PyTorch CONFERENCE EUROPE 2026

Bridging the Gap: Engineering Compliant “Glass Box” Medical AI

Muhammad Saqib Hussain

Muhammad Saqib Hussain

Background: 4th-Year Medical Student (M.D. Candidate) — Bratislava, Slovakia.

Clinical Focus: Building AI systems that augment, rather than replace, clinical expertise.

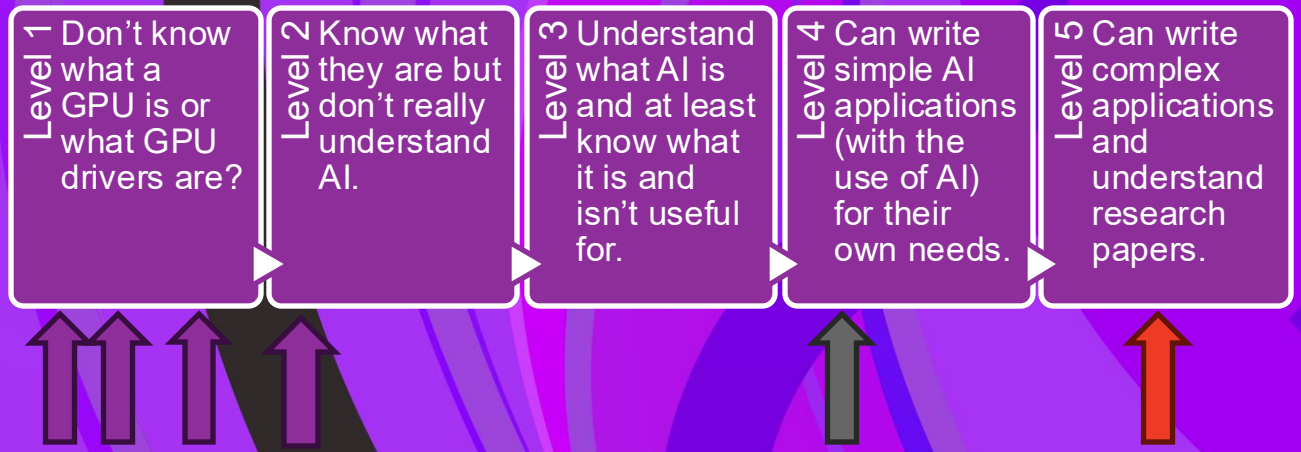
Technical Background: Self-taught in AI and ML.

Honored to be recognized and supported by the Pytorch and Linux Foundation to share this cross-disciplinary work today.

What I will talk about today

- **The Clinical Dilemma:** Why the EU AI Act makes "Black Box" medical AI un-deployable.
- **The Architecture:** A brief look at NeuroBOLT (translating 1D EEG to 4D fMRI).
- **The Glass Box Engine:** Using PyTorch and Captum (`IntegratedGradients`) to audit the model.
- **The Clinical Proof:** Mapping mathematical feature attribution back to human anatomy.
- **The Next Step:** Engineering actionable dashboards for "Human-in-the-Loop" compliance.

A Quick Question about our users first



The Results do not look good.

The BlackBox problem in Healthcare

State-of-the-art models now achieve mathematical excellence.
The problem is that they are clinical black boxes.

(47) AI systems could have an adverse impact on the health and safety of persons, in particular when such systems operate as safety components of products. Consistent with the objectives of Union harmonisation legislation to facilitate the free movement of products in the internal market and to ensure that only safe and otherwise compliant products find their way into the market, it is important that the safety risks that may be generated by a product as a whole due to its digital components, including AI systems, are duly prevented and mitigated. For instance, increasingly autonomous robots, whether in the context of manufacturing or personal assistance and care should be able to safely operate and performs their functions in complex environments. Similarly, in the health sector where the stakes for life and health are particularly high, increasingly sophisticated diagnostics systems and systems supporting human decisions should be reliable and accurate.

Article 13

Transparency and provision of information to deployers

1. High-risk AI systems shall be designed and developed in such a way as to ensure that their operation is sufficiently transparent to enable deployers to interpret a system's output and use it appropriately. An appropriate type and degree of transparency shall be ensured with a view to achieving compliance with the relevant obligations of the provider and deployer set out in Section 3.

Article 14

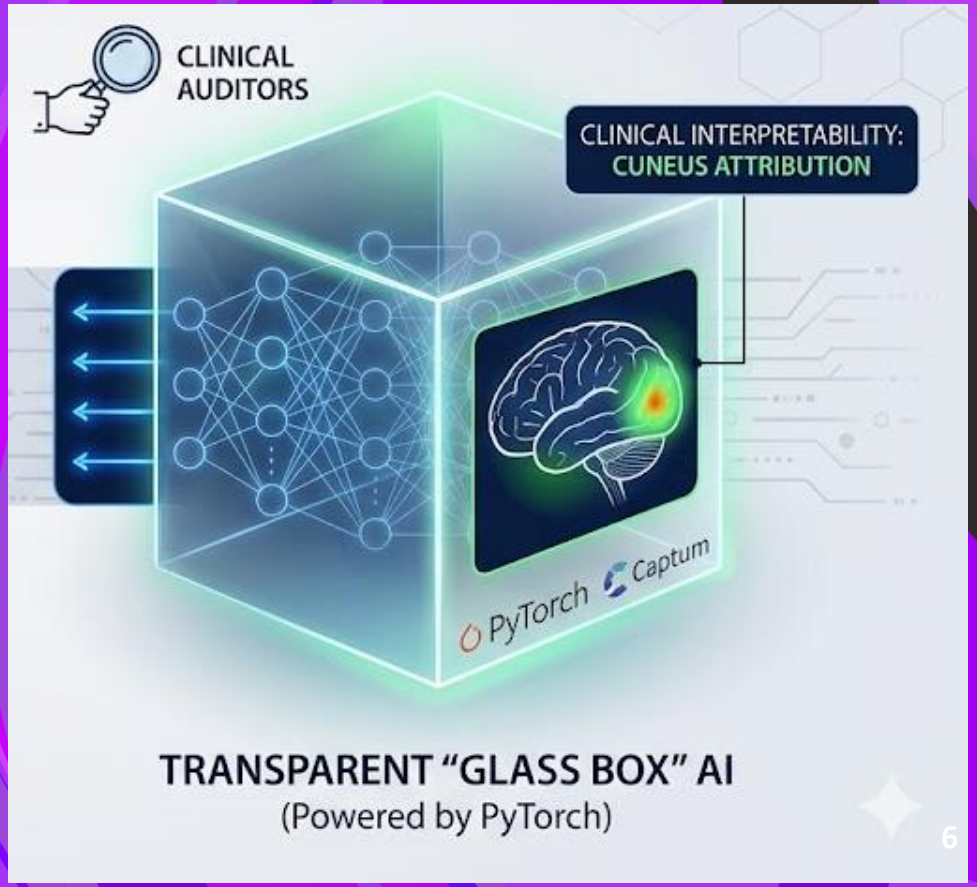
Human oversight

1. High-risk AI systems shall be designed and developed in such a way, including with appropriate human-machine interface tools, that they can be effectively overseen by natural persons during the period in which they are in use.

- EU AI Act

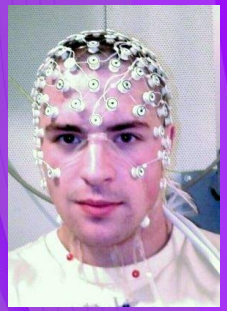
The Pytorch Glassbox Solution

- **The Goal:** Clinically-explainable, auditable AI ("Glass Boxes").
- **The Engine:** PyTorch's **Captum** (`IntegratedGradients`).
- **The Output:** Mapping deep learning math back to physical human anatomy.

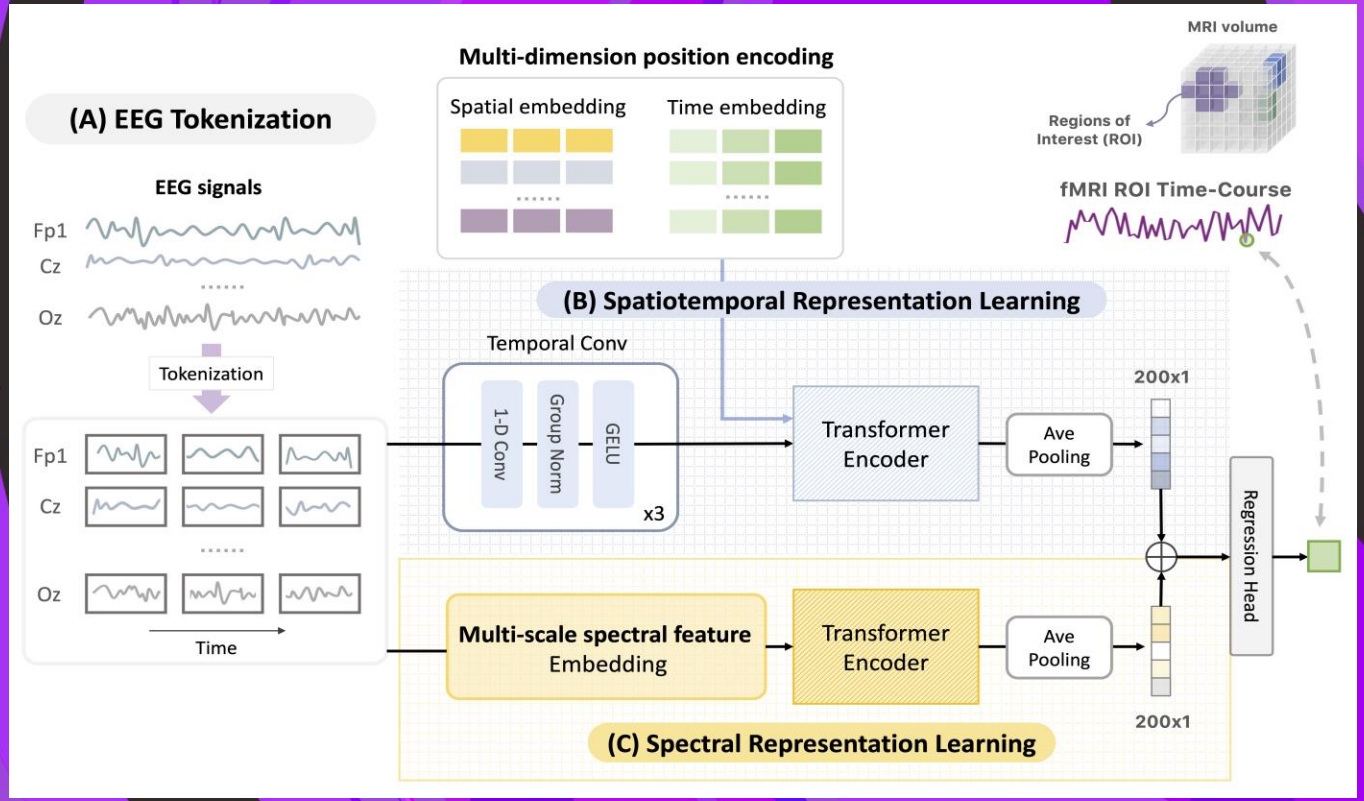


EEG Signals → Model → Predictions → IntegratedGradients

First off, Our Model: NeuroBOLT



Electrodes on a subject and the EEG readings



https://soupeeli.github.io/NeuroBOLT/

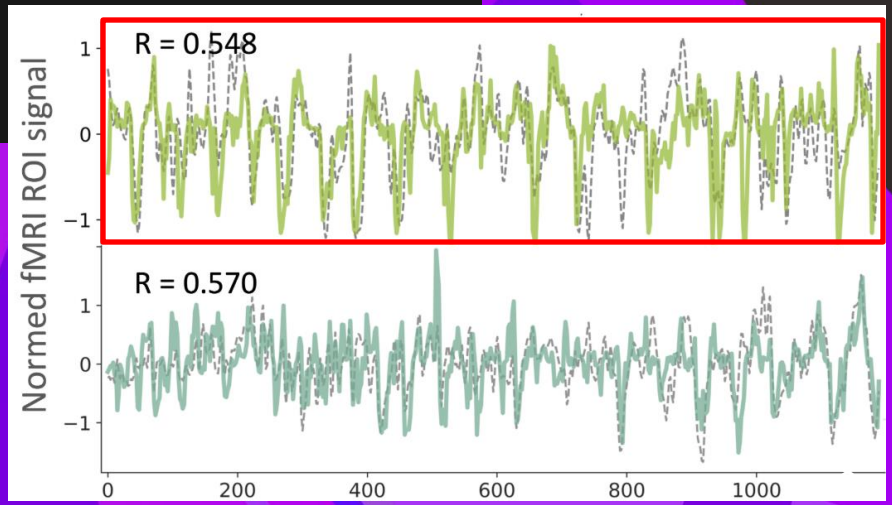
Predictions

```python
model = NeuroBOLTransformer(
    EEG_length=TIME_STEPS, EEG_channel=EEG_CHANNELS,
    patch_size=PATCH_SIZE, num_roi=ROI_COUNT, init_values=0.0
)
```

```python
checkpoint_path = hf_hub_download(repo_id="ssssssup/NeuroBOLT", filename="checkpoints/Cuneus.pth")
checkpoint = torch.load(checkpoint_path, map_location='cpu', weights_only=False)
state_dict = checkpoint.get('model_state_dict', checkpoint.get('model', checkpoint))
model.load_state_dict(state_dict, strict=False)
model.eval()

with torch.no_grad():
    prediction = model(eeg_input, input_chans=input_chans)
    pred_val = prediction[0, 0].item()
```

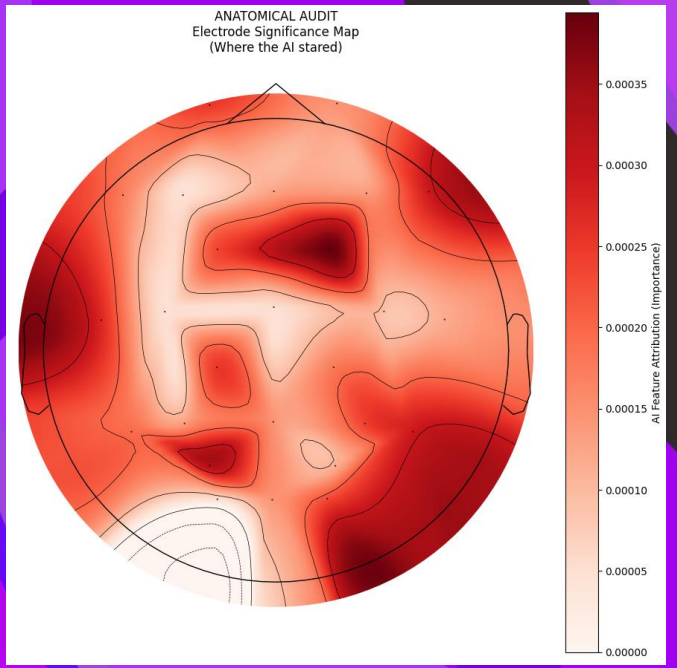
```
pred_val:
0.513
0.010
-0.327
0.020
```



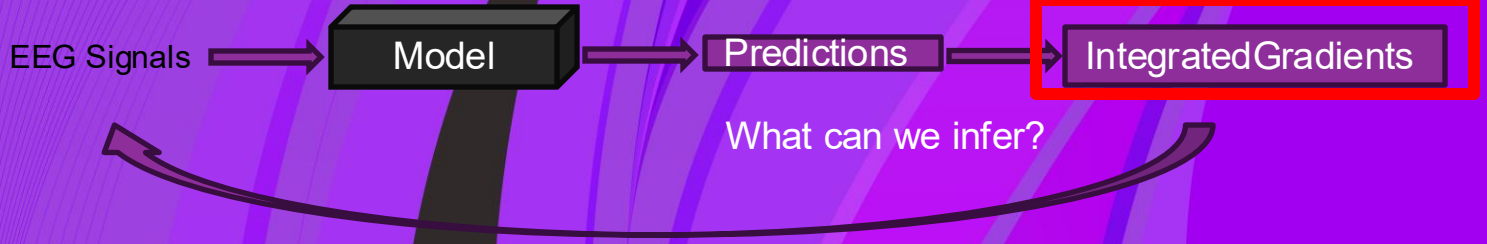

```python
ig = IntegratedGradients(model)
# internal_batch_size=1 prevents the tensor mismatch error
attr = ig.attribute(
    inputs=eeg_input, target=0,
    additional_forward_args=(input_chans,),
    n_steps=10, internal_batch_size=1
)
```

```python
channel_importance = np.mean(np.abs(attr_data), axis=1)
im, _ = mne.viz.plot_topomap(channel_importance, info, axes=ax2, show=False, cmap='Reds')
```

```
C3: 0.0002100
Cz: 0.0002275
C4: 0.0002100
T8: 0.0001750
CP1: 0.0002100
CP2: 0.0002100
P7: 0.0001750
P3: 0.0002100
Pz: 0.0002275
P4: 0.0002100
P8: 0.0001750
PO3: 0.0002800
PO4: 0.0002800
O1: 0.0003325
Oz: 0.0003325
O2: 0.0003325
```



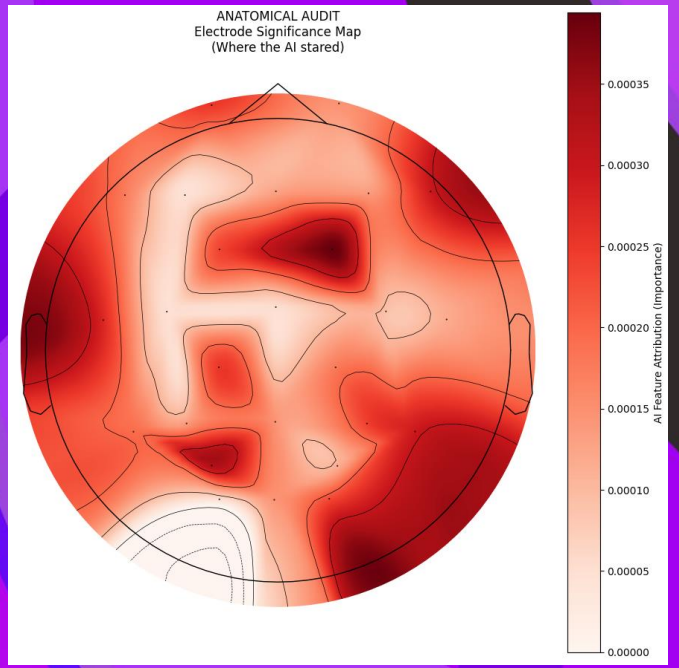
PyTorch CONFERENCE EUROPE 2026



The Expected: High attention on the occipital electrodes (aligning with Cuneus anatomy).

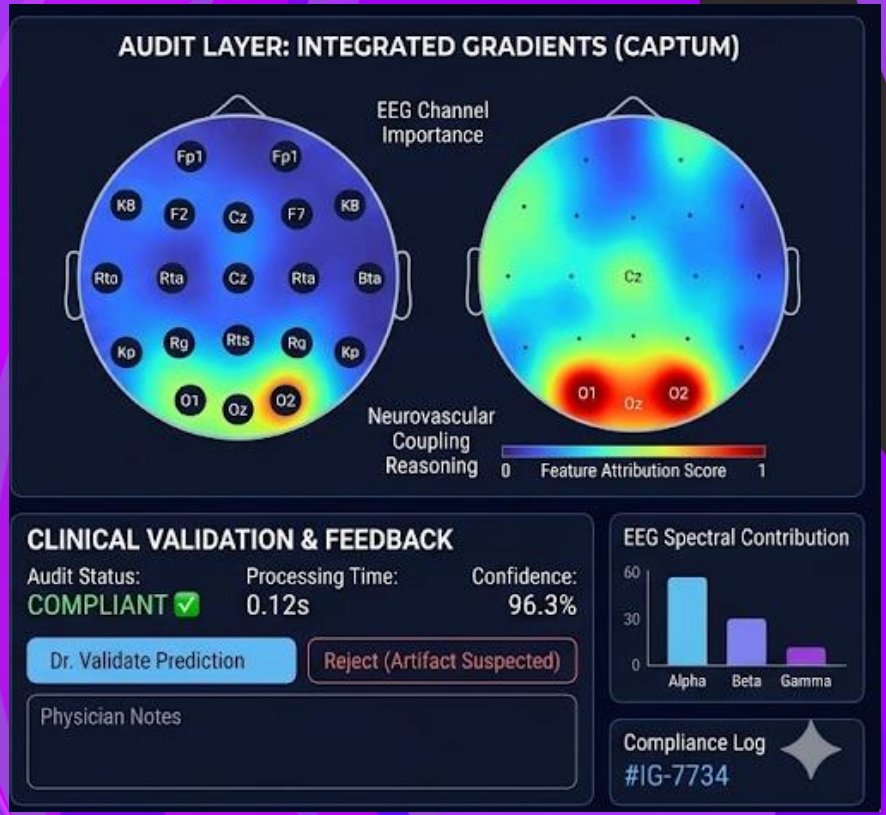
The Unexpected: Significant focus on temporal and frontal regions.
Hypothesis A (Biology): The model is successfully tracking connected visual and attention neural networks.
Hypothesis B (Artifact): The model is "cheating" by using frontal eye blinks as a proxy for visual activity.

The Verdict: This ambiguity is exactly why clinical "Glass Boxes" and human oversight are mandatory.



What is missing?

1. **Clinician-in-the-Loop:** Active clinical validation by neurologists, not just engineers/researchers.
2. **Data Transparency:** Clear clinician visibility into training data distributions and biases.
3. **Actionable Clinical UI:** Translating backend math (Integrated Gradients) into intuitive, visual dashboards, for example with streamlit.



Thank you!

The code is available at:
https://github.com/clinexplain/MVP

The code for NeuroBOLT is available at
https://soupeeli.github.io/NeuroBOLT/

GITHUB

PyTorch CONFERENCE EUROPE 2026