Hybrid Privacy Scheme: Combining Local Differential Privacy and Multi-Party Computation to Optimize Secure Support Vector Machine Training

Yavor Litchev and Abigail Thomas

Mentor: Yu Xia

The Importance of Privacy

- Increase in digitized data (ex. PII, PHI)
- Data breaches can result in discrimination, harassment, etc.
- Cyber security is important
- Many existing algorithms with costs and benefits

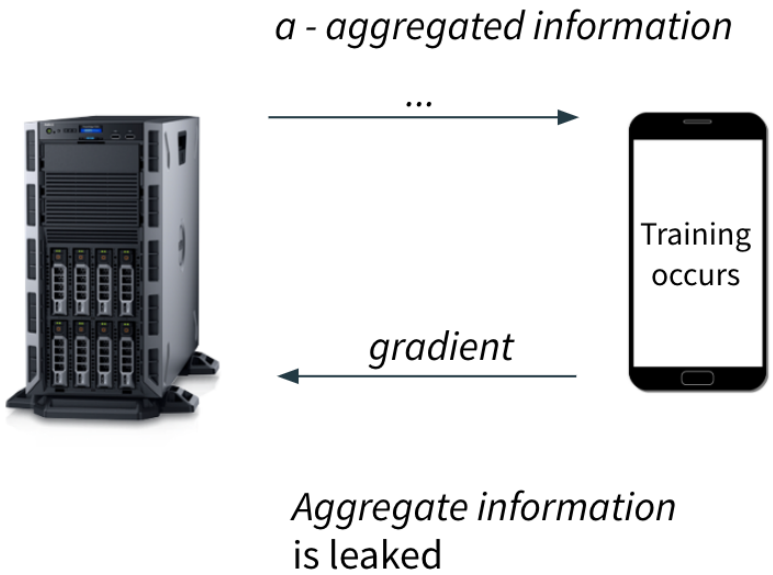
Our Project Goal

- Artificial intelligence is very computationally intensive
- Costs and benefits are magnified
- In this project, we seek to combine existing algorithms in order to make an
- optimized, hybrid machine learning model
- Higher accuracy (0.9247 vs 0.8975)
- Faster ($\sim$ 20 hrs vs $\sim$ 3790 hrs)

Existing Algorithms

Federated Learning

- Protects user privacy by training a model on a separate machine, then sends updated model back
- Raw data is not leaked, but gradients may be leaked.



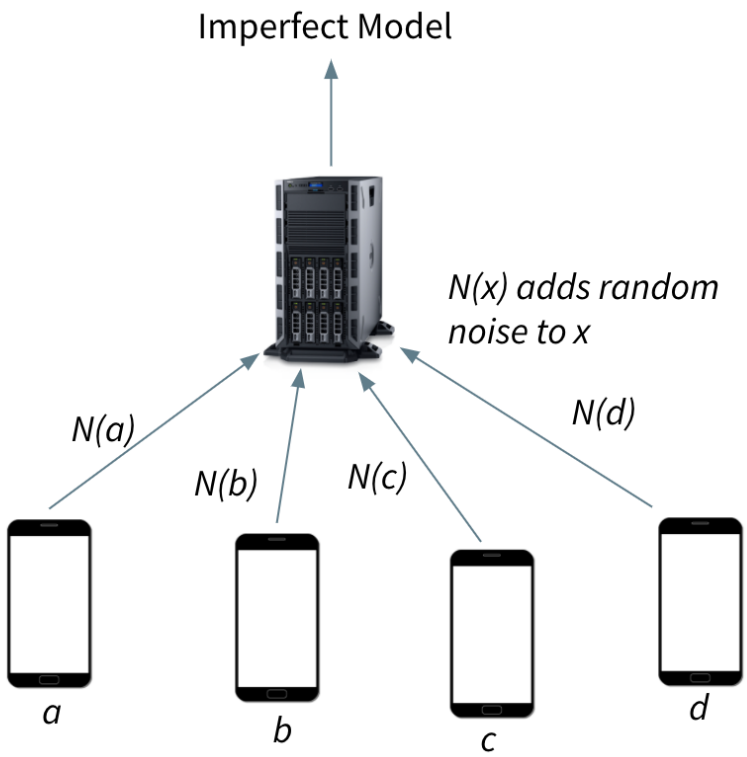
Existing Algorithms

Data Encryption/Homomorphic Encryption

- Data Encryption allows one person to send data to another without a 3rd party intercepting
  - Computations cannot be done on encrypted data
- Homomorphic encryption solves this, allows for computations on encrypted data
  - However, it is computationally expensive and very slow

Local Differential Privacy

- This algorithm seeks to protect user privacy by adding random noise to the inputs of the users in order to give people plausible deniability as to what the true values are.
- Ex. a = 1, a' = 0; b = 2, b' = 3; c = 3, c' = 2; d = 4, d' = 5



Local Differential Privacy

Definition

We say that an algorithm π satisfies ϵ-Local Differential Privacy where $\epsilon > 0$ if and only if for any input v and v'

$$\forall y \in Range(\pi) : \frac{\Pr[\pi(v) = y]}{\Pr[\pi(v') = y]} \leq e^{\epsilon}$$

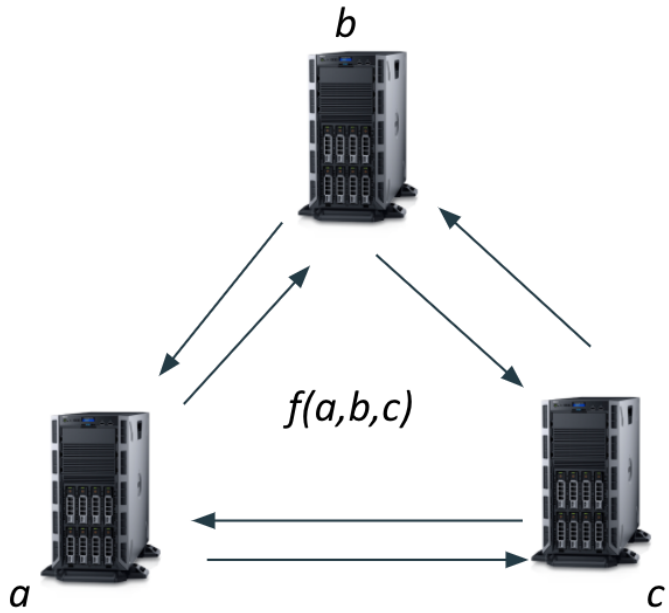
where $Range(\pi)$ denotes every possible output of the algorithm π.

Local Differential Privacy

- Pros
  - Very fast (computations can be done directly on noisy data because it is protected by noise)
  - High standard of security
- Cons
  - Accuracy decreases as privacy increases
  - Ex. 0.9457 vs 0.8975 (for $\epsilon = \ln(3)$)

Multi Party Computation

- Class of algorithms where parties compute a function jointly whilst preserving the privacy of the parties and their inputs to the function.



Multi Party Computation

Definition

A protocol is called secure if there exists an efficient simulator S such that the simulated message transcript

$$S(\{x_j, y_j\}_{P_j \in C}) = \{\overline{view_j}\}_{P_j \in C}$$

and the real leaked values

$$\{view_j\}_{P_j \in C}$$

have the same distribution.

Multi Party Computation

- Pros
  - Very powerful and high standard of security
  - Accuracy of computations is perfect (since no noise is added)
- Cons
  - Excruciatingly slow, takes months ($\sim$3790 hrs for 4000 images)

LDP vs MPC

LDP Pros:

- Very Fast
- Fairly Secure

LDP Cons:

- Somewhat inaccurate

MPC Pros:

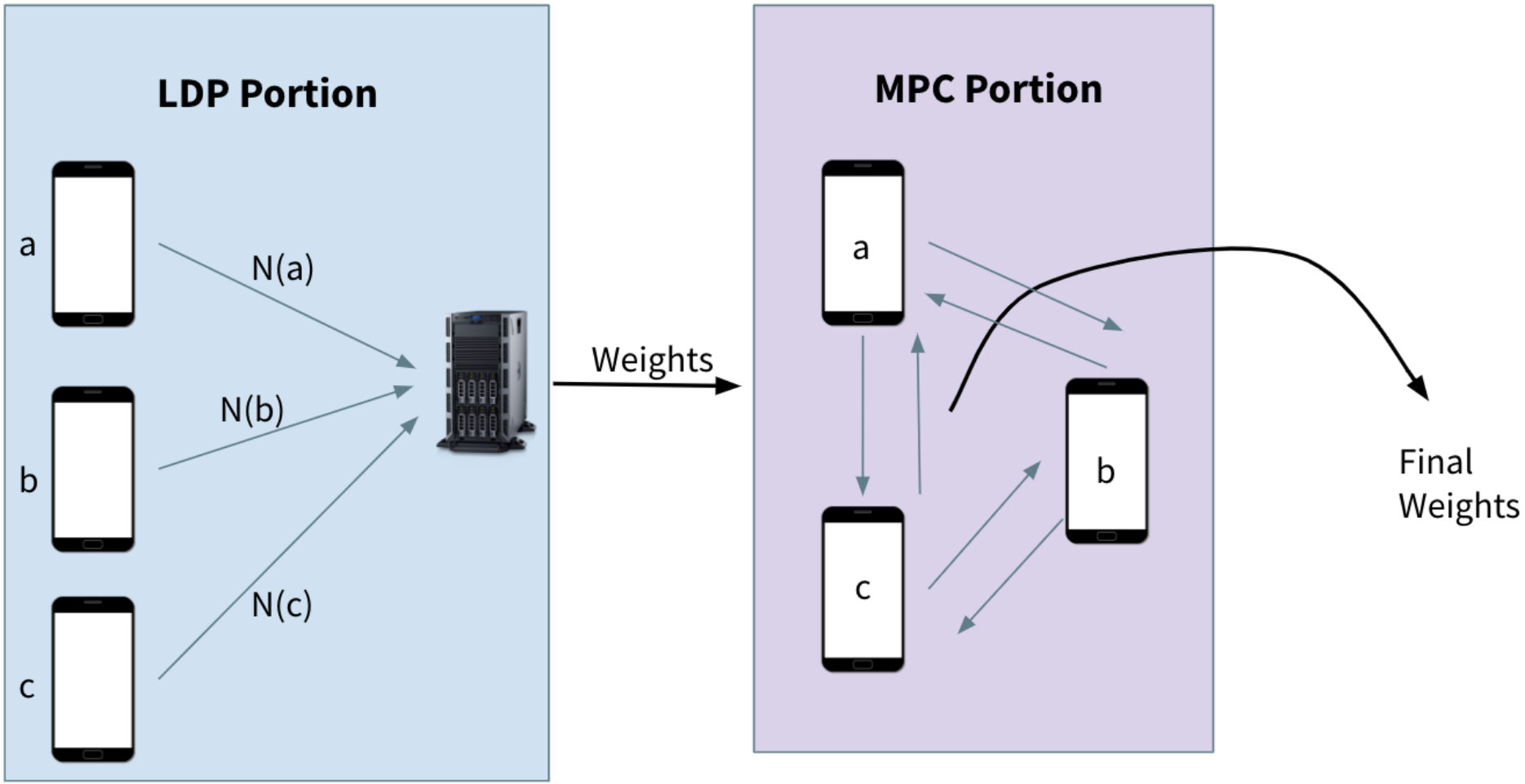
- Very Accurate
- Very Secure

MPC Cons:

- Extremely Slow

Our Hybrid Model

We seek to combine LDP and MPC



Support Vector Machines (SVMs)

- We trained an SVM on the MNIST dataset to implement our model
- MNIST: images of handwritten digits
- SVM: machine learning algorithm that classifies data using hyperplane
- SVM classifies images based on their digit

Framework of Program

- 4 parties (1-4) with their own data to simulate multiple parties
- Only communicate by sending messages
- There is an initial party 0 that downloads the MNIST data and distributes shares to each party

LDP

- Each party adds random noise to their data and sends it all to party 4 (central server) to do the training
- ϵ-LDP is defined with an adjustable privacy parameter ϵ
- Implemented an adjustable parameter to make LDP more/less secure by adding more/less noise
- Party 4 runs the SVM on all the noisy data to get some weights

The Switchpoint

- After LDP training, party 4(who has the weights) directly submits them to the MPC. Therefore, the weights are not leaked until the end of the program.
- No other information is leaked and thus the switchpoint is secure

MPC

- MP-SPDZ framework
- MASCOT Protocol
- 4 parties each submits 1 image to the protocol, 1 party submits the weights
- Trains on all 4 images at once
- This process is repeated 5 times

Security of Our Protocol (Intuition)

- It is known that LDP and MPC are secure
- Weights remain with party 4 after LDP training occurs, then party 4 securely submits the weights directly to the MPC, therefore no extra information is leaked
- MPC algorithm protects users data and the weights as it updates the weights, weights do not have to be revealed
- Therefore, our protocol is as secure as the LDP and MPC used

Results

SVM Training*	Baseline	Hybrid Model
LDP Portion		
Test Cost	108,398	112,097
Test Accuracy	0.8975	0.9097
Runtime	31.310 seconds	21.799 seconds
MPC Portion		
Cost	485.90	2,449.83
Accuracy	0.9457	0.9247
Runtime	3790.406 hours	19.049 hours

Cost Function:

$$J(w) = \frac{1}{2}\|\boldsymbol{w}\|^2 + C\left[\frac{1}{N}\sum_{i}^{n}\max(0, 1 - y_i * (\boldsymbol{w} \cdot x_i + b))\right]$$

*Data retrieved from training on computer with 8th gen core i7, 16GB RAM, no GPU

Why Does This Matter?

- Hybrid model is both fast and accurate
- Security is not compromised
- Individuals can confidently use their data to train an SVM in the real world

Further Research

- Use CNNs in the future instead of SVMs
- Look into more MPC algorithms

Acknowledgements

We would like to thank:

- Our mentor: Yu Xia
- The PRIMES program
- Our families