VCU College of Engineering

340 Privacy Preserving Machine Learning

Final Design Report

Prepared for

Hong-Sheng Zhou

VCU

By

Bryan Soerjanto, Minh Nguyen, David Tran, & Amaris Young-Diggs

Under the supervision of

Hong-Sheng Zhou

Date

5/1/2024

Executive Summary

This project focuses on developing a Privacy-Preserving Machine Learning (PPML) system that ensures the protection of sensitive data throughout the entire machine learning lifecycle. Using advanced cryptographic techniques such as homomorphic encryption and secure multi-party computation, this system allows organizations to extract valuable insights from data while maintaining privacy. This is particularly relevant for industries like healthcare, finance, and marketing, where the privacy of personal data is critical.

Table of Contents

Section A. Problem Statement

As machine learning becomes widely adopted across various industries, there are growing concerns about privacy leakage and unauthorized data access. Sensitive information, such as healthcare records, financial transactions, and personal identifiers, is increasingly being utilized to train machine learning models. However, this creates risks, as organizations need to analyze data without violating privacy laws or exposing sensitive information. With regulations like GDPR and HIPAA enforcing strict data protection policies, there is an urgent need for solutions that balance data utility and privacy.

Traditional approaches to privacy in machine learning often involve data anonymization or aggregation, but these methods are not foolproof and can lead to the re-identification of individuals. To address these challenges, cryptographic techniques like homomorphic encryption and secure multi-party computation have emerged as powerful methods that allow computations on encrypted data without exposing it. Additionally, frameworks like **TenSeal**, an open-source library designed to perform homomorphic encryption operations on tensors, enabling computations on encrypted data, provide a practical foundation for implementing privacy-preserving machine learning.

This project aims to leverage TenSeal's homomorphic encryption and secure computation to design a system that allows organizations to extract valuable insights from their data without compromising privacy. The focus will be on achieving a balance between privacy preservation and model performance, ensuring that the system is scalable, user-friendly, and compliant with global data privacy standards.

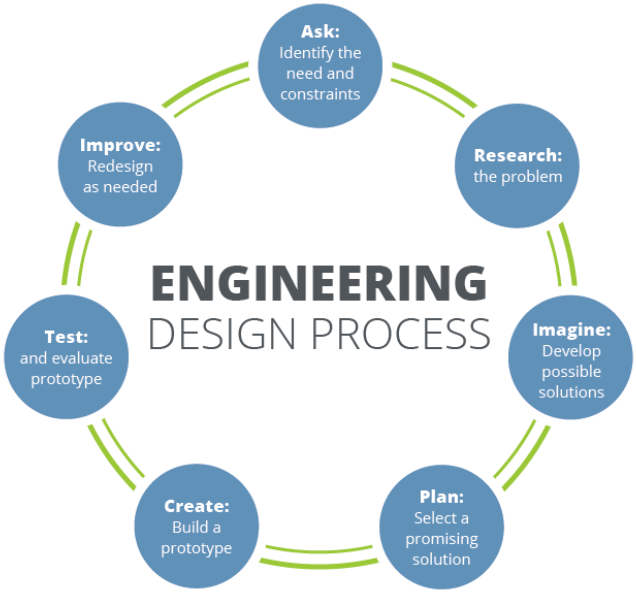

Figure 1. The iterative nature of the engineering design process [2].

Section B. Engineering Design Requirements

B.1 Project Goals (i.e. Client Needs)

- Design a privacy-preserving machine learning system that protects sensitive data during computation.
- Utilize **homomorphic encryption** to ensure computations can be performed on encrypted data without decryption, maintaining privacy throughout the process.
- Ensure compliance with data privacy regulations such as GDPR and HIPAA.
- Minimize the impact of privacy techniques on model performance.

B.2 Design Objectives

- Implement homomorphic encryption and secure multi-party computation to protect data.
- Achieve a balance between data utility and privacy preservation.
- Design an architecture that allows for scalability across multiple data sources.
- Ensure usability and accessibility for non-expert users.

B.3 Design Specifications and Constraints

- Data privacy: Ensure that no sensitive data is exposed during training or inference.
- Performance: Maintain model accuracy with less than a 10% reduction due to privacy-preserving methods.
- Compliance: Adhere to regulatory standards like GDPR and HIPAA.
- Scalability: The system must handle large datasets with minimal latency increase.
- Usability: Users should be able to deploy models with minimal cryptography knowledge.

B.4 Codes and Standards

- **GDPR** (General Data Protection Regulation)
- **HIPAA** (Health Insurance Portability and Accountability Act)
- **ISO/IEC 27001**: Information security standards
- **FIPS 140-2**: Cryptography standards for sensitive data protection

Section C. Scope of Work

C.1 Deliverables

- Privacy-preserving machine learning software.
- Technical documentation and user manuals.
- Test results showing the system's performance under different data privacy scenarios.

C.2 Milestones

- Literature review and requirement gathering (Month 1)
- Initial system architecture design and cryptographic framework selection (Month 2)
- Implementation of homomorphic encryption and secure multi-party computation (Month 4)
- Testing and validation (Month 5)
- Final deployment and report (Month 6)

C.3 Resources

- Computational resources: Cloud-based infrastructure for large-scale model training.
- Cryptographic libraries: Homomorphic Encryption Libraries like TenSeal.
- Domain expertise: Access to cryptography, machine learning, and data privacy specialists.

Section D. Concept Generation

Initial concepts for the project include:

- A centralized model using homomorphic encryption for secure data processing.
- A decentralized model using federated learning for privacy-preserving model training.
- A hybrid approach combining both homomorphic encryption and federated learning for optimal performance.

Section E. Concept Evaluation and Selection

Criterion	Homomorphic Encryption	Federated Learning	Hybrid Approach
Data Privacy	High	Medium	High
Performance Impact	High (slower)	Low	Medium
Scalability	Medium	High	Medium
Implementation Complexity	High	Medium	High

After evaluating these factors, the **hybrid approach** was selected for its balance between privacy, performance, and scalability.

Section F. Design Methodology

F.1 Computational Methods

Homomorphic Encryption (HE): Used for performing computations on encrypted data without decrypting it. The CKKS encryption scheme, implemented via libraries like TenSEAL, is utilized for efficient operations on encrypted numeric tensors.

F.2 Experimental Methods

Datasets: Real-world datasets such as healthcare records (e.g., MIMIC-III), financial transactions, and public benchmarks like MNIST and CIFAR-10 for scalability and versatility testing.

Simulations: Simulate various threat models, including honest-but-curious adversaries and malicious actors, to evaluate robustness.

F.3 Architecture/High-level Design

The system adopts a modular and scalable client-server architecture for privacy-preserving machine learning workflows:

Client-Side Encryption: Data owners encrypt sensitive data locally using homomorphic encryption or secure multi-party computation protocols. Private encryption keys are never shared with the server to ensure data confidentiality.

Server-Side Secure Computation: A server processes encrypted inputs using privacy-preserving algorithms, enabled by TenSEAL libraries.

Intermediate results remain encrypted throughout computations.

Client-Side Decryption: The encrypted outputs are returned to the client for decryption and interpretation.

Federated Learning Module: A federated learning coordinator manages model aggregation and ensures secure communication between clients.

Compliance Layer: The architecture includes mechanisms for real-time privacy policy enforcement, ensuring adherence to GDPR and HIPAA standards.

F.4 Technical Implementation

Core Technologies:

- TenSEAL for homomorphic encryption and encrypted tensor computations.
- PyTorch for neural network implementation and training.
- Docker containers to ensure reproducibility and ease of deployment.

Development Tools:

- Jupyter Notebooks for rapid prototyping.
- CI/CD pipelines for automated testing and deployment of the system.
- Monitoring tools like Prometheus and Grafana to track performance and security.

F.5 Validation Procedure

Performance Comparison:

- Baseline Models: Train and evaluate models on raw data to establish accuracy and efficiency benchmarks.
- PPML Models: Compare performance metrics (accuracy, computational efficiency, and scalability) of models trained and deployed using the proposed framework.

Privacy Assurance:

- Validate compliance with privacy standards (GDPR, HIPAA) through regular audits and penetration tests.
- Use adversarial testing to ensure resistance against inference attacks and cryptographic vulnerabilities.

End-to-End Validation:

- Conduct user studies to validate usability and system transparency for non-technical stakeholders.
- Measure client-side computational burden to ensure practicality for real-world applications.

Section G. Results and Design Details

G.1 Modeling Results

The initial modeling phase will involve:

- Training and testing models on both encrypted and unencrypted datasets to compare performance metrics such as accuracy, precision, recall, and Matthews Correlation Coefficient (MCC).
- Evaluating the impact of privacy-preserving techniques, such as homomorphic encryption and differential privacy, on model performance and computational efficiency.
- Simulating potential trade-offs, such as reduced accuracy or increased computation time, due to privacy-preserving constraints.

Goals:

- Quantify accuracy loss when using privacy-preserving techniques.
- Ensure the system achieves baseline accuracy targets (e.g., $\geq 85\%$ for classification tasks) while maintaining privacy guarantees.

G.2 Experimental Results

Datasets:

- Utilize public datasets (e.g., MNIST, CIFAR-10) for initial testing.
- Plan to incorporate domain-specific datasets, such as healthcare records or financial transactions, in subsequent phases.

Metrics to Measure:

- Accuracy: Compare encrypted vs. plaintext models.
- Computational Overhead: Assess time taken for training and inference using encrypted data.
- Privacy Metrics: Measure the effectiveness of privacy mechanisms (e.g., resistance to adversarial attacks or differential privacy epsilon values).

Goals:

- Demonstrate the system's ability to protect sensitive information while maintaining acceptable performance levels.
- Identify bottlenecks in encrypted computation and propose optimizations

G.3 Prototyping and Testing Results

Prototyping:

- Build an initial version of the client-server architecture TenSEAL for secure computations.
- Implement key features, including local encryption, federated learning, and encrypted model updates.

Testing:

- Simulate a distributed system with multiple clients to evaluate scalability.
- Test system robustness under different workloads, measuring latency and throughput.

Goals:

- Validate that the prototype can handle real-world use cases with minimal performance degradation.
- Establish proof-of-concept functionality for privacy-preserving federated learning.

G.4. Final Design Details/Specifications

Privacy-Preserving Techniques:

- Homomorphic encryption for secure computation on a central server.
- Differential privacy for protecting sensitive data during training and inference.

Federated Learning:

- Decentralized model updates using secure aggregation to prevent exposure of individual contributions.

Compliance and Security:

- Automated anonymization of sensitive data to ensure compliance with privacy regulations.
- Implementation of security protocols, such as encryption-at-rest and role-based access control (RBAC).

Deployment Plan:

- Containerized deployment using Docker to ensure portability and scalability.
- Cloud-based hosting for the central server to facilitate multi-client operations.

Goals:

- Finalize a modular design that supports privacy-preserving machine learning workflows.
- Demonstrate regulatory compliance and practical usability across diverse domains.

Section H. Societal Impacts of Design

H.1 Public Health, Safety, and Welfare

Protects sensitive health data, fostering trust in AI/ML applications in critical areas like healthcare, finance, and public infrastructure.

H.2 Societal Impacts

Encourages responsible and transparent AI adoption by addressing ethical concerns surrounding privacy and data security.

Mitigates fears about surveillance or misuse of personal information in AI systems.

H.3 Political/Regulatory Impacts

Facilitates compliance with global and regional privacy laws such as GDPR, HIPAA, CCPA, and PIPEDA, ensuring that AI systems operate within legal frameworks.

Supports policy discussions by demonstrating the feasibility of balancing innovation with data protection.

H.4 Economic Impacts

Reduces potential financial liabilities from data breaches, fines, and lawsuits, enabling organizations to safely leverage AI for economic growth.

H.5 Environmental Impacts

Optimizes computational efficiency, minimizing the environmental footprint of encrypted machine learning workflows.

Reduces the need for localized hardware by leveraging shared cloud infrastructure, thereby conserving physical resources.

H.6 Global Impacts

Encourages equitable access to privacy-preserving technologies across industries and geographies, fostering innovation in underserved areas.

Promotes global collaboration by enabling secure cross-border data sharing for research and development.

H.7 Ethical Considerations

Ensures that AI systems align with ethical principles by protecting individual autonomy and respecting data ownership.

Balances privacy with the societal benefits of AI, ensuring that marginalized groups are not disproportionately impacted by technological advancements.

Section I. Cost Analysis

- Development costs: $100,000 (estimated)
- Infrastructure costs: $20,000 (cloud computing resources)
- Maintenance and updates: $15,000 annually
- Potential savings: Reduced legal liability and compliance costs

Section J. Conclusions and Recommendations

This project addresses the critical need for privacy-preserving machine learning solutions that comply with privacy regulations and ensure data confidentiality. By adopting the hybrid approach using homomorphic encryption and federated learning, the proposed system offers a viable solution for industries such as healthcare and finance. Moving forward, we recommend scaling the system for broader adoption and conducting additional tests on larger datasets.

Appendix 1: Project Timeline

Milestone	Timeline
Literature Review and Research	Month 1
Initial Design and Planning	Month 2
Implementation and Development	Month 3-4
Testing and Validation	Month 5
Final Report and Deployment	Month 6

Appendix 2: Team Contract (i.e. Team Organization)

Step 1: Get to Know One Another. Gather Basic Information.

Task: This initial time together is important to form a strong team dynamic and get to know each other more as people outside of class time. Consider ways to develop positive working relationships with others, while remaining open and personal. Learn each other's strengths and discuss good/bad team experiences. This is also a good opportunity to start to better understand each other's communication and working styles.

Team Member Name	***Strengths each member bring to the group***	***Other Info***	***Contact Info***
Amaris Young-Diggs	*Organized, communication, leadership, inclusive, time management*	*I don't have too much experience with this type of project, but am willing to learn*	*youngdiggsac@vcu.edu*
Example: Minh Nguyen	*Communication, industry experience, problem solving, progressive*	*I enjoy being a part of a team and meeting new people. Contact through discord.*	*Nguyenms2@vcu.edu*
David Tran	*Likes to work ahead of time, good with time management and organization.*	*I like this project because it'll help with my experience for future projects and teamwork experience.*	*trand11@vcu.edu*
Bryan Soerjanto	*Team-oriented, process-driven, problem solving, organization*	*Yes*	*soerjantobj@vcu.edu*

Other Stakeholders	***Notes***	***Contact Info***
Hong-Sheng Zhou	*Our sponsor and advisor*	*hszhou@vcu.edu*

Sponsor, Mentor, etc. (Add rows if necessary)		

Step 2: Team Culture. Clarify the Group's Purpose and Culture Goals.

Task: Discuss how each team member wants to be treated to encourage them to make valuable contributions to the group and how each team member would like to feel recognized for their efforts. Discuss how the team will foster an environment where each team member feels they are accountable for their actions and the way they contribute to the project. These are your Culture Goals (left column). How do the students demonstrate these culture goals? These are your Actions (middle column). Finally, how do students deviate from the team's culture goals? What are ways that other team members can notice when that culture goal is no longer being honored in team dynamics? These are your Warning Signs (right column).

Resources: More information and an example Team Culture can be found in the Biodesign Student Guide "Intentional Teamwork" page (webpage | PDF)

Culture Goals	***Actions***	***Warning Signs***
Being on time to every meeting	*- Set up meetings in Discord* *- Send reminder Discord message in day before meeting*	*- Student misses first meeting, warning is granted* *- Student misses meetings afterwards – issue is brought up with faculty advisor*
Informing the group of any delays in completing assignments	*- Stay up to date with each other's project responsibilities* *- Set reasonable deadlines and note when an extension is needed*	*- Student shows up for weekly meeting with no considerable work done*

Respectful and Open Communication - Each team member wants to be treated with respect, listened to, and given space to express their ideas.	- Give regular updates and check-ins on task completion. - Ownership of Mistakes, Team members own up to mistakes, see them as opportunities for growth	-Set the norm that mistakes are part of the process and not to be hidden -Members isolating themselves, working without checking in or asking for feedback.

Step 3: Time Commitments, Meeting Structure, and Communication

Task: Discuss the anticipated time commitments for the group project. Consider the following questions (don't answer these questions in the box below):

- What are reasonable time commitments for everyone to invest in this project?
- What other activities and commitments do group members have in their lives?
- How will we communicate with each other?
- When will we meet as a team? Where will we meet? How Often?
- Who will run the meetings? Will there be an assigned team leader or scribe? Does that position rotate or will same person take on that role for the duration of the project?

Required: How often you will meet with your faculty advisor advisor, where you will meet, and how the meetings will be conducted. Who arranges these meetings?
See examples below.

Meeting Participants	***Frequency*** ***Dates and Times / Locations***	***Meeting Goals*** ***Responsible Party***
Students Only	*Thursday*	*Update group on day-to-day challenges and accomplishments*
Students Only	*Communicate through Discord on other days*	*Actively work on project*
Students + Faculty advisor	*TBD*	*Update faculty advisor and get answers to our questions*

Project Sponsor	*TBD*	*Update project sponsor and make sure we are on the right track*

Step 4: Determine Individual Roles and Responsibilities

Task: As part of the Capstone Team experience, each member will take on a leadership role, ***in addition to*** contributing to the overall weekly action items for the project. Some common leadership roles for Capstone projects are listed below. Other roles may be assigned with approval of your faculty advisor as deemed fit for the project. For the entirety of the project, you should communicate progress to your advisor specifically with regard to your role.

- **Before meeting with your team**, take some time to ask yourself: what is my "natural" role in this group (strengths)? How can I use this experience to help me grow and develop more?
- **As a group,** discuss the various tasks needed for the project and role preferences. Then assign roles in the table on the next page. Try to create a team dynamic that is fair and equitable, while promoting the strengths of each member.

Communication Leaders

Suggested: Assign a team member to be the primary contact for the client/sponsor. This person will schedule meetings, send updates, and ensure deliverables are met.

Suggested: Assign a team member to be the primary contact for faculty advisor. This person will schedule meetings, send updates, and ensure deliverables are met.

Common Leadership Roles for Capstone

1. **Project Manager:** Manages all tasks; develops overall schedule for project; writes agendas and runs meetings; reviews and monitors individual action items; creates an environment where team members are respected, take risks and feel safe expressing their ideas.
   Required: On Edusourced, under the Team tab, make sure that this student is assigned the Project Manager role. This is required so that Capstone program staff can easily identify a single contact person, especially for items like Purchasing and Receiving project supplies.
2. **Logistics Manager:** coordinates all internal and external interactions; lead in establishing contact within and outside of organization, following up on communication of commitments, obtaining information for the team; documents meeting minutes; manages facility and resource usage.
3. **Financial Manager:** researches/benchmarks technical purchases and acquisitions; conducts pricing analysis and budget justifications on proposed purchases; carries out

team purchase requests; monitors team budget.

4. **Systems Engineer:** analyzes Client initial design specification and leads establishment of product specifications; monitors, coordinates and manages integration of sub-systems in the prototype; develops and recommends system architecture and manages product interfaces.
5. **Test Engineer:** oversees experimental design, test plan, procedures and data analysis; acquires data acquisition equipment and any necessary software; establishes test protocols and schedules; oversees statistical analysis of results; leads presentation of experimental finding and resulting recommendations.
6. **Manufacturing Engineer:** coordinates all fabrication required to meet final prototype requirements; oversees that all engineering drawings meet the requirements of machine shop or vendor; reviews designs to ensure design for manufacturing; determines realistic timing for fabrication and quality; develops schedule for all manufacturing.

Team Member	***Role(s)***	***Responsibilities***
Minh Nguyen	*Test Engineer*	● *Designing and Developing Test Plans: Create comprehensive test plans that outline the objectives, methodologies, and procedures for conducting experiments. Ensure that these plans align with project goals and specifications.* ● *Ensuring Compliance: Verify that all tests meet industry standards, regulatory requirements, and protocols, ensuring the integrity and safety of the testing process.* ● *Collaborating with Cross-functional Teams: Work closely with other roles to align test strategies with product requirements and goals.*
Bryan Soerjanto	*Test Engineer* *System Engineer*	● *Integration management and coordination* ● *Identify design requirements* ● *Ensure design meets specified requirements* ● *Oversee design, testing, and procedures* ● *Data collection, interpretation, and analysis*
David Tran	*Project Manager,* *Logistics Manager*	● *Manages all tasks; develops overall schedule for project. Contacting advisor for instructions and assistance if needed.* ● *Writes agendas and runs meetings; reviews and monitors individual action items; creates an environment where team members are respected, take risks and feel safe expressing their ideas.* ● *Coordinates all internal and external interactions. Obtaining information for the team and following up on communication commitments.*

Amaris Young-Diggs	*Systems Engineer, Financial Manager*	• *Analyzes Client initial design specification and leads establishment of product specifications; monitors, coordinates and manages integration of sub-systems in the prototype; develops and recommends system architecture and manages product interfaces.* • *Researches/benchmarks technical purchases and acquisitions; conducts pricing analysis and budget justifications on proposed purchases; carries out team purchase requests; monitors team budget.*

Step 5: Agree to the above team contract

Team Member: Signature: Amaris Young-Diggs

Team Member: Signature: Minh Nguyen

Team Member: Signature: Bryan Soerjanto

Team Member: Signature: David Tran

References

TF Encrypted. (2024, September 29). *TF encrypted: Encrypted deep learning in TensorFlow*. https://tf-encrypted.io/

Facebook Research. (n.d.). *CrypTen* [GitHub repository]. GitHub. https://github.com/facebookresearch/CrypTen

Wikipedia contributors. (n.d.). *Homomorphic encryption*. Wikipedia. https://en.wikipedia.org/wiki/Homomorphic_encryption