ENHANCING CUSTOMER ENGAGEMENT WITH PRIVACY PRESERVING AI

IMDA PET SANDBOX –
ANT INTERNATIONAL CASE STUDY

INFOCOMM MEDIA DEVELOPMENT AUTHORITY

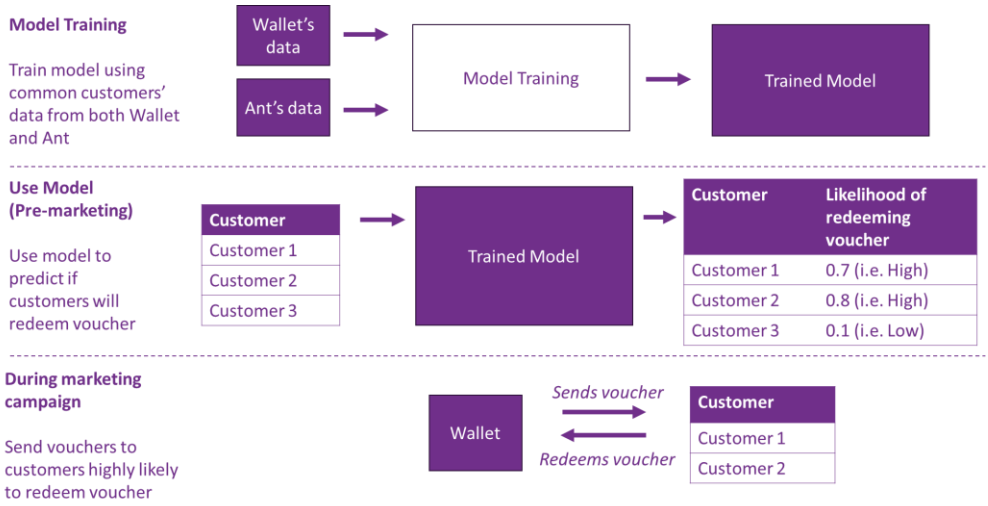
Contents

Business Case

1. Ant International "Ant" as a global infrastructure and network company, operates a rewards system for its partners, e.g. digital wallets "Wallet" and retailers. If retailers can be connected to Wallet's customers through the rewards system, this can increase revenue for retailers and increase customer engagement with Wallet. One such way would be to offer relevant promotions (e.g. vouchers) from the retailers through Ant's reward system, to better serve Wallet's customers' needs. These can be predicted through a model that is trained based on customer voucher redemption history and customer preferences.

2. In this use case, Ant's reward programme collects voucher redemption data about the customers and Wallet collects purchase history and preference data of the same customer, including additional demographics data (e.g. age, gender) that are not collected by Ant's rewards programme.

3. If Ant could access common customers data from both their rewards programme and Wallet, they could train a more accurate prediction model on likelihood of customer redemption of voucher. This allows them to pre-select a group of customers for marketing campaign and offer them relevant promotions that would better serve their needs.



Challenges

4. Ant & Wallet cannot reveal or share common identifiers to identify the group of common customers between them due to personal data concerns. In addition, both entities do not allow customers data to leave their original environment due to business sensitivities.

Methodology

5. Through a combination of Federated Learning (FL), Multi-Party-Computing (MPC), and Homomorphic Encryption (HE), Ant and Wallet will be able to identify their common customers, while protecting the underlying personal data. At the same time, their data continues to reside in their own environment while co-training the prediction model.

6. For this POC, model was co-trained using 500,000 rows and ~200 columns of historical data. Hardware configuration used per party consists of 4 local nodes, each comprising of 29 CPU cores and 75GB memory, and was sufficient for the model training to be completed over 12 hours.

7. After completing the model training, model effectiveness is tested by sending vouchers to a sample group of 70,000 customers over a period of 2.5 weeks. The percentage of vouchers redeemed will be used to measure the effectiveness of the model in selecting customers. The higher the redemption rate, the more effective the model. This will be compared against percentage of vouchers redeemed by customers selected using business rules (e.g. age, gender), which is a common methodology in selecting customers for marketing campaign.

Solution Overview

8. The solution comprises of 2 stages
    a. Model training stage, which comprises of
        - Identifying common customers
        - Training model using common customer data
    b. Using model stage, where the trained model is used to predict and generate each customer's likelihood of redeeming a voucher received

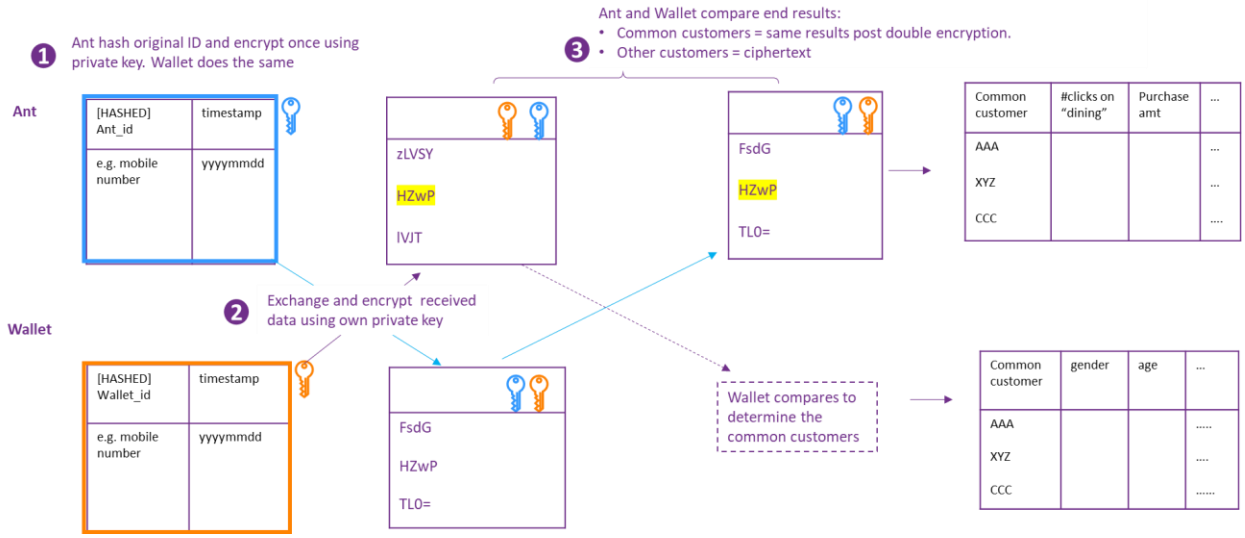
The following segments on explaining how the model works has been simplified, intended to help readers (e.g. business users, non-technically trained individuals) to understand the conceptual workings. Readers interested in the in-depth technical explanations can refer to ***Annex A – Technical Explanation of Solution.***

Train Model

1: Identify common customers

9. In order to meaningfully co-train a model, both parties will need to first identify the data they each have belonging to common customers. However, identifying common customers usually entails sharing of a list of user identities, which are considered personal data. At the same time, this process may also inadvertently reveal the identities of other customers within their customer base.

10. By using Private Set Intersection (PSI) a Multi-Party Computing (MPC) technique, both parties will only learn who are their common customers (ie. the intersection of their customer base), without learning anything else on each other's data.



11. In the Private Set Intersection (PSI) methodology used in this POC, first both Ant and Wallet will each encrypt the user identities with their own secret key respectively. The encrypted data are then exchanged and each uses their private key to encrypt the received dataset again. Ant and Wallet then identify their common customers by finding records with same encrypted user identities from the 2 doubly-encrypted customer lists they each hold.

12. For this POC, both parties generated their secret key based on the Elliptic Curve Diffie-Hellman (ECDH) protocol. It offers advantages such as : computation efficiency due to the mathematical structure (which makes it computationally difficult to solve) ; and efficient mathematical operations (allowing smaller keys). These provides strong security guarantees with smaller key sizes compared other key cryptosystems.

2: Train model using common customers

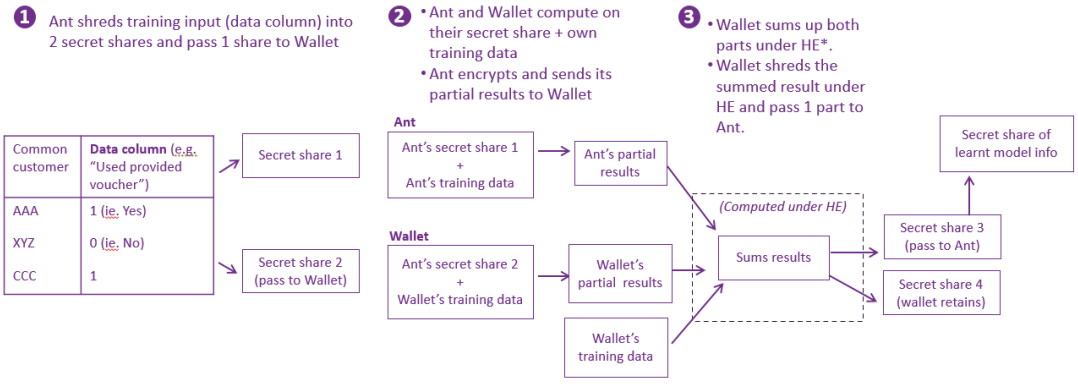
13. After identifying the common customers, Ant and Wallet will use their data on the common customers to begin co-training a decision tree-based model.

14. As an overview, the entire training process is based on Federated Learning (FL) structure, where Ant and Wallet can collaboratively build a model using their data without sharing any original data with each other or sending data to a central server for training. Generally during FL, each party will train the model within their own environment and send across model updates learnt on their own data to each other to create and update the model within their own environment. However, model updates may reveal information about the data and customers used to train the model.

15. For additional security, this POC also utilises Multi-Party Computing (MPC) throughout the training process. This is done by generating and using a random number to shred data into secret shares, and the secret shares are distributed among parties and computed by each other using the secret shares and their own data. The parties then recombine their results and recompute to obtain the outcome of their secret sharing. In this case, the steps of shredding, distributing, computing and recombining requires a lot of communication across both parties, which translates to higher network requirement. Therefore, Homomorphic Encryption (HE) is also used to improve the computational efficiency by reducing the number of steps required to recombine and recompute the results.

16. Consequently, in this POC, as the model is trained using a combination of FL, MPC and HE, each party will only hold a partial model and partial model information.

17. The following paragraphs explain in detail, how the model is trained using the combination of MPC, FL and HE:

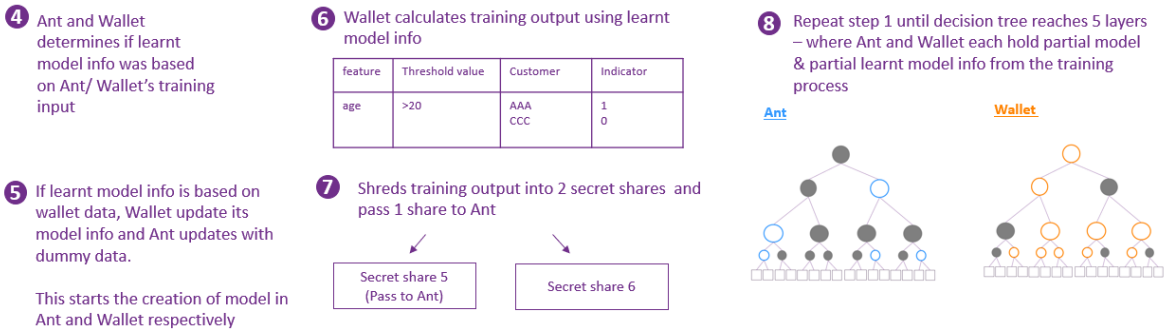


a. Step 1 : Ant generates a random number in the range of 2^{64} or 2^{128} to shred the data ("Used provided voucher") into 2 secret shares. The shares are then distributed to Ant and Wallet

b. Step 2 – 3 : Ant and Wallet compute using the secret share and their data to obtain the partial result, where the partial result resides within their own environment. Ant encrypts its partial result using its private key and sends the encrypted partial result, together with a public key to Wallet. This is necessary for Wallet to perform encrypted computation.

c. Wallet then encrypts its partial results with the public key received. Then Wallet combines its encrypted partial results, its training data, and Ant's encrypted partial result under Homomorphic Encryption (HE) to obtain an encrypted summed result. This summed result contains learnt model info, which Wallet shreds the learnt model info into 2 secret shares and passes one encrypted share to Ant. 1

d. In this step, HE is used to increase computation efficiency instead of improving the security of the solution. Without HE, the same summed result can be obtained through implementing additional steps of shredding partial results into secret shares, exchanging the shares, computing and recombining. This incurs additional communication costs and is unscalable for large datasets. Therefore, HE mitigates this challenge by reducing the number of steps using encrypted summation.



e. Step 4 -5 : Ant decrypts the learnt model info and determines together with Wallet, if Ant or Wallet's training data would increase the effectiveness of the model. In the case where Wallet's data was deemed to train a more efficient model, based on the learnt model info, Wallet would update its model. At the same time, Ant will update its own model with dummy data. Thus, starting the creation of the partial model for both Ant and Wallet respectively.

f. Step 6 - 8 : In order to continue training the model, Wallet will use the learnt model info from the previous step on its data to obtain a training output. Another random number is generated to shred the training output into 2 secret shares and Wallet

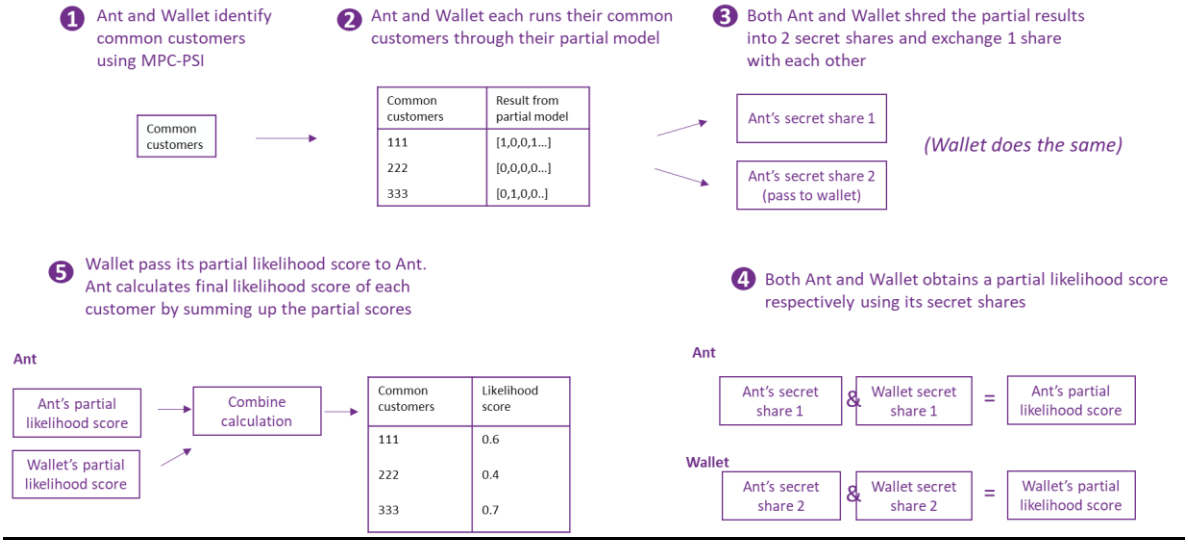
[1] Wallet will also perform the same actions that Ant performs in Steps 2 – 3 at the same time.

passes 1 secret share to Ant. With this secret share, Ant iteratively repeats step 2 - 7 together with Wallet until the model is optimised, eventually resulting in Ant and Wallet each holding a partial model and partial learnt model info based on their own data.

g. In this way, both parties do not know the training data contributed by the other party, and yet is able to co-train a model based on the data from both parties.

Use Model

3: Generate likelihood score



18. Before launching a marketing campaign, Ant and Wallet will use the trained model to predict the likelihood of common customers using a voucher received. Here, MPC is also used to protect the results from their partial model, which may also reveal information on the individual.

    a. Step 1- 3 : Both Ant and Wallet use PSI to identify a common group of customers. Then, they use their own partial model on this common group of customers within their own environment and obtain a partial result in the form of 1's and 0's. Here, they shred their own partial result into 2 shares and exchange 1 share with each other.

    b. Step 4 -5 : Then, using the partial model info that they hold, both Ant and Wallet calculate a partial likelihood score and Ant combines both scores to obtain the final prediction score. The higher the score, the higher the likelihood of the customer redeeming a voucher if they receive one. Ant will then select the customers with the higher scores to receive the voucher during the marketing

campaign, hence increasing the probability of redemption and therefore achieving higher marketing efficiency.

POC Outcomes

19. POC results indicate that model selected customers had >90% relative increase to the baseline conversion rate, and was comparable or better than industry standards. This allowed Ant to verify that by co-training a model with data from both Ant and Wallet, it was able to give a better prediction if customers are likely to redeem a voucher. Therefore by selecting these customers to receive the voucher, the conversion rate increases, which improves the marketing campaign efficiency.

User Group	**Selection method**	**Conversion rate (vouchers claimed/ vouchers sent)**
A	Based on A+ business rules only (e.g. A+ payments for "dining" category)	Baseline
B	Based on A+ & wallet business rules (e.g. A+ payments for "dining" category + wallet clicks on food-related items)	7% relative increase (to baseline)
C	Based on model predicted likelihood (e.g. likelihood > 75)	>90% relative increase (to baseline)

Technical Learnings

20. Communication latency affects model training and prediction time. Although this POC was conducted for offline computations, preliminary tests for real-time computations show that within 0.5 – 1sec after a customer logs into wallet, the model can predict his likelihood of accepting the voucher and send him the voucher. This response time is considered acceptable by industry standards.

21. As the randomness of numbers used to shred the data is from a sufficiently large range (2^{64} or 2^{128}), it provides security for the model training process. Additionally, if there is a need to retrain the model, keys used for HE and randomness of number used are regenerated. This provides another layer of security for subsequent model re-training.

Preliminary Regulatory Guidance

22. PDPC is in the process of reviewing the technical architecture and regulatory compliance of the implementation and has provided the following preliminary regulatory guidance specific to the implementation of MPC and HE for model training. A Practical Guidance from PDPC on the end-to-end implementation will be provided at a later date.

23. Implementation of MPC to generate secret shares of training output prevents either party from constructing any meaningful information about the individual common customers, unless any party is able to access both pair of secret shares. Such secret shares on their own would generally not be considered as personal data, and parties would not be considered to have shared personal data during the model training phase.

24. In Ant's case, MPC is used in combination with HE. These implementations are premised on the assumption that each party must ensure that their secret shares and secret keys are not disclosed, by putting in place industry-recognised governance processes and technical standards (e.g., IEEE 2842-2021 Recommended Practice for Secure Multi-Party Computation).

25. To reduce the risk of an individual or small group of individuals being singled out in the decision tree model, Ant may wish to consider implementing safeguards such as the following, where relevant:
    a. Remove outliers prior to model training,
    b. Use a sufficiently large training sample size with a maximum cap on the number of layers in the decision tree, and/or
    c. Enforce rules to ensure there is a minimum number of individuals for each predicted score.

Conclusions and Next Steps

26. The POC proved that the solution could access more data for model training while preserving the privacy and utility of the underlying data, and helped Ant and Partner pre-select customers likely to use the vouchers received during a marketing campaign

27. Moving ahead, as the model continues to improve with more training data, Ant intends to shift into live production. As such, Ant anticipates that more digital wallets may be willing to join their rewards programme to enhance their marketing campaign effectiveness, creating a multiplier effect that could forge more PET-based data partnerships in the ecosystem or value chain. Additionally, similar AI algorithms may be

applied to other business scenarios (e.g. fraud detection), thus saving time for future model development.

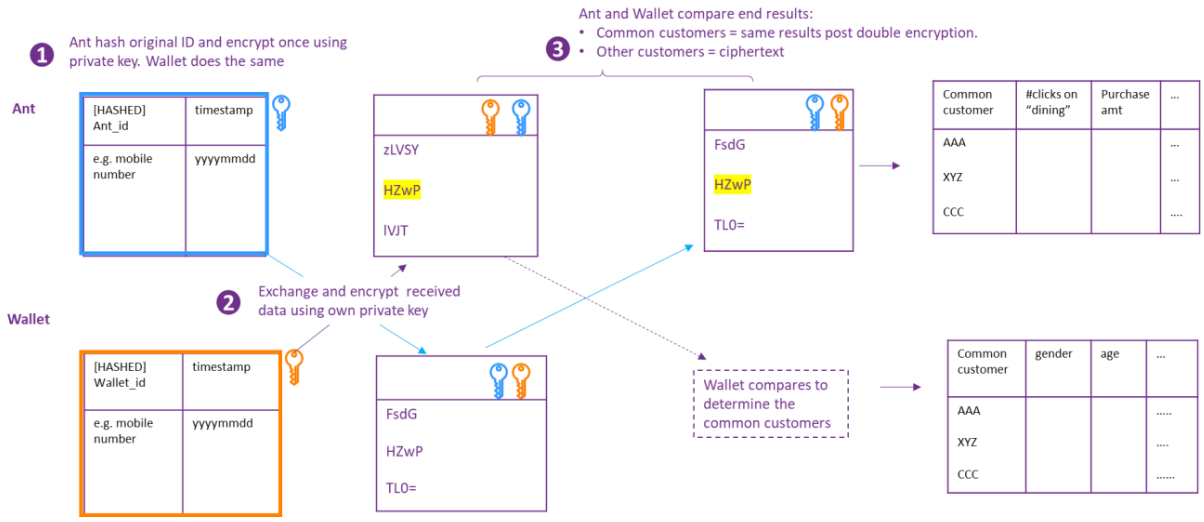
28. The models will be made available on Ant's open-source PET platform. With more PET success stories, Ant anticipates further business developments around their PET platform, such as wider spectrum of new applications that would drive more technical developments. For example, enhancements to their platform's data handling capabilities, efficiency and security of its algorithms, as well as user friendliness for business users. This would allow more companies which may not have the necessary technical in-house capabilities to access PET-enabled solutions to enable them to access and use new sources of data in a trusted and safe manner.

Annex A – Technical Explanation of Solution

This segment is provided by Ant International and intended for readers who are familiar with decision tree-based model and the technical terms used.

Train Model

1: Identify common customers



1. Private Set Intersection (PSI) allows multiple parties, each holding a private dataset, to find the intersection of their datasets without revealing the non-intersecting elements. In this project, we implement a semi-honest ECDH-based two-party PSI protocol following paper [2], which combines Elliptic-Curve Cryptography (ECC) and the Diffie-Hellman Key Exchange Protocol to achieve Private Set Intersection. At the beginning of PSI, Ant and Wallet agree on an elliptic curve E. Commonly used groups include subgroups of the multiplicative group of a finite field and elliptic curve groups. In practice, we choose elliptic curves like Secp256k1 which is in line with stds [3].

   Note that at the beginning of ECDH-PSI protocol, we assume the input data from both Ant and Wallet are shuffled.

   a. Step 1: Ant randomly generates a private key α, for each element x_i in its set, Ant applies the hash function and then encrypts it using its key to get $H(x_i)^{\alpha}$. Ant then sends $H(x_i)^{\alpha}\ (i = 1, \dots n_1)$ to Wallet.

      For each element y_i in Wallet's set, Wallet applies the hash function and then encrypts it using its key β, thus computing $H(y_i)^{\beta}$. Wallet then sends the set $H(y_i)^{\beta}\ (i = 1, \dots n_2)$ to Ant.

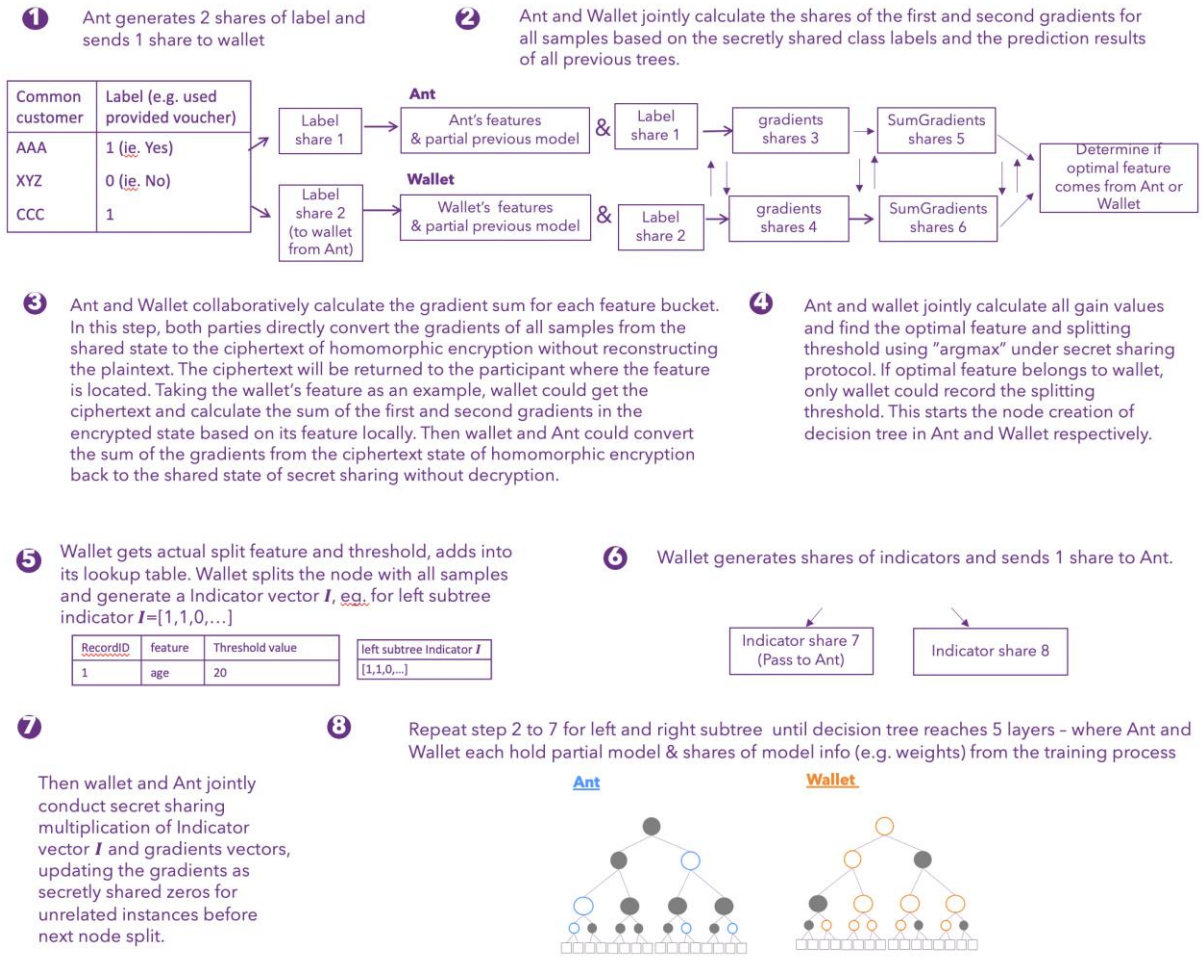
b. Step 2: For each element $\mathrm{H}(\mathrm{x}_i)^{\alpha}$ received from Ant in the previous step, Wallet encrypts it using its key β, computing $\mathrm{H}(\mathrm{x}_i)^{\alpha\beta}$. Wallet sends $\mathrm{H}(\mathrm{x}_i)^{\alpha\beta}$ $(\mathrm{i} = 1, \ldots \mathrm{n}_1)$ to Ant.

   For each element $\mathrm{H}(\mathrm{y}_i)^{\beta}$ received from Wallet, Ant encrypts it using its key α, computing $\mathrm{H}(\mathrm{y}_i)^{\beta\alpha}$. Ant then sends $\mathrm{H}(\mathrm{y}_i)^{\beta\alpha}$ $(\mathrm{i} = 1, \ldots \mathrm{n}_2)$ to Wallet.

c. Step3: Ant and Wallet identify their common customers by comparing two sets $\mathrm{H}(\mathrm{x}_i)^{\alpha\beta}$ $(\mathrm{i} = 1, \ldots \mathrm{n}_1)$ and $\mathrm{H}(\mathrm{y}_i)^{\beta\alpha}$ $(\mathrm{i} = 1, \ldots \mathrm{n}_2)$ to get intersection.

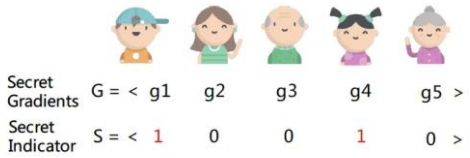
2: Train model using common customers

2. The specific steps are shown in the following figure:



3. Note 1: We can complete step 3 (SumGradients) using only Secret Sharing. However, this approach faces the issue of excessive communication, making it challenging to handle large-scale datasets. The reason for this increased communication is the introduction of redundant zeros in the secret indicator and the additional computations required in

secure SumGradients to confuse the adversary. Consequently, while this method achieves security, it does so at the cost of heavy communication overheard.



4. Additive Homomorphic Encryption (Additive HE) enables computations on encrypted data without requiring the secret key. Employing this form of encryption, we can efficiently perform the summation of encrypted grandients g_1 and g_4. This approach is equal to the calculation of the dot product between gradients and an indicator, thereby streamlining the computational process.

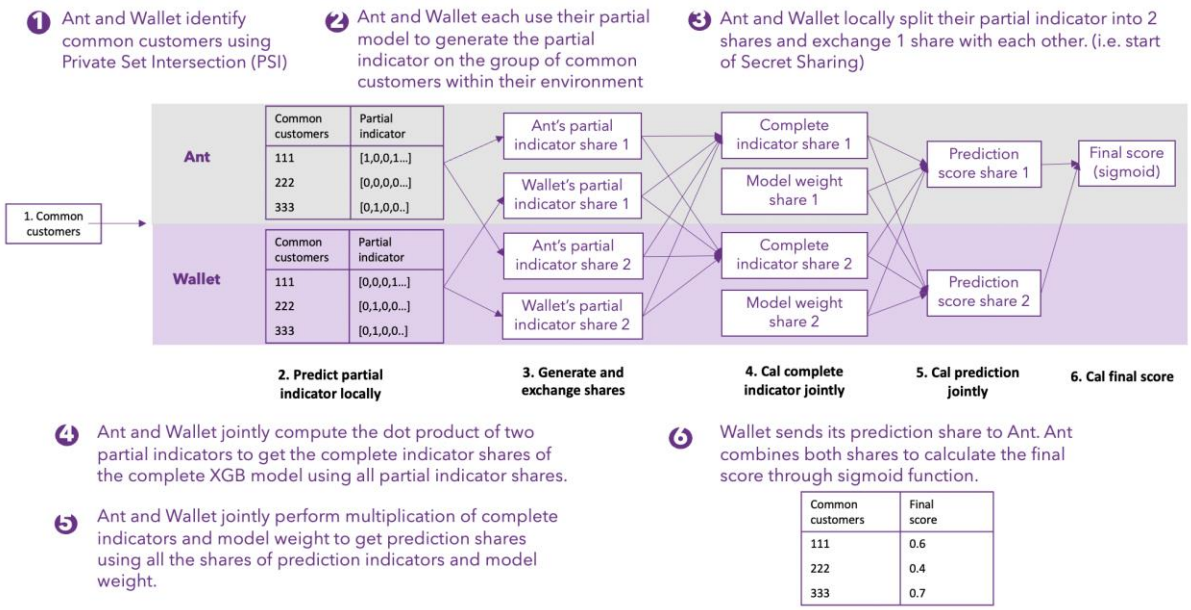
5. When applied this step to the model, we need mainly three steps. That is, (1) party A and party B transform the gradient vector from secret sharing scheme to HE scheme (SS2HE), after which A holds a ciphertext vector since A owns the ciphertext of indicator, (2) A executes addition on the ciphertext vector, and (3) A and B transform the permuted ciphertext vector from HE scheme to secret sharing scheme (HE2SS). This methodology substantially reduces communication overhead in this protocol, thereby enhancing the efficiency.

6. Note 2：For security, we distribute the model to two parties after training. Both parties share the same tree structures, but none of them has the whole split information. Specifically, the split information on a single node will only be visible to the party who owns the corresponding feature, and will be non-visible (dummy node) to the other party. Moreover, both parties collaboratively store the leaf weights in secret sharing scheme. To this end, we can avoid potential information leakage from the trained model.

Use Model

3: Generate probability score

7. The specific steps are shown in the following figure:



Reference

[1] Fang W, Zhao D, Tan J, et al. Large-scale secure XGB for vertical federated learning[C]//Proceedings of the 30th ACM International Conference on Information & Knowledge Management. 2021: 443-452.

[2] Bernardo A. Huberman, Matt Franklin, and Tad Hogg. Enhancing privacy and trust in electronic communities. In ACM CONFERENCE ON ELECTRONIC COMMERCE. ACM, 1999.

[3]Standards for Efficient Cryptography (SEC) <http://www.secg.org/sec2-v2.pdf>