HEPrune: Fast Private Training of Deep Neural Networks with Encrypted Data Pruning

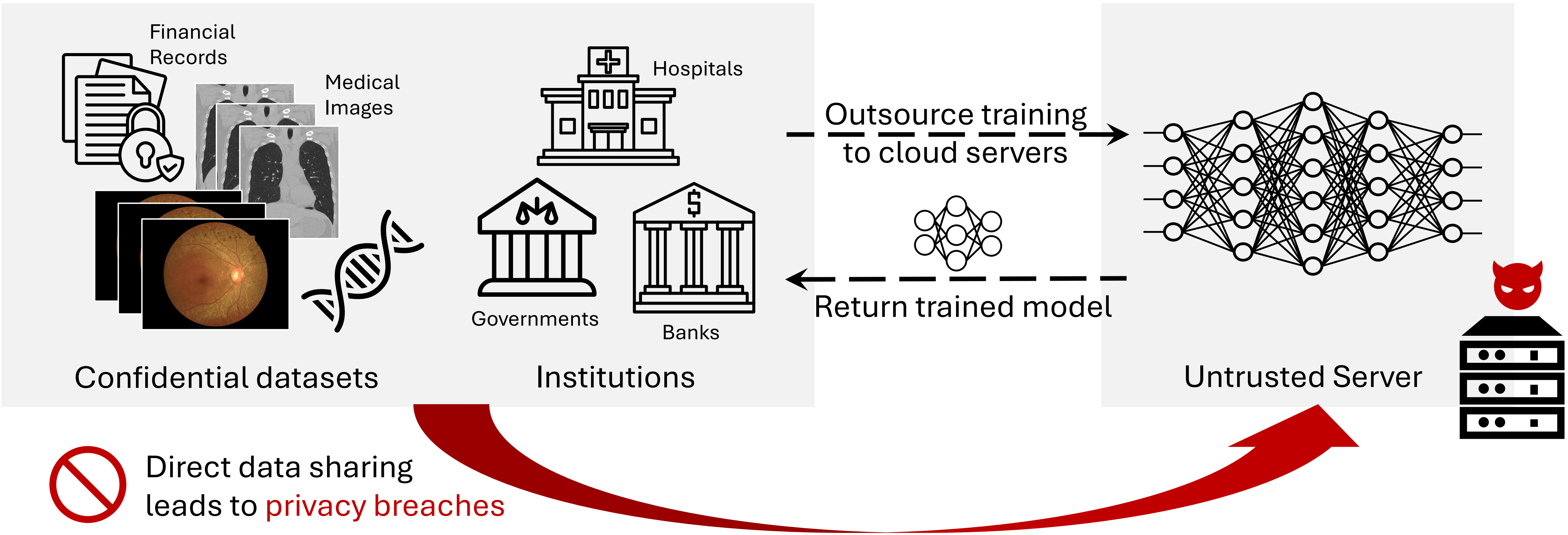
Yancheng Zhang[1], Mengxin Zheng[1] , Yuzhang Shang[2] , Xun Chen[3], Qian Lou[1]

[1]University of Central Florida
[2]Illinois Institute of Technology
[3]Samsung Research America

NeurIPS'24
12th Nov. 2024

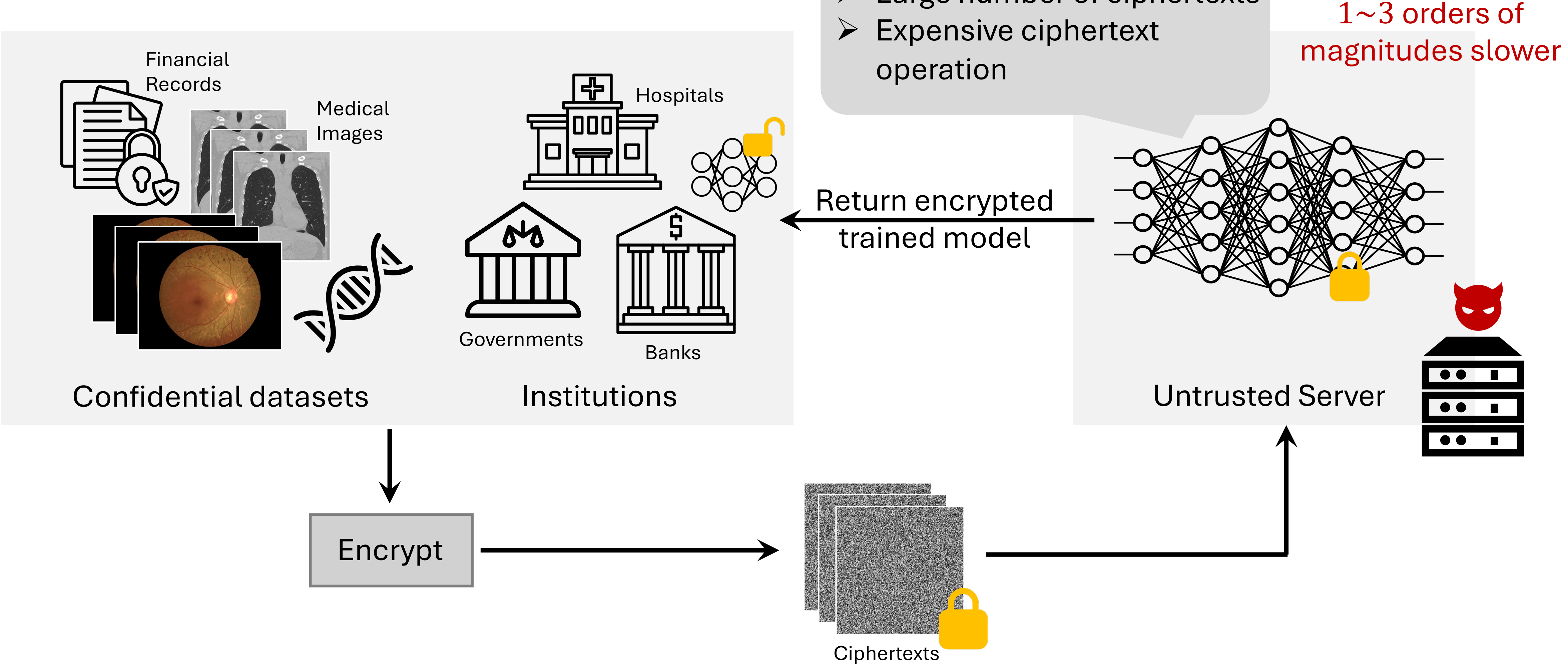
Data Privacy is Important in Neural Network Training

Deep neural networks are widely applied across domains such as healthcare, finance, and law enforcement.



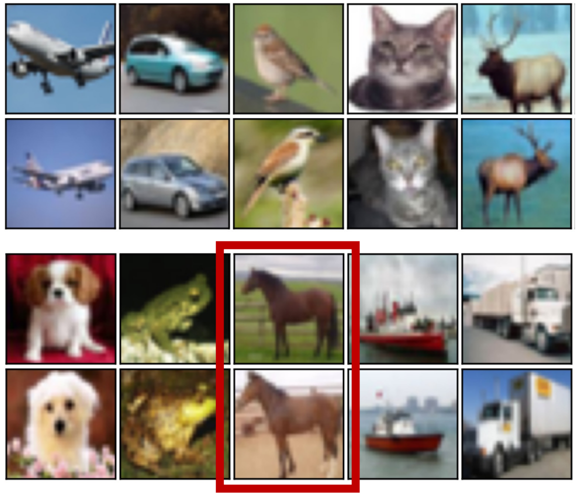
Private Training is Secure but Slow

FHE-based private training offers strong data privacy guarar

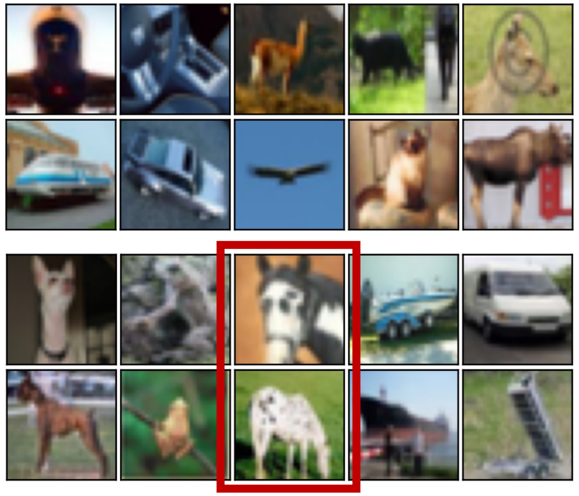


Our Motivation

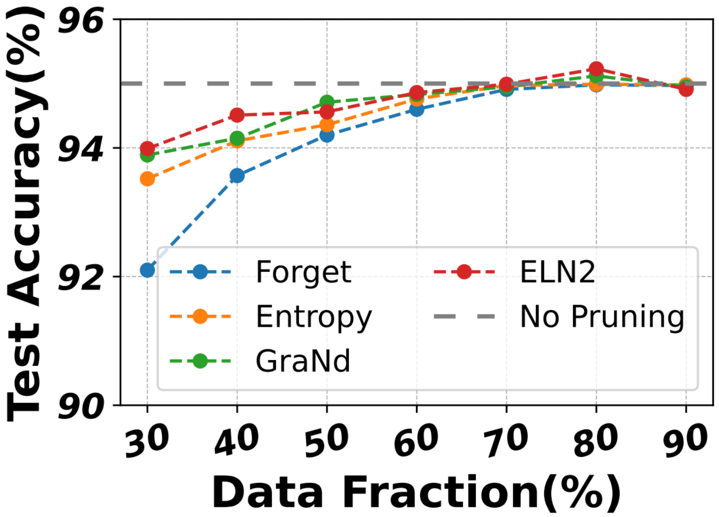
Can we reduce the number of ciphertexts, i.e., encrypted data samples, during private training without compromising accuracy?



Less informative samples



More informative samples



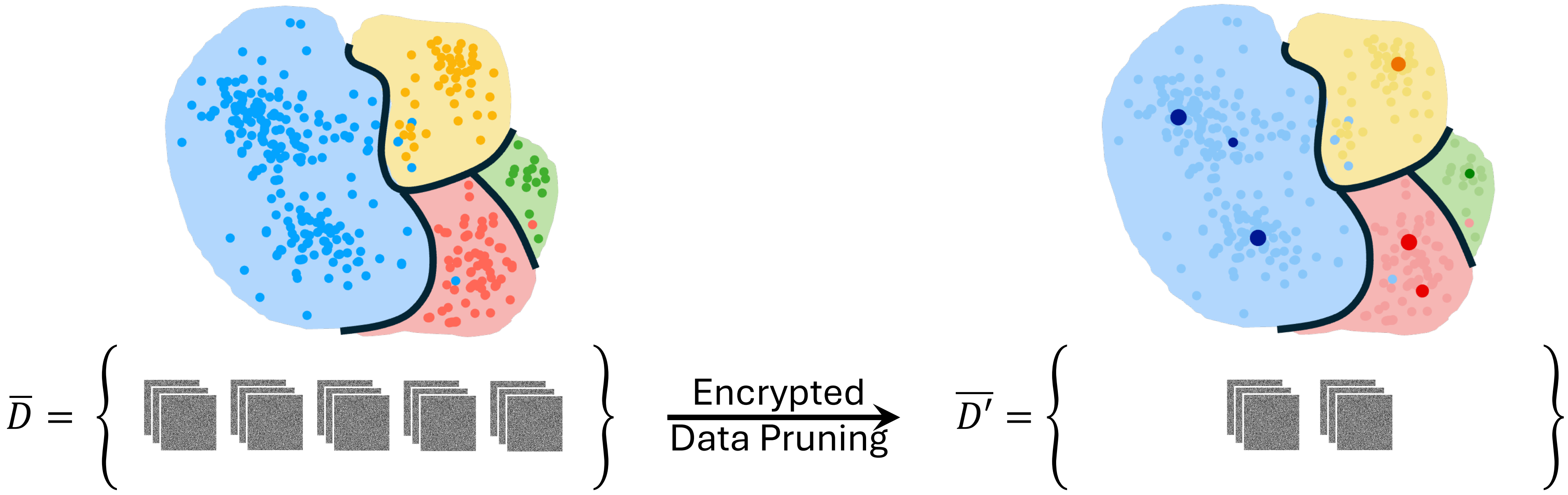
- Redundant
- Easy to learn
- ...

- Diverged
- Challenging
- ...

Training on a subset of samples barely compromise the accuracy in the plaintext

Problem Statement

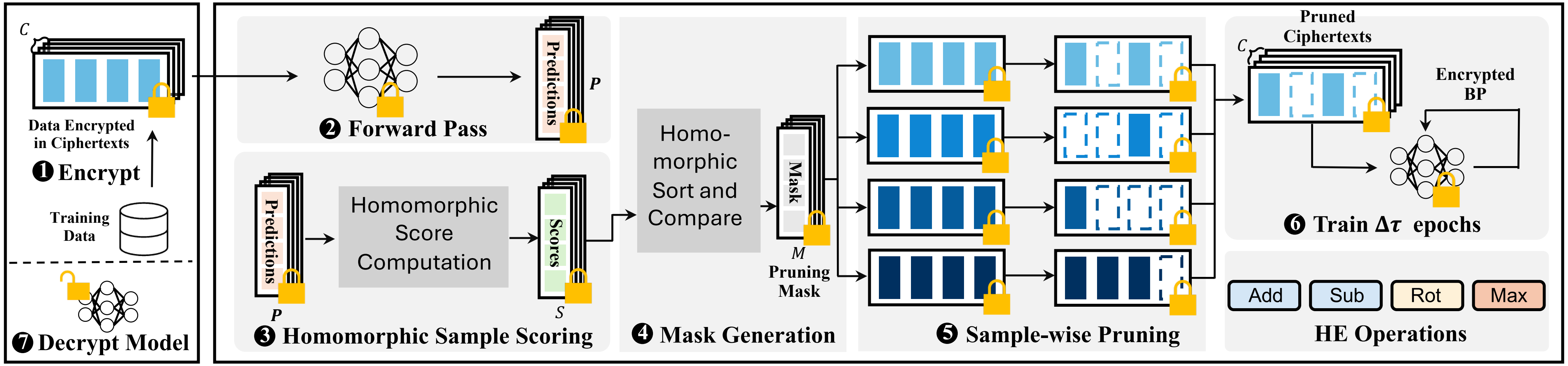
The server choose the most salient subset of samples $\overline{D'}$ from the encrypted dataset $\overline{D}$.



- ➢ Security. The server should not learn the training data or model weights during pruning.
- ➢ Accuracy. The chosen subset should have a close accuracy compared to full dataset.
- ➢ Efficiency. Encrypted data pruning should speedup private training.

Naïve Encrypted Data Pruning

Directly applying data pruning methods in the plaintext to private training is impractical.



Complex non-linear score

- Complex non-linear functions are needed, e.g., in EL2N.
  $\mathbb{E}_{w_t}\|p(x;w_t) - y\|_2$
- A single homomorphic square root function can take up to 2 minutes.

Massive homomorphic sorting

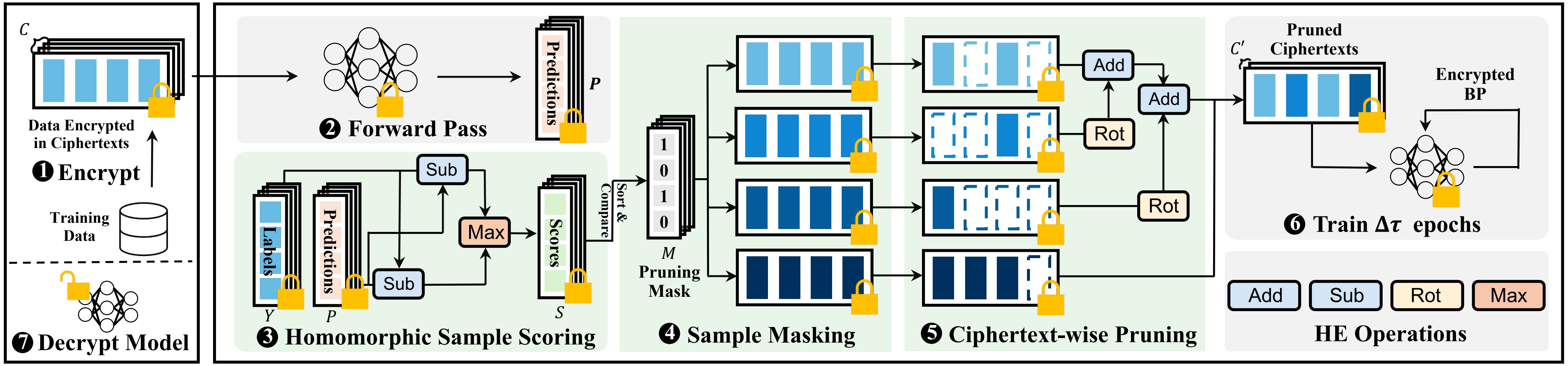
- $O(N^2)$ homomorphic comparisons are needed to sort the score and generate the pruning mask.

Sample-wise pruning

- The number of ciphertexts cannot be effectively reduced.

HEPrune Framework

HEPrune enables encrypted data pruning with HE-friendly score, client-aided masking and ciphertext-wise pruning.



Complex non-linear score

- Complex non-linear functions are needed.

$$\mathbb{E}_{w_t}\|p(x;w_t)-y\|_2$$

- A single homomorphic square root function can take up to 2 minutes.

→ HE friendly score

Massive homomorphic sorting

- $O(N^2)$ homomorphic comparisons are needed to sort the score and generate the pruning mask.

→ Client-aided masking

Sample-wise pruning

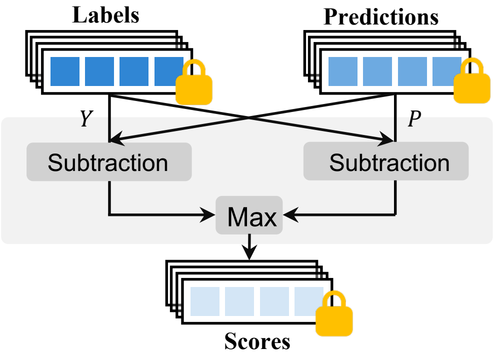
- The number of ciphertexts cannot be effectively reduced.

→ Ciphertext-wise pruning

HE-friendly Importance Score

The HE-friendly importance score (HEFS) is easy to compute in the encrypted state.

Computing HEFS for one ciphertext takes less than 2 seconds.



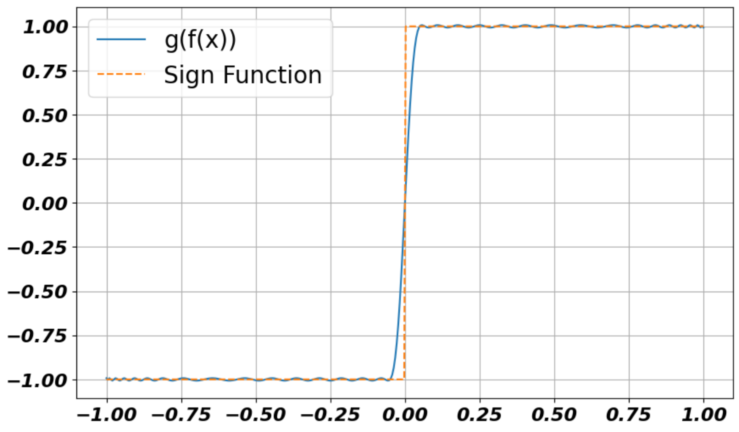
Streamlined circuit

$$score = \text{HE.Max}\big((Y \boxminus P), (P \boxminus Y)\big) = (Y \boxminus P)\,\text{HE.Sign}(Y \boxminus P)$$

$$\text{HE.Max}(u, v) = \frac{(u+v)+(u-v)\text{HE.Sign}(u-v)}{2}$$

$$\text{HE.Sign}(x) = g(f(x))$$

Lightweight computation



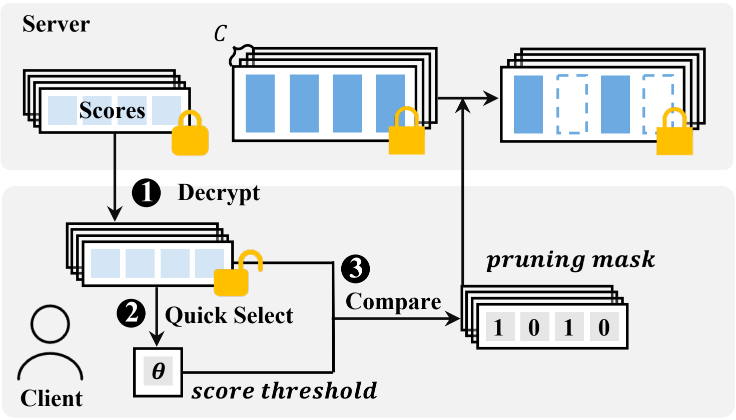
Low approximation error

$$f(x) = 8.83133072x - 46.45750399x^3 + 83.02822347x^5 - 44.99284778x^7$$

$$g(x) = 3.94881885x - 12.91030110x^3 + 28.08653622x^5 - 35.59691490x^7 + 26.51593709x^9 - 11.41848894x^{11} + 2.62558444x^{13} - 0.24917230x^{15}$$

Client-aided Masking

Client-aided masking avoids expensive homomorphic sorting without leaking data privacy.



$N = 2^{16}$	$L=0$	$L=1$	$L=2$	$L=3$	$L=4$	$L=5$
Ciphertext Size(MB)	1.01	2.03	3.02	4	5.02	6

Size of a CKKS ciphertext at different level L

Security.
The training data and model weights remain encrypted.
The privacy of data and model is protected.

Efficiency.

- Runtime

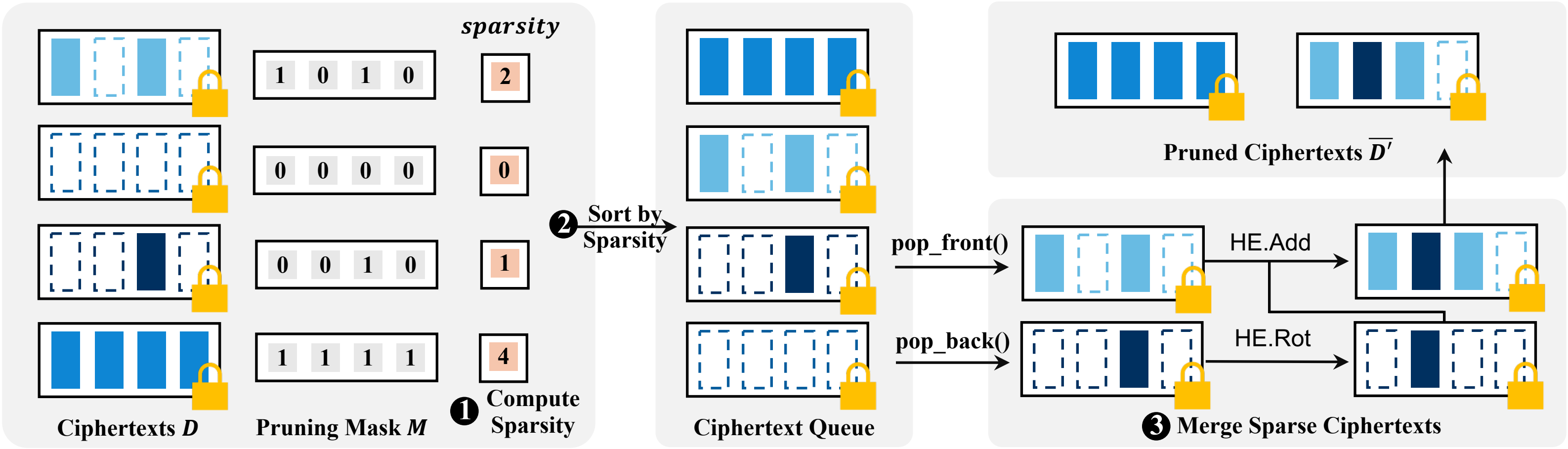
Generating the pruning mask needs only $O(N)$ time on the client side (15 ms for the CIFAR-10 dataset).

- Communication

Before sending the scores, the server can set the score to a low multiplicative level to improve communication.

Ciphertext-wise Pruning

Ciphertext-wise pruning (CWP) effectively removes the sparse ciphertexts and reduces the number of ciphertexts in private training.



Encrypted Data Pruning on Different Datasets

We set the pruning ratio as $p = 0.9$ (only 10% of the dataset is kept) on different datasets.
Encrypted data pruning speedup the training time by around 6.6 times.

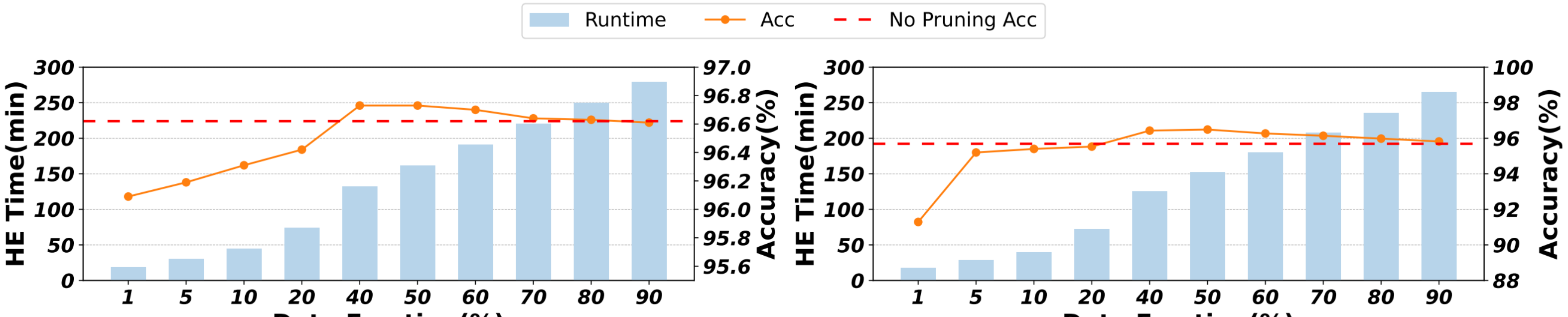
Method		MNIST	CIFAR-10	Face Mask Detection	DermaMNIST	SNIPS
Unencrypted	Acc(%)	$95.69_{\pm 0.02}$	$96.62_{\pm 0.02}$	$95.46_{\pm 0.06}$	$75.91_{\pm 0.11}$	$94.43_{\pm 0.05}$
HETAL	Acc(%)	$96.27_{\pm 0.02}$	$96.57_{\pm 0.04}$	$95.46_{\pm 0.05}$	$76.06_{\pm 0.18}$	$95_{\pm 0.08}$
	Runtime(h)	276.75	293.3	32.88	101.55	113.7
Ours	Acc(%)	$95.54_{\pm 0.05}$	$96.31_{\pm 0.06}$	$95.21_{\pm 0.06}$	$75.86_{\pm 0.15}$	$95.14_{\pm 0.08}$
	Runtime(h)	41.89	44.76	5.02	15.5	17.36

The proposed methods effectively improves the performance over the baselines.

Method	Accuracy(%)	Runtime(h)	Speedup	Communication(MB)
Full Data(HETAL)	$96.57_{\pm 0.04}$	293.3	$\times 1$	18.1
Prune Baseline	$95.98_{\pm 0.12}$	488.91	$\times 0.6$	18.1
+Client Aided	$96.16_{\pm 0.07}$	196.91	$\times 1.49$	22
+HEFS	$96.31_{\pm 0.06}$	105.57	$\times 2.78$	22
+Ciphertext-wise Pruning	$96.31_{\pm 0.06}$	44.76	$\times 6.55$	22

Different Pruning Ratios and Training from Scratch

We experiment with different pruning ratio on the CIFAR-10 and MNIST dataset. Training with 40%~70% of the dataset has even high accuracy than training with the full dataset.



(a) Different pruning ratios on CIFAR-10 **(b) Different pruning ratios on MNIST**

The encrypted data pruning can also be applied to the training-from-scratch setting.

Method		1%	5%	10%	20%	40%	50%	60%	70%	80%	90%
Acc.	Acc(%)	93.23	97.12	97.39	98.38	98.52	98.55	98.5	98.48	98.45	98.45
	$\Delta Acc.$	-5.26	-1.37	-1.1	-0.11	+0.03	+0.06	+0.01	-0.01	-0.04	-0.04
Runtime(h)	Time(h)	32.25	110.61	208.56	404.46	796.26	992.16	1188.06	1383.94	1579.88	1775.72
	speed up	60.8×	17.2×	9.4×	4.8×	2.5×	1.9×	1.7×	1.4×	1.2×	1.1×

Thank you!

Code

Paper

Poster