THE UNIVERSITY OF HONG KONG

DEPARTMENT OF
COMPUTER SCIENCE

A Repository of Jupyter Notebooks on Unlearning in Federated Learning

Final Year Project
Final Report

Sheng "Victor" HUANG

Supervised by **Prof. S.M. YIU**

Abstract

In recent years, the growing importance of privacy and data protection has fueled interest in machine unlearning, a process that enables the removal of specific data points from trained machine learning models. This heightened focus on privacy has also led to the emergence of a decentralized learning technique called federated learning. This report presents a comprehensive overview of a final year project that aims to organize and study materials related to machine unlearning in the context of federated learning. We introduce the background knowledge, define the problem, outline the project objectives and methodologies, and present and evaluate the results. Additionally, the report explores promising directions for future research in this area. The findings of this project can serve as a valuable resource for researchers and practitioners seeking to develop more robust, efficient, and privacy-preserving machine learning systems in the federated learning context.

Keywords— machine unlearning, federated learning, GDPR, right to be forgotten, privacy, security, data deletion, memory loss, amnesia

HONG KONG
Tue 18th Apr 2023

❤️ *To my loving parents.*

Contents

List of Figures

List of Tables

List of Algorithms

1 Introduction

The primary objective of this project is to explore the intersection between two emerging areas in the field of machine learning (ML), namely machine unlearning and federated learning (FL). The focus will be on understanding how to effectively apply machine unlearning techniques within the context of federated learning. Both machine unlearning and federated learning have emerged as innovative approaches to enhancing privacy and security in machine learning applications, and as a result, they have become the subjects of extensive research and rapid development in recent years.

Given the fast-paced nature of advancements in these areas, it can be challenging for students, researchers, and other individuals new to the field to locate appropriate resources to study the foundational concepts, as well as to keep abreast of the latest developments. This project seeks to address this issue by creating a comprehensive collection of resources aimed at facilitating the study of machine unlearning and federated learning, with a particular emphasis on their intersection.

To achieve this goal, the project will compile a repository of Jupyter Notebooks that cover various aspects of the subject matter, ranging from background knowledge to cutting-edge research findings. The materials will be organized in a systematic and user-friendly manner, making it easy for newcomers to navigate and learn from the resources provided.

By offering a well-structured and accessible compilation of information on machine unlearning and federated learning, this project aims to support and encourage the growth of the research community in these domains. In doing so, it hopes to contribute to the ongoing efforts to improve privacy and security in machine learning applications, ultimately leading to more robust and trustworthy systems that can be confidently used in a wide range of real-world scenarios.

1.1 Machine Unlearning

ML is an increasingly prevalent technology that often involves using data of a personal and sensitive nature, including medical records [1], biometric identifiers [2], and geolocation information [3]. In order to ensure the continued accuracy and relevance of ML models, it is sometimes necessary to incorporate new data using various methods such as incremental learning, online learning, and data stream learning [4]. Simultaneously, there is a growing need to remove certain data and the influence of this data from ML models for various reasons.

One reason for removing data from ML models is the potential for adversarial attacks, where training data may be contaminated or poisoned by malicious data [5, 6]. Such attacks can cause ML models to malfunction or produce undesirable results. Additionally, data from unlikely scenarios may contribute to inefficient storage and even lead to inaccurate predictions [6]. Furthermore, biased data could result in discriminatory and unfair models that exacerbate existing inequalities related to race, sex, and religion [7]. In these cases, removing some data and its influence may be necessary to repair the model and mitigate negative outcomes.

Privacy concerns are another driving factor behind the need to remove data from ML models. The introduction of privacy regulations in various jurisdictions, such as the European Union's General Data Protection Regulation (GDPR) [8] and the California Consumer Privacy Act of 2018 (CCPA) [9], has led to the establishment of the **Right to be Forgotten (RtbF)** or **right to erasure.**. Under these regulations, entities may be required to erase data concerning certain individuals upon request. While deleting data from back-end databases is often straightforward, the process becomes more complex when it involves ML models that may have memorized the

data [10]. In such instances, companies may be required to remove an individual's data from their ML models [6].

The process of making ML models "forget" that they have learned from certain data is known as **machine unlearning** or simply **unlearning**. This concept was first proposed by [11] in 2015. Due to the widespread application of ML technology in today's world, the topic of machine unlearning has become increasingly important. By understanding and addressing the various reasons for removing data from ML models, researchers and practitioners can work towards developing more robust, secure, and privacy-aware systems that better serve the needs of individuals and organizations alike.

1.2 Federated Learning

FL is an innovative, privacy-aware collaborative learning method that was first proposed by [12] in 2017. The primary objective of FL is to enable participants to jointly train a machine learning (ML) model without having to share their individual datasets. This approach aims to protect the privacy of the data while still benefiting from the collective knowledge and insights held by all participants.

The core concept of FL revolves around the distribution of datasets among participants, each of whom generates a sub-update by training on their own data. Subsequently, these sub-updates are aggregated to build a comprehensive ML model. This aggregation process can take place either centrally [13] or decentrally [13, 14], depending on the specific implementation and requirements of the system.

FL offers numerous advantages, including accelerated model training speeds and the prevention of direct privacy leakage [15]. By allowing participants to retain control over their own data, FL minimizes the risks associated with data sharing and helps maintain compliance with data protection regulations, such as the GDPR and the CCPA.

However, as FL introduces significant changes to the traditional ML system architecture, it is essential to thoroughly investigate the implications of these changes on the application of existing privacy-preserving techniques within the context of FL. Researchers and practitioners must explore how established ML privacy-preserving methods can be adapted or extended to effectively function within an FL environment. This may involve examining the trade-offs between privacy protection, model accuracy, and computational efficiency, as well as developing novel techniques specifically designed for federated settings.

By studying the intersection of FL and privacy-preserving ML techniques, the research community can work towards creating more robust, secure, and privacy-aware learning systems that harness the power of collaborative learning while safeguarding sensitive information. This, in turn, can help accelerate the adoption of FL across various industries and promote the development of trustworthy artificial intelligence (AI) systems that respect individual privacy and promote data security.

1.3 Contributions

The contribution of this project lies in its potential to serve as a valuable resource for researchers, practitioners, and students in the field of machine learning. By consolidating the current state-of-the-art knowledge, this project aims to facilitate a deeper understanding of privacy and

security concerns in machine learning and to inspire further research and innovation in the areas of machine unlearning and FL.

By offering a well-structured and accessible overview of the existing work in this domain, this project can help the research community to identify knowledge gaps, recognize emerging trends, and explore potential research directions. In doing so, it contributes to the ongoing efforts to improve privacy and security in machine learning applications, ultimately leading to the development of more robust and trustworthy systems that can be used across various real-world scenarios.

2 Outline

The rest of this report is structured as follows: First, we will outline the specific problem that this project aims to address. Next, we will present the project's objectives and elaborate on the technologies and resources employed to achieve these goals. This will be followed by a comprehensive literature review that provides context for the project.

Subsequently, we will showcase the results obtained from the project and engage in a thorough discussion of the work carried out. We will then draw conclusions based on the project's findings and explore potential avenues for future research. Additional details pertaining to the project will be included in the appendices following the reference list.

3 Problem Definition

Given that FL is not entirely immune to the privacy vulnerabilities that may be present in other ML techniques [16], and considering that FL cannot replace machine unlearning since the influence of data is recorded in the sub-updates that are exchanged during the FL process, it becomes crucial to integrate unlearning techniques within the FL framework. While there have been attempts to unify the field of machine unlearning [6, 17–21] and a growing body of research investigating the application of unlearning methods to FL [15, 22–27], there remains a noticeable gap in the literature when it comes to a comprehensive and in-depth analysis of the developments and advancements within this specific sub-field.

Addressing this gap is essential for researchers, practitioners, and students to better understand the challenges, opportunities, and potential solutions related to the intersection of machine unlearning and FL. A detailed and well-structured study that introduces and summarizes the progress made in this area can contribute to the ongoing efforts to enhance privacy and security in federated learning systems and inspire further research and innovation in this promising domain.

4 Project Objectives

This project focuses on several educational and research objectives aimed at promoting and enhancing the understanding of unlearning in FL, a relatively new area that has witnessed significant research developments in recent years. The project seeks to consolidate the knowledge in this sub-field and explore the latest research advancements.

4.1 Educational Objective

The primary educational objective of this project is to offer comprehensive materials for learning about unlearning in FL. These materials, which include documentation, experiment instructions, code, and scripts, will bring together machine learning, security, and privacy insights from recent research publications in an interactive and well-organized manner. This approach will enable readers of varying expertise to learn about this topic, from fundamental concepts to cutting-edge research studies. The target audience for this project includes machine learning developers, researchers, technology lawyers, managers, and operators, all of whom can benefit from the insights provided.

4.2 Research Objectives

The research objectives for this project encompass several key aspects. The primary objective is to examine the methods proposed in recent publications for unlearning in FL, comparing their approaches to machine unlearning within the FL context. This project aims to draw conclusions from these studies and present the findings in a clear and detailed manner.

Additionally, the project seeks to identify and analyze the challenges and limitations associated with current unlearning techniques in FL, exploring potential areas for improvement or optimization. The effectiveness of different unlearning methods in FL across various application domains and scenarios will be evaluated, providing a comparative analysis of their performance and suitability.

Moreover, the project also delves into researching other privacy-preserving techniques, such as differential privacy. By examining these techniques, the project aims to investigate potential synergies between unlearning in FL and these methods, ultimately enhancing the overall privacy and security of FL systems.

5 Methodology

The methodology for this project can be divided into several stages, each contributing to the overall understanding of unlearning in FL and the exploration of other privacy-preserving techniques.

5.1 Literature Review

Throughout the duration of this project, literature searches were conducted to identify relevant research publications focusing on unlearning in federated learning (FL). The search process involved using Google Scholar, the most widely used academic search engine, with the keywords "federated" and "unlearning." Some of the relevant publications found in the search results were sourced from databases and repositories such as arXiv, ACM Digital Library, IEEE Xplore, SpringerLink, and official proceedings of conferences such as AAAI, NeurIPS, and PMLR. Access to these resources, if paywalled, was facilitated through the use of HKUL E-resources E-resources.

Upon collecting the literature, the publications were systematically reviewed and categorized based on the techniques, application domains, and challenges addressed in each study. This categorization enabled a comprehensive analysis of the various methods proposed for unlearning

in FL, as well as the exploration of other privacy-preserving techniques, such as differential privacy and secure multi-party computation. By identifying similarities, differences, and potential synergies between these methods, the review process contributed to a deeper understanding of unlearning in FL and its relationship with other privacy-preserving techniques.

The insights gained from the literature reviews were then used to inform a comparative evaluation of the different unlearning methods and privacy-preserving techniques. This evaluation assessed each method's performance, suitability, and effectiveness across various application domains and scenarios, taking into consideration their respective strengths and weaknesses, as well as identifying potential areas for improvement or optimization.

By combining these elements, this final report provides a comprehensive examination of unlearning in federated learning and its interplay with other privacy-preserving techniques. The findings are presented in a clear and detailed manner, highlighting the key contributions, challenges, and future research directions in this promising field.

5.2 Notebooks

In this project, Jupyter Notebooks were employed as the primary means of documenting and presenting the background knowledge, key concepts, framework designs, theoretical analyses, and experiment setup and results from various works. Jupyter Notebooks are a versatile web application designed for creating and sharing computational documents that offer a user-friendly, document-centric experience. This approach not only facilitates the presentation of essential information from the respective publications, such as key figures and tables, but also enables the incorporation of code, instructions for setting up the environment, installing required packages, and running the code for some of the works.

By using Jupyter Notebooks, users can easily explore different combinations of unlearning techniques, frameworks, datasets, and models by interacting with the provided code and modifying it as needed. This interactive format allows users to build upon the provided notebooks to design their own unlearning frameworks or even adapt these academic outcomes for real-world applications.

To further enhance the accessibility and user experience, all notebooks have been made compatible with Google Colab. This compatibility allows users to conveniently view, edit, and run the notebooks within the Google Colab environment without needing to set up a local environment.

It is worth noting that due to the decentralized nature of FL, some code may not be suitable for direct execution within Jupyter Notebooks. In such cases, instructions have been provided for running the code using the terminal, ensuring that users can still interact with and understand the underlying processes and techniques effectively.

5.3 Experiments

This project carried out experiments on the HKUCS GPU Farm and using GPU sessions on Google Colab, as ML systems typically train more quickly on GPUs due to the large volume of tensor calculations involved. In order to facilitate the experimental process, testing and development environments were constructed using technologies such as Miniconda, CUDA Toolkit, and packages employed in related publications.

The project conducted comparative experiments using various techniques proposed in different publications and applied them to datasets summarized by [6], which include MNIST, CIFAR,

SVHN, and Adult. By modifying the code provided in the reviewed publications, the project aimed to eliminate bugs and test the performance of the code under different parameter settings and to achieve the performance claimed in their repective publications. The resulting code or instructions to create the resulting code are documented in Jupyter Notebooks in more detail.

5.4 Repository and Website

This project's Jupyter Notebooks are hosted in a GitHub repository[1]. Within the repository, the notebooks are stored in the `notebooks` folder. In addition to this folder, the `README.md` file provides a description of the project and the repository, as well as information on the content of the notebooks and related references.

The project website[2] is also hosted within this repository, utilizing GitHub Pages for deployment. To streamline the process of updating and maintaining the website, GitHub Actions, a world-class CI/CD (Continuous Integration/Continuous Deployment) tool, is employed. The website is built using Hugo, one of the most popular open-source static site generators that support the Markdown language.

The chosen set of technologies for this project, including GitHub, GitHub Pages, GitHub Actions, and Hugo, were selected due to their widespread use in the industry, ease of use, and powerful functionality. By utilizing these tools, the project ensures a seamless experience for users who wish to access the notebooks, while also maintaining a high standard of quality and organization.

6 Literature Review

6.1 Machine Unlearning

6.1.1 Goal of Machine Unlearning

A simplistic approach to machine unlearning involves retraining the model from scratch using the remaining data, which includes all data except the ones targeted for erasure. While this method is straightforward, it is also computationally expensive and inefficient. However, by examining this scenario, we can glean the desired outcomes for unlearning. Specifically, we want models that have undergone unlearning to be "equivalent" to, or within the same distribution as, models trained from scratch using the dataset $S \setminus X$, where S represents the original dataset, and X denotes the data to be erased [28].

It is important to recognize the subtle differences between machine unlearning and data deletion. Machine unlearning focuses on the model perspective, while data deletion centers on the data itself [6]. To put it in simpler terms, the goal of machine unlearning is to induce "amnesia" in machine learning models, effectively causing them to lose specific memories. Ideally, this process should be efficient and precise, allowing for the preservation of privacy, security, usability, and fidelity in machine learning systems. By accomplishing this, we can ensure that the models remain reliable and effective, while also adhering to the increasing demands for privacy and data protection in today's digital landscape.

[1] https://github.com/vicw0ng-hk/feul
[2] https://vicw0ng-hk.github.io/feul/

6.1.2 Comparison with Differential Privacy

There is another approach for addressing privacy concerns in machine learning, known as **differential privacy (DP)** [29]. However, it is crucial to understand that DP and machine unlearning are distinct concepts, each providing different privacy guarantees. In the case of machine unlearning, the specific unlearning process ensures that any influence the unlearned data had on the model is completely eliminated [6]. In contrast, ε-differential privacy, for any non-zero ε, provides a bounded influence on any data point, but this bound remains non-zero. Essentially, while it limits the influence of individual data points, it does not entirely remove their impact.

A 0-differential privacy, which theoretically achieves zero influence, would render the learning algorithm incapable of learning anything from the data, thus defeating the primary purpose of machine learning [6, 28]. Consequently, it is essential to distinguish between these two privacy-preserving techniques and recognize that they serve different purposes in the broader context of machine learning. By understanding their unique characteristics and applications, researchers and practitioners can make more informed decisions when selecting the appropriate method to protect privacy and uphold data integrity within their machine learning systems.

6.1.3 Challenges of Machine Unlearning

Much like inducing memory loss with precision and efficiency is a challenging task, it is important to acknowledge that machine unlearning also presents significant difficulties [6]. There are numerous challenges that need to be addressed in order to make machine unlearning a viable solution.

Firstly, there is the issue of **stochasticity of training**. It is not a straightforward process to identify and map the impact of a single data point on the training, particularly in complex models such as deep neural network (DNN). This complexity is further exacerbated by the inherent randomness that occurs during the training process [28].

Secondly, there is the challenge of **incrementality of training**. The influence that data X_0 has on the model when trained at time t_0 persists and continues to impact subsequent training at t_i, where $t_0 < t_i$ with X_i. Conversely, the model training with X_0 at t_0 is also influenced by training that occurred prior to t_0. Determining the precise influence that needs to be removed from the model in order to effectively unlearn X_0 is a complex and challenging problem [6].

Moreover, recent studies have found that unlearned models tend to underperform compared to models trained from scratch using the remaining data. This performance degradation worsens as more data is unlearned [30, 31]. This phenomenon, known as **catastrophic unlearning** [31], is difficult to prevent and poses a significant obstacle to the successful implementation of machine unlearning.

In conclusion, the process of machine unlearning is fraught with challenges that must be addressed before it can become a reliable and efficient solution for privacy preservation in machine learning. Researchers must continue to explore and develop innovative techniques to overcome these hurdles and improve the effectiveness of machine unlearning.

6.1.4 Overview of Machine Unlearning

The comprehensive 2022 survey by Nguyen *et al.* [6] offers a valuable summary of recent advancements in the field of machine unlearning. In their work, they categorize machine unlearn-

ing strategies into three distinct approaches: model-agnostic, model-intrinsic, and data-driven methods. Each of these approaches is examined in the context of various unlearning scenarios, design requirements, and unlearning requests, providing a broad perspective on the current state of the art.

Furthermore, the authors delve into the applications of machine unlearning, including its application to federated learning, which is the primary focus of this project. The survey not only sheds light on the latest research in federated unlearning [15, 22, 24, 26] but also explores the future direction of the field as a whole.

The insights provided by [6] serve as a valuable high-level overview of the topic of machine unlearning, enabling researchers to better understand the current landscape and potential avenues for improvement. By considering the various methods and strategies presented in the survey, researchers may find inspiration to adapt and apply ideas from other forms of ML to FL, further enhancing the development of novel techniques and approaches in federated unlearning.

In summary, the 2022 survey by Nguyen *et al.* [6] represents a pivotal resource for researchers in the field of machine unlearning, offering a comprehensive perspective on the latest developments and the challenges that lie ahead. By focusing on the intersection of machine unlearning and federated learning, this project aims to contribute to the ongoing exploration of innovative techniques and strategies for enhancing privacy and security in the realm of ML.

6.2 Unlearning in FL

6.2.1 Additional challenges

Unlearning in the context of FL presents an even greater challenge than unlearning in centralized ML, due to a variety of factors that add to its complexity. One such factor is the use of aggregation instead of raw gradients to compute global weights in FL. This process can become increasingly difficult to manage as the number of participating clients grows, further complicating the unlearning process [25].

Another factor contributing to the increased complexity of unlearning in FL is the issue of data partitioning and statistical heterogeneity. In particular, the distinctions between vertical and horizontal FL, as well as the presence of non-IID (independent and identically distributed) data, pose additional challenges for unlearning techniques in this setting [32].

Furthermore, the potential for overlapping data among participating clients introduces yet another layer of complexity to the unlearning process in FL. While most proposed methods make the assumption that the data to be removed is exclusive to a single client, this may not always be the case in practice [6, 15, 22, 26]. The presence of shared or overlapping data among clients must be carefully considered when developing and implementing unlearning techniques for FL.

In summary, unlearning in FL is a considerably more complex and challenging endeavor than unlearning in centralized ML systems. The unique characteristics of FL, such as aggregation, data partitioning, statistical heterogeneity, and overlapping data among clients, all contribute to the increased difficulty of effectively implementing unlearning techniques in this environment. As research continues to explore and address these challenges, it is essential to develop innovative strategies and methods that can effectively handle the intricacies of unlearning in FL.

6.2.2 Methods

Liu *et al.* [22] pioneered the development of an unlearning method for FL, which relies on calibration training to isolate the contributions of a target client to the central model. While this approach represents an important first step in addressing unlearning in FL, it has certain limitations in terms of scalability. In particular, its performance on more complex models, such as DNNs, is less than satisfactory.

Recognizing the need for more scalable solutions, Wu *et al.* [26] proposed an alternative unlearning method based on knowledge distillation. This approach does not require the direct use of participating clients' data during the unlearning process, making it more compatible with complex models like DNNs. As a result, this method represents a significant improvement in terms of both flexibility and applicability.

However, the quest for effective unlearning methods in FL did not stop there. Liu *et al.* [23] introduced a rapid retraining approach designed to fulfill the requirements of unlearning. This method leverages the L-BFGS (Limited-memory Broyden-Fletcher-Goldfarb-Shanno) algorithm to compute a Hessian approximation using historical parameter updates. Although this approach offers certain advantages in terms of retraining speed, it also suffers from scalability issues similar to those encountered with the method proposed by Liu *et al.* [22].

These are just some examples of early attempts at solving the problem. Continued research and exploration of innovative techniques will be essential in order to overcome these limitations and unlock the full potential of unlearning in FL. More work done in this area will be discussed later in this report.

6.2.3 Verification

In traditional ML scenarios, there are several methods available to verify unlearning, including membership inference attacks [6, 22] and backdoor attacks [6, 15, 27]. However, the situation becomes more complex in the context of FL, where the introduction of adversarial attacks for verification purposes may inadvertently compromise the security features inherent to FL. Additionally, the participation of various clients can have subtle, yet significant effects on the output space, further complicating the verification process [25].

Recognizing these challenges, Gao *et al.* [25] proposed a novel verification mechanism specifically tailored to the FL setting. This approach takes advantage of the same communication channels employed during the training phase and is capable of verifying unlearning with just a few rounds of communication. By using this method, the verification process can be conducted more efficiently and securely, without compromising the unique features of FL.

This innovative verification mechanism highlights the need for continued research and development of specialized techniques for unlearning in the context of FL. As the field continues to evolve, it will be crucial to identify and address the unique challenges associated with unlearning verification in order to ensure the integrity, security, and effectiveness of FL systems.

6.3 Related Papers

6.3.1 FedEraser [22]

The proposed FedEraser methodology is the first design to remove the influence of a federated client's data on the global FL model while significantly reducing the time used for constructing the unlearned FL model. The authors develop novel storage-and-calibration techniques to tackle

the forward coupling of information in parameter updates, which can provide a significant speed-up to the reconstruction of the unlearned model while maintaining its efficacy. They also propose a new indicator that uses the layer parameter's deviation between the unlearned model and the retrained model to measure FedEraser's effectiveness on the global model.

The authors evaluate FedEraser's performance on four realistic datasets and compare it with two baselines. The results demonstrate that FedEraser can remove the influence of target client data with an expected speed-up of 4x compared with retraining from scratch. The experiments show that FedEraser achieves high accuracy in removing client data while maintaining overall model performance.

One key advantage of FedEraser is its non-intrusive nature, which allows it to serve as an opt-in component inside existing FL systems. It does not involve any information about the target client, enabling unwitting unlearning processes. This feature makes it particularly useful for scenarios where privacy concerns or malicious attacks require efficient removal of specific client-level data from trained models.

Overall, this paper presents a novel solution to an important problem in FL by proposing a methodology that enables efficient removal of specific training data from trained models. The results demonstrate that FedEraser achieves high accuracy in removing client data while maintaining overall model performance and provides significant speed-ups compared with retraining from scratch. These findings have significant implications for practical applications of FL, particularly in scenarios where privacy concerns or malicious attacks require efficient removal of specific client-level data from trained models.

6.3.2 RevFRF [15]

RevFRF is a novel framework that enables cross-domain random forest training with revocable FL. It introduces a suite of homomorphic encryption-based secure protocols to implement federated random forest, which allows efficient distributed machine learning without direct revealing of private participant data. The framework addresses the participant revocation problem in FL by implementing two levels of revocation. For the first-level revocation, RevFRF ensures that the data of an honest revoked participant in the trained RF model cannot be utilized by the remaining participants. For the second-level revocation, RevFRF further ensures that if a revoked participant is dishonest, it cannot get back to utilize the data of remaining participants memorized by the trained RF model.

The experiment results prove that RevFRF is effective and secure under the curious-but-honest model. The authors conducted experiments on three real-world datasets and compared RevFRF with other state-of-the-art methods. The results show that RevFRF achieves comparable or better performance while ensuring privacy and security. Overall, RevFRF provides a practical solution for cross-domain random forest training with revocable FL in real-world scenarios.

6.3.3 Bayesian Federated Unlearning [33]

This paper proposes a novel approach to federated learning and unlearning in decentralized networks within a Bayesian framework. The authors develop federated variational inference (VI) solutions based on the decentralized solution of local free energy minimization problems within exponential-family models and on local gossip-driven communication. The proposed protocols are shown to yield efficient unlearning mechanisms, allowing agents to exercise their right to be forgotten and have their contribution to the jointly trained model deleted.

The authors demonstrate the effectiveness of their approach through numerical simulations, showing that their method outperforms existing approaches in terms of both accuracy and communication efficiency. They also discuss potential real-world applications of federated learning and unlearning, such as in healthcare or finance, where data privacy is crucial but collaboration is necessary for accurate decision-making. Overall, this paper presents an innovative solution to the challenges of collaborative machine learning in decentralized networks while maintaining data privacy and security.

6.3.4 Knowledge Distillation [26]

This paper proposes a novel federated unlearning method that eliminates a client's contribution by subtracting the accumulated historical updates from the model and leveraging the knowledge distillation method to restore the model's performance without using any data from the clients. The proposed method does not have any restrictions on the type of neural networks and does not rely on clients' participation, making it practical and efficient in the FL system. The authors also introduce backdoor attacks in the training process to help evaluate the unlearning effect.

The authors conducted empirical studies on three canonical datasets, including MNIST, CIFAR-10, and EMNIST, to demonstrate the effectiveness and efficiency of their proposed method. They compared their method with two existing methods: Fine-tuning and Re-initialization. The results show that their proposed method outperforms both existing methods in terms of accuracy, convergence speed, and communication efficiency.

Moreover, they evaluated their proposed method under different scenarios, such as varying numbers of clients and different percentages of malicious clients. The results show that their proposed method is robust against malicious clients' attacks and can effectively eliminate a client's contribution without affecting other clients' performance.

In conclusion, this paper proposes a practical and efficient federated unlearning method that can help FL models comply with recent legislation on the right to be forgotten. The authors demonstrate its effectiveness through empirical studies on three canonical datasets and under different scenarios. Their proposed method outperforms existing methods in terms of accuracy, convergence speed, and communication efficiency while being robust against malicious clients' attacks.

6.3.5 Class-Discriminative Pruning [24]

This paper proposes a novel approach to selectively forgetting categories from trained convolutional neural network (CNN) classification models in federated learning. The authors define the problem of selectively forgetting categories and propose a scrubbing procedure that cleans information about particular categories from the trained model without requiring global access to the training data or retraining from scratch. The proposed method is evaluated on two datasets, CIFAR-10 and CIFAR-100, and compared to the gold standard model trained from scratch without any knowledge of the target category data. The results show that the proposed method outperforms other methods in terms of unlearning speed while maintaining accuracy. This work contributes as a complementary block for compliance with legal and ethical criteria in real-world applications. The authors envision this work as an early step towards federated unlearning, which has potential implications for various machine learning models beyond CNNs.

6.3.6 Rapid Retraining [23]

This paper presents a novel approach to machine unlearning in FL, which enables data holders to erase their data from a trained model even when full access to all training data is unavailable. The proposed approach leverages the first-order Taylor expansion approximation technique to customize a rapid retraining algorithm based on diagonal experience Fisher Information Matrix (FIM). To further boost model utility, the momentum technique is introduced into the unlearning update strategy to alleviate the negative impact caused by approximation errors.

The authors conducted extensive experiments over four datasets in terms of the ability to achieve unlearning, efficiency, model performance, and parameter sensitivity. Compared to the baseline retraining, their approach achieved a speedup of 9.1x with an accuracy loss of $2.235x10^{-3}$ on the large dataset CelebA. The experimental results demonstrate that their approach can effectively and efficiently achieve machine unlearning in FL scenarios.

The paper also provides a comprehensive theoretical analysis of the proposed algorithm and its convergence properties. The authors prove that their algorithm converges linearly under mild assumptions and provide an upper bound on its convergence rate.

Overall, this paper presents an efficient and effective approach to machine unlearning in FL scenarios that can be applied to any machine learning model. The experimental results demonstrate that their approach achieves significant speedup while maintaining high accuracy levels compared to baseline retraining methods. This work has important implications for privacy-sensitive applications of machine learning where data holders may want to remove their data from trained models without compromising overall model performance.

6.3.7 Forget-SVGD [34]

This paper proposes a novel Bayesian federated unlearning method called Forget-SVGD, which leverages non-parametric Bayesian approximate inference to develop a flexible and effective approach to unlearning. The method builds on the existing SVGD and DSVGD methods, using particle-based approximate Bayesian inference schemes with gradient-based deterministic updates.

In the proposed method, local SVGD updates are carried out at agents whose data needs to be "unlearned" interleaved with communication rounds with a parameter server. The authors demonstrate the effectiveness of Forget-SVGD through numerical experiments on both synthetic and real-world datasets. The results show that Forget-SVGD outperforms existing methods in terms of accuracy and efficiency in unlearning tasks.

Overall, this paper presents a promising approach to federated unlearning that can help address privacy concerns in machine learning by allowing agents to selectively forget their data while preserving the overall model accuracy.

6.3.8 VeriFi [25]

This paper proposes a unified framework for federated unlearning and verification, called VeriFi, which allows for the systematic analysis of the verifiable amount of unlearning with different combinations of unlearning and verification methods. The authors conduct the first systematic study in the literature for verifiable federated unlearning, with seven unlearning methods (including newly proposed ^{u}S2U) and five verification methods (including newly proposed ^{v}EM and ^{v}FM), covering existing, adapted, and newly proposed ones for both unlearning and verification.

The authors prompt the concept of verifiable federated unlearning, which is an important aspect that has been overlooked in the current literature. They propose VeriFi as a solution to this problem. In VeriFi, a leaving participant is granted the right to verify (RTV), meaning that they notify the server before leaving and then actively verify the unlearning effect in the next few communication rounds. The unlearning is done at the server-side immediately after receiving the leaving notification, while the verification is done locally by the leaving participant via two steps: marking and checking.

The authors evaluate their framework on two datasets: MNIST and CIFAR-10. They show that their proposed ^{u}S2U method outperforms existing methods in terms of accuracy while achieving comparable or better privacy protection. They also demonstrate that their proposed ^{v}EM method can effectively detect malicious participants who do not follow RTV protocol.

Overall, this paper presents an important contribution to federated learning research by addressing an overlooked aspect of implementing RtbF in federated learning - independent verification of unlearning effects. The authors' proposed framework provides a practical solution to this problem while achieving high accuracy and privacy protection. Their results demonstrate that their framework can effectively detect malicious participants who do not follow RTV protocol while outperforming existing methods in terms of accuracy.

6.3.9 Efficient Client Erasure [27]

This paper explores the problem of removing any client's contribution in FL. During FL rounds, each client performs local training to learn a model that minimizes the empirical loss on their private data. The authors propose to perform unlearning at the target client (to be erased) by training a model to maximize the empirical loss. They formulate the unlearning problem as a constrained maximization problem by restricting to an ℓ_2-norm ball around a suitably chosen reference model. The average of the other clients' local models (except the local model of the target client) in the last round of FL training is an effective reference model, as it helps to retain some knowledge learned from other clients' data. The authors evaluate their approach on MNIST digit recognition task and show that it outperforms existing methods for removing clients from FL. Their proposed method can help ensure privacy and compliance with regulations such as GDPR by allowing users to exercise their right to be forgotten.

6.3.10 FedRecover [35]

This paper proposes a novel approach to recover the global model in federated learning systems that have been subjected to poisoning attacks. The proposed method, called FedRecover, uses historical information to identify and remove malicious clients and restore the global model without having to train from scratch.

The authors first introduce the concept of federated learning and explain how it is vulnerable to poisoning attacks, where malicious clients send poisoned model updates to the server. They then present FedRecover as a solution that can effectively detect and remove these malicious clients while preserving the accuracy of the global model.

FedRecover uses several strategies such as warm-up, periodic correction, abnormality fixing, and final tuning to optimize the recovery process. The server asks clients to compute and communicate their exact model updates, which are then used to recover the global model. The authors show that under certain assumptions, the global model recovered by FedRecover is close or identical to that recovered by training from scratch.

The authors evaluate FedRecover on four datasets and three federated learning methods with both untargeted and targeted poisoning attacks. The results show that FedRecover is both accurate and efficient in recovering from these attacks. In particular, it outperforms existing defense mechanisms such as robust federated learning methods and detecting malicious clients when there are a large number of them.

Overall, this paper presents an innovative approach for recovering from poisoning attacks in federated learning systems. By leveraging historical information and optimizing the recovery process through various strategies, FedRecover can effectively detect and remove malicious clients while preserving the accuracy of the global model. The results demonstrate its effectiveness in mitigating poisoning attacks on federated learning systems.

6.3.11 Federated Clusters [36]

This paper proposes a novel unlearning mechanism for federated clustering with privacy criteria, allowing for efficient data removal at both the client and server level. The approach combines special initialization procedures with quantization methods to enable secure aggregation of estimated local cluster counts. The proposed method achieves good clustering performance, low computational and communication complexity while meeting the privacy criteria that are commonly used in classical FL literature. Furthermore, an intuitive and efficient exact unlearning algorithm is proposed to accommodate data removal requests within the federated clustering framework. Empirical studies on seven benchmark datasets established fundamental performance-complexity trade-offs between unlearning and complete retraining. The results show that the proposed method outperforms existing methods in terms of efficiency and accuracy while ensuring data security. This work has significant implications for practical applications such as personalized recommender systems and healthcare systems where machine unlearning is important due to recent laws ensuring the "right to be forgotten".

6.3.12 Informed Federated Unlearning (IFU) [37]

The paper introduces a novel approach to federated unlearning, IFU, that identifies the optimal iteration of Federated Learning (FL) from which FL has to be reinitialized. The approach comes with theoretical quantification of its effectiveness and is extended to account for sequential unlearning requests. The authors experimentally demonstrate the effectiveness of IFU on a series of benchmarks and show that it leads to more efficient unlearning procedures compared to basic re-training and state-of-the-art Federated Unlearning approaches. The results show that IFU can unlearn a series of forgetting requests while satisfying the unlearning guarantees. Overall, the paper presents an innovative approach to federated unlearning that can be applied in various real-world scenarios, making it an important contribution to the field of ML.

6.3.13 General Pipeline [38]

This paper proposes a general pipeline for efficient federated unlearning against three types of requests: class, client, and sample. The authors revisit how training data affects the final FL model performance and propose a framework that empowers the proposed pipeline with reverse stochastic gradient ascent (SGA) and elastic weight consolidation (EWC). The goal of this work is to address the challenges associated with fully retraining FL models, including accuracy, unlearning privacy, model agnostic, and unlearning efficiency.

The authors conduct various experiments to verify the effectiveness of their proposed method in both aspects of unlearning efficacy and efficiency. The results show that their proposed

method outperforms existing methods in terms of accuracy and efficiency. Additionally, they demonstrate the effectiveness of their method in real-world scenarios where federated unlearning may be necessary. Overall, this paper provides a promising approach for efficient federated unlearning that can help address privacy concerns related to FL models while maintaining high accuracy.

6.3.14 Subspace-based [39]

This paper introduces a novel Subspace-based Federated Unlearning (SFU) framework that allows for the removal of a specific client's contribution in federated learning without requiring additional storage. The main insight of SFU is to constrain the gradients generated by the target client's gradient ascent to the input subspace of other clients to remove their contribution from the global model. The authors conducted extensive experiments to determine the effectiveness of SFU in removing target clients' contributions while ensuring model accuracy and robustness to data heterogeneity and training degree. The results show that SFU can effectively remove target clients' contributions while maintaining model accuracy and has strong robustness to data heterogeneity and training degree. Additionally, the authors designed a differential privacy method to prevent possible privacy leakage caused by transmitting client representation matrices during unlearning. Overall, SFU provides a promising solution for federated unlearning that can be used in scenarios where server storage resources are constrained.

6.3.15 FedLU [40]

The paper proposes a novel Federated Learning (FL) framework for heterogeneous Knowledge Graph (KG) embedding learning and unlearning, called FedLU. The framework addresses the challenges of data heterogeneity and knowledge forgetting in FL by using mutual knowledge distillation to transfer local knowledge to global and absorb global knowledge back. Additionally, the authors present a KG embedding unlearning method that combines retroactive interference and passive decay to achieve knowledge forgetting.

The authors conducted extensive experiments on newly-constructed datasets with varied numbers of clients, showing that FedLU outperforms state-of-the-art methods in both link prediction and knowledge forgetting. The results demonstrate the effectiveness of the proposed framework in addressing the challenges of data heterogeneity and knowledge forgetting in FL for KG embedding learning. Overall, this paper presents a promising approach for federated KG embedding that can fully take advantage of knowledge learned from different clients while preserving the privacy of local data.

6.3.16 Federated Recommendation Unlearning (FRU) [41]

This paper proposes a novel approach to address data privacy concerns in recommendation systems. The authors introduce Federated Recommender Systems (FedRecs) that enable users to learn personal interests and preferences from their on-device interaction data while also allowing for efficient erasure of specific users/clients' influence. The proposed method, FRU, stores each client's historical changes locally on their devices, and uses negative sampling and importance-based update selection mechanisms to improve storage space efficiency on resource-constrained devices. FRU rolls back FedRecs to erase the target users/clients' influence and fast recovers FedRecs by calibrating the historical model updates.

The authors evaluate the proposed approach using two real-world datasets, demonstrating that FRU can effectively erase the influence of specific clients while maintaining high recommendation

accuracy. The results show that FRU outperforms existing methods in terms of both unlearning effectiveness and efficiency. Overall, this paper presents a promising solution to address data privacy concerns in recommendation systems while still providing accurate recommendations.

6.3.17 Knot [42]

This paper presents Knot, a novel solution to the challenging problem of machine unlearning in the context of federated learning. Knot allows clients to request the erasure of their private data without incurring prohibitive retraining costs. The authors evaluate Knot's performance on a variety of datasets and tasks and show clear evidence that it outperforms state-of-the-art federated unlearning mechanisms by up to 85% in the context of asynchronous federated learning.

The authors also demonstrate that Knot is robust against membership inference attacks, which are commonly used to evaluate the effectiveness of approximation algorithms. Overall, this paper presents an important contribution to the field of federated learning, as it addresses a crucial challenge related to privacy and data protection. Knot has potential applications in real-world scenarios where users have the right to erasure regarding their own private data used for training neural network models.

7 Results

7.1 Repository and Website

The repository[3] and website[4] are created according to section 5.4. Figures 1 and 2 show screenshots of the repository and website, respectively, before this report and final presentation slides have been finalized.

Table 1: Contents of the repository.

`.github/workflows`	`pages.yml`: file needed for the GitHub Action as described in section 5.4.
`notebooks`	See section 7.2.
`site`	Files for the website built according to section 5.4.
`.gitignore`	File to ignore certain files when doing git version control.
`.gitmodules`	File telling git to include the module of the Hugo theme used in the website.
`LICENSE`	The Unlicense.
`README.md`	A README file describing the project and the contents of the repository.

7.2 Notebooks

The Jupyter Notebooks of this project are organized under the `notebooks` folder of the repository. One can use the `README.me` file to check the structure of the notebooks and open any notebook using the link provided there. There is an emoji before each link to a notebook, specifically the open book emoji 📖, representing that the notebook is a review of the paper, and the laptop emoji 💻, representing that the notebook is instructions on how to run the code for the paper. Depending on the operating system and emoji font of your device, they may look different but the concept should be clear. Clicking on these emojis will open the correcponding notebook in Google Colab. References to the notebooks are also noted in `README.md`.

[3] https://github.com/vicw0ng-hk/feul
[4] https://vicw0ng-hk.github.io/feul/

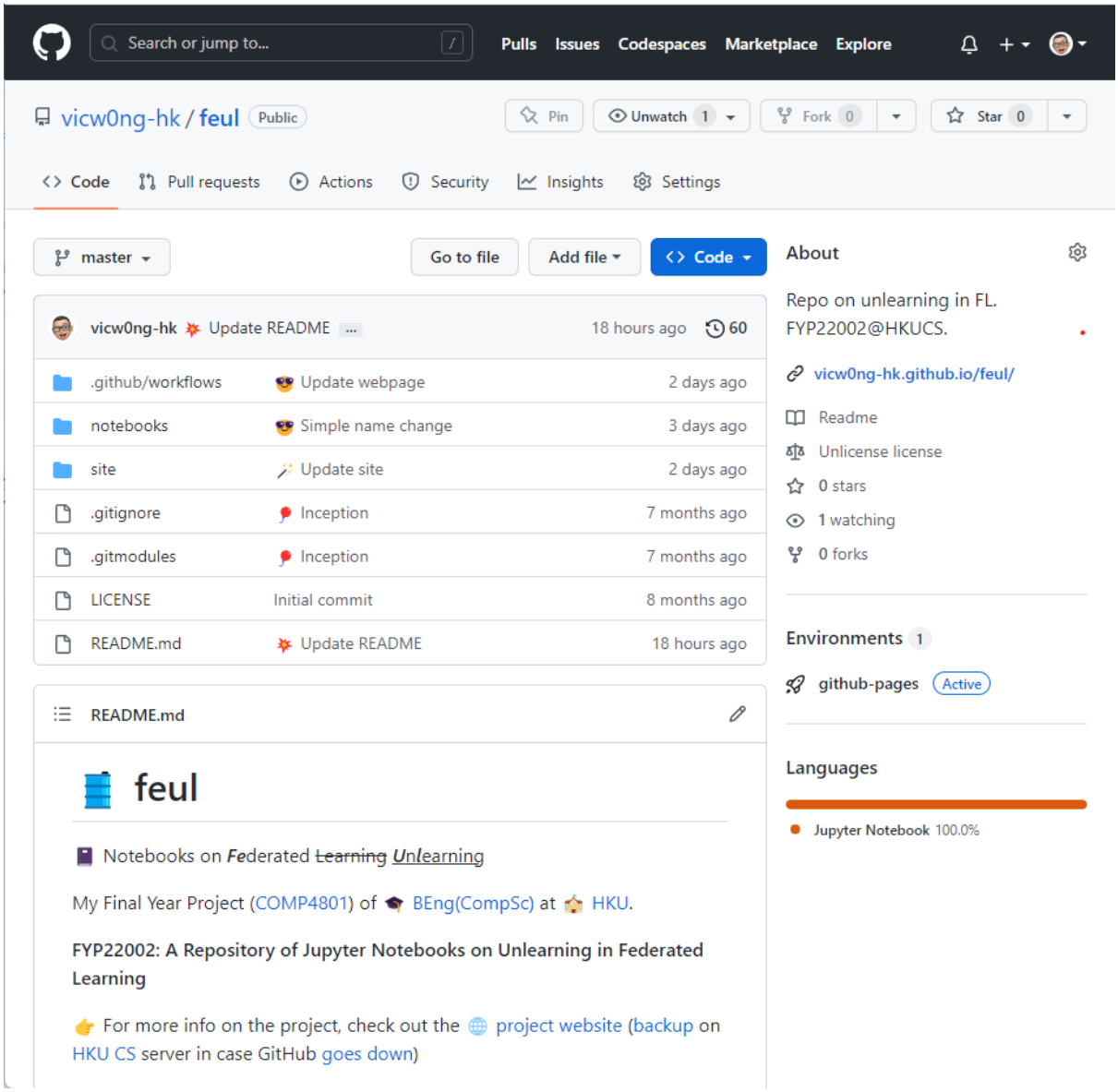
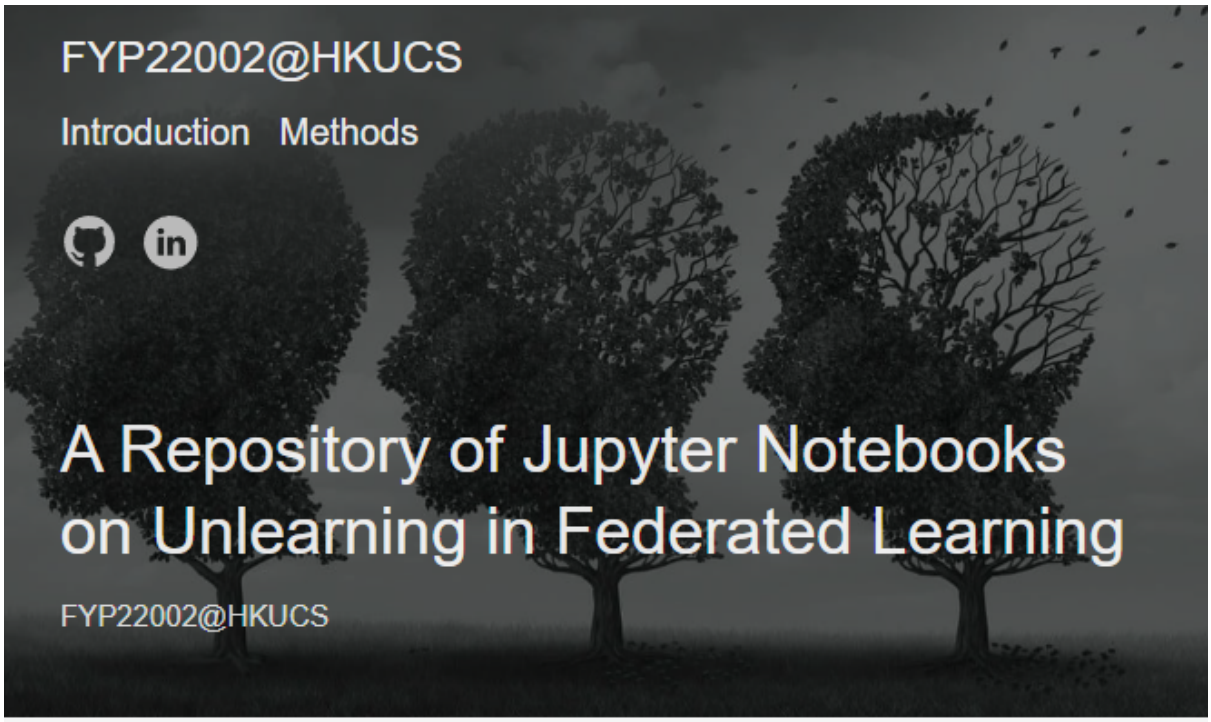


Figure 1: Repository of Jupyter Notebooks hosted on GitHub, before Final Report and Presentation slides have been updated.



Welcome to the BEng(CompSc) FYP of Victor S. HUANG, supervised by Prof. S.M. YIU.

🎉 👉 Check out the repo 👈 🎉

Introduction | Methods

Update

1st Presentation
Final Presentation

Project Plan | Interim Report | Final Report

Made with ❤️ by Victor

Figure 2: Website of this project, before Final Report and Presentation slides have been updated.

Table 2: Contents of the `notebooks` folder.

Folder	Contents	References
`00-intro-ul-fl`	Introduction and code on unlearning and FL.	[6, 11, 12, 28, 32, 43–46]
`01-ul-more`	More details on unlearning.	[6]
`02-un-in-fl`	More details on unlearning in FL.	[6, 12]
`03-liu+21a`	Detailed paper and code review.	[22]
`04-liu+21b`	Detailed paper review.	[15]
`05-gsk21`	Detailed paper review.	[33]
`06-wzm22`	Detailed paper review.	[26]
`07-wan+22`	Detailed paper review.	[24]
`08-liu+22`	Detailed paper and code review.	[23]
`09-gon+22`	Detailed paper review.	[34]
`10-gao+22`	Detailed paper review.	[25]
`11-hal+22`	Detailed paper review.	[27]
`12-cao+22`	Detailed paper review.	[35]
`13-pan+22`	Detailed paper review.	[36]
`14-fra+22`	Detailed paper review.	[37]
`15-wu+22`	Detailed paper review.	[38]
`16-li+23`	Detailed paper review.	[39]
`17-zlh23`	Detailed paper review.	[40]
`18-yua+23`	Detailed paper review.	[41]
`19-sl23`	Detailed paper and code review.	[42]

7.2.1 `00-intro-ul-fl`

There are 4 notebooks under this folder: `intro-unlearning.ipynb`, `intro-fl.ipynb`, `code-amnesiac-ml.ipynb` and `code-flwr.ipynb`. The first two are concept reviews and the last two are code examples to create and run unlearning and FL. Figures 3 to 6 are screenshots of these notebooks opened using GitHub's preview function.

7.2.2 `01-ul-more`

There are 2 notebooks under this folder: `unlearning-definition.ipynb` and `unlearning-framework.ipynb`. The first discusses the definitions od the unlearning problem that are usually assume in most related papers, while the second talks about common unlearning frameworks proposed and used. Tests were carried out on Google Colab using a GPU session, as well as on HKUCS GPU Farm using a GPU instance.

7.2.3 `02-un-in-fl`

There is one notebook under this folder: `ul-in-fl.ipynb`. This notebook talks about the early efforts made to extend unlearning into the realm of FL, showcasing how research progressed.

7.2.4 `03-liu+21a`

There are 2 notebooks under this folder: `federaser.ipynb` and `code-federaser.ipynb`. The first discusses the work done by [22] (see section 6.3.1) and the second provides instructions on how to run the code for the design proposed in the paper. Tests were carried out on Google

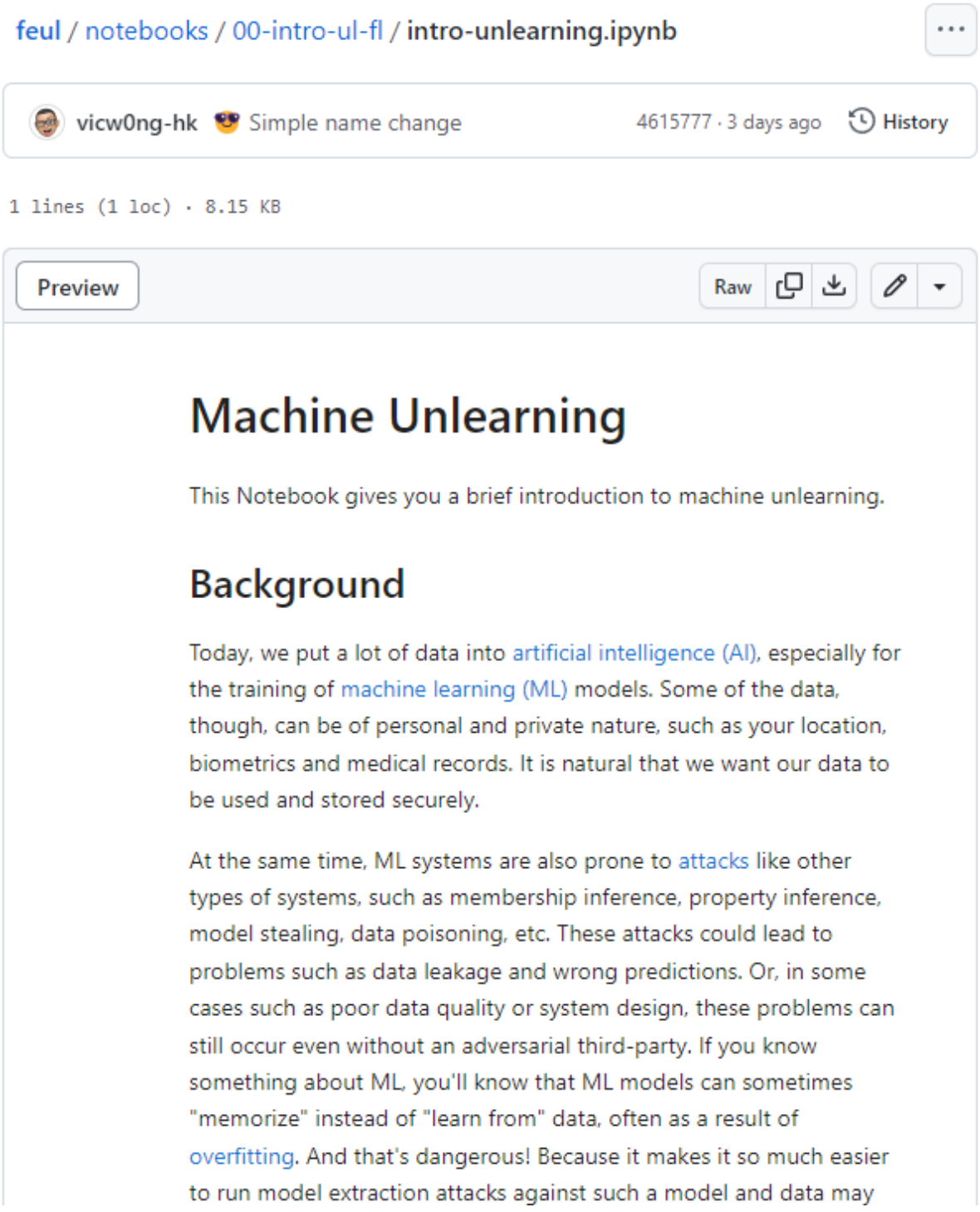


Figure 3: `intro-unlearning.ipynb`.

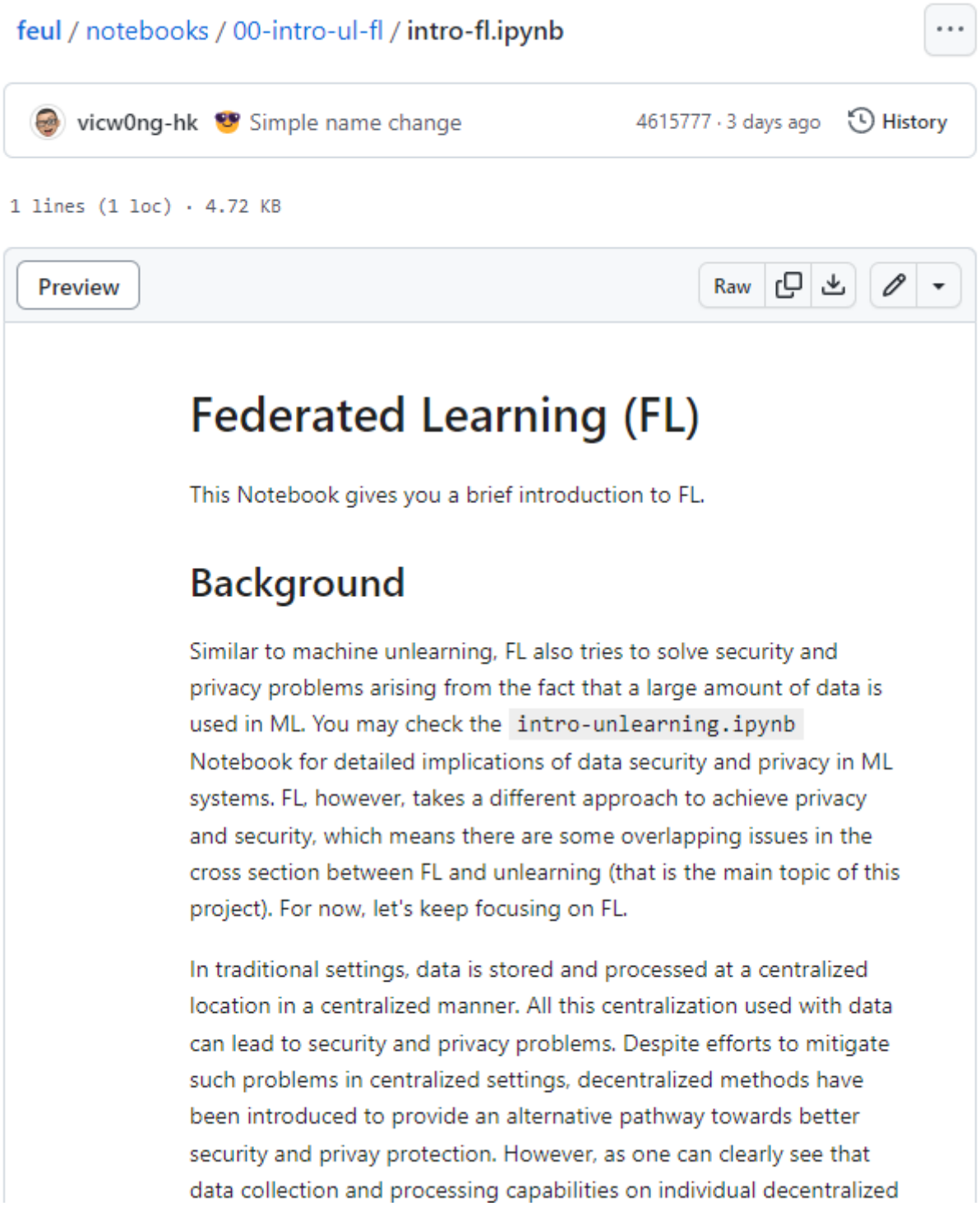


Figure 4: `intro-fl.ipynb`.

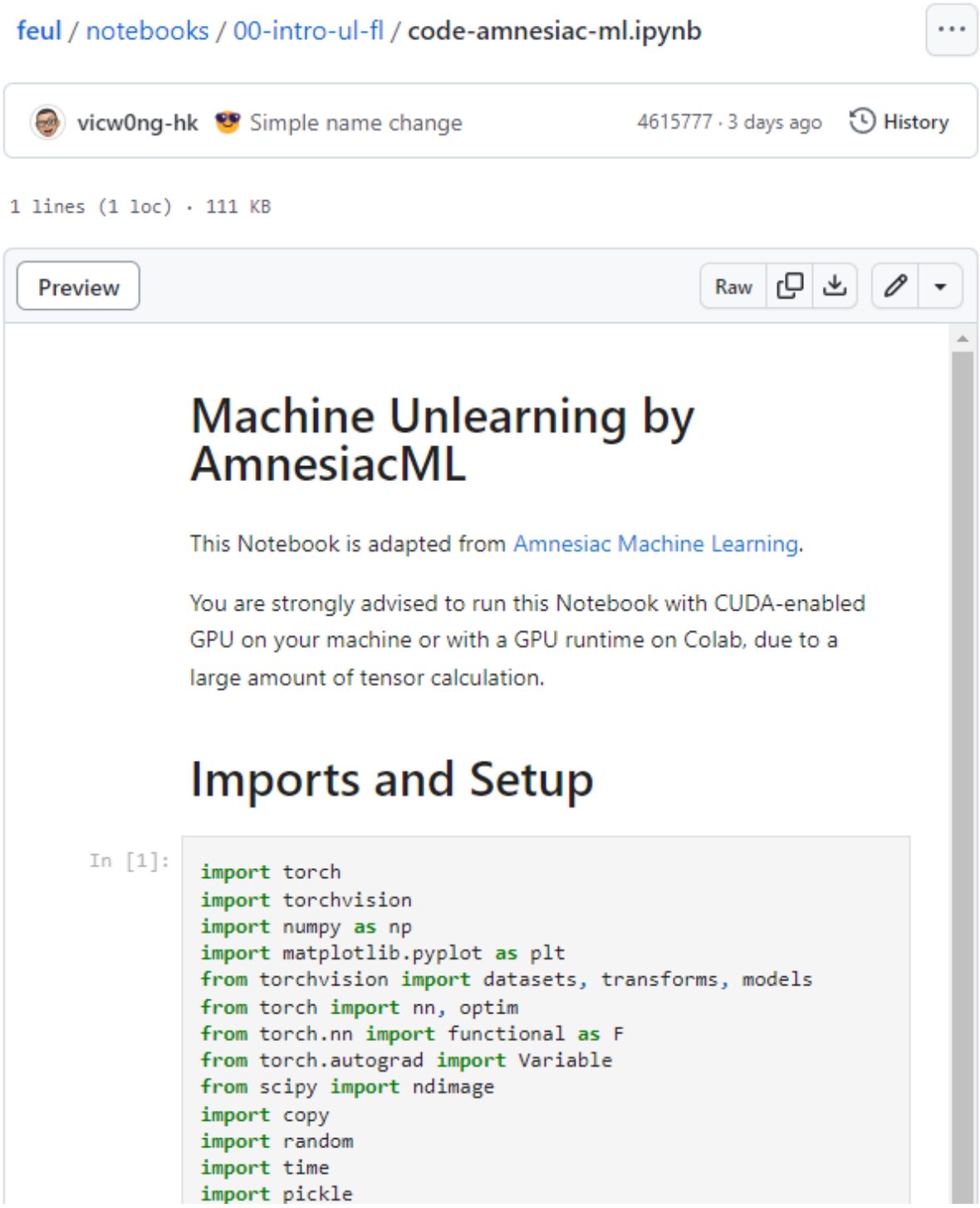


Figure 5: `code-amnesiac-ml.ipynb`.

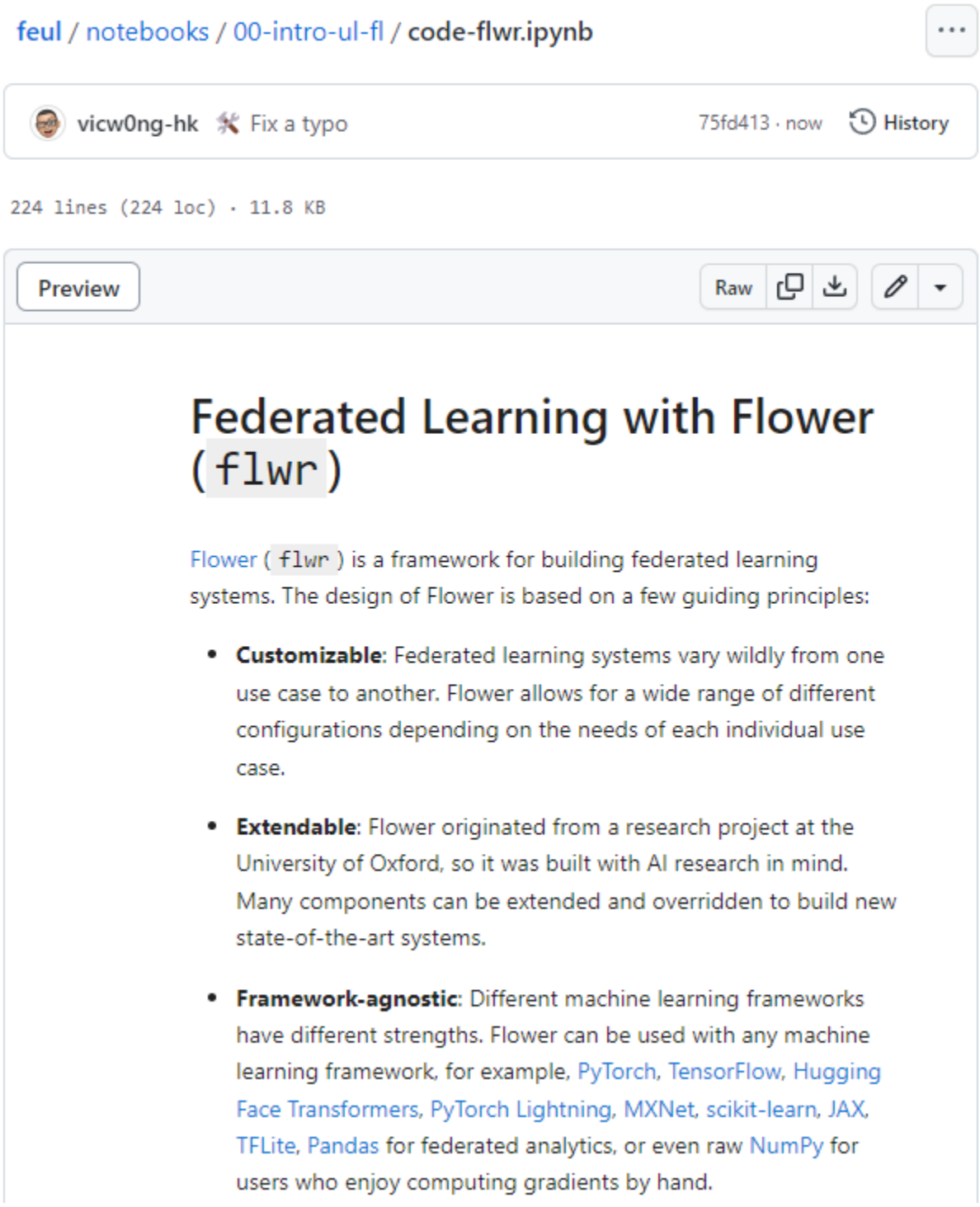


Figure 6: `code-flwr.ipynb`.

Colab using a GPU session, as well as on HKUCS GPU Farm using a GPU instance, and bugs were fixed.

7.2.5 04-liu+21b

There is one notebook under this folder:`revfrf.ipynb`. This notebook discusses the work done by [15] (see section 6.3.2).

7.2.6 05-gsk21

There is one notebook under this folder:`bayesian-variational.ipynb`. This notebook discusses the work done by [33] (see section 6.3.3).

7.2.7 06-wzm22

There is one notebook under this folder:`distillation.ipynb`. This notebook discusses the work done by [26] (see section 6.3.4).

7.2.8 07-wan+22

There is one notebook under this folder:`channel-prune.ipynb`. This notebook discusses the work done by [24] (see section 6.3.5).

7.2.9 08-liu+22

There are 2 notebooks under this folder: `rapid-retrain.ipynb` and `code-rapid-retrain.ipynb`. The first discusses the work done by [23] (see section 6.3.6) and the second provides instructions on how to run the code for the design proposed in the paper. Tests were carried out on Google Colab using a GPU session, as well as on HKUCS GPU Farm using a GPU instance, and bugs were fixed.

7.2.10 09-gon+22

There is one notebook under this folder:`forget-svgd.ipynb`. This notebook discusses the work done by [34] (see section 6.3.7).

7.2.11 10-gao+22

There is one notebook under this folder:`verifi.ipynb`. This notebook discusses the work done by [25] (see section 6.3.8).

7.2.12 11-hal+22

There is one notebook under this folder:`opt-out-unlearning.ipynb`. This notebook discusses the work done by [27] (see section 6.3.9).

7.2.13 12-cao+22

There is one notebook under this folder:`fedrecover.ipynb`. This notebook discusses the work done by [35] (see section 6.3.10).

7.2.14 13-pan+22

There is one notebook under this folder:`unlearning-cluster.ipynb`. This notebook discusses the work done by [36] (see section 6.3.11).

7.2.15 14-fra+22

There is one notebook under this folder:`sequential-informed.ipynb`. This notebook discusses the work done by [37] (see section 6.3.12).

7.2.16 15-wu+22

There is one notebook under this folder:`federated-unlearning.ipynb`. This notebook discusses the work done by [38] (see section 6.3.13).

7.2.17 16-li+23

There is one notebook under this folder:`subspace.ipynb`. This notebook discusses the work done by [39] (see section 6.3.14).

7.2.18 17-zlh23

There is one notebook under this folder:`heterogeneous-kg-embedding.ipynb`. This notebook discusses the work done by [40] (see section 6.3.15).

7.2.19 18-yua+23

There is one notebook under this folder:`on-device-recommend.ipynb`. This notebook discusses the work done by [41] (see section 6.3.16).

7.2.20 19-sl23

There are 2 notebooks under this folder: `knot.ipynb` and `code-knot.ipynb`. The first discusses the work done by [42] (see section 6.3.17) and the second provides instructions on how to run the code for the design proposed in the paper. Tests were carried out on Google Colab using a GPU session, as well as on HKUCS GPU Farm using a GPU instance, and bugs were fixed. One bug in the original code was reported to the research team at the University of Toronto[5].

8 Discussions

8.1 Prerequisites

As mentioned in the `README.md` file, having a strong foundation in machine learning, computer security, and operating systems (OS) is highly recommended for effectively utilizing this repository and conducting research in this area. Additionally, having a background in cryptography and distributed systems can further enhance one's experience and understanding of the subject matter.

Machine unlearning lies at the intersection of ML and security, making a solid understanding of both domains essential for grasping the intricacies of unlearning techniques. Furthermore, FL,

[5] https://github.com/TL-System/plato/issues/304

which is another focus of this project, is fundamentally a decentralized system. Possessing a basic knowledge of OS is crucial for navigating the engineering challenges associated with such systems.

Cryptography not only helps in developing a deeper understanding of threat modeling and general security concepts, but it also plays a role in the design of unlearning frameworks, as demonstrated in the work of Liu *et al.* [15]. Familiarity with cryptographic techniques can enable researchers to devise more secure and robust unlearning solutions.

In addition to cryptography, having knowledge of decentralized and distributed systems can provide valuable insights into the principles and design choices underlying FL. By understanding the inner workings of these systems, researchers can better appreciate the challenges and opportunities associated with implementing machine unlearning in federated learning contexts.

A strong foundation in machine learning, computer security, operating systems, cryptography, and distributed systems is highly beneficial for effectively engaging with this repository and conducting research in machine unlearning and federated learning. Gaining expertise in these areas will enable researchers to develop more innovative and effective solutions to the challenges posed by unlearning in decentralized ML systems.

8.2 How to Use

There are multiple ways to interact with and utilize the repository, depending on one's background and objectives. Viewing the repository on GitHub is the most straightforward approach. For individuals who have a strong foundation in the prerequisite subjects but have not yet delved into this specific area, going through the notebooks in the order listed in the repository will provide a comprehensive introduction to the topic. The first three sections, which cover background knowledge and early attempts to address the problem, will be particularly beneficial.

For non-experts who are simply looking to gain insights into this area, focusing on the first three sections should suffice in providing a solid understanding of the fundamental concepts. Seasoned researchers interested in staying up-to-date with the latest developments can go through the remaining notebooks one by one to explore the progress of research in the field.

If users are interested in learning about specific works or techniques, they can simply navigate to the relevant sections and explore those particular topics. For those who wish to build upon the project or incorporate the notebooks into their own research, they can download the entire project or specific notebooks and open them in Google Colab or any other preferred environment to continue their work.

The repository offers a flexible and diverse range of options for users with varying backgrounds and goals. By providing a structured and accessible learning resource, the repository can serve as a valuable tool for individuals looking to expand their knowledge in the area of machine unlearning and federated learning.

8.3 Significance

The significance of this project lies in its exploration and organization of materials related to machine unlearning in federated learning. As privacy concerns become increasingly crucial in the development and deployment of AI and ML systems, this project contributes to the understanding of the complex interplay between privacy preservation, data removal, and decentralized learning approaches.

By providing a comprehensive overview of the current state of machine unlearning in FL, the project serves as a valuable resource for researchers and practitioners in the field. It highlights the challenges and opportunities that arise when combining these techniques and fosters a deeper understanding of the nuances and trade-offs associated with achieving privacy and efficiency in ML systems.

The project helps pave the way for more innovative solutions that can be adapted to a wide range of ML applications and use cases, ultimately contributing to the development of more secure, robust, and privacy-aware AI systems.

8.4 Limitations

There are several limitations to this work that should be acknowledged. Firstly, the absence of a uniform benchmark across different publications, coupled with the fact that many papers do not provide their code, makes it challenging to quantitatively compare the various methods proposed in the literature. This limitation can impede efforts to draw clear conclusions and build on previous work.

Secondly, many papers focus on highly specific unlearning requests, models, data partitions, and other factors, which may limit the applicability of their proposed methods to the broader realm of federated learning. The narrow scope of these studies can make it difficult to generalize their findings or adapt their techniques to different contexts or use cases within the field of federated learning.

Thirdly, this work concludes in April 2023, and as such, any developments in this rapidly-evolving field that occur after this date will not be included. Given the fast-paced nature of research in machine unlearning and federated learning, it is possible that this project may become outdated within just a few years. Users should be aware of this limitation and consider supplementing the information provided here with the most recent research findings.

While this work provides a valuable resource for understanding the current state of machine unlearning in federated learning, it is essential to recognize its limitations and supplement the information provided with up-to-date research and practical considerations to ensure a comprehensive understanding of the field.

9 Conclusion

In conclusion, this project provides a comprehensive overview of machine unlearning in the context of FL. It serves as a valuable resource for both newcomers and seasoned researchers in the field, offering a collection of Jupyter Notebooks that document the background knowledge, key concepts, framework designs, theoretical analyses, and experiment instructions and expected results. Despite the limitations, the repository remains a helpful tool for understanding the current state of the art in machine unlearning in FL.

This project emphasizes the importance of addressing privacy, security, usability, and fidelity in machine learning systems, and highlights the challenges associated with efficiently and precisely unlearning data in FL. As researchers continue to explore new techniques and methodologies, it is essential to keep up with the latest advancements and adapt accordingly. It is our hope that this project will serve as a foundation for future research in the area of machine unlearning and FL, fostering innovation and collaboration in the pursuit of more robust and secure machine learning systems.

10 Future Work

As the field of machine unlearning, particularly in the context of FL, continues to evolve rapidly, there are numerous opportunities for future work and exploration. Some of the key areas for future research and development include:

10.1 Unified Benchmark for Unlearning Techniques

One of the limitations in the current state of research is the lack of a uniform benchmark for comparing different unlearning techniques. Future work could focus on establishing a unified benchmark that considers various aspects of unlearning, such as efficiency, precision, scalability, and applicability to different FL scenarios. This would allow researchers to more effectively evaluate and compare the performance of different techniques, fostering innovation and collaboration within the community.

10.2 Expanding Applicability of Unlearning Methods

Many existing unlearning methods focus on specific unlearning requests, models, or data partitions, which may limit their applicability to the broader realm of FL. Future research should aim to develop more generalizable unlearning techniques that can be adapted to different scenarios, data distributions, and model architectures. This could involve exploring new optimization algorithms, incorporating transfer learning or meta-learning techniques, or developing novel unlearning frameworks that account for various FL settings.

10.3 Addressing the Challenges of Unlearning Verification

Verifying the success of an unlearning process, particularly in the context of FL, is a challenging problem that warrants further investigation. Future work could explore new verification mechanisms that can efficiently and securely determine the effectiveness of unlearning without compromising the privacy and security guarantees of FL.

10.4 Exploring Unlearning in the Context of Emerging ML Paradigms

As machine learning continues to advance, new paradigms such as edge learning, continual learning, and reinforcement learning present unique challenges and opportunities for unlearning. Future research should investigate how unlearning techniques can be adapted and applied to these emerging paradigms, taking into account their specific characteristics and constraints.

10.5 Ethical and Regulatory Considerations in Machine Unlearning

Machine unlearning has significant implications for privacy, fairness, and accountability in machine learning systems. Future work should delve deeper into the ethical and regulatory aspects of unlearning, examining how it can be incorporated into existing legal frameworks such as the GDPR, and exploring its potential impact on algorithmic fairness and bias mitigation.

By addressing these areas of future work, researchers and practitioners can contribute to the development of more robust, secure, and efficient machine learning systems that better respect

user privacy, comply with data regulations, and improve overall system performance.

10.6 ChatGPT, Federated Learning, and Machine Unlearning

Exploring the intersection of ChatGPT, FL, and machine unlearning presents an exciting and promising direction for future research. The integration of these concepts has the potential to create a more privacy-preserving generative AI model that can still leverage the remarkable capabilities of ChatGPT while addressing the privacy concerns associated with the use of massive amounts of data.

Incorporating FL into ChatGPT training allows for a decentralized approach that mitigates privacy risks by aggregating knowledge learned from diverse data sources without directly accessing the raw data. Adding machine unlearning into this framework provides further privacy guarantees by enabling the efficient removal of specific data points' influence from the trained model, ensuring compliance with privacy regulations and user requests.

However, combining FL and machine unlearning with a large-scale generative AI model like ChatGPT presents several challenges, including handling the stochastic nature of the training process, addressing incrementality, preventing catastrophic unlearning, and managing communication overhead and unlearning complexity in the context of generative AI models.

Future research should focus on developing novel strategies and techniques tailored to generative AI models in an FL setting that address these challenges. This may involve creating new unlearning methods specifically designed for generative AI models in FL, optimizing communication protocols to minimize overhead, and devising approaches to verify the effectiveness of unlearning in such complex models.

Integrating ChatGPT, FL, and machine unlearning has the potential to create a new generation of AI models that respect user privacy without sacrificing effectiveness. Overcoming the challenges associated with this integration will open up new possibilities for robust, efficient, and privacy-preserving generative AI systems.

11 Acknowledgement

This project owes much gratitude to the supervisor, Prof. S.M. YIU, Associate Head of Department of Computer Science, HKU (HKUCS), as well as the second examiner, Dr. Tao YU, Assistant Professor at HKUCS.

This project was partially inspired by an HKUCS FYP in 2020-21, *Building a code and data repository for teaching algorithmic trading* by Woo, Wu and Lee, which won champion of 2020-21 FYP competition at HKUCS.

References

[1] J. A. Castellanos-Garzón, E. Costa, J. L. Jaimes S. and J. M. Corchado, 'An evolutionary framework for machine learning applied to medical data,' *Knowledge-Based Systems*, vol. 185, p. 104 982, 2019, ISSN: 0950-7051. DOI: 10.1016/j.knosys.2019.104982. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0950705119304046.

[2] B. Arslan, E. Yorulmaz, B. Akca and S. Sagiroglu, 'Security perspective of biometric recognition and machine learning techniques,' in *2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 2016, pp. 492–497. DOI: 10.1109/ICMLA.2016.0087.

[3] P. Zola, P. Cortez and M. Carpita, 'Twitter user geolocation using web country noun searches,' *Decision Support Systems*, vol. 120, pp. 50–59, 2019, ISSN: 0167-9236. DOI: https://doi.org/10.1016/j.dss.2019.03.006. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167923619300442.

[4] H. M. Gomes, J. Read, A. Bifet, J. P. Barddal and J. Gama, 'Machine learning for streaming data: State of the art, challenges, and opportunities,' *SIGKDD Explor. Newsl.*, vol. 21, no. 2, pp. 6–22, Nov. 2019, ISSN: 1931-0145. DOI: 10.1145/3373464.3373470. [Online]. Available: https://doi-org.eproxy.lib.hku.hk/10.1145/3373464.3373470.

[5] P. McDaniel, N. Papernot and Z. B. Celik, 'Machine learning in adversarial settings,' *IEEE Security & Privacy*, vol. 14, no. 3, pp. 68–72, 2016. DOI: 10.1109/MSP.2016.51.

[6] T. T. Nguyen, T. T. Huynh, P. L. Nguyen, A. W.-C. Liew, H. Yin and Q. V. H. Nguyen, *A survey of machine unlearning*, 2022. DOI: 10.48550/ARXIV.2209.02299. [Online]. Available: https://arxiv.org/abs/2209.02299.

[7] S. Verma, M. Ernst and R. Just, *Removing biased data to improve fairness and accuracy*, 2021. DOI: 10.48550/ARXIV.2102.03054. [Online]. Available: https://arxiv.org/abs/2102.03054.

[8] European Parliament and Council of the European Union, *Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation) (Text with EEA relevance)*, 2016. [Online]. Available: https://eur-lex.europa.eu/eli/reg/2016/679/oj.

[9] California State Legislature, *An act to add title 1.81.5 (commencing with section 1798.100) to part 4 of division 3 of the civil code, relating to privacy.* 2018. [Online]. Available: https://leginfo.legislature.ca.gov/faces/billTextClient.xhtml?bill_id=201720180AB375.

[10] S. Chatterjee, 'Learning and memorization,' in *Proceedings of the 35th International Conference on Machine Learning*, J. Dy and A. Krause, Eds., ser. Proceedings of Machine Learning Research, vol. 80, PMLR, Jul. 2018, pp. 755–763. [Online]. Available: https://proceedings.mlr.press/v80/chatterjee18a.html.

[11] Y. Cao and J. Yang, 'Towards making systems forget with machine unlearning,' in *2015 IEEE Symposium on Security and Privacy*, 2015, pp. 463–480. DOI: 10.1109/SP.2015.35.

[12] B. McMahan, E. Moore, D. Ramage, S. Hampson and B. A. y. Arcas, 'Communication-Efficient Learning of Deep Networks from Decentralized Data,' in *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, A. Singh and J. Zhu, Eds., ser. Proceedings of Machine Learning Research, vol. 54, PMLR, Apr. 2017, pp. 1273–1282. [Online]. Available: https://proceedings.mlr.press/v54/mcmahan17a.html.

[13] P. Kairouz *et al.*, 'Advances and open problems in federated learning,' *Foundations and Trends® in Machine Learning*, vol. 14, no. 1–2, pp. 1–210, 2021, ISSN: 1935-8237. DOI: 10.1561/2200000083. [Online]. Available: http://dx.doi.org/10.1561/2200000083.

[14] S. R. Pokhrel and J. Choi, 'Federated learning with blockchain for autonomous vehicles: Analysis and design challenges,' *IEEE Transactions on Communications*, vol. 68, no. 8, pp. 4734–4746, 2020. DOI: 10.1109/TCOMM.2020.2990686.

[15] Y. Liu, Z. Ma, Y. Yang, X. Liu, J. Ma and K. Ren, 'Revfrf: Enabling cross-domain random forest training with revocable federated learning,' *IEEE Transactions on Dependable and Secure Computing*, pp. 1–1, 2021. DOI: 10.1109/TDSC.2021.3104842.

[16] B. Hitaj, G. Ateniese and F. Perez-Cruz, 'Deep models under the gan: Information leakage from collaborative deep learning,' in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '17, Dallas, Texas, USA: Association for Computing Machinery, 2017, pp. 603–618, ISBN: 9781450349468. DOI: 10.1145/3133956.3134012. [Online]. Available: https://doi.org/10.1145/3133956.3134012.

[17] S. Mercuri *et al.*, *An introduction to machine unlearning*, 2022. DOI: 10.48550/ARXIV.2209.00939. [Online]. Available: https://arxiv.org/abs/2209.00939.

[18] S. Schelter, '"amnesia" - machine learning models that can forget user data very fast,' in *10th Conference on Innovative Data Systems Research, CIDR 2020, Amsterdam, The Netherlands, January 12-15, 2020, Online Proceedings*, www.cidrdb.org, 2020. [Online]. Available: http://cidrdb.org/cidr2020/papers/p32-schelter-cidr20.pdf.

[19] S. Shintre, K. A. Roundy and J. Dhaliwal, 'Making machine learning forget,' in *Privacy Technologies and Policy*, M. Naldi, G. F. Italiano, K. Rannenberg, M. Medina and A. Bourka, Eds., Cham: Springer International Publishing, 2019, pp. 72–83, ISBN: 978-3-030-21752-5.

[20] M. Veale, R. Binns and L. Edwards, 'Algorithms that remember: Model inversion attacks and data protection law,' *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 376, no. 2133, p. 20 180 083, 2018. DOI: 10.1098/rsta.2018.0083. eprint: https://royalsocietypublishing.org/doi/pdf/10.1098/rsta.2018.0083. [Online]. Available: https://royalsocietypublishing.org/doi/abs/10.1098/rsta.2018.0083.

[21] E. F. Villaronga, P. Kieseberg and T. Li, 'Humans forget, machines remember: Artificial intelligence and the right to be forgotten,' *Computer Law & Security Review*, vol. 34, no. 2, pp. 304–313, 2018, ISSN: 0267-3649. DOI: https://doi.org/10.1016/j.clsr.2017.08.007. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0267364917302091.

[22] G. Liu, X. Ma, Y. Yang, C. Wang and J. Liu, 'Federaser: Enabling efficient client-level data removal from federated learning models,' in *2021 IEEE/ACM 29th International Symposium on Quality of Service (IWQOS)*, 2021, pp. 1–10. DOI: 10.1109/IWQOS52092.2021.9521274.

[23] Y. Liu, L. Xu, X. Yuan, C. Wang and B. Li. 'The right to be forgotten in federated learning: An efficient realization with rapid retraining.' arXiv: 2203.07320. (2022).

[24] J. Wang, S. Guo, X. Xie and H. Qi, 'Federated unlearning via class-discriminative pruning,' in *Proceedings of the ACM Web Conference 2022*, ser. WWW '22, Virtual Event, Lyon, France: Association for Computing Machinery, 2022, pp. 622–632, ISBN: 9781450390965. DOI: 10.1145/3485447.3512222. [Online]. Available: https://doi.org/10.1145/3485447.3512222.

[25] X. Gao *et al.* 'Verifi: Towards verifiable federated unlearning.' arXiv: 2205.12709. (2022).

[26] C. Wu, S. Zhu and P. Mitra. 'Federated unlearning with knowledge distillation.' arXiv: 2201.09441. (2022).

[27] A. Halimi, S. Kadhe, A. Rawat and N. Baracaldo, *Federated unlearning: How to efficiently erase a client in fl?* 2022. DOI: 10.48550/ARXIV.2207.05521. [Online]. Available: https://arxiv.org/abs/2207.05521.

[28] L. Bourtoule *et al.*, 'Machine unlearning,' in *2021 IEEE Symposium on Security and Privacy (SP)*, 2021, pp. 141–159. DOI: 10.1109/SP40001.2021.00019.

[29] C. Dwork, 'Differential privacy: A survey of results,' in *Theory and Applications of Models of Computation*, M. Agrawal, D. Du, Z. Duan and A. Li, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 1–19.

[30] Q. P. Nguyen, R. Oikawa, D. M. Divakaran, M. C. Chan and B. K. H. Low, 'Markov chain monte carlo-based machine unlearning: Unlearning what needs to be forgotten,' ser. ASIA CCS '22, Nagasaki, Japan: Association for Computing Machinery, 2022, pp. 351–363, ISBN: 9781450391405. DOI: 10.1145/3488932.3517406. [Online]. Available: https://doi.org/10.1145/3488932.3517406.

[31] Q. P. Nguyen, B. K. H. Low and P. Jaillet, 'Variational bayesian unlearning,' in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan and H. Lin, Eds., vol. 33, Curran Associates, Inc., 2020, pp. 16 025–16 036. [Online]. Available: https://proceedings.neurips.cc/paper/2020/file/b8a6550662b363eb34145965d64d0cfb-Paper.pdf.

[32] C. Zhang, Y. Xie, H. Bai, B. Yu, W. Li and Y. Gao, 'A survey on federated learning,' *Knowledge-Based Systems*, vol. 216, p. 106 775, 2021, ISSN: 0950-7051. DOI: https://doi.org/10.1016/j.knosys.2021.106775. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0950705121000381.

[33] J. Gong, O. Simeone and J. Kang, 'Bayesian variational federated learning and unlearning in decentralized networks,' in *2021 IEEE 22nd International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, 2021, pp. 216–220. DOI: 10.1109/SPAWC51858.2021.9593225.

[34] J. Gong, O. Simeone, R. Kassab and J. Kang, *Forget-svgd: Particle-based bayesian federated unlearning*, 2021. DOI: 10.48550/ARXIV.2111.12056. [Online]. Available: https://arxiv.org/abs/2111.12056.

[35] X. Cao, J. Jia, Z. Zhang and N. Z. Gong, *Fedrecover: Recovering from poisoning attacks in federated learning using historical information*, 2022. DOI: 10.48550/ARXIV.2210.10936. [Online]. Available: https://arxiv.org/abs/2210.10936.

[36] C. Pan, J. Sima, S. Prakash, V. Rana and O. Milenkovic, *Machine unlearning of federated clusters*, 2022. DOI: 10.48550/ARXIV.2210.16424. [Online]. Available: https://arxiv.org/abs/2210.16424.

[37] Y. Fraboni, R. Vidal, L. Kameni and M. Lorenzi, *Sequential informed federated unlearning: Efficient and provable client unlearning in federated optimization*, 2022. DOI: 10.48550/ARXIV.2211.11656. [Online]. Available: https://arxiv.org/abs/2211.11656.

[38] L. Wu, S. Guo, J. Wang, Z. Hong, J. Zhang and Y. Ding, 'Federated unlearning: Guarantee the right of clients to forget,' *IEEE Network*, vol. 36, no. 5, pp. 129–135, 2022. DOI: 10.1109/MNET.001.2200198.

[39] G. Li, L. Shen, Y. Sun, Y. Hu, H. Hu and D. Tao, *Subspace based federated unlearning*, 2023. DOI: 10.48550/ARXIV.2302.12448. arXiv: 2302.12448 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2302.12448.

[40] X. Zhu, G. Li and W. Hu, *Heterogeneous federated knowledge graph embedding learning and unlearning*, 2023. DOI: 10.48550/ARXIV.2302.02069. arXiv: 2302.02069 [cs.LG]. [Online]. Available: https://arxiv.org/abs/2302.02069.

[41] W. Yuan, H. Yin, F. Wu, S. Zhang, T. He and H. Wang, 'Federated unlearning for on-device recommendation,' in *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*, ser. WSDM '23, Singapore, Singapore: Association for Computing Machinery, 2023, pp. 393–401, ISBN: 9781450394079. DOI: 10.1145/3539597.3570463. [Online]. Available: https://doi.org/10.1145/3539597.3570463.

[42] N. Su and B. Li, 'Asynchronous federated unlearning,' IEEE, 2023.

[43] Q.-V. Dang, 'Right to be forgotten in the age of machine learning,' in *Advances in Digital Science*, T. Antipova, Ed., Cham: Springer International Publishing, 2021, pp. 403–411, ISBN: 978-3-030-71782-7.

[44] N. Pitropakis, E. Panaousis, T. Giannetsos, E. Anastasiadis and G. Loukas, 'A taxonomy and survey of attacks against machine learning,' *Computer Science Review*, vol. 34, p. 100 199, 2019, ISSN: 1574-0137. DOI: https://doi.org/10.1016/j.cosrev.2019.100199. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1574013718303289.

[45] R. Shwartz-Ziv and N. Tishby, *Opening the black box of deep neural networks via information*, 2017. DOI: 10.48550/ARXIV.1703.00810. arXiv: 1703.00810 [cs.LG]. [Online]. Available: https://arxiv.org/abs/1703.00810.

[46] L. Graves, V. Nagisetty and V. Ganesh, 'Amnesiac machine learning,' *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 13, pp. 11 516–11 524, May 2021. DOI: 10.1609/aaai.v35i13.17371. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/17371.

A Useful Resources

- Privacy and Security in ML Seminars - Privacy & Security in Machine Learning (PriSec-ML) Interest Group[6]
- Virtual Seminar Series - Challenges and Opporunities for Security & Privacy in Machine Learning[7]
- IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)[8]
- the cleverhans blog - a blog by Ian Goodfellow and Nicolas Papernot about security and privacy in machine learning[9]
- ECE1784H/CSC2559H: Trustworthy Machine Learning Fall 2022 - University of Toronto[10]
- Awesome Machine Unlearning - GitHub Repo[11]

B Related HKUCS FYPs

- FYP22019: Quantitative Performance and Security Evaluation of Federated Learning on open-sourced platforms (Industry-based Project) by Fong, 2022-23[12]
- FYP20009: Building a code and data repository for teaching algorithmic trading by Woo, Wu and Lee, 2020-21[13]

[6] https://prisec-ml.github.io/
[7] https://vsehwag.github.io/SPML_seminar/
[8] https://satml.org/
[9] http://www.cleverhans.io/
[10] https://www.papernot.fr/teaching/f22-trustworthy-ml.html
[11] https://github.com/tamlhp/awesome-machine-unlearning
[12] https://wp.cs.hku.hk/2022/fyp22019/
[13] https://awoo424.github.io/algotrading_fyp/

C More Details on FL [12]

C.1 The `FederatedAveraging` Algorithm

Algorithm 1 `FederatedAveraging`. The K clients are indexed by k; B is the local minibatch size, E is the number of local epochs, and η is the learning rate.

Server executes:
initialize w_0
for each round $t = 1, 2, \dots$ **do**
  $m \leftarrow \max(C \cdot K, 1)$
  $S_t \leftarrow$ (random set of m clients)
  for each client $k \in S_t$ **in parallel do**
    $w_{t+1}^k \leftarrow \text{ClientUpdate}(k, w_t)$
  $w_{t+1} \leftarrow \sum_{k=1}^{K} \frac{n_k}{n} w_{t+1}^k$

ClientUpdate(k, w): // *Run on client* k
$\mathcal{B} \leftarrow$ (split $\mathcal{P}_k$ into batches of size B)
for each local epoch i from 1 to E **do**
  for batch $b \in \mathcal{B}$ **do**
    $w \leftarrow w - \eta \nabla \ell(w; b)$
return w to server

C.2 Federated Optimization

Federated optimization has several key properties that differentiate it from a typical distributed optimization problem:

- **Non-IID** The training data on a given client is typically based on the usage of the mobile device by a particular user, and hence any particular user's local dataset will not be representative of the population distribution.
- **Unbalanced** Similarly, some users will make much heavier use of the service or app than others, leading to varying amounts of local training data.
- **Massively distributed** We expect the number of clients participating in an optimization to be much larger than the average number of examples per client.
- **Limited communication** Mobile devices are frequently offline or on slow or expensive connections.

D More Details on Unlearning

Aprt from section D.1, these are some detailed information on proposed unlearning methods in FL that are more generalizable.

D.1 Unlearning Framework [6]

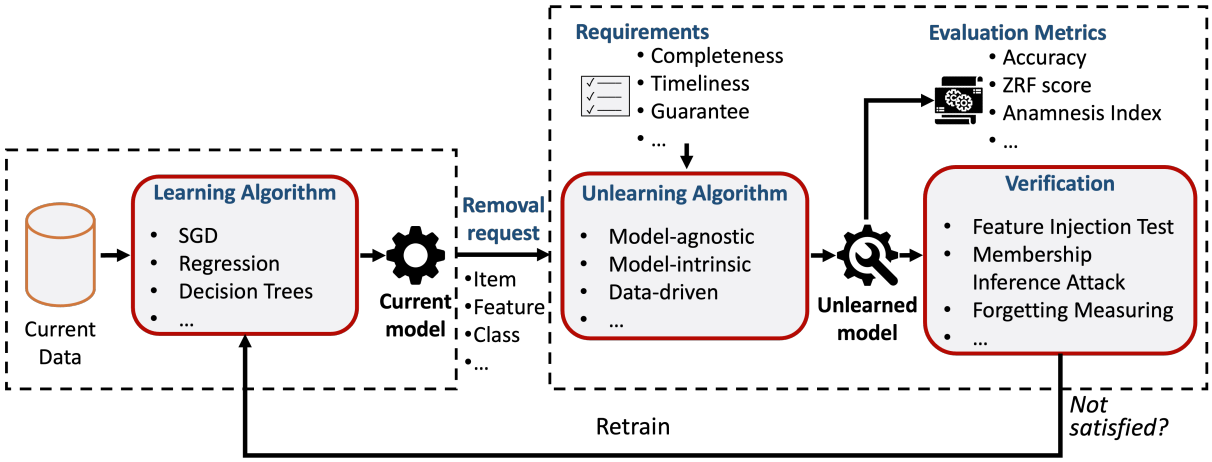


Figure 7: A Machine Unlearning Framework

D.2 FedEraser [22]

Algorithm 2 FedEraser

Require: Initial global model $\mathcal{M}^1$; retained client updates U
Require: Target client index k_u
Require: Number of global calibration round T
Require: Number of local calibration training epoch E_{cali}
Central server executes:
  for each round $R_{t_j}, j \in \{1, 2, \cdots, T\}$ **do**
    for each client $C_{k_c}, k_c \in \{1, 2, \cdots, K\} \setminus k_u$ **in parallel do**
      $\widehat{U}_{k_c}^{t_j} \leftarrow \text{CaliTrain}(C_{k_c}, \widetilde{\mathcal{M}}_{k_c}^{t_j}, E_{cali})$
      $\widetilde{U}_{k_c}^{t_j} \leftarrow |U_{k_c}^{t_j}| \frac{\widehat{U}_{k_c}^{t_j}}{||\widehat{U}_{k_c}^{t_j}||}$ {Update Calibrating}
    end
    $\widetilde{\mathcal{U}}^{t_j} \leftarrow \frac{1}{(K-1)\sum w_{k_c}} \sum_{k_c} w_{k_c} \widetilde{U}_{k_c}^{t_j}$ {Update Aggregating}
    $\widetilde{\mathcal{M}}^{t_{j+1}} \leftarrow \widetilde{\mathcal{M}}^{t_j} + \widetilde{\mathcal{U}}^{t_j}$ {Model Updating}
  end

CaliTrain$(C_{k_c}, \widetilde{\mathcal{M}}_{k_c}^{t_j}, E_{cali})$: // Run on client C_{k_c}
  for each local training round j from 1 to E_{cali} **do**
    $\widetilde{\mathcal{M}}_{k_c}^{t_j}|_{j+1} \leftarrow Train(\widetilde{\mathcal{M}}_{k_c}^{t_j}|_j, D_{k_c})$
  end
  $\widehat{U}_{k_c}^{t_j} \leftarrow$ Calculating Update$(\widetilde{\mathcal{M}}_{k_c}^{t_j}|_{E_{cali}}, \widetilde{\mathcal{M}}_{k_c}^{t_j}|_1)$
  return $\widehat{U}_{k_c}^{t_j}$ to the central server

D.3 Federated Unlearning with Knowledge Distillation [26]

$$M_F = M_1 + \sum_{t=1}^{F-1} \Delta M_t$$

$$\Delta M_t = \frac{1}{N} \sum_{i=1}^{N} \Delta M_t^i = \frac{1}{N} \sum_{i=1}^{N-1} \Delta M_t^i + \frac{1}{N} \Delta M_t^N$$

$$\Delta M_t' = \frac{1}{N-1} \sum_{i=1}^{N-1} \Delta M_t^i = \frac{N}{N-1} \Delta M_t - \frac{1}{N-1} \Delta M_t^N$$

Assume client N still participated in the training process but set his updates $\Delta M_t^N = 0$ for all rounds.

$$\Delta M_t' = \frac{1}{N} \sum_{i=1}^{N-1} \Delta M_t^i = \Delta M_t - \frac{1}{N} \Delta M_t^N$$

A combination of the above formula with Equation 1 gives us the unlearning result of the final global model M_F'.

$$\begin{aligned} M_F' &= M_1 + \sum_{t=1}^{F-1} \Delta M_t' + \sum_{t=1}^{F-1} \epsilon_t \\ &= M_1 + \sum_{t=1}^{F-1} \Delta M_t - \frac{1}{N} \sum_{t=1}^{F-1} \Delta M_t^N + \sum_{t=1}^{F-1} \epsilon_t \\ &= M_F - \frac{1}{N} \sum_{t=1}^{F-1} \Delta M_t^N + \sum_{t=1}^{F-1} \epsilon_t \end{aligned}$$

Algorithm 3 Federated Unlearning with Knowledge Distillation

Input: Global model M_F, Total number of clients N
Input: Historical updates ΔM_t^A of target client A at round t
Input: Outsourced unlabelled dataset X
Parameter: Distillation epoch k, Temperature T
Output: The unlearning model M_F'

1: $M_F' \leftarrow M_F - \frac{1}{N} \sum_{t=1}^{F-1} \Delta M_t^A$
2: **for** *epoch* $= 1, 2, \ldots, k$ **do**
3: $\quad y_{teacher} \leftarrow M_F(X), T$
4: $\quad y_{student} \leftarrow M_F'(X), T$
5: $\quad$ Calculate $loss_{distillation}$ of $y_{teacher}$ and $y_{student}$
6: $\quad$ Back-propagate model M_F'
7: **return** unlearning model M_F'

D.4 Rapid Retraining [23]

Algorithm 4 Federated Rapid Retraining Algorithm

Input: Local training dataset $\mathcal{D}_k = \{x_i, y_i\}_{i=1}^{n_k}$, model ω, mini-batch size B, unlearned dataset $\mathcal{U}_k$, learning rate η, and local epoch E_{local}.
Output: Unlearned global model ω^u.
Deletion Operation: ▷Run on client k
for each unlearned client **in parallel do**
    Perform batch data deletion operations, i.e., $\mathcal{D}_k^u \leftarrow \mathcal{D}_k \backslash \mathcal{U}_k$
Rapid Retraining Stage: ▷ Start unlearning stage
Server Executes:
Reinitialize the global model and send it to all clients, i.e., $\omega_0^{k_u} \leftarrow \omega$
Client Executes:
for all clients **in parallel do**
    for unlearned client **in parallel do**
        for $t = 0, 1, \ldots, T$ **do**
            LocalTraining $(k_u, \omega_t^{k_u}, E_{local}, \mathcal{D}_k^u)$
    for each normal client **in parallel do**
        for $t = 0, 1, \ldots, T$ **do**
            LocalTraining $(k_c, \omega_t^{k_u}, E_{local}, \mathcal{D}_k)$
Server Executes:
Update unlearned global model parameter ω_{t+1}^u
LocalTraining $(k, \omega, E_{local}, \mathcal{D})$ ▷Run on client k
for local epoch i from 1 to E_{local} **do**
    $\mathcal{D}_k^u \leftarrow \mathcal{D}_k \backslash \mathcal{U}_k$ % Current unlearned dataset
    $\Delta_k \leftarrow \nabla \ell_k(\omega_t^{k_u}, \mathcal{D}_k^u)$ % Current step gradient
    $\Gamma_k \leftarrow diag(\Gamma_k)$ % Current step estimated diagonal FIM
    Update $\bar{G}_t$
    Update m_t, v_t
    $\omega_{t+1}^{k_u} = \omega_t^{k_u} - \frac{\eta}{B - \Delta B_t} m_t / v_t$
return ω^u

D.5 FedRecover [35]

Algorithm 5 FedRecover

Input: $n-m$ remaining clients $\mathbf{C}_r = \{C_i | m+1 \leq i \leq n\}$; original global models $\bar{\boldsymbol{w}}_0, \bar{\boldsymbol{w}}_1, \cdots, \bar{\boldsymbol{w}}_T$ and original model updates $\bar{\boldsymbol{g}}_0^i, \bar{\boldsymbol{g}}_1^i, \cdots, \bar{\boldsymbol{g}}_{T-1}^i (m+1 \leq i \leq n)$; learning rate η; number of warm-up rounds T_w; periodic correction parameter T_c; number of final tuning rounds T_f; buffer size s of the L-BFGS algorithm; abnormality threshold τ; and aggregation rule $\mathcal{A}$.
Output: Recovered global model $\hat{\boldsymbol{w}}_T$.
$\hat{\boldsymbol{w}}_0 \leftarrow \bar{\boldsymbol{w}}_0$
for $t = 0, 1, \cdots, T_w - 1$ **do**
  $\hat{\boldsymbol{w}}_{t+1} \leftarrow \text{ExactTraining}(\mathbf{C}_r, \hat{\boldsymbol{w}}_t, \eta, \mathcal{A})$
for $t = T_w, T_w + 1, \cdots, T - T_f - 1$ **do**
  update the buffers $\Delta \boldsymbol{W}_t$ and $\Delta \boldsymbol{G}_t^i$ if needed
  if $(t - T_w + 1) \bmod T_c == 0$ **then**
    $\hat{\boldsymbol{w}}_{t+1} \leftarrow \text{ExactTraining}(\mathbf{C}_r, \hat{\boldsymbol{w}}_t, \eta, \mathcal{A})$
  else
    for $i = m+1, m+2, \cdots, n$ **do**
      $\tilde{\boldsymbol{H}}_t^i(\hat{\boldsymbol{w}}_t - \bar{\boldsymbol{w}}_t) \leftarrow \text{L-BFGS}(\Delta \boldsymbol{W}_t, \Delta \boldsymbol{G}_t^i, \hat{\boldsymbol{w}}_t - \bar{\boldsymbol{w}}_t)$
      $\hat{\boldsymbol{g}}_t^i = \bar{\boldsymbol{g}}_t^i + \tilde{\boldsymbol{H}}_t^i(\hat{\boldsymbol{w}}_t - \bar{\boldsymbol{w}}_t)$
      if $\|\hat{\boldsymbol{g}}_t^i\|_\infty > \tau$ **then**
        server sends $\hat{\boldsymbol{w}}_t$ to the ith client
        ith client computes $\boldsymbol{g}_t^i = \frac{\partial \mathcal{L}_i(\hat{\boldsymbol{w}}_t)}{\partial \hat{\boldsymbol{w}}_t}$
        ith client reports $\boldsymbol{g}_t^i$ to the server
        $\hat{\boldsymbol{g}}_t^i \leftarrow \boldsymbol{g}_t^i$
    $\hat{\boldsymbol{w}}_{t+1} \leftarrow \hat{\boldsymbol{w}}_t - \eta \cdot \mathcal{A}(\hat{\boldsymbol{g}}_t^{m+1}, \hat{\boldsymbol{g}}_t^{m+2}, \cdots, \hat{\boldsymbol{g}}_t^n)$
for $t = T - T_f, T - T_f + 1, \cdots, T - 1$ **do**
  $\hat{\boldsymbol{w}}_{t+1} \leftarrow \text{ExactTraining}(\mathbf{C}_r, \hat{\boldsymbol{w}}_t, \eta, \mathcal{A})$
return $\hat{\boldsymbol{w}}_T$