spectro-hemodynamic feature fusion. **Head.** Fold-fitted logistic head.	**Input.** Workload retrieval linked spectral EEG and hemodynamic responses. **Fusion.** Method retrieval supported concatenating complementary feature blocks. **Head.** Fold checks kept fitting inside the released group split.
TU simultaneous	**Input.** Raw event-analysis boundary $[0, 12]$ s; the selected derived representation then yields three-second modality slices (28×600 EEG and 72×30 HbO/HbR) plus lagged fNIRS views. **Fusion.** Coordinate-aware spatial-prior token, width 5,362. **Head.** Agent-prior MLP with two logits.	**Input.** Event identity resolved the 200/10-Hz rate mismatch. **Fusion.** Distributed cross-modal evidence supported the spatial-prior token. **Head.** Verification required fold-local scaling and a two-class output.
HEFMI	**Input.** Baseline-relative delayed HbO/HbR/tHb windows. **Fusion.** Fold-local delay and feature selection. **Head.** fNIRS linear head.	**Input.** Delayed hemodynamic evidence was sufficient for this task. **Fusion.** Retrieval selected informative delayed features instead of forcing a multimodal path. **Head.** The selected fNIRS head remained fold-fitted.

Table 21: Selected programs and the retrieval decisions that support their input representation, fusion operation, and prediction head.

F.4 TASK-CONDITIONED HYPOTHESIS FORMATION AND REVISION

The selected programs in Table 21 begin as scientific hypotheses rather than as fixed tool sequences. The Planner first states a cross-modal relation, its expected signatures and falsifiers, and an early–middle–late realization. Verification then reconciles that proposal with the released representation and the available executable contracts. The structured examples below show how this process produces different hypotheses for mental arithmetic, word generation, motor execution, clinical motor imagery, and directional motor imagery.

Structured planning output: TU HBCI

Profile cues: mental arithmetic versus rest; fast EEG workload oscillations; slower prefrontal fNIRS response; different sampling rates and physical units.
Hypothesis: `hyp_ma_rest_spectral_hemodynamic_middle_fusion`. Mental arithmetic engages prefrontal and frontoparietal control networks, producing EEG theta/alpha changes and a complementary hemodynamic response; a trial-level combination should improve workload discrimination.
Initial plan: preserve the two modalities at the signal level, form EEG spectral and fNIRS response features in the middle stage, and fit a regularized classifier in the late stage.
Verification update: replace unsupported HbO/HbR conversion with released optical-density features, map the proposal to the workload-reference feature bundle, and keep fusion and prediction fitting inside each training fold.

The TU HBCI proposal is therefore driven by an *shared workload* relation, but it does not force raw signals onto a common clock. Its executable form combines modality-specific summaries only after each signal has been expressed in a representation supported by the release.

Structured planning output: TU simultaneous

Candidate hypotheses: `wg_eeg_frontal_alpha_fnirs_oxy_contrast` links EEG alpha/beta contrast to a delayed fNIRS oxy–deoxy contrast; `fnirs_frontal_hemodynamic_independent_discrimination` tests whether the fNIRS component contributes independent task information; and `cross_modal_spatial_prior_products_enhance_wg_detection` tests explicit cross-modal products over matched spatial summaries.
Selected hypothesis: `wg_eeg_frontal_alpha_fnirs_oxy_contrast`. Word generation is expected to produce task-related EEG spectral changes together with a frontal hemodynamic response, and the spatial-prior token retains their temporal, spectral, and coarse spatial contrasts.
Stage plan: construct the token in the early stage, skip an otherwise redundant middle adapter, and fit a fold-local nonlinear head.
Verification decision: approve without a scientific amendment because the early token directly satisfies the late head's input contract.

Here the candidate set separates three questions that a generic concatenation would collapse: whether both modalities carry directional evidence, whether fNIRS adds an independent component, and whether their spatially matched interaction is itself informative. The selected program operationalizes the first relation while retaining product features that can test the third.

Structured planning output: Figshare

Profile cues: synchronized right- versus left-hand motor-execution trials; EEG and two-wavelength fNIRS intensity streams; no verified fNIRS hemisphere, ROI, source–detector-distance, or wavelength metadata.
Hypothesis: `eeg_fNIRS_motor_lateralization_erd_hbr`. Sensorimotor EEG mu/beta modulation and the slower fNIRS response provide complementary evidence for movement side.
Initial plan: extract event-locked modality-specific features, fuse them after fold-local normalization and selection, and use a regularized linear head.
Verification update: retain the motor-lateralization hypothesis but replace HbO/HbR conversion and assumed fNIRS laterality with per-channel optical-density features; remove redundant probability calibration and bind the three stages to registered release-compatible operations.

This change narrows the claim without discarding its task-specific content. The resulting program still tests complementary motor EEG and fNIRS responses, but it does so using quantities that can actually be derived from the released signals. Appendix M.1 gives the corresponding DataProfile, evidence packet, and original Planner response.

Structured planning output: HEFMI

Candidate hypotheses: H1 estimates a neurovascular delay between EEG mu/beta modulation and the fNIRS response; H2 combines independent modality predictions by late fusion; H3 learns an early cross-modal representation.
Initial recommendation: H1, because motor imagery should produce a rapid electrophysiological response followed by a slower hemodynamic response.
Verification finding: the proposal marked early and middle stages as skipped while assigning operations to them, and its active late stage ended at delay estimation without a classifier or decision-fusion endpoint.
Revision direction: make delayed hemodynamic features an explicit representation, perform feature selection within the training fold, and attach a valid prediction head rather than treating delay estimation as a terminal output.

HEFMI illustrates that physiological plausibility alone is not sufficient. The temporal-lag idea becomes testable only after its expected output is linked to an executable representation and a prediction endpoint. This produces a delayed hemodynamic route rather than the spatial-prior or motor-lateralization routes selected for the preceding tasks.

Selected program and three-seed evaluation on MIND. The MIND program used in the main comparison assigns EEG the primary directional representation and uses fNIRS to control and interact with that representation: both modalities are standardized within the training fold, encoded asymmetrically, combined with an fNIRS-conditioned EEG gate and an element-wise interaction, and passed to a LogisticRegression probe. The main-table estimate is reported over seeds 0, 1, and 2; BA is .2994.

Selected MIND program and evaluation

Evaluation result: The selected program has BA `0.2994444444444444`.
Executable path: EEG/fNIRS asymmetric encoding, fNIRS-conditioned EEG gate, multiplicative EEG–fNIRS interaction, additive projection, and LogisticRegression.
Evaluation protocol: the frozen path is aggregated over seeds 0, 1, and 2 under the registered four-class group-wise five-fold protocol.

Field	Setting	Executable path	Value
Program	Selected MIND path	`joint_mind_asymmetric_interaction_upstream` → `joint_replay_logistic_probe`	–
Seeds	0, 1, 2	The same frozen path under the registered group-wise five-fold protocol	–
Main-table BA / Macro-F1	.2994 / .2549	Mean over three seeds	.2994 / .2549

Table 22: Selected MIND program and evaluation settings. The same program is evaluated over three seeds under the group-wise five-fold protocol.

The MIND result averages three executions of the same frozen program under the same evaluation protocol. Its asymmetric encoding, fNIRS-conditioned gate, and interaction operator are fixed across seeds, allowing the reported result to characterize this selected fusion design.

Selected task-conditioned programs. Table 23 connects the preceding hypothesis records to the programs used in the main evaluation. These endpoints differ in representation, interaction stage, modality direction, and prediction head.

Task	Selected scientific relation	Executable program	BA
TU HBCI	Fast workload spectra and slower optical-density responses provide complementary mental-arithmetic evidence.	Event windows → workload OD features → fold-safe assembly → linear probe	.7411
Figshare	Motor-related EEG spectral changes and fNIRS optical-density summaries jointly distinguish movement side.	Motor-lateralization pair → fold-local feature concatenation → linear probe	.7160
TU simultaneous	Word-generation EEG and fNIRS contrasts are organized by a coordinate-aware spatial prior.	Spatial-prior token → Agent-prior MLP head	.6873
HEFMI	Delayed HbO/HbR/tHb dynamics provide the stable task representation.	Delayed hemodynamic representation → fold-local selection → fNIRS linear head	.6656
MIND	An asymmetric EEG–fNIRS representation uses fNIRS as a gate on an EEG-anchored motor representation.	Asymmetric encoding → fNIRS gate → multiplicative interaction → logistic head	.2994

Table 23: Hypotheses and selected programs for five representative tasks. BA follows the task-specific evaluation protocol used in the main comparison.

Together, the examples show adaptation at the level of the scientific proposal, not only at the level of fitted weights. Planning assigns different roles to the modalities—shared workload evidence, spatially organized interaction, motor complementarity, delayed hemodynamics, or asymmetric reliability—and Verification turns those roles into data-compatible stage contracts. Review can then preserve the hypothesis while repairing an interface, reject a physiologically plausible but incomplete program, or change the interaction itself in the next attempt. The resulting diversity of executable programs is the mechanism by which EFiMAS conditions fusion on the task.

G Qualitative Component-Ablation Trace

Table 24 follows the first decision divergence between dual evidence and a merged evidence packet on the TU simultaneous word-generation task. Both conditions start from the same target-free task profile and preserve the same subject–session protocol and ablation budget. The table connects the resulting retrieval packet to the hypothesis, verified plan, executable graph, and output, complementing the aggregate ablation results in Table 2.

Stage	Full agent	Merged evidence packet	Where the path changes
Task observation	Word generation versus baseline; EEG at 200 Hz and fNIRS at 10 Hz with independent native clocks.	The same task profile and clock constraint.	No divergence yet.
Retrieval packet	Separate method evidence (feature concatenation and held-out evaluation) and neuroscience evidence (word-generation EEG–fNIRS physiology and delayed hemodynamics).	A single mixed packet in which the selected support is dominated by generic fusion and evaluation evidence.	Mechanism and implementation evidence now compete in one ranking, so their distinct roles are not preserved.
Falsifiable hypothesis	Event-identity-aligned early fusion of fold-safe EEG temporal/spectral features and delayed, baseline-corrected fNIRS response features will add complementary information.	A broader event-aligned EEG–fNIRS fusion proposal, without an explicit delayed-response mechanism.	The hypothesis loses a measurable physiological signature and becomes harder to falsify mechanistically.
Verification	Replaces timestamp synchronization with event identity; uses released HbO/HbR directly; fixes the qualitative trajectory's derived $[-2, 12]$ s task/response feature window and fold-local fitting; records four concrete amendments.	Repairs input and privacy contracts, but does not add a neuro-specific timing constraint; the plan falls back to a direct feature route.	Verification changes from testing a mechanism to making a generic route executable.
Executed graph	Event materialization $\rightarrow$ modality-specific features $\rightarrow$ fold-safe fusion $\rightarrow$ prototypical head.	Event materialization $\rightarrow$ unavailable preprocessing branch $\rightarrow$ two reference-feature routes $\rightarrow$ fold-safe fusion $\rightarrow$ linear probe.	The ablated graph is longer, contains a redundant extraction route, and uses a generic head.
Observed output	All 78 participant–session units complete within the three subject–session folds, with finite late-stage logits and probabilities.	All 78 participant–session units complete within the three subject–session folds, but the out-of-range preprocessing branch is bypassed and the late stage is skipped after middle-stage classification.	The ablation remains executable, but its stage structure no longer directly tests the proposed physiological timing relation.
Predictive outcome	Full dual evidence: hidden BA / macro-F1 0.6873 / 0.7602.	Merged evidence: hidden BA / macro-F1 0.6244 / 0.6133.	Separate evidence packets improve BA by 0.0629 and macro-F1 by 0.1469.

Table 24: Decision trace for completed runs under the same task and protocol. The comparison identifies the first divergence and follows its propagation through Verification, execution, and prediction.

The important contrast is therefore not simply that the ablated score is lower. With dual evidence, the retrieved physiology is converted into a delayed-response hypothesis, and the Verifier turns that hypothesis into an event-identity and windowing constraint that the Executor can test. With merged evidence, execution still succeeds, but the missing role separation weakens the physiological signature, leaves a generic fusion hypothesis, and produces a less targeted executable graph. This is one mechanism consistent with the component sensitivity summarized by the merged-evidence row of Table 2.

H IMPLEMENTATION DETAILS

The anonymized implementation is available at https://anonymous.4open.science/r/efimas-anonymous-code-30E6.

H.1 WORKFLOW PSEUDOCODE

Algorithm 1 summarizes the planning–verification loop, local execution revisions, and Reviewer feedback described in Sections 3.2–3.4. The applicable stage and global budgets are checked before every research-stage retrieval, model call, and tool execution. No further call is issued once either budget is exhausted; if the required plan, output, or Review decision has not been obtained, the workflow returns explicit failure.

Algorithm 1 EFiMAS

Require: task τ, DataProfile D_τ, two evidence indexes, registered tool library, and research budget
Ensure: Reviewer-accepted frozen fusion program $G_\tau^\star$, or explicit failure

repeat
  repeat
    $E_P \leftarrow \textsc{DualRAG}(D_\tau, \text{feedback})$
    $P_\tau \leftarrow \textsc{Plan}(D_\tau, E_P, \text{feedback})$
    $E_V \leftarrow \textsc{DualRAG}(D_\tau, P_\tau, \text{support, counterevidence})$
    $(\widehat{P}_\tau, \text{feedback}) \leftarrow \textsc{Verify}(D_\tau, P_\tau, E_V)$
  until a plan is verified or an applicable budget is exhausted
  if no plan is verified **then**
    return explicit failure
  end if
  Initialize G_τ from the ordered operations in $\widehat{P}_\tau$
  repeat
    for each ordered operation in $\widehat{P}_\tau$ **do**
      $E_M \leftarrow \textsc{MethodRAG}(\widehat{P}_\tau, \text{current operation})$
      $T \leftarrow \textsc{MatchRegisteredTool}(\text{operation contract}, E_M)$
      if $T = \emptyset$ **then**
        Generated-tool sandbox: generate one typed operation from E_M → source, interface, leakage, and resource checks → temporary registration as T.
      end if
      repeat
        Execute the operation with T and its method evidence E_M
        Compare its observed output with the expected output
        if the mismatch is local **then**
          Revise the current operation
        end if
      until the operation can advance or an applicable budget is exhausted
      if the operation cannot advance **then**
        return explicit failure
      end if
    end for
    $E_R \leftarrow \textsc{DualRAG}(G_\tau, \text{stage reports})$
    $d \leftarrow \textsc{Review}(G_\tau, \text{stage reports}, E_R)$
  until $d \neq \textsc{Reexecute}$ or an applicable budget is exhausted
  if $d = \textsc{Reexecute}$ **then**
    return explicit failure
  end if
  if $d = \textsc{Replan}$ **then**
    Use the Reviewer's report as Planning feedback
  end if
until $d = \textsc{Accept}$ or the research budget ends
if $d \neq \textsc{Accept}$ **then**
  return explicit failure
end if
Freeze the Reviewer-accepted program as $G_\tau^\star$
Evaluate $G_\tau^\star$ with private targets
return $G_\tau^\star$

H.2 DATAPROFILE EXAMPLE

We use the word-generation task from the TU simultaneous EEG–NIRS dataset (Shin et al., 2018) to illustrate D_τ. The dataset contains simultaneous recordings from 26 participants over three sessions. Table 25 shows the input presented to the Planner after data preparation. The raw event-analysis boundary is $[0, 12]$ s at the released native rates (2,400 EEG samples and 120 fNIRS samples per

event). The preparation uses its event epoch and lag grid for feature construction, then derives three-second modality slices (600 EEG samples and 30 fNIRS samples) and lagged fNIRS views for the spatial-prior token; the three-second values below therefore describe final derived inputs, not a replacement for the raw boundary.

Profile item	TU simultaneous word-generation task
Task	Word generation versus baseline
Evaluation protocol	Three-fold split by subject and session
Participants and sessions	26 participants and 3 sessions
Event windows	1,560 event-window examples
Derived EEG input	28 channels sampled at 200 Hz; 600 points per three-second slice
Derived fNIRS input	36 HbO and 36 HbR channels sampled at 10 Hz; 30 points per three-second slice
Signal form	EEG signals and released oxy- and deoxyhemoglobin concentrations
Cross-modal time	Event-locked windows paired by trial and event identifiers
Data quality	No missing values in either modality; finite fraction 1.0

Table 25: DataProfile example for the TU simultaneous word-generation task. The values describe the Planner input after data preparation; the original dataset is reported by Shin et al. (2018).

H.3 Dual-evidence retrieval details

The method and neuroscience indexes are searched separately and retain separate scores and source labels. The same role query starts both searches. Before the neuroscience search, it is expanded with fixed terms for EEG electrophysiology, HbO/HbR responses, neurovascular coupling, temporal delay, and spatial correspondence. The active configuration uses task anchoring and limits the evidence packet by role. Planning and Verification receive at most 16 method items, Execution receives at most 12, and Review receives at most 8. Their neuroscience limits are 8, 8, and 4, respectively; Execution receives none. Verification reserves up to four neuroscience positions for counterevidence or limitations. The configuration uses lexical retrieval with evidence-specific reranking and keeps dense retrieval disabled. Each retrieval result retains its query, component, score, and source label.

Table 26 gives representative full-text sources from the two indexes.

Index	Example source	Decision supported
Method	Channel Selection and Early Temporal Features (Li et al., 2017)	EEG and fNIRS processing, feature construction, and fold-local selection
Method	Bidirectional Cross-Modal Transformer (Si et al., 2025)	Cross-modal fusion and its required temporal and spatial inputs
Method	Cross-Modal Transfer from EEG to fNIRS (Wang et al., 2022a)	Directional transfer, training procedure, and reported limitations
Neuroscience	Hemodynamic Correlates of EEG (Kilner et al., 2005)	Physiological support for relating EEG spectra to hemodynamic activity
Neuroscience	EEG–NIRS during Motor Preparation (Zama & Shimada, 2015)	A task-specific temporal relation between electrical and hemodynamic responses
Neuroscience	Neurovascular Coupling during Auditory Stimulation (Muñoz et al., 2023)	Supporting observations and counterevidence for Verification

Table 26: Representative sources from the method and neuroscience indexes.

H.4 Registered tool examples

The runtime catalog contains 227 unique executable tools and 28 registered linear combinations. Table 27 shows representative entries from this catalog; each entry declares a typed input/output artifact contract, prerequisites, fit scope, execution mode, and fallback.

Stage	Catalog scope	Pre-registered examples	Execution / fitting scope
Early examples	Alignment, preprocessing, QC, shallow fusion, and neurovascular features	`event_window_crop`, `physical_lag_shift`, `fnirs_detrend_filter`, `eeg_reference`, `channel_qc_mask`, `neurovascular_lag_stack`, `motor_lateralization_pair`, `early_graph_alignment`	Trial-local unless the declared operation requires training-fold fitting
Middle examples	Feature fusion, directional interaction, distribution alignment, and temporal fusion	`concat_mlp`, `cca`, `eeg_to_fnirs_cross_attention`, `fnirs_to_eeg_cross_attention`, `bidirectional_cross_attention`, `bipartite_graph_fusion`, `quality_gated_mid_fusion`	Closed-form or training-fold fit according to the tool card
Late examples	Decision/representation fusion, calibration, and classifier heads	`logit_average`, `linear_stacking`, `temperature_calibrated_fusion`, `linear_probe`, `mlp_head`, `prototypical_head`, `evidential_head`	Calibration and supervised heads use training rows and labels only
Path-specific examples	Reproducibility tools for the TU simultaneous path	Spatial-prior token construction; training-fold Agent-prior prediction	Label-free representation construction / training-fold prediction head

Table 27: Representative entries from the runtime tool library. The catalog contains 227 executable tools and 28 registered combinations.

The Executor retrieves method evidence for the current operation and searches this catalog by required artifacts, output contract, scientific capability, and fitting rule. For example, an event-locked delayed-response plan can compose `event_window_crop` / `physical_lag_shift` / `neurovascular_lag_stack`; a spatial asymmetric plan can use `early_graph_alignment` / `fnirs_to_eeg_cross_attention`; and a fold-safe prediction path can end with `linear_probe`, `mlp_head`, or `prototypical_head`. A generated tool is considered only when no registered entry or valid composition satisfies the verified contract.

H.5 Stage output checks

Each tool returns an observed output to the runtime. The output check compares it with the expected output in the verified plan. For arrays, the check covers trial alignment, dimensions, finite values, and required metadata. The stage report records the expected output, observed output, and check result. Execution repeats the current stage when a required check fails. Hidden labels remain outside this process.

H.6 Generated-tool sandbox

Generated-tool sandbox boundary. The Executor can generate code only after method RAG and a targeted catalog search establish that no registered tool or valid composition satisfies the required operation. The generated tool supplies one typed processing, fusion, or prediction operation, is registered only for the current run, and cannot modify the source repository. Before invocation it must pass the source, interface, dependency, leakage, and resource checks below.

Configuration. Table 28 summarizes the checks applied before a generated tool can be used. The fitting rule determines whether an operation can process one trial, use an unlabeled subject distribution, or fit parameters on the current training fold. Held-out labels are never provided.

Configuration item	Runtime rule
Generation trigger	No registered tool can perform the required operation
Allowed dependencies	NumPy, SciPy, scikit-learn, PyTorch, MNE, MNE-NIRS, pyRiemann, and selected standard-library modules
Source checks	Reject dynamic execution, unrestricted file access, network calls, multiprocessing, and undeclared dependencies
Interface checks	Test the declared inputs, outputs, array shapes, and finite numeric values on synthetic data
Fitting access	Per trial, unlabeled subject data, or the current training fold
Leakage check	Change held-out labels and reject code whose output changes
Execution	Python isolated mode in a separate process, with no shell or standard input
Execution metadata	Retain source, validation result, and invocation report

Table 28: Configuration used to validate a generated tool before registration.

Resource	Default limit
Wall-clock time	30 seconds
CPU time	20 seconds
Virtual memory	16 GB
Single file	64 MB
Generated source	256 KB
Returned arrays	128 MB
Open file descriptors on POSIX	at most 64

Table 29: Default limits for one generated-tool worker. Wall-clock timeout is enforced by the parent process. POSIX workers also use operating-system resource limits.

Security boundary. Generated code runs in an application-level restricted Python process. The runtime limits imports, file access, execution time, memory use, and access to training labels. It does not provide a container, a separate file-system namespace, or operating-system network isolation. The security boundary therefore depends on source checks, the dependency allowlist, Python isolated mode, resource limits, and the controlled input interface.

H.7 Correspondence between method and implementation

Table 30 maps each method component to the corresponding implementation output and decision record.

Method component	Implementation output	Decision record
Dataset-conditioned Planning	Task profile and proposed fusion plan	Planning decision and selected-path summary
Role-specific dual retrieval	Separate method and neuroscience retrieval calls with role budgets	Retrieval evidence and source identifiers
Independent Verification	A new query and evidence packet for each candidate hypothesis	Planner and Verifier decision records
Hypothesis-linked execution	Program nodes retain expected outputs and evidence identifiers	Executable graph and stage reports
Output-guided revision	Stage judgments record observed gaps and the next decision	Stage reports and Reviewer decisions

Table 30: Correspondence between method components, implementation outputs, and decision records.

I QUALITATIVE EXECUTION TRACE

Table 31 documents the output-guided revision component on TU simultaneous. The trace starts with event-window construction, identifies the missing feature-matrix output, and adds modality processing, a delayed feature adapter, and a fold-local classifier. The TU simultaneous program evaluated in the main table combines a coordinate-aware spatial-prior token with an Agent-prior MLP head (Section F.1).

Step	Program state	Operation and observed output	Decision
1. Planning	EEG at 200 Hz and fNIRS at 10 Hz used independent native clocks	Planner proposed event-aligned EEG features and a delayed HbO/HbR representation	Verifier approved the timing and training-fold requirements
2. Execution	The released continuous signals had not been divided into trials	An event-window operation produced finite paired signal windows	Continue to modality processing
3. Output check	The plan expected EEG and fNIRS feature matrices	The operation returned signal windows without the expected feature matrices	Mark the expected output as unmet
4. Revision	The fusion hypothesis remained supported	Executor added EEG and HbO/HbR processing and a training-fold delayed feature adapter	Repeat Execution with the revised operations
5. Review	The completed program produced predictions and passed the output checks	Reviewer accepted and froze the program	Evaluate the frozen program once on private targets

Table 31: Output-guided revision on TU simultaneous. Each step connects an observed program state to the next execution decision.

The trace records how an observed output changes the next executable operation: event-window construction is followed by modality-specific processing, a generated delayed feature adapter, and a training-fold classifier. The adapter tests delays from 2 to 12 seconds using training-fold data. This execution trace illustrates how output-guided revision turns a missing feature-matrix output into a complete prediction program while retaining the proposed temporal relation.

J ROBUSTNESS ACROSS LANGUAGE MODELS

We evaluate the robustness of the fusion workflow to the choice of language model. The experiment compares three models across all ten EEG–fNIRS tasks, measuring the availability of evaluated programs and the predictive performance achieved under the same robustness-specific interaction budget.

This robustness experiment is separate from the main study. Each model–task pair receives at most three global attempts here, whereas the main study uses its separately stated ten-attempt budget. The three models share the robustness-specific budget, but these results are not budget-matched to the main-study results.

J.1 EXPERIMENTAL SETUP

We use `gpt-5.6-sol`, `gpt-5.6-terra`, and `gpt-5.5` under the same task inputs, role structure, retrieval interfaces, tool access, and robustness-specific interaction budget. Each model performs Planning, Verification, Execution, and Review, producing one evaluated trajectory per task and 30 model–task trajectories in total. Table 32 summarizes the shared settings.

The model receives the task summary, class semantics, data format and scale, sampling rates, channel information, event and window summaries, quality observations, evidence cards, and tool descriptions. These inputs support hypothesis formation and tool selection throughout the four stages.

For each model–task pair, the evaluator freezes a completed candidate using the research-phase decision record. Selection uses the accepted program or the completed program with the highest research-partition BA. The selected program is then evaluated once on held-out data.

Experimental factor	Shared setting
Tasks	The ten EEG–fNIRS tasks used in the main study, with the same task-specific inputs and evaluation protocols across models.
Model input	DataProfile, task and signal summaries, retrieved evidence, tool descriptions, and execution observations. Held-out labels and outcomes remain with the evaluator.
Workflow	Planner, Verifier, Executor, and Reviewer; Review can accept the program, request re-execution, or request replanning.
Stage budget	At most 8 RAG queries, 16 tool searches, and 16 observation loops per stage.
Task budget	At most 32 tool calls, 2 generated tools, 3 global attempts, and 4M input tokens per task; each model call allows at most 65,536 output tokens.
Candidate selection	An accepted program or the completed program with the highest research-partition BA is frozen before one held-out evaluation.
Predictive evaluation	BA and macro-F1 from the frozen program, with the task-specific chance level used for normalization.
Trajectory assessment	`claude-opus-4-6` assesses all 30 trajectories with the same rubric. Four dimensions are scored from 0 to 5; their mean is multiplied by 20 to give a reasoning score from 0 to 100.

Table 32: Controlled settings for the comparison across language models. Each model uses the same information, workflow, and robustness-specific research budget; this budget is separate from the ten-attempt budget of the main study.

We report the number of tasks with an evaluated BA, the mean BA over the ten tasks, and a 10% Winsorized mean that replaces the lowest and highest task scores with their adjacent order statistics. To account for different class counts, we also report chance-normalized BA,

$$\widetilde{\mathrm{BA}}_{m,d} = \min\left(1, \frac{\mathrm{BA}_{m,d}}{2c_d}\right),$$

where c_d is the chance level of task d. This normalization maps chance performance to 0.5; its mean gives equal weight to each task. The reasoning assessment scores scientific grounding, correctness of the fusion plan, tool-trajectory quality, and consistency between observations and subsequent decisions. The judge reads the task summary and the trajectory frozen before held-out evaluation.

J.2 Aggregate performance across models

Evaluated BA is available for all 30 model–task pairs, covering all ten tasks for each model (Table 33). This complete evaluation coverage allows all three models to be compared on the same task set. Mean BA ranges from .4653 to .4899, a spread of 2.46 percentage points. The corresponding chance-normalized means range from .5517 to .5782.

Model	Tasks with BA	Mean BA	10% Winsorized mean BA	Chance-normalized mean BA	Mean reasoning score (0–100)
gpt-5.6-sol	10/10	0.4849	0.4924	0.5763	64.15
gpt-5.6-terra	10/10	0.4899	0.4991	0.5782	64.55
gpt-5.5	10/10	0.4653	0.4768	0.5517	57.15

Table 33: Evaluation coverage and predictive performance across three language models under identical interaction budgets. Each aggregate summarizes ten tasks; reasoning scores use the same judge and rubric.

`gpt-5.6-terra` has the highest mean BA (.4899), followed by `gpt-5.6-sol` (.4849) and `gpt-5.5` (.4653). Both the Winsorized and chance-normalized summaries retain this ordering. The agreement across summaries shows that the aggregate pattern persists when extreme task scores are downweighted or class counts are taken into account.

J.3 Task-level sensitivity and trajectory quality

Table 34 resolves the aggregate results by dataset. Six tasks have a maximum BA spread below 1.5 percentage points across the three models: FJMI, MIND, ds004022, ds004514, Stroop, and

TU HBCI. The larger differences occur on HYGRIP (15.81 points) and TU simultaneous (8.08 points). These task-level measurements identify where predictive performance is most sensitive to the language model used to construct the fusion program.

Dataset	gpt-5.6-sol BA	gpt-5.6-terra BA	gpt-5.5 BA	BA range (pp)
FJMI	0.1353	0.1248	0.1213	1.40
HEFMI	0.5347	0.5337	0.5653	3.16
HYGRIP	0.6178	0.6858	0.5277	15.81
MIND	0.2559	0.2657	0.2631	0.98
ds004022	0.2517	0.2425	0.2375	1.42
ds004514	0.4909	0.4973	0.4980	0.71
Stroop	0.5072	0.5098	0.5153	0.81
Figshare	0.6900	0.6900	0.6500	4.00
TU HBCI	0.6345	0.6337	0.6238	1.07
TU simultaneous	0.7314	0.7160	0.6506	8.08

Table 34: Task-level performance under different language models. BA range is the difference between the highest and lowest model BA on the task, expressed in percentage points (pp).

The best-performing model also varies by task. `gpt-5.6-sol` is strongest on FJMI, ds004022, TU HBCI, and TU simultaneous; `gpt-5.6-terra` is strongest on HYGRIP and MIND; and `gpt-5.5` is strongest on HEFMI, ds004514, and Stroop. The two `gpt-5.6` models tie on Figshare. The strongest model therefore depends on the task, while all three models yield evaluated programs across the full task set.

Trajectory assessment provides a complementary view of these results. `gpt-5.6-terra` receives the highest mean reasoning score (64.55), followed by `gpt-5.6-sol` (64.15) and `gpt-5.5` (57.15). Across all 30 trajectories, the Pearson correlation between reasoning score and chance-normalized BA is $r = -0.079$. The differing orderings show why both the decision process and its predictive outcome are measured when assessing model robustness. Taken together, the evaluation coverage, aggregate scores, and task-level comparisons support the workflow's applicability across the three models and characterize how model choice affects its performance.

K Dual RAG Knowledge Bases

The system uses two evidence stores because the scientific question and the implementation question are related but not interchangeable. The neuroscience store supports claims about neurovascular timing, regional responses, and task mechanisms. The method store supports concrete feature, fusion, fitting, and evaluation operations. Each record retains its source identifier, paper kind, evidence role, fusion stage, task assignments, and limitations. Retrieval applies metadata filtering, lexical search, optional dense retrieval, reciprocal-rank fusion, evidence reranking, and role/task boundary filtering before compact decision cards are passed to a model.

K.1 Knowledge-base construction and quality control

The construction pipeline starts from searches of PubMed, Semantic Scholar, and arXiv. We normalize the metadata and remove duplicate records. We then assign each candidate paper p a deterministic quality score

$$S(p) = S_{\text{mod}} + S_{\text{evi}} + S_{\text{task}} + S_{\text{venue}} + S_{\text{journal}} + S_{\text{cite}} + S_{\text{type}} + S_{\text{year}} + S_{\text{access}}, \tag{23}$$

where the nine terms sum to at most 31 points. Table 35 gives the complete allocation.

Criterion	Range	Exact rule
Modality relevance	0–5	5: explicit EEG–fNIRS fusion, BCI, or neurovascular coupling; 4: joint recording or analysis; 3: both modalities are discussed without a strong joint relation; 2: one modality with relevant physiological evidence; 1: weak single-modality relevance; 0: neither modality.
Evidence coverage	0–5	One point for each matched family, capped at 5: EEG marker, fNIRS marker, brain region, temporal relation, cross-modal relation, and artifact or confound.
Task match	0–3	3: target task or neurovascular coupling; 2: adjacent cognitive or clinical task; 1: general signal processing or machine learning; 0: no task match.
Venue	0–4	4: core neuroscience or neuroengineering venue; 3: secondary domain venue; 2: another peer-reviewed journal; 1: venue identified but quality unknown; 0: venue missing. The two venue lists are fixed before scoring.
Journal standing	0–4	JCR Q1/Q2/Q3/Q4 gives 4/3/2/1. Without JCR quartile, JIF percentile $\geq 90/75/50/ < 50$ gives 4/3/2/1, or impact factor $\geq 10/6/3/ > 0$ gives 4/3/2/1. SJR Q1/Q2/Q3–Q4 gives 3/2/1.
Citations	0–4	< 10: 0; 10–49: 1; 50–199: 2; 200–499: 3; ≥ 500: 4. A paper from 2024 onward receives 1 point when citation data are missing and its venue or journal score is at least 3.
Paper type	0–3	3: review, meta-analysis, method, dataset, benchmark, or experimental study; 2: applied, clinical, BCI, or otherwise empirical study; 1: short or preliminary paper; 0: editorial, commentary, or letter.
Recency	0–2	2: 2020 onward; 1: 2010–2019; 0: before 2010. A pre-2010 paper with at least 200 citations receives 1 point.
Access	0–1	1: an open PDF, open-access flag, or PMC record is available; 0: none is available.

Table 35: Deterministic scoring rubric used to prioritize candidate papers.

The modality score is also a gate. Papers scoring 0 or 1 on modality relevance are rejected. A single-modality paper with a modality score of 2 must reach 26 points to remain eligible. Among the remaining papers, scores of 22–31 enter the must-download queue, scores of 17–21 enter the candidate queue, and lower scores are rejected. Strong EEG–fNIRS papers on neurovascular coupling, BCI, or motor imagery, as well as reviews and papers from the highest venue tier, can be routed to manual review when their total is below 17. For example, FGANet receives $5+3+3+4+3+2+3+2+1 = 26$ points and therefore enters the must-download queue. The scoring pass covers 414 candidates, and the frozen metadata snapshot retains 409 records.

The 31-point score is rule-based; Qwen does not generate or alter it. We use a local Qwen3.5-27B model only to review new arXiv candidates after scoring. The model runs at temperature zero and reads the title, abstract, keywords, source, and PDF link. It assigns a modality label and confidence, extracts short EEG and fNIRS evidence phrases, identifies joint recording or analysis, and returns a download, review, or skip action. This pass reviews 43 papers. It identifies 19 EEG–fNIRS papers, four EEG-only papers, two fNIRS-only papers, and 18 unrelated papers. The resulting actions comprise 18 downloads, one manual review, and 24 skips.

The full-text corpus initially contains 282 documents. SHA-256 deduplication removes 69 repeated copies and leaves 213 unique papers. We build two indexes from this shared source pool. The method branch registers 188 paper dossiers. The neuroscience branch registers 47 source papers. A paper can support both branches, so these two counts describe index membership rather than distinct papers.

Each accepted full text is converted into a dossier and a set of typed components. The component types cover acquisition, preprocessing, EEG and fNIRS features, fusion, classifiers, evaluation, task mechanisms, and counterevidence. A component records its inputs, ordered operations, outputs, reported parameters, fitting scope, label visibility, applicable tasks, failure conditions, and limitations. Every material field links to an exact PDF span, including its page, section, bounding box, and verbatim text.

A separate semantic audit compares every proposed method component with its linked spans. The audit processes 911 candidate cards. It accepts 684, revises 211, and rejects 16. Accepted and corrected cards are reconstructed from their cited spans; rejected cards are excluded from retrieval.

The final method index combines 895 audited cards with 88 previously validated cards, giving 983 records. The neuroscience index contains 232 quote-preserving records.

Evidence layer	Papers or dossiers	Semantic records/cards	Runtime use
Method RAG	188 method dossiers	983 method records	Feature extraction, fusion, fitting scope, output contracts, and implementation evidence.
Neuroscience RAG	47 source papers	232 quote-preserving records	Physiological mechanisms, timing, regions, task contrasts, directionality, and falsifiers.
Full-text source corpus	213 unique papers	282 source documents before duplicate removal	Exact source spans for component construction and audit.
Candidate inventory	409 retained metadata records	Bibliographic metadata	Bibliographic basis for constructing and updating the two evidence stores.

Table 36: Dual-RAG asset sizes and their roles in program construction.

Table 37 lists representative sources from established neuroscience and neuroengineering venues. The examples cover both the scientific relations used to form a hypothesis and the operations used to implement it.

Evidence branch	Representative source	Venue	Evidence contributed to the knowledge base
Method	A transfer-learning hybrid EEG–fNIRS BCI for intracerebral hemorrhage rehabilitation (Chen et al., 2025a)	*Advanced Science*, 2025	Cross-subject transfer, modality-specific feature construction, fusion, and clinical evaluation.
Neuroscience	Neurovascular coupling during attention in visual working memory (Zhang et al., 2024)	*iScience* (Cell Press), 2024	Task-dependent theta and hemodynamic relations in frontoparietal regions.
Neuroscience	Hemodynamic correlates of EEG: a heuristic (Kilner et al., 2005)	*NeuroImage*, 2005	A foundational account of the relation between electrophysiological activity and hemodynamic signals.
Neuroscience	Simultaneous EEG and NIRS during cognitive tasks (Shin et al., 2018)	*Scientific Data* (Nature Portfolio), 2018	Acquisition, task timing, signal organization, and validation for a reusable multimodal dataset.
Method	FGANet: fNIRS-guided attention for hybrid EEG–fNIRS interfaces (Kwak et al., 2022)	*IEEE TNSRE*, 2022	A directional fusion pattern in which fNIRS spatial information guides EEG attention.

Table 37: Representative sources retained in the dual-RAG knowledge bases.

Evidence records and tools play complementary roles. RAG records support hypotheses and implementation choices, whereas tools instantiate admissible operations after their input representation and fitting scope have been checked. The following examples show the model-facing structure used for both evidence stores.

K.2 Example neuroscience evidence card

The following quote-preserving neuroscience card is supplied to Planning. It contains the claim, task, modality, source location, and quotation status used by downstream roles.

RAG Card: Neuroscience Evidence

Title: Simultaneous measurement of electroencephalography and near-infrared spectroscopy during voluntary motor preparation
Evidence type: fNIRS finding; **scope:** core mechanism
Claim: "Using sliding window analysis on the haemodynamic response, the oxyHb was found to significantly start to increase in the contralateral premotor cortex (BA6, ch-8: [-41, 11, 60]) at about -900 ms ($t(14) = 3.84$, $p = 0.0009$)."
Task: voluntary motor preparation; **modalities:** EEG and fNIRS/NIRS
Source location: page 2, Results
Evidence status: direct quotation preserved in the source record; eligible for neuroscience planning.

This card contributes a timing and premotor-region mechanism. The Planner combines it with a compatible method record and states a falsifier, while the Verifier checks whether the proposed representation is available in the current release.

K.3 Example method evidence card

The following method record translates a published operation into an implementation constraint for feature construction and fusion.

RAG Card: Method Evidence

Title: Enhancing Performance of a Hybrid EEG–fNIRS System Using Channel Selection and Early Temporal Features
Component type: fusion method; **workflow phase:** middle fusion
Operation: Normalize the modality-specific trial features and concatenate the EEG and fNIRS blocks.
Reported parameters: 44-dimensional EEG block; four-dimensional fNIRS block; 48-dimensional hybrid vector; each block rescaled to $[0, 1]$
Dataset assignments: Figshare, TU HBCI, HEFMI, and TU simultaneous; **fit scope:** trial-local feature construction
Source evidence: The normalized 44-dimensional EEG features and four fNIRS features were concatenated into a 48-dimensional hybrid feature vector for each trial.

The assignment list guides retrieval, while the current DataProfile, sampling rate, task, and release representation determine applicability. Consequently, the same paper component can be used with an adapter on one task and rejected on another.

L Tool Library and Tool-Card Contract

The runtime catalog contains 227 unique executable tools and 28 registered linear combinations. The model-facing catalog used in the robustness experiment (Appendix J) provides 185 capability cards in 23 categories and 23 workflow bundles. These cards give the models compact descriptions of available operations and their input and output requirements.

Tool construction follows four steps. First, an operation is extracted from method evidence or a validated implementation. Second, it is assigned a typed input/output schema, required metadata, applicability tags, fitting scope, and failure behavior. Third, contract tests check shapes, finite values, units, stage compatibility, and leakage boundaries on controlled inputs. Fourth, the tool is registered with a stable runtime identifier and a compact usage guide. New generated tools enter only after targeted evidence retrieval and the same interface checks; source and invocation metadata are retained with the execution.

L.1 Typical tool source

As a concrete example, the registered early-stage tool `workload_theta_hemodynamic_pair` implements an asymmetric workload representation. It preserves an EEG theta summary

as the fast neural term, a compact fNIRS hemodynamic summary as the slow vascular term, and their interaction. The excerpt below shows the registered executable implementation.

```
if tool_id == "workload_theta_hemodynamic_pair":
    roi_indices = payload.metadata["roi_indices"]
    eeg_rois = dict(roi_indices["eeg"])
    fnirs_rois = dict(roi_indices["fnirs"])
    if "frontal" not in eeg_rois or "frontal" not in fnirs_rois:
        raise ValueError("workload_theta_hemodynamic_pair requires frontal ROI indices")
    frontal_eeg = eeg[:, np.asarray(eeg_rois["frontal"], dtype=int)]
    frontal_fnirs = fnirs[:, np.asarray(fnirs_rois["frontal"], dtype=int)]
    theta = np.mean(
        _fft_filter(frontal_eeg, payload.signals.eeg_rate_hz, 4.0, 8.0) ** 2,
        axis=(1, 2),
    )
    hemo = np.column_stack(
        [
            frontal_fnirs.mean(axis=(1, 2)),
            frontal_fnirs[..., -1].mean(axis=1) - frontal_fnirs[..., 0].mean(axis=1),
        ]
    )
    return _set_fused(payload, np.column_stack([theta, hemo, theta[:, None] * hemo]))
```

L.2 Example usage instruction supplied to the model

The matching model-facing instruction exposes the same operation as a compact typed contract:

Tool Card: workload_theta_hemodynamic_pair

Purpose: Couple a fast EEG workload-band summary to a compact hemodynamic fNIRS summary while preserving the asymmetric modality roles.
Operation: Read frontal ROI indices, filter EEG in the 4–8 Hz band, compute mean fNIRS level and end-minus-start response, and append EEG×fNIRS interaction terms.
Input: `signals.eeg` and `signals.fnirs`, each a trial-by-channel-by-time NumPy array, plus `metadata.roi_indices` with `eeg.frontal` and `fnirs.frontal`.
Output: `features.fused` with one EEG theta feature, two fNIRS hemodynamic features, and their two interaction features.
Parameters: The EEG band is 4–8 Hz; frontal ROI indices are required and missing indices raise a contract error.
Fit scope: `trial_local`; **execution:** `closed_form`; labels are not needed for feature construction.

The source defines the operation, while the usage card provides the compact object used for tool selection. The Executor binds the card to the current artifact, and the Verifier checks the next-stage input before execution.

M Three Feedback Loops in an Accepted Trajectory

The workflow contains three distinct feedback mechanisms. The Planner–Verifier loop tests whether a scientific proposal is supported and executable before any tool is run. The Execution/ReAct loop responds to observed artifacts and runtime errors without silently changing the scientific hypothesis. Finally, Review either accepts the completed attempt or returns outcome-linked feedback that opens another global planning attempt.

We illustrate all three mechanisms with an accepted Figshare trajectory that reaches BA 0.8000 on its research partition. The sequence contains 21 ordered turns from Planning to Review. The cards reproduce the original model-written decision fields verbatim, with line wrapping and omission of repeated nondecision metadata for readability. Table 38 summarizes the complete sequence, including context assembly, retrieval, stage handoffs, and final acceptance.

M.1 From a released dataset to a Planner hypothesis

Before the first turn, the data adapter converts release-level facts into a target-free DataProfile supplied to the Planner. The following card shows its decision-relevant fields; repeated channel descriptors are abbreviated at the marked ellipses.

Planner Input: Figshare DataProfile

task:
{dataset_id: figshare_smr, task_name: "right versus left hand motor execution", class_names: [right_hand, left_hand], modalities: [eeg, fnirs], protocol_id: subject_fixed_11_2_2, target_unit: trial}
dataset overview:
{subject_count: 15, session_count: 15, trial_count: 750, eeg_channel_count: 21, fnirs_channel_count: 68, missingness: {eeg: 0.0, fnirs: 0.0}}
cross-modal timing:
{common_time_origin: released_shared_event_clock, eeg_rate_hz: 250.0, fnirs_rate_hz: 10.42, offset_seconds: 0.0, synchronized: true}
release boundary:
{source_representation: continuous_release_streams, precomputed_trial_signals_used: false, scientific_processing_deferred_to_tools: [event_window_extraction, explicit_resampling, modality_conversion, feature_extraction], private_targets_exposed: false}
metadata preview:
{eeg: [{name: F3, hemisphere: null, roi: null}, ..., {name: C4, hemisphere: null, roi: null}], fnirs: [{name: fnirs_wavelength_1_intensity:Fc3A, hb_type: released_intensity, hemisphere: null, roi: null, wavelengths_nm: []}, ...]}

This input does three jobs before any fusion choice is made. It establishes that both modalities are synchronized continuous streams, exposes the physical sampling-rate difference, and states that scientific processing remains an Executor decision. It also makes missing spatial metadata explicit. The Planner can reason about motor physiology, but it cannot treat channel names as verified hemisphere or ROI labels.

The retrieval service constructs the Planner query from the task portion of this profile. The returned packet contains 16 ranked cards; the two cards cited by the Planner are shown below.

Planner Retrieval Request

role: planner
query: "right versus left hand motor execution Classify released right- and left-hand motor-execution events using continuous synchronized EEG and fNIRS wavelength-intensity streams."
returned packet size: 16 evidence cards

Evidence Cited by the Planner

rank 7; neuroscience evidence; DOI 10.1117/1.JBO.20.10.106013: "A relationship between the brain rhythmic activity and the hemodynamic response was studied using the simultaneous measurement of electroencephalogram (EEG) and the functional near-infrared spectroscopy (fNIRS) during a motor task (self-paced right finger movements) for 10 subjects." "During the task an increase in oxyhemoglobine (HbO) was accompanied by a decrease in deoxyhemoglobine (HbR) concentration and a decrease in amplitudes (desynchronisation) of alpha (8–13 Hz) and beta (13–30 Hz) EEG rhythms."

rank 12; method evidence; DOI 10.1088/1741-2560/12/3/036004: "The BCI utilized simultaneously recorded electroencephalographic (EEG) and functional near-infrared spectroscopy (fNIRS) signals." "The time-phase-frequency feature was extracted from EEG, whereas the HbD [the difference of oxy-hemoglobin (HbO) and deoxy-hemoglobin (Hb)] feature was used to improve the classification accuracy of fNIRS."

card limitation: "Method parameters require full-text confirmation."

The first card supplies a testable electrophysiology–hemodynamics relation; the second supplies an executable multimodal feature pattern. The Planner then binds those two roles to the observed task and signal interface. Its original response below includes the hypothesis, expected signatures, falsifiers, and stage allocation that entered Verification.

Original interaction: Planner hypothesis response

next_action: `submit_plan`
recommended_hypothesis_id: `h1_contralateral_erd_hbo_fusion`
evidence_component_ids: [`doi:10.1117/1.jbo.20.10.106013`, `doi:10.1088/1741-2560/12/3/036004`]
statement: "Right- versus left-hand motor execution trials can be distinguished above chance by subject-wise cross-validated classification using early-fused features derived from contralateral EEG mu/beta event-related desynchronization and fNIRS optical-density changes in motor cortical channels."
mechanism: "Motor execution produces lateralized sensorimotor cortex activation: EEG mu (8–13 Hz) and beta (13–30 Hz) power decrease over the contralateral hemisphere during movement planning and execution, while fNIRS intensity in contralateral motor regions shows a task-related hemodynamic response. These complementary electrophysiological and hemodynamic signatures, when extracted per trial and concatenated, provide sufficient class-discriminative information for a linear classifier under subject-wise validation."
expected_signatures: "Contralateral EEG mu and beta bandpower decrease relative to baseline in C3/C4 and adjacent central channels"; "Contralateral fNIRS optical-density slope or peak amplitude increase in C3*/C4*/Cp3*/Cp1* channels relative to baseline."
falsifiers: "Subject-wise cross-validated validation AUC falls within chance-level confidence interval"; "Per-subject separability diagnostics show no stable contralateral feature effect in either modality"; "Fused feature block performs no better than the best unimodal block in fold-safe model selection."
stage_modes: early = `active`; middle = `skip`; late = `classifier_only`.

This response illustrates both the value and the limit of Planner reasoning. The DataProfile and retrieved evidence support a motor-related spectral plus hemodynamic hypothesis and an early-fusion test. However, the response also maps that mechanism to contralateral channels even though the DataProfile records null ROI and hemisphere fields. The next loop is therefore necessary: the Verifier preserves the evidence-supported complementarity claim but removes the unsupported spatial commitment before execution.

Turn	Role or interaction	Decision/status	State transition
1	Planner	`submit_plan`	Submit one motor-lateralization hypothesis.
2	Verifier input	verification context assembled	Bind the proposed plan to independent evidence and tool-search state.
3	Verifier	`search_tools; approved=false`	Block the plan on a baseline-window mismatch, unsupported fNIRS laterality, and missing capabilities.
4	Verifier	`search_tools; approved=false`	Search for feature concatenation, event filtering, and fold-safe preparation.
5	Verifier	`search_tools; approved=false`	Refine executable alternatives after the first search results.
6	Verifier	`search_tools; approved=false`	Retain a feasible late head while searching for missing early features.
7	Verifier	`search_tools; approved=false`	Resolve the remaining EEG and fNIRS feature contracts.
8	Verifier	`submit_verification; approved=true`	Approve an amended, metadata-compatible path.
9	Executor	`tool_search`	Request the missing epoch materializer and feature-concatenation tools.
10	Retrieval	tool cards returned	Add the two executable contracts to the Executor context.
11	Executor	`act`	Materialize trials, extract modality-specific features, and concatenate them.
12	Executor observation	`advance; expectation_met=true`	Confirm the early fused artifact and feature manifest.
13	Early handoff	`completed`	Publish the early-stage artifact collection.
14	Executor	`mode=skip`	Apply the verified middle-stage skip.
15	Middle handoff	`skipped`	Pass the early fused representation to the late stage unchanged.
16	Executor	`act`	Run fold-safe feature finalization and a linear probe.
17	Executor observation	`revise; expectation_met=false`	Detect that finalization consumed the pre-concatenation collection.
18	Executor	`act`	Rebind finalization to the concatenated feature collection.
19	Executor observation	`advance; expectation_met=true`	Confirm finite finalized features and predictions.
20	Late handoff	`completed`	Publish logits and probabilities in runtime row order.
21	Reviewer	`accept`	Freeze the completed graph and terminate the attempt.

Table 38: Complete chronological index of the accepted Figshare trajectory. The detailed cards below quote the response fields that caused each state transition.

M.2 Planner–Verifier loop

The Planner began from a physiological relation between lateralized motor EEG and a slower fNIRS response, then made that relation falsifiable rather than requesting a generic multimodal classifier.

Trace: Planner response

next_action: `submit_plan`
recommended_hypothesis_id: `h1_contralateral_erd_hbo_fusion`
statement: "Right- versus left-hand motor execution trials can be distinguished above chance by subject-wise cross-validated classification using early-fused features derived from contralateral EEG mu/beta event-related desynchronization and fNIRS optical-density changes in motor cortical channels."
stage_modes: early = `active`; middle = `skip`; late = `classifier_only`
falsifiers: "Subject-wise cross-validated validation AUC falls within chance-level confidence interval"; "Per-subject separability diagnostics show no stable contralateral feature effect in either modality"; "Fused feature block performs no better than the best unimodal block in fold-safe model selection."
rationale: "Early fusion is used because the literature suggests combining EEG spectral features and fNIRS hemodynamic features improves decoding; middle fusion is skipped to avoid unnecessary complexity."

The first verification pass did not merely endorse this plausible mechanism. It found that the proposed fNIRS window excluded the baseline required by the same plan, that fNIRS laterality was not established by the released metadata, and that no retrieved tool contracts yet implemented three required capabilities.

Trace: Initial Verifier response

next_action: `search_tools`; **approved:** false
blocking finding: "The early-stage plan sets fNIRS epoch window to [0,12] s but computes optical-density baseline using I_0 from [-2,0] s; this leaves no baseline samples. The epoch window must be [-2,12] s or the baseline definition must be changed."
metadata finding: "Dataset preview shows fNIRS hemisphere=null and source/detector metadata absent, but channel names include standardized EEG-like positions. Laterality can be used only after confirming the mapping in channel_metadata."
implementation finding: "The current tool_search_results are empty and no registered tool IDs are available for the required feature extraction and classification capabilities; feasibility cannot be established without dedicated searches."
amendment: "Remove fold-split manifest and validation indices from generated tool input contracts; subject-wise grouping is provided by the runtime protocol and the classifier only receives subject identifiers to avoid leakage."

Across turns 3–7, the Verifier searched separately for EEG spectral features, fNIRS optical-density summaries, task-event filtering, feature concatenation, fold-safe feature preparation, and a classifier head. This separation matters: each query repaired a named contract gap instead of broadening the hypothesis after seeing a score. The final response retained the motor mechanism but removed unsupported spatial claims and selected operations compatible with the released metadata.

Trace: Final Verifier response

next_action: `submit_verification`; **approved:** true
selected_hypothesis_id: `h1_contralateral_erd_hbo_fusion`
finding: "EEG channel metadata has roi=null; the dedicated ERD tool requires metadata.roi_indices.eeg, which is unavailable. The proposed ERD percent-change pathway cannot execute without a generated ROI mapping."
finding: "fNIRS channel metadata lacks hemisphere, coordinate, source/detector, and wavelength labels. Laterality cannot be inferred from names like C3/C4 because suffixes are not documented scalp anchors."
amendment: "Replace EEG ERD percent-change extraction with sensorimotor Welch bandpower from the release sensorimotor bundle."
amendment: "Use symmetric fNIRS optical-density slope/peak features across all wavelength-intensity channels; remove contralateral C3*/C4* channel requirement."
amendment: "Insert fold-safe feature finalization before the linear probe to satisfy fold-safe standardization/feature selection."

The loop therefore changed the executable claim before training: the accepted plan still tested complementary sensorimotor activity, but it no longer pretended that absent ROI or hemisphere metadata were known. Verification also made the middle-stage skip meaningful by requiring the early fused feature block to satisfy the late-stage interface directly.

M.3 Execution/ReAct loop

Execution first retrieves the two missing contracts and then submits an early action batch. The action dependencies specify the complete program rather than only its final tools.

Trace: Early execution action

turn_decision: `act`
materialize_trials: tool = `raw_event_epoch_materializer`; depends on = []; expected output = EEG and fNIRS trial arrays, rates, trial identity, and epoch alignment.
extract_features: tool = `figshare_smr_release_od_sensorimotor_bundle`; depends on = [`materialize_trials`]; expected output = separate EEG and fNIRS feature blocks.
concatenate_features: tool = `trial_feature_concatenation`; depends on = [`extract_features`]; expected output = `features.fused` and a feature manifest.

Trace: Early execution observation

decision: `advance`; **expectation_met:** true
observed_results: "Execution status completed"; "materialize_trials completed with 1/1 folds"; "extract_features completed with 1/1 folds"; "concatenate_features completed with 1/1 folds"; "Output references present for materialize_trials, extract_features, and concatenate_features collections."
unmet_expectations: []

Because the Verifier approves a direct early-to-late handoff, the middle stage records `mode=skip` and `middle:verified_skip_without_processing`. This explicit skip passes the early fused representation directly to the late stage.

The first late action exposes an interface mistake: feature finalization is bound to the feature-extraction collection rather than the subsequent concatenation collection. The observation identifies the missing fused artifact, so the Executor preserves the hypothesis and classifier while correcting the dependency.

Trace: Late execution observation and repair

decision: `revise`; **expectation_met:** false
observed_results: "fold_safe_feature_finalize_bundle: Missing prerequisites: features.fused"; "linear_probe: upstream_dependency_unavailable: Missing runtime output for dependency finalize_fused_features."
unmet_expectations: "finalize_fused_features did not run because input collection lacked features.fused"; "No output references for features.fused, predictions.logits, or predictions.probabilities."
revision_instruction: "Rerun fold_safe_feature_finalize_bundle with the early concatenate_features collection, keep parameters default, and rerun linear_probe depending on it with ridge 0.01."

Trace: Repaired late execution observation

decision: `advance`; **expectation_met:** true
observed_results: "execution_status: completed"; "executed actions: finalize_fused_features completed, fit_linear_probe completed"; "output references: finalize_fused_features collection and fit_linear_probe collection present"; "execution evidence: finite output for both actions"; "runtime_expectation_met: true."
unmet_expectations: []

This local revision changes the artifact binding while preserving the sensorimotor hypothesis and representation. The repaired late stage then emits finite logits and probabilities in runtime row order.

M.4 Review and global feedback

Review received the verified plan, completed graph, and stage reports. Its response closed the attempt rather than requesting another planning cycle; held-out predictive metrics were not part of this context.

Trace: Reviewer response

action: `accept`; **failed_gates:** []
rationale: "All stage expectations were met: early materialization/feature extraction/concatenation completed; middle stage was skipped by verified design; late finalization and linear probe completed with finite outputs and zero failed folds."
tool_path_interpretation: "materialize_trials produced event-locked EEG and fNIRS trial arrays."; "figshare_smr_release_od_sensorimotor_bundle extracted EEG sensorimotor Welch bandpower and fNIRS optical-density slope/peak features."; "trial_feature_concatenation fused the separate feature blocks into features.fused with a manifest."; "fold_safe_feature_finalize_bundle applied fold-safe standardization/selection within the training scope."; "linear_probe fitted a ridge classifier and emitted finite runtime-row-order predictions."

Review accepts the attempt after all stage expectations are satisfied. Across the sequence, Planning forms a testable physiological proposal, Verification removes the unsupported spatial commitment and completes the typed tool chain, Execution repairs the artifact dependency while preserving the hypothesis, and Review confirms a complete graph with finite predictions. The example therefore shows how the three feedback loops transform a DataProfile and retrieved evidence into an accepted executable program.

N Analysis of the Selected MIND Program

We analyze the selected MIND program used in the main comparison. MIND has four classes and uses the registered group-wise five-fold protocol for seeds 0, 1, and 2. The program achieves BA 0.299444, rounded to .2994. The frozen path is an asymmetric EEG–fNIRS encoder with an fNIRS-conditioned gate, multiplicative interaction, and LogisticRegression head.

Stage	Trajectory evidence	Effect on the resulting program
Planning	The selected plan assigns EEG the primary directional representation and uses fNIRS as a task-conditioned reliability and interaction signal.	The executable hypothesis is fixed as asymmetric encoding, fNIRS gating, multiplicative interaction, and a LogisticRegression probe.
Execution	The frozen path performs train-fold modality standardization, asymmetric EEG/fNIRS encoding, the fNIRS-conditioned EEG gate, element-wise interaction, additive projection, and fold-local LogisticRegression fitting.	The same registered path is run under the group-wise five-fold protocol for seeds 0, 1, and 2, preserving the declared input, fit scope, and class order.
Aggregation	The evaluator returns the candidate results only after each frozen seed run completes.	The three-seed main-table aggregation is reported as BA .2994 and Macro-F1 .2549; these values do not feed back into planning or path selection.

Table 39: Three-seed evaluation of the MIND program used in the main table. Planning fixes the scientific relation, Execution instantiates the frozen program, and Aggregation reports the seed-level predictive result only after the protocol is complete.

The selected program is justified by the following rationale:

Trace: Selected MIND program

"The promoted candidate preserves the declared asymmetric EEG–fNIRS relation, fits all supervised operations within the training fold, and emits the four-class LogisticRegression output required by the evaluator. The frozen path is aggregated over seeds 0, 1, and 2 only after the three runs complete."

The three stages connect the scientific hypothesis to its predictive outcome. Planning selects the asymmetric relation, Execution implements it as a frozen program, and evaluation measures that program across three seeds. The fNIRS gate and multiplicative interaction specify how the selected design combines the two modalities.

O SELECTED PATHS VERSUS EXISTING ALGORITHMS

The main tables compare predictive performance. We additionally compare the executable structure of four successful or comparatively strong task-conditioned paths with the fixed algorithms used as baselines. The BA values follow the same evaluation protocol as the main tables, while the structural comparison examines representation, fusion, and head choices.

Dataset	EFiMAS	DAMF	DeepSync	BiSTF	Strongest baseline (BA)	Δ
TU HBCI	0.7411	0.7235	0.6818	0.6923	CBraMod PT+FT (0.7605)	-0.0194
Figshare	0.7160	0.6000	0.5707	0.6347	fNIRS Transformer (0.7027)	$+0.0133$
TU simultaneous	0.6873	0.5718	0.5442	0.5160	CBraMod PT+FT (0.6353)	$+0.0520$
HEFMI	0.6656	0.5160	0.5162	0.5271	EEG-Conformer (0.5510)	$+0.1146$

Table 40: Quantitative comparison on four representative tasks under the main-table protocols. Δ is EFiMAS BA minus the strongest baseline BA.

Dataset	Selected path	Difference from fixed algorithms
TU HBCI	Event windows $\rightarrow$ workload OD features $\rightarrow$ fold-safe assembly $\rightarrow$ linear probe	Task-specific workload/OD summaries and a compact fold-local probe replace a fixed multimodal architecture; the path is not the learned DAMF block or the synchronized DeepSync/BiSTF representation.
Figshare	Motor-lateralization pair $\rightarrow$ fold-local feature concatenation $\rightarrow$ linear probe	A hand-task lateralization hypothesis, canonical sensorimotor pairing, and fold-local feature selection are explicit; the strongest fixed comparator is an fNIRS Transformer, not this path.
TU simultaneous	Coordinate-aware spatial-prior token $\rightarrow$ Agent-prior MLP head	A spatial-prior token is constructed before a task-specific nonlinear head; this differs from generic deep fusion and EEG-pretrained fixed entries.
HEFMI	Delayed HbO/HbR/tHb representation $\rightarrow$ fold-local selection $\rightarrow$ fNIRS linear head	The selected representation is delayed and asymmetric, and the path permits fNIRS-only prediction instead of forcing symmetric multimodal fusion or the fixed EEG Conformer.

Table 41: Structural comparison of the same four paths. Differences are reported at the representation, fusion-stage, and prediction-head levels.

For a compact quantitative comparison, each selected path and each named fixed algorithm is encoded on eight executable axes: signal representation, EEG operator, fNIRS operator, temporal handling, spatial prior, fusion operator, fusion direction, and prediction head. Across the twelve path–baseline pairs, no pair matches on all eight axes; the normalized distance ranges from 0.875 to 1.000 with mean 0.9583. Every pair therefore differs on at least seven of the eight axes, quantitatively confirming the representation, fusion, and head differences in Table 41.

Across these four tasks, EFiMAS exceeds DAMF-Net in all four cases ($+0.0176$, $+0.1160$, $+0.1155$, and $+0.1496$ BA), and exceeds the strongest baseline in three cases. TU HBCI provides the exception: the task-conditioned path remains above DAMF-Net but below the strongest foundation-model entry. Together with the high structural distances, these results show that the system constructs task-specific executable programs rather than replaying a fixed baseline architecture.