# Technical Appendix

# Extended Methods

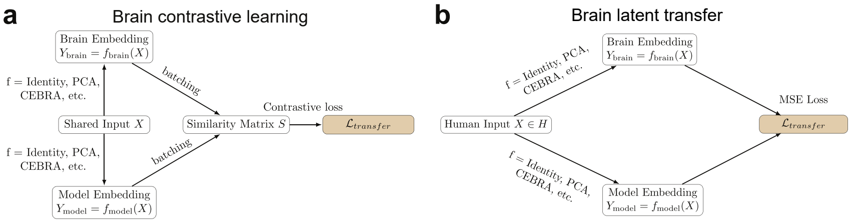
a
Brain contrastive learning
Brain Embedding $Y_{\text{brain}} = f_{\text{brain}}(X)$
f = Identity, PCA, CEBRA, etc.
Shared Input $X$
f = Identity, PCA, CEBRA, etc.
Model Embedding $Y_{\text{model}} = f_{\text{model}}(X)$
batching
batching
Similarity Matrix $S$
Contrastive loss
$\mathcal{L}_{transfer}$
b
Brain latent transfer
Human Input $X \in H$
f = Identity, PCA, CEBRA, etc.
Brain Embedding $Y_{\text{brain}} = f_{\text{brain}}(X)$
f = Identity, PCA, CEBRA, etc.
Model Embedding $Y_{\text{model}} = f_{\text{model}}(X)$
MSE Loss
$\mathcal{L}_{transfer}$

Figure 3: Obtaining $\mathcal{L}_{transfer}$ in B2M. (a) Brain contrastive learning. A shared input is transformed into both brain embeddings and artificial model embeddings of equal dimensionality. From input batches containing several examples, we compute a similarity matrix between brain and model embeddings of different inputs, which is utilized to obtain a transfer loss value. We designate neural-model embedding pairs computed from the same input example as positive pairs, while pairs computed from separate examples are designated negative pairs. (b) Brain latent transfer learning. An input $X$ from the human train set $\mathcal{H}$ is used to produce brain embeddings and artificial embeddings, produced from a model trained with an artificial train set $\mathcal{A}$. These embeddings are subsequently compared with a mean square error distance to obtain a transfer loss.

## Brain Contrastive Loss

One goal of B2M is to maximize the mutual information between neural representations and their counterparts in artificial models. This aims to assist artificial learning by encouraging models to find advantageous sensorimotor representational subspaces that might have been acquired by the human brain over the course of task learning and, ultimately, evolution. We propose to achieve this by adapting contrastive learning via an InfoNCE loss framework, such as the one adopted in SimCLR, [43, 44], which aims to maximize mutual information between similar data points (Fig. 3a).

Given a sensory or contextual input $X \in \mathcal{I}$, with $X \in \mathbb{R}^{d_1 \times d_2 \times \ldots \times d_n}$, where $\mathcal{I}$ is a train set presented to both humans and artificial models, we propose aligning brain and artificial representations of X by maximizing the mutual information between embeddings $Y_{brain} \in \mathbb{R}^E$ and $Y_{model} \in \mathbb{R}^E$, where $E$ is the embedding dimensionality. These embeddings are produced by non-linear transfer functions such that $Y_{brain} = f_{brain}(X)$ and $Y_{model} = f_{model}(X)$, indicating compressed neural and artificial representations of sensorimotor inputs, respectively. Any applicable dimensionality reduction method can be utilized as $f$, or even the identity function, as long as the dimensionality between brain and model embeddings is matched.

For this, we define a batch of $b$ examples $X \in \mathcal{I}$, $[X_1, X_2, ..., X_b]$, and their respective neural and artificial embeddings $B_{neural} = [Y_{(neural,1)}, Y_{(neural,2)}, ..., Y_{(neural,b)}]$ and $B_{model} = [Y_{(model,1)}, Y_{(model,2)}, ..., Y_{(model,b)}]$. From these, given a temperature hyperparameter $\tau$, we obtain a similarity matrix $\mathcal{S} = \frac{1}{\tau}(B_{neural} \times B_{model}^T)$, which represents the similarity between neural and artificial example pairs, including both pairs obtained from the same example $X_i$, but also pairs obtained from different examples. Then, we define positive contrastive pairs $Y_{(neural,i)}$ and $Y_{(model,i)}$, and negative contrastive pairs $Y_{(neural,i)}$ and $Y_{(model,j)}$ for all $i \neq j$. With these, we define, for a given anchor example $i$, with $\mathcal{L}_{transfer} = \sum_i \mathcal{L}_{transfer,i}$:

$$\mathcal{L}_{transfer,i} = -\log \frac{exp(S_{i,i})}{exp(S_{i,i}) + \sum_{i \neq j} exp(S_{i,j})} \tag{2}$$

## Brain Latent Transfer Loss

In decision-making tasks, future states are often dependent on decisions made in past episodes, as well as the outcomes that occurred in them. For this reason, it is potentially challenging to assemble a train set of episodes and environment states for an artificial agent that will exactly match the train sets experienced by human participants in a task, given that the agent is free to act differently from their human counterparts during training. To circumvent this, instead of exact episode matching, we propose matching brain embeddings obtained during exposure to an input $X$ to the artificial embeddings obtained during exposure to the same input $X$, as long as $X$ is an approximation of the examples contained in the artificial train set (Fig. 3b).

Concretely, given a sensory or contextual input $X \in \mathcal{H}$, with $X \in \mathbb{R}^{d_1 \times d_2 \times \ldots \times d_n}$, where $\mathcal{H}$ is the train set presented to humans, we assume brain activity produces an embedding $Y_{brain}$ via a non-linear transfer function $f_{brain}$, such that $Y_{brain} = f_{brain}(X)$, with $Y_{brain} \in \mathbb{R}^E$, in which $E$ is the embedding dimension. Additionally, we assume the artificial model learns from examples $X' \in \mathcal{A}$, $X' \in \mathbb{R}^{d_1 \times d_2 \times \ldots \times d_n}$, where $\mathcal{A}$ is the train set presented to the artificial model, related but not necessarily equal to $\mathcal{H}$. Throughout its learning process, the artificial model finds a non-linear transfer function $f_{model}$ which produces its own embedding $Y'_{model} = f_{model}(X')$, with $Y'_{model} \in \mathbb{R}^E$.

Then, we propose to achieve brain-to-model transfer by performing a latent transfer between brain and model embeddings. Concretely, for an input $X \in \mathcal{H}$ previously presented to humans, we obtain its embedding produced by the artificial model $Y_{model} = f_{model}(X)$ and minimize its mean square error distance to its known corresponding brain embedding $Y_{brain} = f_{brain}(X)$:

$$\mathcal{L}_{transfer} = \frac{1}{E} \sum_{i=1}^{E} (Y_{model,i} - Y_{brain,i})^2 \tag{3}$$

## Memory Task Extended Methods

Invasive neural data was acquired with IRB approval and informed consent, utilizing the Blackrock system. Standard spike sorting was performed with WaveClus [45]. We applied the following pre-processing steps to temporally align neural data and RNN activity: (1) we aligned all spikes to stimulus presentation and created two time periods: pre-stimulus (-1s to 0, resolution: 10ms) and post-stimulus (0s to reaction time). (2) We normalized post-stimulus spikes by reaction time to always fit into a 100-element rate vector, and concatenated pre-stimulus and post-stimulus normalized spikes into a spiking rate vector. (3) We filtered spike rates by convolving the rate vector of each step with a causal exponential filter (kernel: 20 zeros followed by $e^{-0.5x}$, $x = [0, 0.5, 1, ..., 9.5, 10]$).

To learn shared representations across multiple behavioral sessions, we used the Contrastive Embedding by Relative Arrangement (CEBRA) framework [40] (Apache License). CEBRA is a self-supervised learning algorithm that maps high-dimensional neural activity to a low-dimensional space by leveraging temporal structure and optional contextual supervision. Below, we describe the full procedure used to generate multi-session embeddings from neural recordings across 17 behavioral sessions.

We aggregated neural activity from all neurons in the 17 distinct recording sessions. Each session contained neural population firing rates stored as 3D arrays with shape $(N_{steps}, N_{times}, N_{neurons})$, where $N_{steps}$ represents the number of unique stimuli presented to a participant, considering each stimulus constitutes a step. For each session, data were reshaped into 2D arrays of shape $(N_{steps} \times N_{times}, N_{neurons})$ and paired with time labels repeated across trials. These reshaped arrays represent temporally ordered sequences of neural activity and serve as the input to CEBRA.

We instantiated a CEBRA model configured for multi-session contrastive learning. The following hyperparameters were used: model architecture: offset10-model, batch size: 512, learning rate: $3 \cdot 10^{-4}$, temperature mode: auto with a minimum temperature of 0.1, embedding dimensionality: 7, maximum training iterations: 15,000, distance metric: cosine similarity, supervision: conditional sampling based on relative time (i.e., time-delta), utilizing time within step as a supervising feature.

## Driving Task Extended Methods

Healthy adults (N=11 sessions in 9 participants, with written informed consent, IRB approved) completed a boundary-avoidance driving task inside a virtual-city environment rendered through an HTC Vive Pro Eye headset. Seated at a Logitech G920 steering wheel with accelerator and brake pedals, participants piloted a virtual car while continuous “fog” opacity dynamically modulated visual uncertainty on a trial-by-trial staircase. Crashes with road boundaries incurred point penalties to encourage timely, accurate steering. Scalp EEG was recorded throughout with a 64-channel BioSemi ActiveTwo system (Ag/AgCl active electrodes, international 10–20 system, $impedances < 50k\Omega$) at 2048Hz; a lossless screen-capture of the VR scene was recorded via the Unity engine; steering-wheel position, pedal inputs, and headset-embedded eye-tracking data were time-synchronized with the EEG stream for later source and connectivity analyses.

Additionally, the VAE model utilized in for visual scene reconstruction in this task consists of an encoder-embedding-decoder architecture (Fig. 2b), that reconstructs input target urban scenes into matching outputs. These scenes were previously obtained in the CARLA driving simulator environment [46] and made publicly available [42], with 12000 preset training images and 2000 test images, which are examples of driving scenes. We changed the embedding dimensionality from the original VAE to 64 dimensions, to directly match the 64 channels recorded in EEG sessions. The encoder and decoder each contain 5 convolutional/leaky ReLU layers and one fully connected layer (encoder dimensions: $[3, 32, 64, 128, 256]$, decoder dimensions: $[256, 128, 64, 32, 3]$, fully connected dimensions: $1024.$). We also adapted the original model to include dropouts ($p = 0.1$) in each convolutional and fully connected layer. The last convolutional encoder layer also contains a batch normalization step. Training visual scenes to be reconstructed were presented in batches (batch size: 32) of $160 \times 80$ pixel images.

# Limitations

The observed improvements in learning performance could still be partially attributable to a regularization effect introduced by structured noise in the neural data, rather than reflecting brain information transfer alone. Future work should explore more refined controlled ablation studies (e.g., using permuted or synthetic neural data with matched noise statistics to disentangle representational transfer from implicit regularization effects).

In a small subset of training runs, Brain Latent Transfer produced instability, occasionally resulting in catastrophic model divergence. This could potentially arise when the neural training data and model training data differ substantially in input distribution, potentially exposing the model to conflicting or out-of-distribution (OOD) latent signals. Addressing this challenge may require more robust alignment techniques, such as OOD detection or reweighting schemes to reconcile mismatched training domains. Additionally, systematic benchmarking across architectures and input domains will be essential for assessing B2M's scalability.

Furthermore, the benefits of B2M were tested on RNNs and VAEs. It remains unclear whether these findings extend to other architectures, such as reinforcement learning agents, or task modalities beyond vision and memory-based decision making. Further work must be done to establish extended generalization. Systematic benchmarking across architectures and input domains will be essential for assessing B2M's scalability.

Finally, while B2M improves testing performance, it is not yet determined what portions of information are being transferred and whether the brain-derived embeddings encode high-level abstractions, low-level features, or task-specific biases. Developing tools to interpret and visualize the aligned latent spaces will be useful for understanding the semantic content of the transfer signal.

# Ethical Considerations

Despite its promise, this line of work introduces important ethical considerations. Critically, the use of human brain data for training artificial models raises issues of privacy, consent, and data stewardship. All neural data used in this study were anonymized and collected under IRB approval with informed consent, and care must be taken that these standards are upheld even if the economic viability of B2M in large-scale projects is demonstrated.

Additionally, techniques that align AI systems with neural representations could, in theory, be misused in contexts such as surveillance or cognitive-behavioral manipulation. Although we do not release pretrained models, any future release will include usage terms to prevent misuse in the context of human participants protection. We encourage the community to proactively discuss ethical governance frameworks and emphasize full transparency, human participants protection, consent, and user autonomy in downstream applications. Additionally, care must be taken that human participants are compensated fairly for the data they provide for building better models, which could ultimately provide significant economic potential at scale.

## Computer Resources

All experiments were performed on a Lambda Labs server, with the following characteristics: 192 CPUs, AMD Ryzen Threadripper PRO 7995WX 96-Cores, 5390MHz (maximum), graphics card: NVIDIA Corporation AD102GL [RTX 6000 Ada Generation], 512GB RAM, 18 TB HD storage.

On this machine, the 220 memory task runs (RNN: B2M and noise) took 14.16h in total, whereas the 220 driving task runs (VAE: B2M and noise) took 113.59h in total.

## Human Participants

For the memory-based decision-making task, we collected intracranial data from 6 human epilepsy patients, undergoing seizure monitoring prior to surgery. Patients read the following text on the screen, substituting target-stim1 and target-stim2 for each episode with the actual target stimuli for that episode:

A scientist is studying different patterns. Your job is to help him. Objects will appear one after another, and he wants you to take a picture when you spot a particular pattern in their sequence. (For example, a Vase followed by a Flower.) We'll always tell you what pattern to look for. Every few objects, there will be a new pattern to watch for. When you spot the pattern, press your photograph button to take a picture. Otherwise, press the appropriate button to skip that object. We'll tell you which button is which on each trial. The objects can appear in any order, including several of the same type in a row. A new object will appear every two seconds or so. Press 'LEFT' to take a picture, and press 'RIGHT' to skip this object. Wait for the first [target-stim1], then take a picture of it. Then wait for the first [target-stim2], and take a picture of it. Alternate taking pictures of one [target-stim1] and one [target-stim2].

For the VR driving task, we collected EEG data from 9 healthy human participants. They had prior driving experience, normal or corrected-to-normal vision, and reported they were not prone to motion sickness. Participants were verbally instructed to drive a car along a road in VR for a fixed amount of time, avoiding collisions. They were told that collisions would deduct an amount of money from their total reward bonus, which was displayed on the car's dashboard. They could accelerate, brake, and steer with realistic input controls. Participants were compensated at a rate of 20 USD/h for 3 hours. EEG contact positioning was the same for all participants, as displayed in Fig. 4

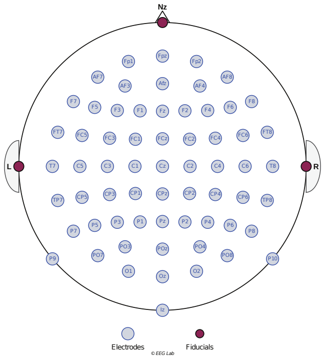

Figure 4: EEG standard contact map. All participants in the VR driving task underwent EEG recording, with electrodes positioned along the same standard contact grid. Nz indicates the direction of the front of the head.