Weighted Federated Learning with Encryption for Diabetes Classification

Present by: Zhiyi Yue

Authors: Puyang Zhao, Zhiyi Yue, Xinhui Liu, Jingjin Wu

UTHealth Houston
School of Public Health

A Global Health Crisis

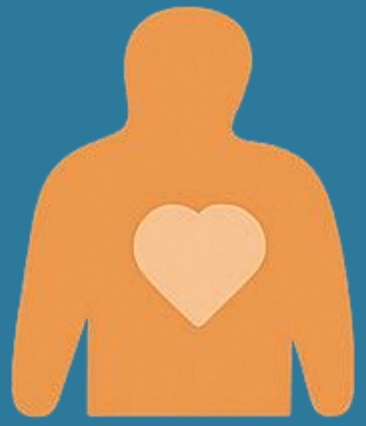
530

Million

Diabetes affects 530+ million adults globally

Diabetes increases the risk of:

Type 2 Diabetes



Heart Disease

Stroke

Motivation & Challenges

Traditional machine learning methods:

- Centralized models trained on pooled datasets
- Examples: Logistic Regression, SVM, Random Forest, Deep Neural Networks

Problems:

- ❌ Require access to all data in one location
- ❌ Privacy laws (e.g., HIPAA, GDPR) prohibit such sharing
- ❌ Lack generalizability due to homogeneous training data

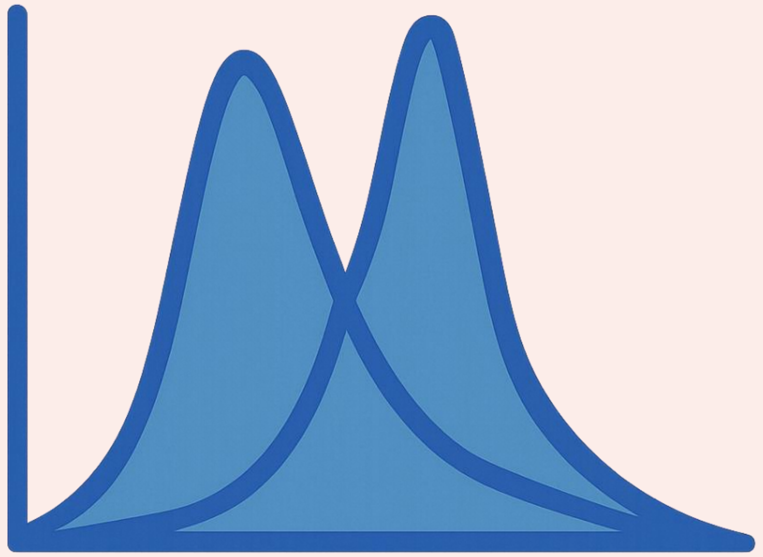
Motivation & Challenges

Why this work?

1. Healthcare data are fragmented across institutions and highly sensitive.
2. Centralized AI models struggle with:

Privacy risk

Data imbalance



Non-IID (heterogeneous) distributions

Goal:

Develop a privacy-preserving, robust, and scalable AI framework for diabetes classification across institutions.

Solution Overview

Proposed Framework:

1. Weighted Federated Learning (FL)
2. Lightweight Encryption
3. Support for both classical ML and deep learning
4. Interpretability via SHAP

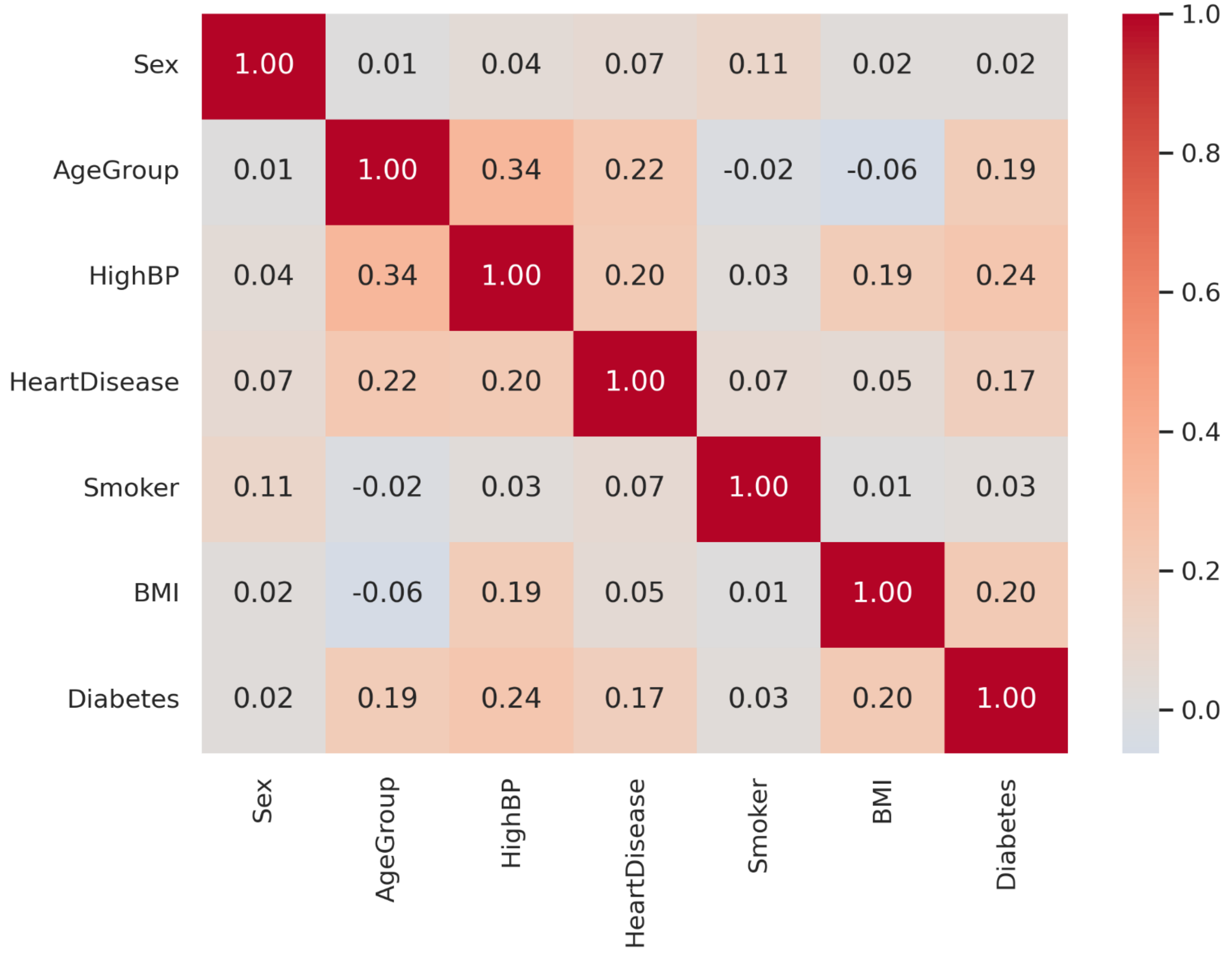
FL trains models collaboratively without sharing raw data.

Dataset & Preprocessing

Source Institutions:

- ADCES: 4,995 samples
- CDC: 5,172 samples
- IDF: 5,180 samples

Total: 15,347 records



Local Preprocessing (per site):

1. Imputation
2. Normalization
3. Anonymization

→ Ensures privacy, standardization, and non-leakage

Feature Risk Profiles

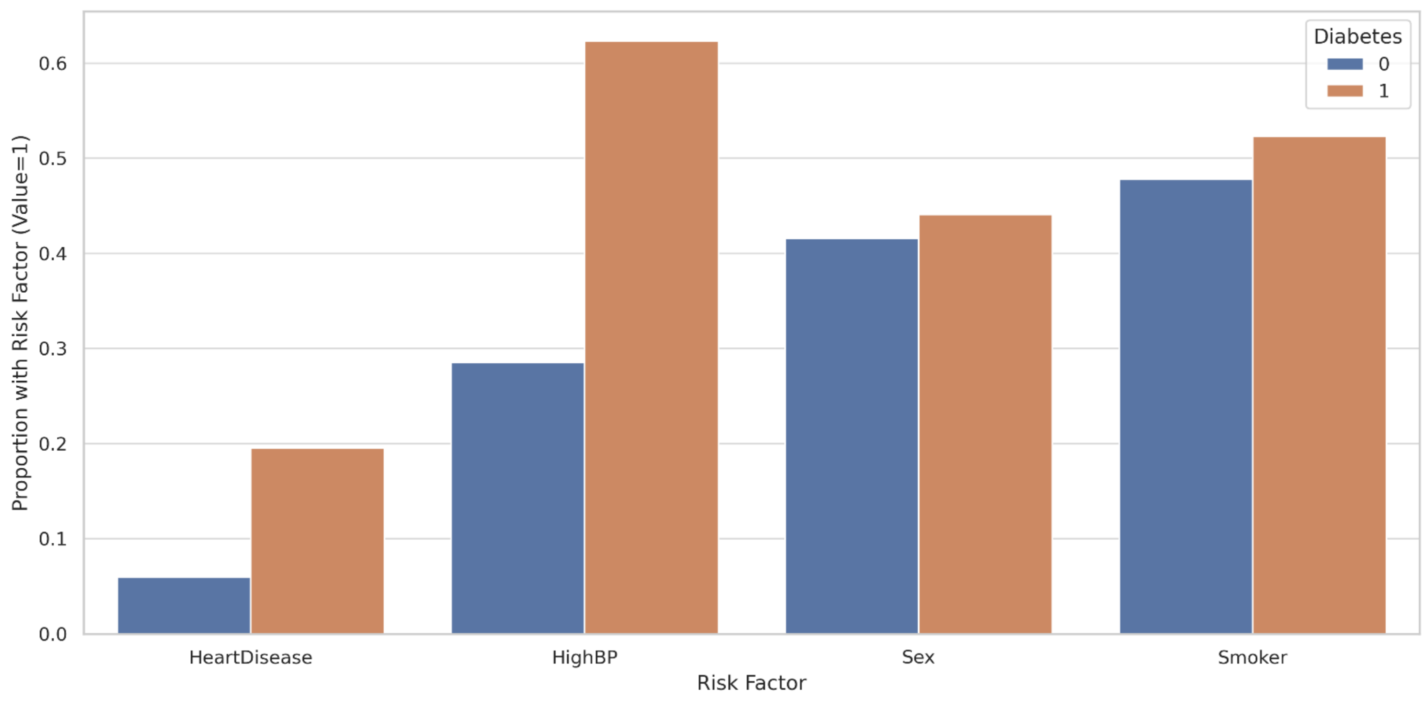
Diabetics more likely to have:

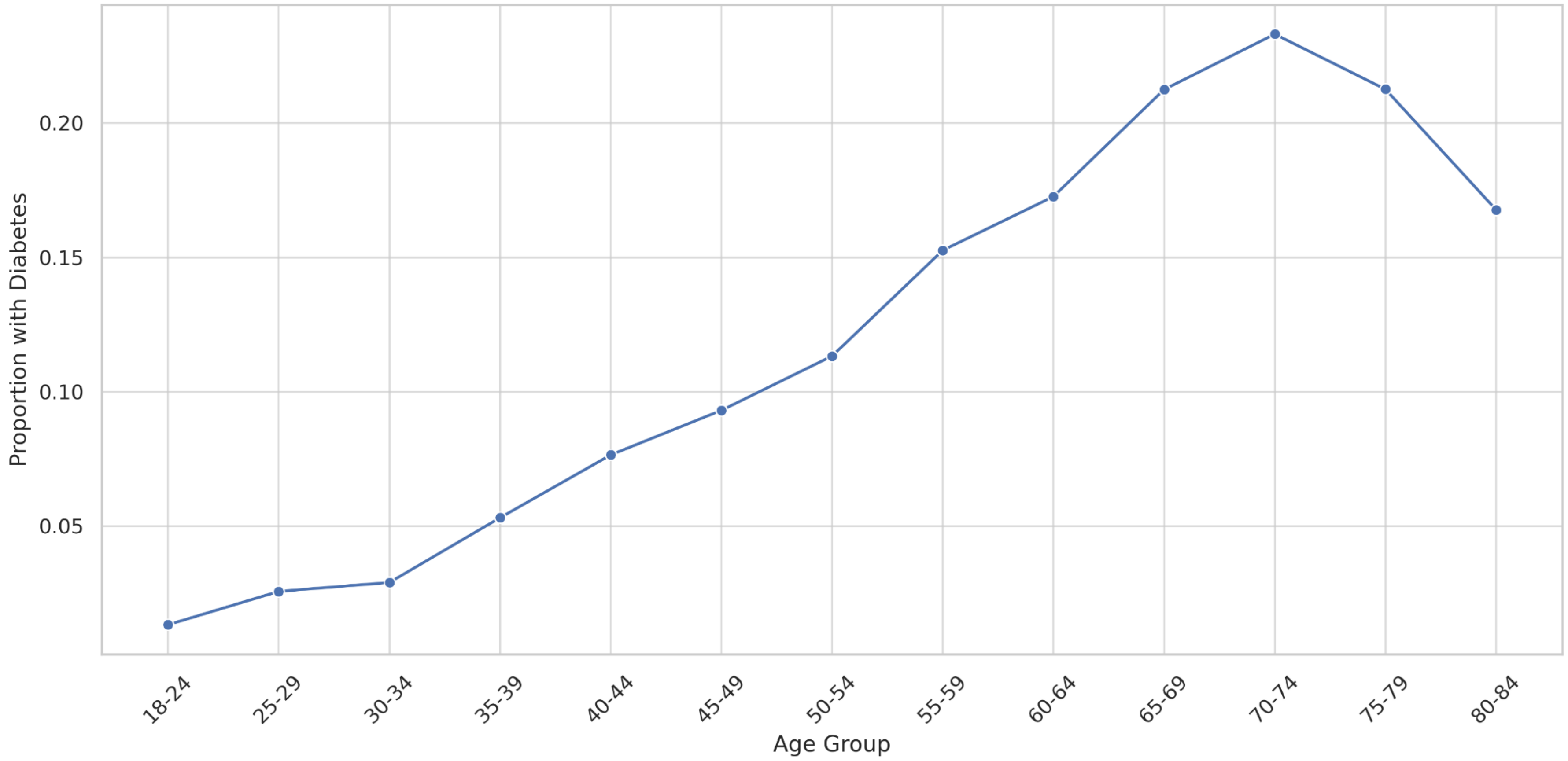
- High blood pressure (61% vs 29%)
- Heart disease (20% vs 7%)
- Smoking history

No major sex imbalance

Sharp increase after age 55

Peak: 70–74 age group





Feature Selection: Laplacion Score

✦ **Similarity graph**

$$S_{ij} = \exp\left(\frac{\left\|x_i - x_j\right\|^2}{a}\right)$$

- x_i, x_j: feature vectors; a: scaling parameter

✦ **Laplacian matrix**

$$L = D - S, \quad D_{ii} = \Sigma_j S_{ij}$$

✦ **Feature Relevance**

$$L_r = \frac{f_r^T L f_r}{f_r^T D f_r}$$ ⇨ **Lower, more important**

- f_r: r-th feature

Weighted FL: Overview

✦ FL basics

- Clients train locally on their data
- Server aggregates local updates
- Raw data never shared → privacy preserved

Challenge
- Clients have **unequal** dataset sizes
- Data distributions are often **non-IID**
- Simple averaging (FedAvg) may bias results

✦ Weighted FL idea

- Clients with larger datasets get more influence
- Use dataset size to assign weights

Weighted FL: Aggregation

Let us consider N clients, and each client i holds a local dataset D_i of size n_i, and $n = \Sigma_{i=1}^{N} n_i$

✦ Aggregation rule

$$\boxed{w_t} = w_{t-1} + \Sigma_{i=1}^{N} \frac{n_i}{n} \Delta w_i^t$$

- w_t: global model at round t
- Δw_i^t: local update from client i

w_t

- Distributed back to all clients for the next training round
- Serves as the final global model for prediction once training converges

Algorithm 1 Weighted FL Algorithm

```
Input: Local datasets {D_i}_{i=1}^N, number of communication rounds T, initial global model parameters w^0.
Output: Trained global model parameters w^T.
for each round t = 1 to T do
    for each client i in parallel do
        Local Training:
        Initialize local model parameters: w_i^t ← w^{t-1}.
        Train the local model on D_i to obtain updated parameters w_i^t.
        Compute local model update: Δw_i^t = w_i^t − w^{t-1}.
    end for
    Server Aggregation:
    Update global model parameters using weighted aggregation:
        w^t = w^{t-1} + Σ_{i=1}^N (n_i / n) Δw_i^t.
end for
Return final global model parameters w^T.
```

Weighted FL: Encryption Phase

To enhance privacy, we incorporate an encryption phase using a **masking** technique during the transmission of local model updates. This method ensures that individual updates remain confidential.

Algorithm 2 Encrypted FL with Masking

Input: Local datasets $\{D_i\}_{i=1}^{N}$, number of rounds T, initial global model w^0, random seeds $\{s_i\}_{i=1}^{N}$.
Output: Trained global model w^T.
for each round $t = 1$ to T **do**
  for each client i in parallel **do**
    Local Training:
    Compute local model w_i^t based on D_i.
    Compute local update $\Delta w_i^t = w_i^t - w^{t-1}$.
    Encryption:
    Generate random mask m_i using seed s_i.
    Masked update: $\widetilde{\Delta w}_i^t = \Delta w_i^t + \boxed{m_i}$. → Add random mask m_i
    Send $\widetilde{\Delta w}_i^t$ to the server.
  end for
  Server Aggregation:
  Aggregate masked updates: $\widetilde{\Delta w}^t = \sum_{i=1}^{N} \widetilde{\Delta w}_i^t$.
  Decryption:
  Compute total mask $M = \sum_{i=1}^{N} m_i$.
  Unmask aggregated update: $\Delta w^t = \widetilde{\Delta w}^t - M$.
  Update global model: $w^t = w^{t-1} + \frac{1}{N}\Delta w^t$.
end for
return w^T

Key points:

- Individual updates remain confidential
- Masks cancel out after aggregation
- Ensures privacy without affecting learning outcome

Machine Learning Models

Centralized learning

- Data from all participating institutions are aggregated into a single dataset
- Trains machine learning models on the full spectrum of data
- Ignores privacy and decentralization concerns

Machine learning models - algorithms benchmarked (both Centralized & Federated):

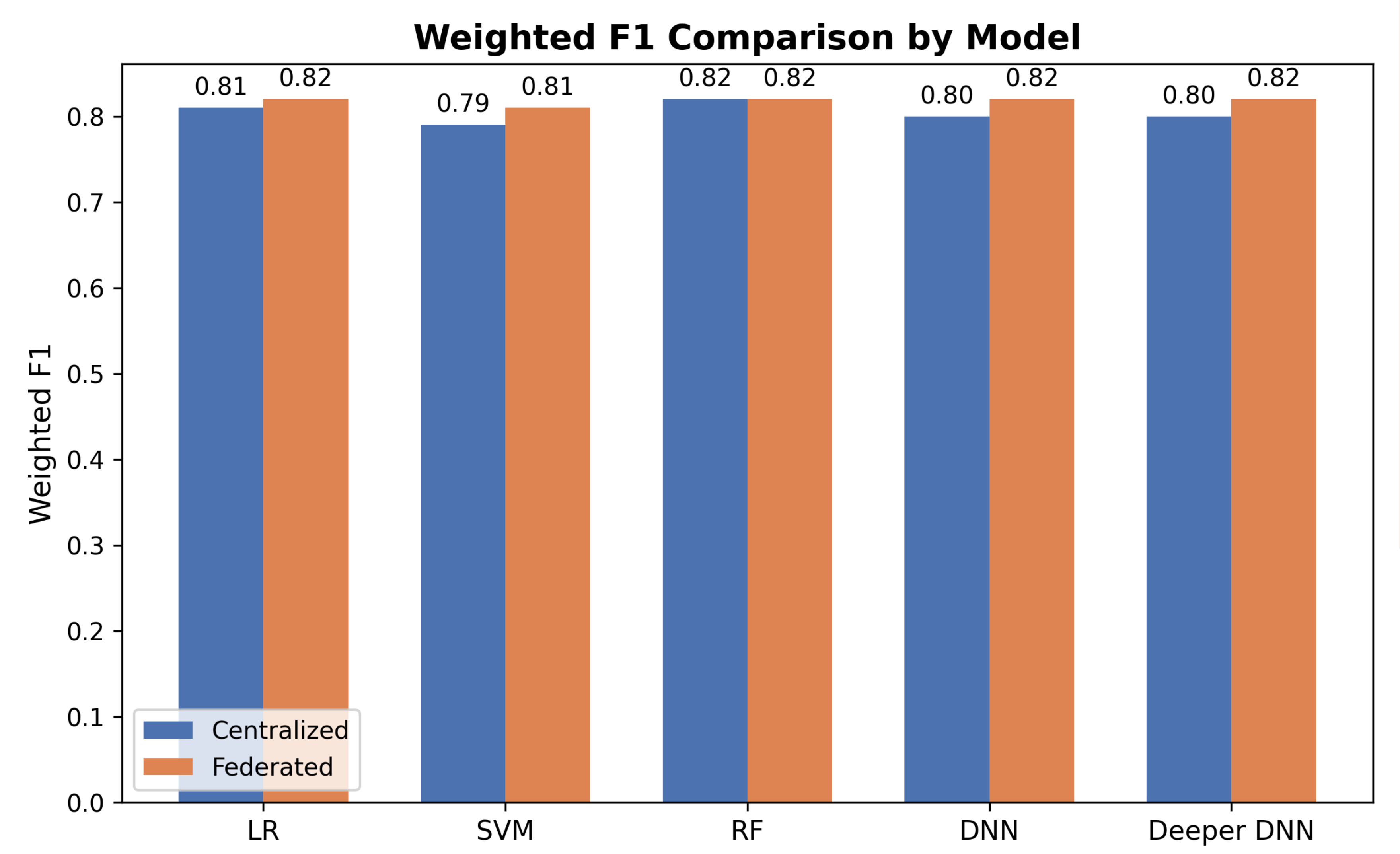
- LR – Logistic Regression: coefficients averaged in FL
- RF – Random Forest: global prediction = average of local forests
- SVM – Support Vector Machine: decision scores averaged in FL
- DNN – 3-layer feed-forward NN (PyTorch, FedAvg)
- Deeper DNN – 5-layer NN with larger hidden dimension (PyTorch, FedAvg)

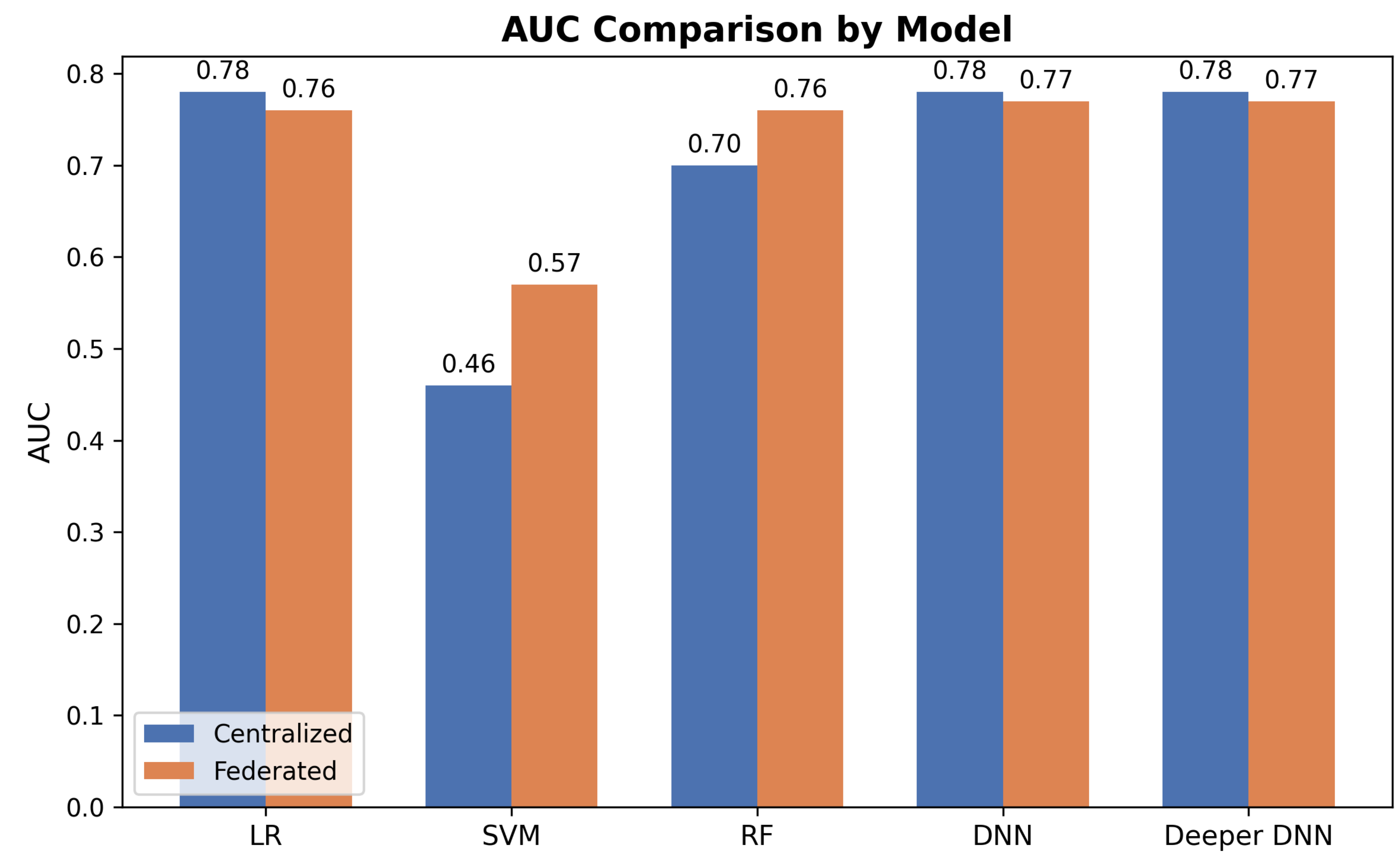
Performance Comparison

Model	Framework	Weighted F1	AUC	Precision	Recall
LR	Centralized	0.81	0.78	0.81	0.86
	Federated (No Enc.)	0.82	0.76	0.82	0.87
	Federated (With Enc.)	0.82	0.76	0.82	0.87
SVM	Centralized	0.79	0.46	0.78	0.79
	Federated (No Enc.)	0.81	0.57	0.89	0.87
	Federated (With Enc.)	0.81	0.57	0.89	0.87
RF	Centralized	0.82	0.70	0.81	0.85
	Federated (No Enc.)	0.82	0.76	0.83	0.87
	Federated (With Enc.)	0.82	0.76	0.83	0.87
DNN (3L)	Centralized	0.80	0.78	0.83	0.86
	Federated (No Enc.)	0.82	0.77	0.83	0.87
	Federated (With Enc.)	0.82	0.77	0.83	0.87
Deeper DNN (5L)	Centralized	0.80	0.78	0.88	0.86
	Federated (No Enc.)	0.81	0.77	0.76	0.87
	Federated (With Enc.)	0.82	0.77	0.83	0.87

- Weighted FL consistently matches or surpasses centralized learning.
- Classical models (SVM, RF) show the strongest improvements.
- Deep models remain stable; encryption introduces no accuracy loss.

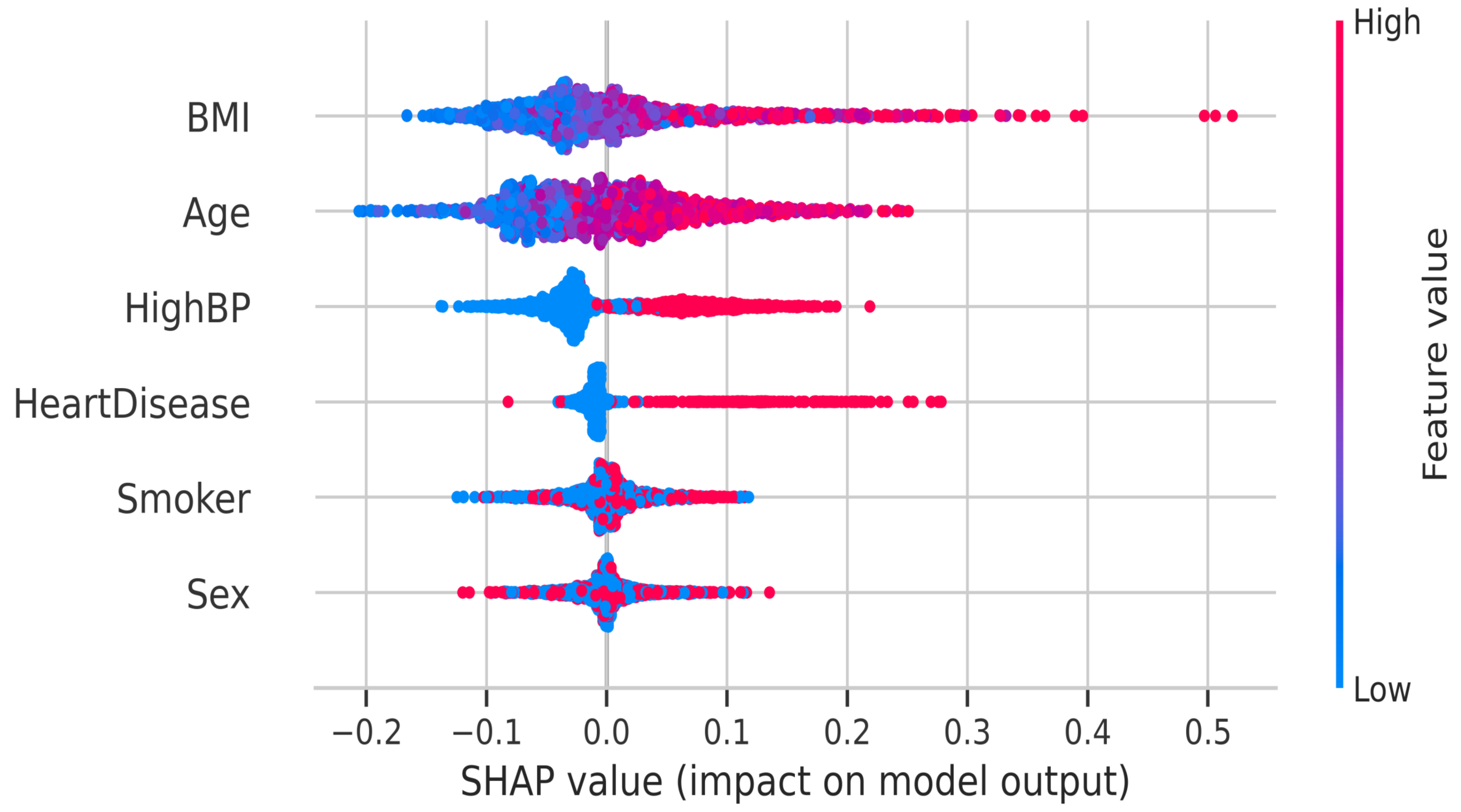
Model-specific Performance Gains





- SVM and RF show the largest AUC gains under federated learning.
- LR achieves a slight F1 improvement while maintaining comparable AUC.
- DNNs remain stable in both AUC and F1, unaffected by encryption.
- Classical models, particularly SVM and RF, benefit most from federated learning, while deep networks maintain stable performance without loss from encryption.

Feature Contributions (SHAP Analysis)



- BMI emerges as the dominant predictor, strongly increasing diabetes risk.
- Age, high blood pressure, and heart disease contribute significantly as secondary risk factors.
- Smoking and Sex show negligible influence across most cases.
- The feature impact patterns mirror established clinical knowledge, reinforcing trust and face validity of the model.

Encryption and Communication Overhead

Metric	Value
Avg. Encryption Time / round	0.0001 s
Avg. Decryption Time / round	0.0013 s
Communication cost / round	0.16 KB

- Encryption and decryption add negligible latency (0.0001–0.0013s/round).
- Communication cost is minimal and stable (≈0.16 KB/round).
- Privacy is preserved without accuracy loss or efficiency trade-off.

Conclusion

- Weighted FL with encryption achieves equal or better performance than centralized learning across classical and deep models.
- Privacy is preserved with negligible encryption overhead, ensuring feasibility for routine clinical networks.
- Model interpretability is supported by clinically meaningful features, led by BMI and followed by key factors such as age, hypertension, and heart disease.
- The framework demonstrates practical scalability for multi-institutional healthcare collaboration.

Limitation & Future Work

- Limitations
  - Conducted on a cross-sectional dataset with limited features, which may restrict generalizability.
  - Site-level differences in coding practices and label quality were not fully addressed.
  - Real-world challenges such as client dropout, unstable networks, and adversarial risks were not modeled.
- Future Work
  - Extend to larger and more diverse multi-institutional datasets, including longitudinal records.
  - Incorporate additional modalities (e.g., imaging, notes) to enrich predictive performance.
  - Explore advanced privacy-preserving techniques (e.g., differential privacy, secure aggregation).
  - Address system-level robustness through simulations of communication delays and client heterogeneity.

THANK YOU!

Thank you for watching our presentation!
Do you have any questions, comments, or suggestions?

GET IN TOUCH

WEBSITE:

https://puyangzhao.github.io//

EMAIL ADDRESS:

jj.wu@ieee.org

Puyang.Zhao@uth.tmc.edu

Zhyi.Yue@uth.tmc.edu

x.liu146@lse.ac.uk