TraceFL: Interpretability-Driven Debugging in Federated Learning via Neuron Provenance

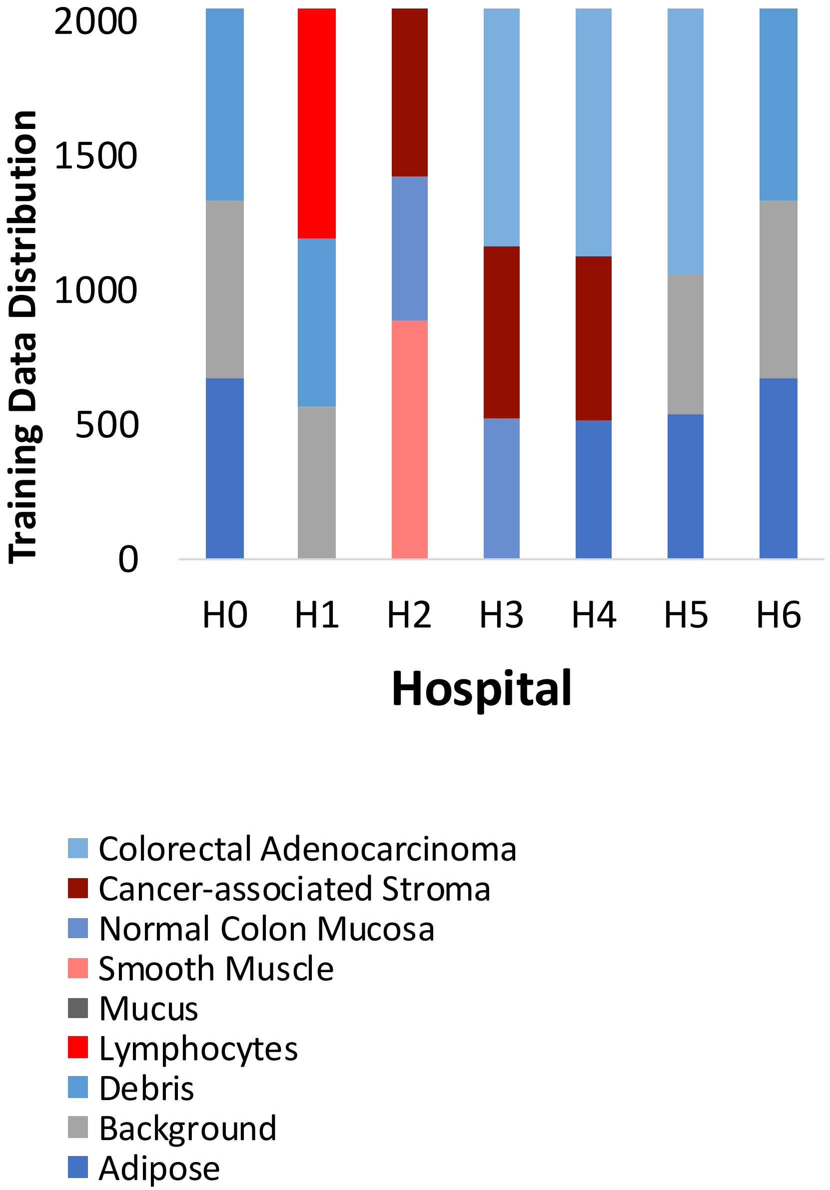
Waris Gill[1], Ali Anwar[2], Muhmmad Ali Gulzar[1]

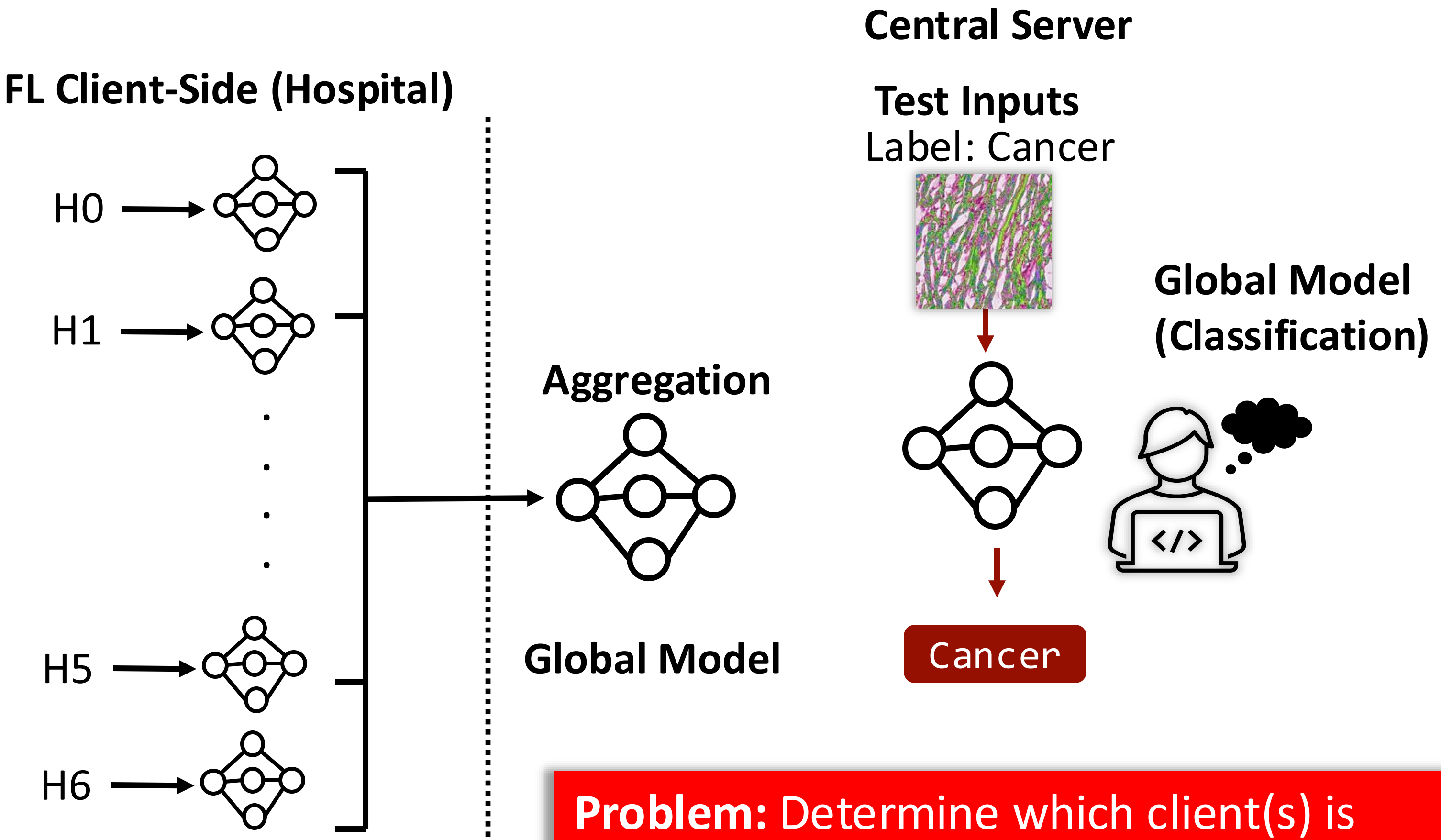
1 VIRGINIA TECH

2 UNIVERSITY OF MINNESOTA

Flower Framework

How can we interpret FL global model output?



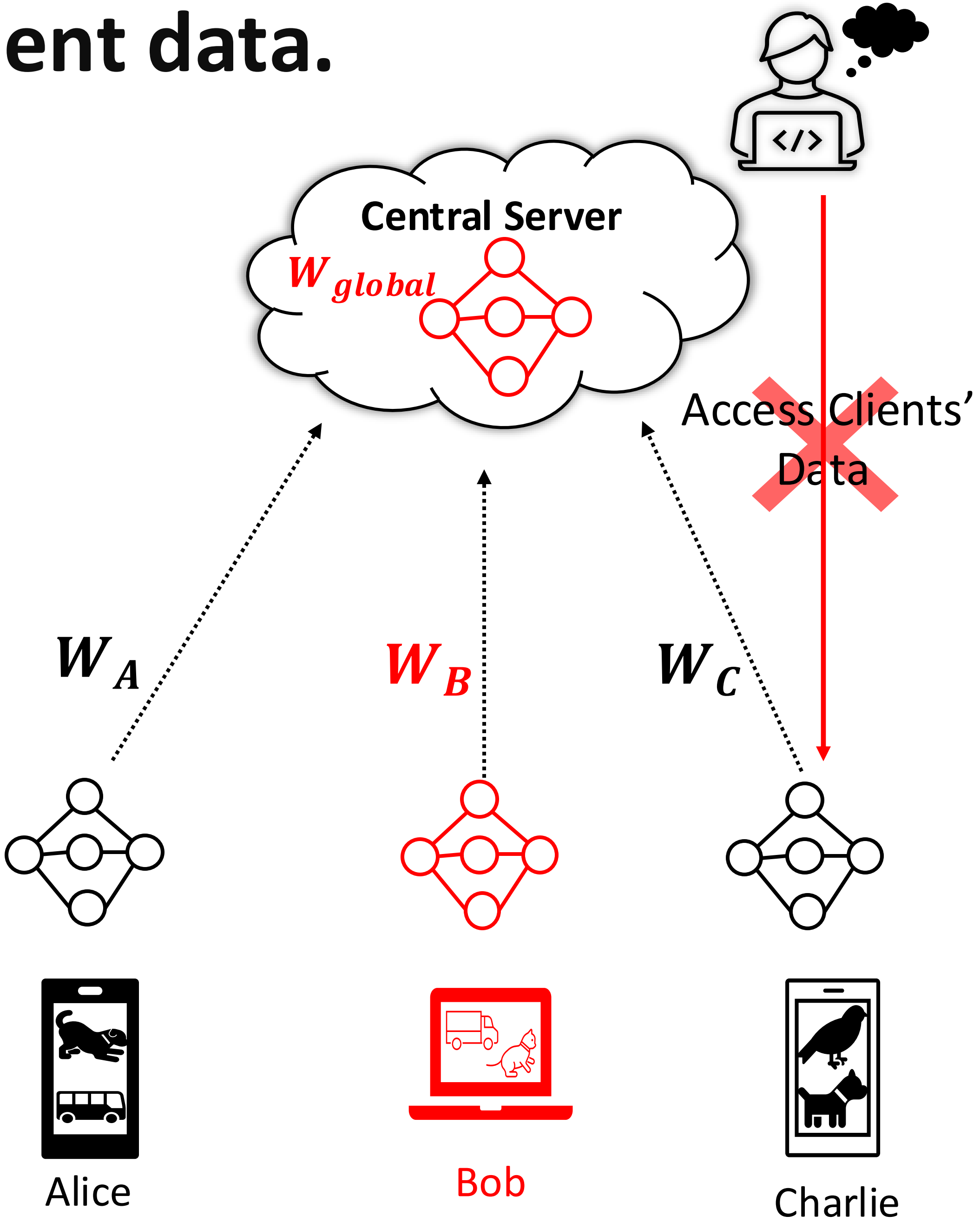


Problem: Determine which client(s) is primarily responsible for the global model output (cancer)?

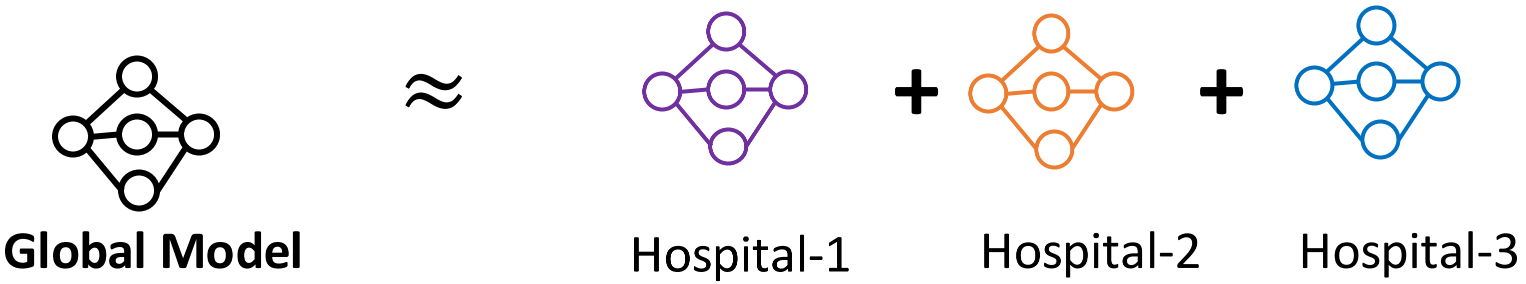
No prior solution is available.

Challenge 1: No direct access to client data.

A **developer** at the central server **cannot access client data** due to FL privacy principles, making it difficult to identify faulty clients before aggregation.

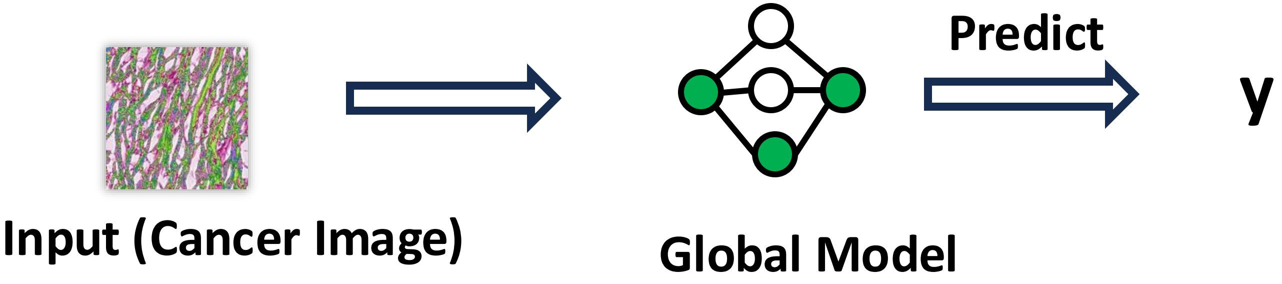


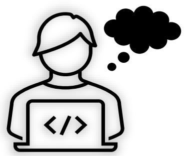
Challenge 2: FL Global Model is Not Directly Trained on Data



Global model is a mixture of many clients' models.

Suppose during production on an **input image**, global model predicts **y**.

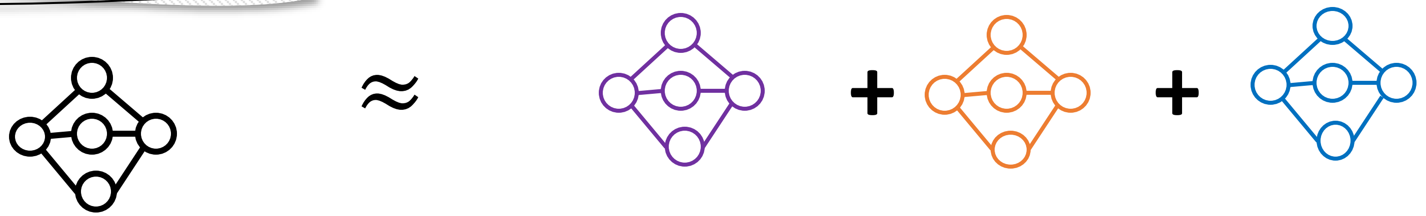




Identifying which client or group of clients caused specific model behaviors (**y**) is difficult.

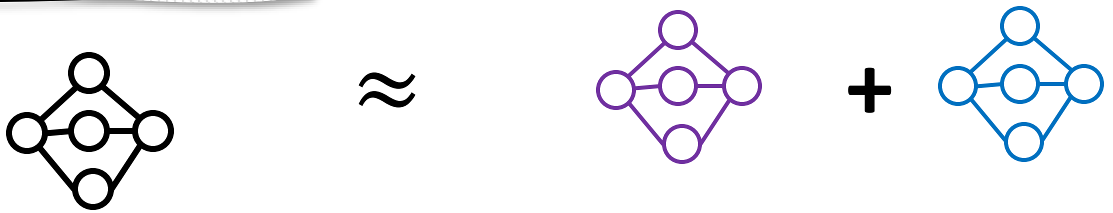
Challenge 3: Clients may not participate in every FL round

FL Training: Round 21



Global Model of Round 21 ≈ Hospital-1 + Hospital-2 + Hospital-3

FL Training: Round 22



Global Model of Round 22 ≈ Hospital-1 + Hospital-3

Hospital-2 is not participating in Round 22.

Possible Reasons:

- Connectivity Issues
- Sometimes clients are randomly sampled in each FL round

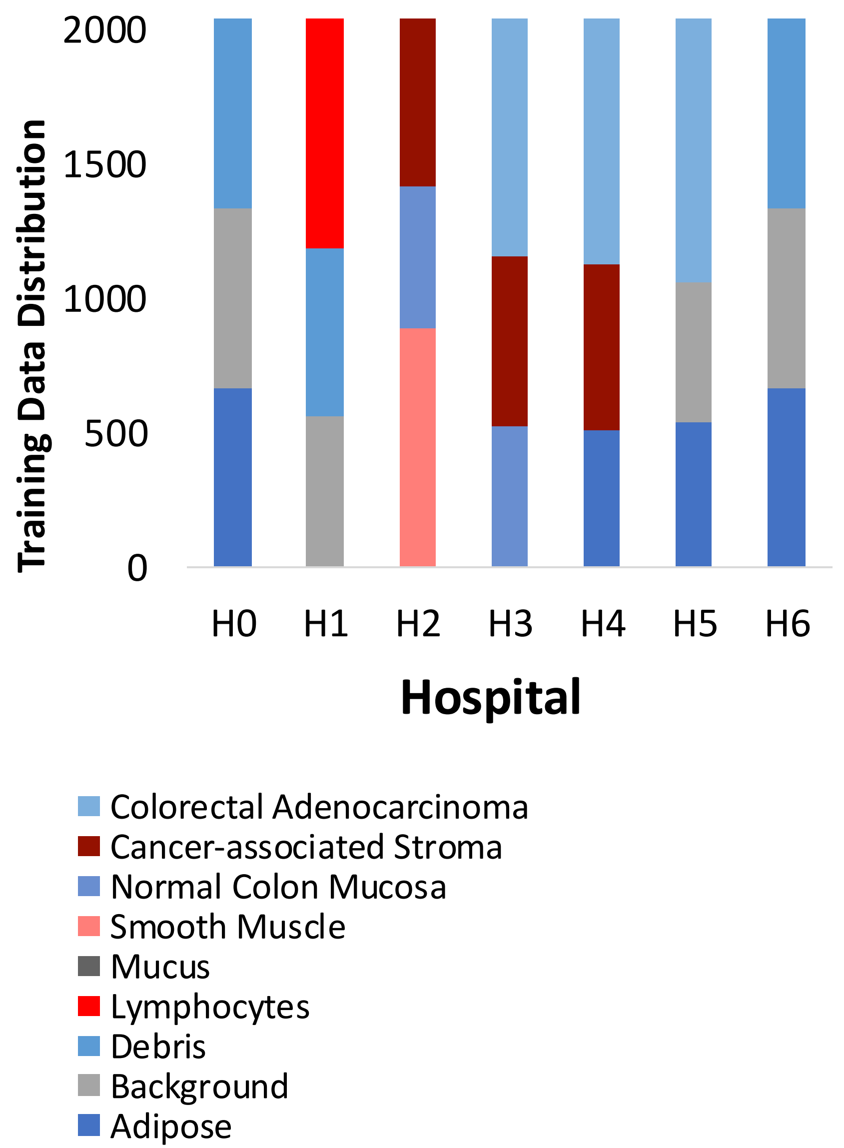
Challenge 4: Clients have Heterogeneous Data Distributions

FL clients have highly diverse and imbalanced data distributions.

- Unequal Data Quantity
- Unequal Labels Distribution

Example: Hospitals 2, 3, 4 have cancer-associated stroma.

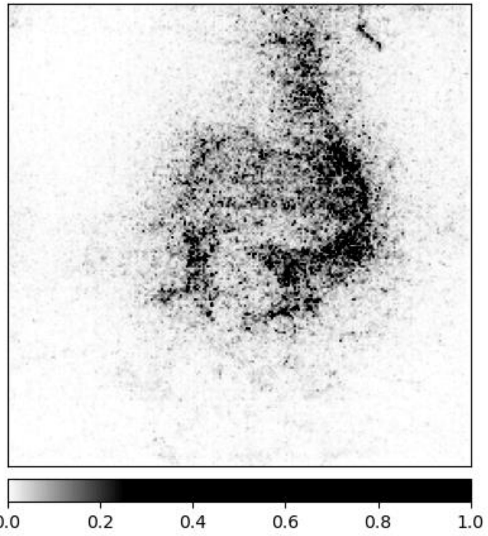
Heterogeneous client data makes it difficult to interpret client contributions and debug prediction errors.



Challenge 5: ML Interpretability Methods Are Inadequate

- **Traditional ML interpretability Methods are not feasible for FL.**
  - **Vision tasks:**
    - Integrated Gradients, Gradient Shap, Occlusion, and LRP **focus on pixel importance.**

Input to ML Model



Pixels Attribution

But our goal is to determine **clients' importance** contributed to specific predictions.

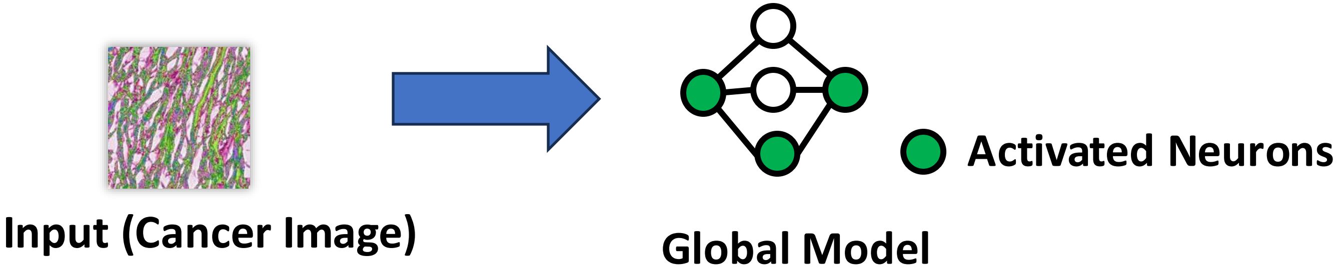
- Thus, traditional ML Interpretability methods are incompatible with FL interpretability problem. FL needs privacy-preserving alternatives for effective debugging and interpretability.
- It is an **open challenge** in FL **(Kairouz et al., 2021)**.

How can we design **debugging** and **interpretability** techniques **for FL,** given the challenges?

Kairouz, Peter, et al. "Advances and open problems in federated learning." *Foundations and trends® in machine learning* 14.1–2 (2021)

TraceFL (Dynamic Neuron-Level Provenance) ICSE 2025

- **Key Idea:** Trace **neuron-level contributions** in the global model to **individual clients** for the given input.



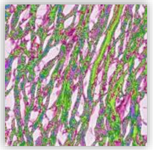
<u>**High-Level Steps of TraceFL:**</u>

1. Identify Activated Neurons in the Global Model
2. Use Gradients to find Influential Neurons
3. Map Client Contributions in an Activated Neuron
4. Rank Clients by Total Contribution

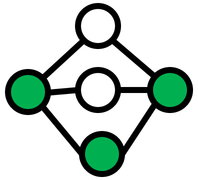
TraceFL recovers **how much each client influenced global neuron outputs**, providing interpretable insights.

Step 1: Identify Neurons Activated by the Input

- Consider **ReLU** ($z = \max(0, \boldsymbol{w_g} \ . \ \boldsymbol{x})$) as activation function in a neuron, where $\boldsymbol{w_g}$ is the global neuron weights and $\boldsymbol{x}$ is the input to the global neuron.



Input (Cancer Image)



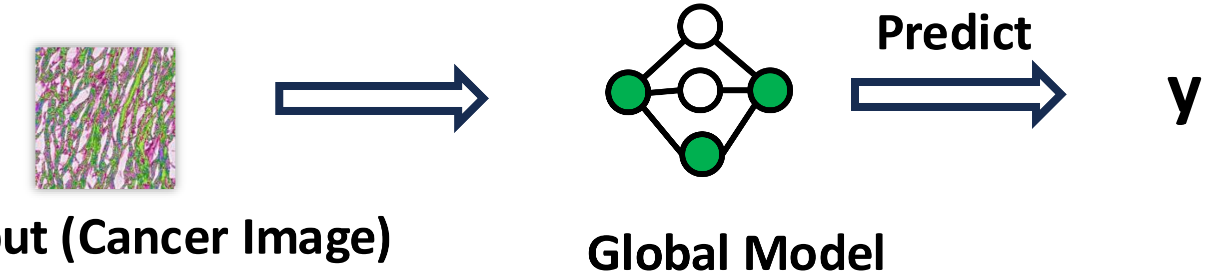
Global Model

○ z = 0, **Inactive Neuron**

● z> 0, **Active Neuron**

Benefit: Focus only on relevant neurons while tracing clients and avoid irrelevant attributions.

Step 2: Influential Neurons via Gradients

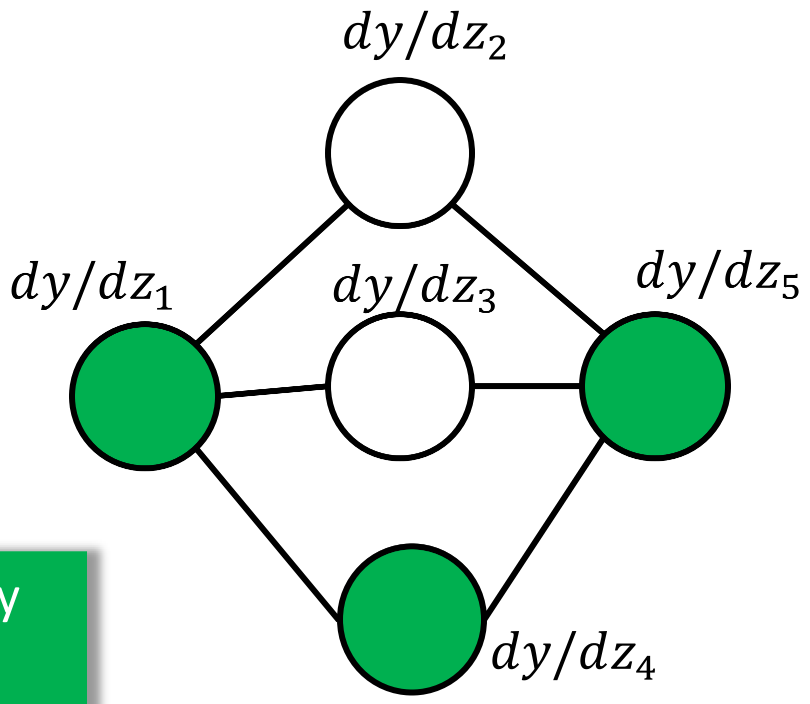


○ z = 0, **Inactive Neuron**

● z> 0, **Active Neuron**

Compute gradient dy/dz_j for each neuron output (z_j) in the global model for given prediction **y**.

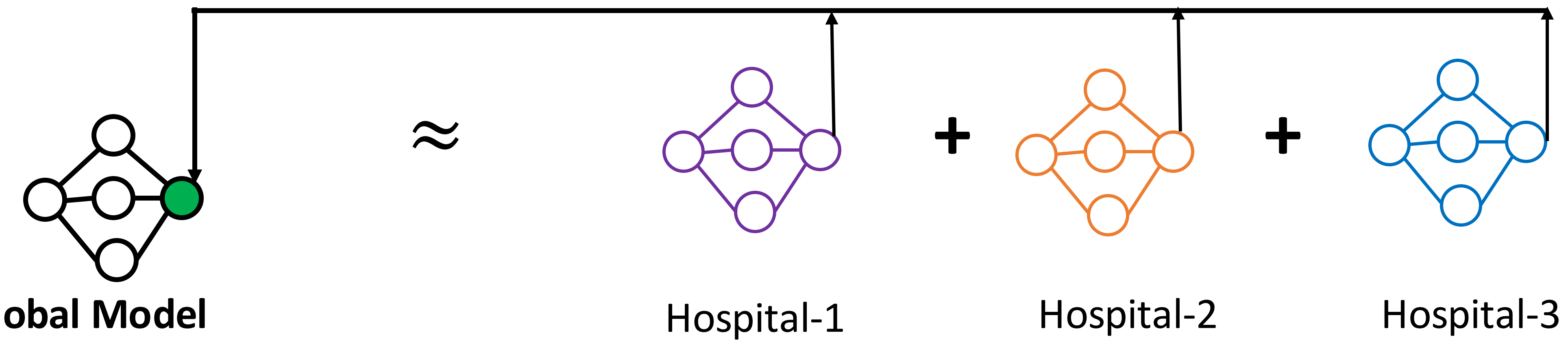
<u>**Insight:**</u> **Neurons with large gradients** (dy/dz_j) significantly influence the prediction. **Neurons with small or zero gradients** have minimal impact.



Global Model

Step 3: Map Client Contributions in an Activated Neuron

$\boldsymbol{w_g}$ is the **aggregation** of the **corresponding clients' neuron weights.**



- **Formally (**ignoring data distribution constant**):**

$$\boldsymbol{w_g} \approx \boldsymbol{w_{h1}} + \mathbf{w_{h2}} + \mathbf{w_{h3}}$$

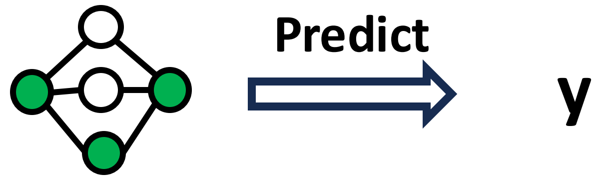
- Suppose **gradient** computed in previous step for **this global neuron** is: $\nabla = \ dy/dz$

- Then, contribution of the **hospital -1** in a **global neuron (**$\boldsymbol{n_j}$**)** is: $t_{h1\_n_j}$ = $\boldsymbol{w_{h1}} . \boldsymbol{x^T}$ x ∇

<u>Key Insight:</u> If **gradient (∇) is large**, neuron strongly impact the final prediction, **increasing the client's partial contribution**. If **∇=0**, it will **ignore** the provenance for that neuron.

Step 4: Rank Clients by total Contribution

- **Total contribution of Hospital-1** in a prediction (**y**) **by the global model** is:



$$For\ Hospital-1{:}\ \ T_{H1} = \ t_{h1\_n_1} + \ t_{h1\_n_2} + t_{h1\_n_3} + \ 0 + 0$$

- Similarly, we can compute the contributions $T_{H2}\ \ and\ \ T_{H3}$ for hospitals 2 and 3.

- **Normalize Attributions:** $\boldsymbol{T_{rank}} = Softmax\ ([T_{H1} + T_{H2} + T_{H3}])$

<u>Key Insight :</u> This step **aggregates client contributions** across all active neurons, providing an overall “responsibility score” for each client.

The **top-ranked client(s), in $\boldsymbol{T_{rank}}$** , are the most significant contributors to the global model’s decision.

Evaluations: General Description about datasets and Models

Datasets

- **Image Classification**
  - CIFAR-10 (10 Classes)
  - MNIST (10 Classes)
- **Medical Imaging**
  - Colon Pathology (9 Classes)
  - Abdominal CT (11 Classes)
- **Text Classification**
  - DBpedia (14 Classes)
  - Yahoo Answers (10 Classes)

Models

- **CNNs** (Image)
  - ResNet
  - DenseNet
- **Transformer** (Text)
  - GPT
  - BERT

FL Clients

- **Client Scaling:** Up to 1000 clients.
- **Sampling Per Round:** 10-50 clients randomly sampled.

Data Distribution Among Clients

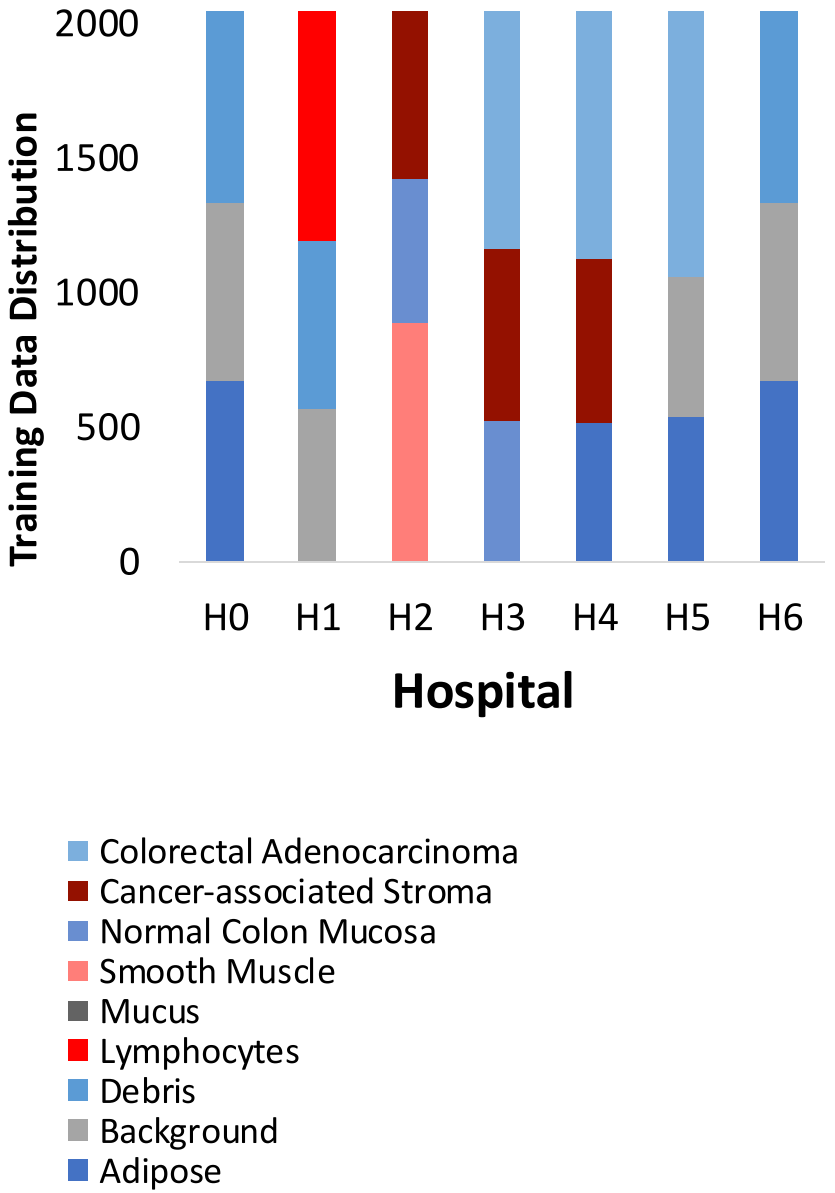
- **Dirichlet Distribution**
  - Commonly used to simulate **non-IID** client data in FL.
- **Default Setting**
  - $\alpha = 0.5$: Standard non-IID configuration.
- **Challenging Setting**
  - $\alpha = 0.3$: Evaluates TraceFL in difficult settings.
- **Stress Test**
  - Vary α from **0.1 to 1** to assess TraceFL's robustness across diverse data

Such combinations of datasets, models, clients, and data distribution settings are rarely seen in existing FL research.

Localization Accuracy

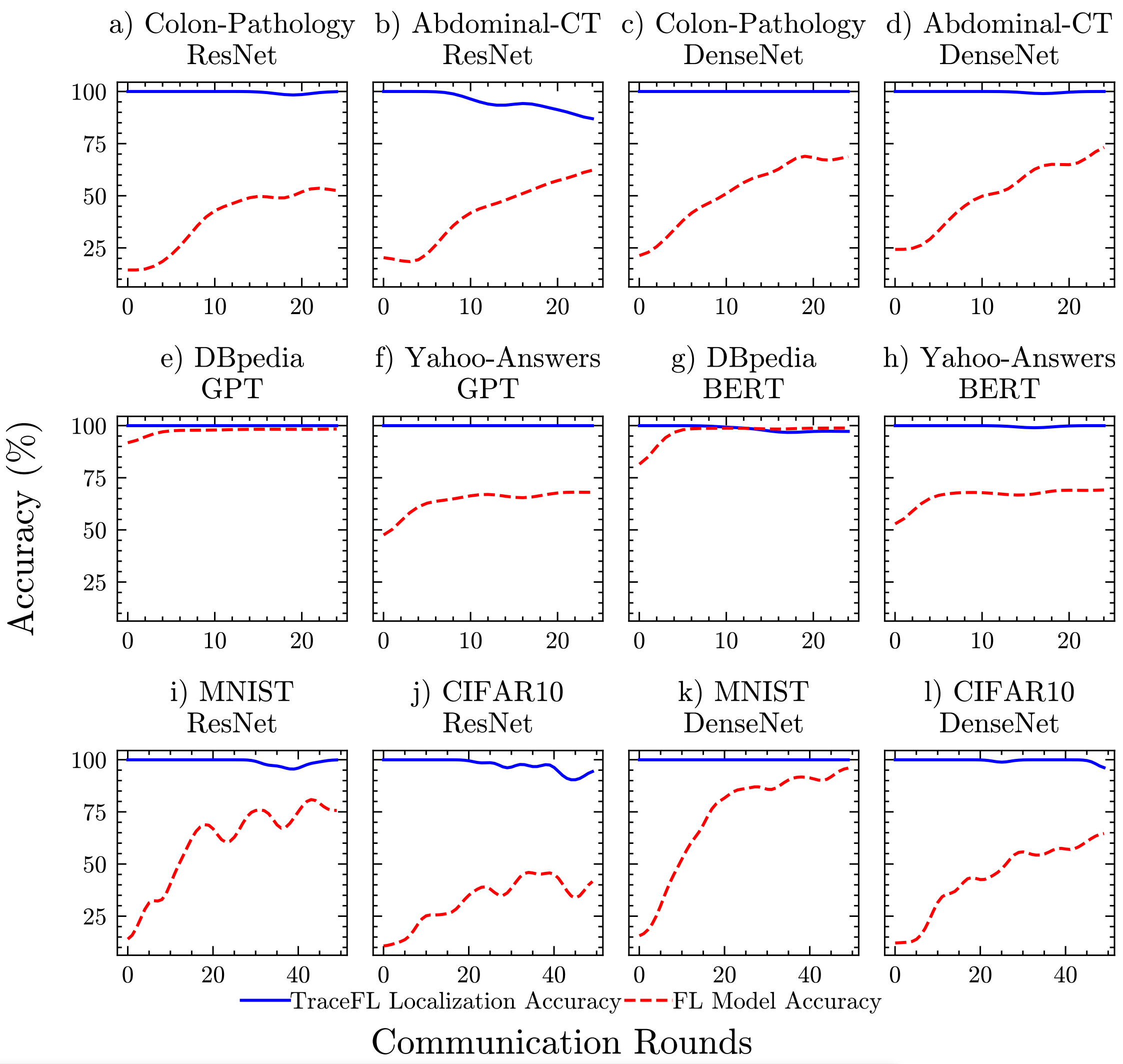
- Given the **z number of test inputs** to the **global model**, if **TraceFL** accurately locates **m times** the **clients responsible** for the **the predictions** then:

$$Localization\ Accuracy = \frac{m * 100}{z}$$



Result 1: 12 FL Configurations (400 FL Rounds)

We include **FL Model Accuracy** to demonstrate training progression, improve with more rounds, and **help calibrate neuron provenance results**.



TraceFL Performance Summary

- **Image Classification:** 98.96% Localization Accuracy
- **Text Classification:** 99% Localization Accuracy

Slight Variation in Resnet. ResNet's simpler architecture may lead to neurons learning less robust features, impacting global model performance compared to DenseNet.

Takeaway: TraceFL is effective for both CNNs and Transformers, performing well on real-world medical imaging and text datasets, and sustaining high accuracy throughout FL training rounds.

Result 2: TraceFL with Differential Privacy enabled FL

DP in FL (McMahan et al., 2018) **adds noise** to the **weights of a model** to protect against stealing or recovering the individual training data points.

GPT and DBpedia FL configuration

FL model's accuracy decreases when the DP noise increases and vice versa.

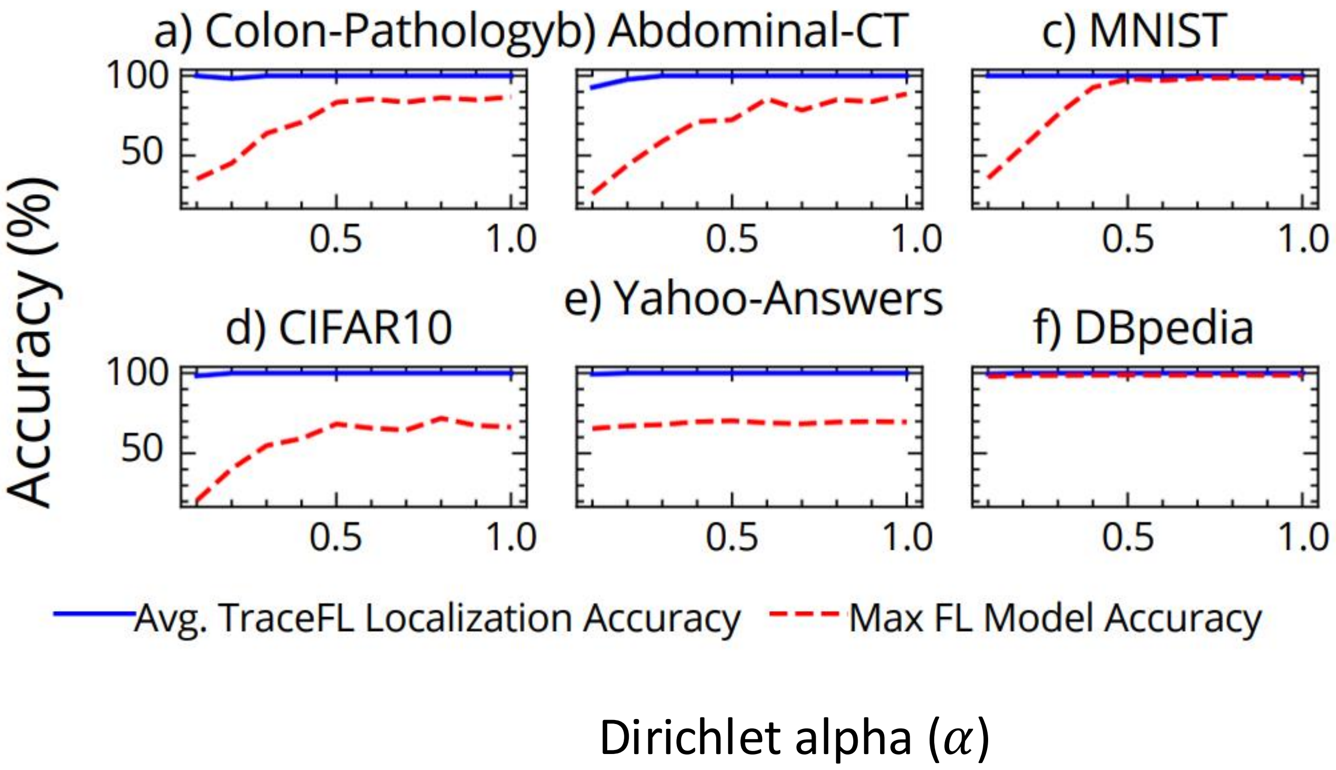
DP Noise	DP Sensitivity	FL Model Accuracy	TraceFL Localization Accuracy
0.003	15	97.36 %	100 %
0.006	10	97.90 %	100 %
0.012	15	**88.81 %**	100 %

Note: TraceFL does not recover the individual clients' data points. It only identifies the responsible clients in ranked order.

Takeaway: TraceFL works with DP enabled FL. DP adds noise to neurons and TraceFL works at neuron level which makes it effective even with DP.

Result 3: TraceFL with Varying Data Distribution

- **Different data distributions** among clients can impact the FL training process.
- To evaluate **TraceFL robustness**, we vary **Dirichlet alpha (α)** from 0.1 (highly challenging scenario) to 1.
- We can see that FL model Accuracy is **very low during challenging scenarios** but **TraceFL performance is constant.**



Takeaway: TraceFL operates effectively under real-world challenging FL settings.

Summary

- **TraceFL** is the **first clients' attribution (interpretability) technique** for FL.
- **Compatible**
  - **HuggingFace's Classification Models (e.g., GPT)**
  - **Flower Datasets**
  - **Differential Privacy**

Flower Framework + Flower Datasets

Complete artifact is available at https://github.com/SEED-VT/TraceFL

The **TraceFL artifact** for ICSE 2025 **has received** the Available, Functional, and Reproducible evaluation badges.

Functional Reusable Available

Thank you everyone :)