Patient Similarity Learning with Selective Forgetting

Wei Qian[1], Chenxu Zhao[2], Huajie Shao[3], Minghan Chen[4], Fei Wang[5], Mengdi Huai[1]
[1]Iowa State University, {wqi, mdhuai}@iastate.edu
[2]The Chinese University of Hong Kong (Shenzhen), chenxuzhao@link.cuhk.edu.cn
[3]William and Mary, hshao@wm.edu
[4]Wake Forest University, chenm@wfu.edu
[5]Weill Cornell Medicine, few2001@med.cornell.edu

2022 IEEE International Conference on Bioinformatics and Biomedicine (BIBM) | 978-1-6654-6819-0/22/$31.00 ©2022 IEEE | DOI: 10.1109/BIBM55620.2022.9995016

***Abstract*—Patient similarity learning aims to use patient information such as electronic medical records and genetic data as input to calculate the pairwise similarity between patients, and it is becoming increasingly important in healthcare applications. However, in many cases, patient similarity learning models also need to forget some patient data. From the perspective of privacy, patients desire a tool to erase the impacts of their sensitive data from the trained patient similarity models. From the perspective of utility, if a patient similarity model's utility is damaged by some bad patient data, the patient similarity model needs to forget such patient data to regain utility. Although some researchers have studied the problem of machine unlearning, existing methods cannot be directly applied to patient similarity learning as they fail to consider the comparative relationships among patients. In addition, they also fail to identify the optimal conditions of the local objective functions. In this paper, we fill in this gap by studying the unlearning problem in patient similarity learning. To unlearn the knowledge of a specific patient, we propose a novel erasable patient similarity learning framework, which enjoys the provable data removal guarantee and achieves high unlearning efficiency while keeping high model utility in patient similarity learning. We also conduct extensive experiments on real-world patient disease datasets to verify the desired properties of the proposed erasable framework.**

***Index Terms*—Patient similarity learning, selective knowledge removal**

I. Introduction

Patient similarity learning [26], [27] aims to learn a clinically meaningful distance metric to measure the similarity between patient pairs represented by their key clinical indicators. With the learnt patient similarity metric, physicians can perform different tasks. For example, in personalized medicine, physicians can retrieve a cohort of similar patients for a target patient to make medical comparisons and make a personalized treatment plan effectively [2], [22], [23]. In disease sub-typing, physicians identify sub-types of diseases (e.g., distinct subtypes of type 2 diabetes) by identifying clinically homogeneous patient subgroups [29]. In personalized medical prediction (e.g., mortality prediction in intensive care units), physicians can utilize patient similarity to boost the power of the model by using only patients most similar to the target patient in model training [9], [17].

However, in many cases, a patient similarity learning model also needs to forget certain sensitive data and its complete lineage. Consider privacy first, recent studies have shown that patients' sensitive information could be leaked from the trained models [15], [18], [30]. In practice, recent privacy legislation such as the European Union's General Data Protection Regulation (GDPR) and the former Right to be Forgotten [8] also give users the right to eliminate their data from the trained model as if they never existed in the training dataset. Nowadays, bad patient data (e.g., polluted data in poisoning attacks [20] or outliers) can seriously degrade the performance of the trained patient similarity models. Once these data are detected, the model needs to forget them to regain utility. Therefore, it is important to design efficient techniques that enable patient similarity models to forget what has been learned from the patient samples to be removed.

To remove patient samples from a trained patient similarity model that need to be forgotten, a straightforward approach is to simply train a new patient similarity model from scratch on the remaining dataset (i.e., excluding the samples that need to be erased) following the original training procedure of patient similarity learning. However, such a retraining method comes at a high computation cost and is thus not practical when adopting large-scale data and accommodating frequent removal requests. To address this problem, several exact machine unlearning methods [1], [4], [5] have been proposed, among which the SISA method proposed in [1] is the most general one. The basic idea of SISA is to randomly split the training dataset into several disjoint shards and train each shard model separately. Upon receiving an unlearning request to remove a specific sample, the model provider only needs to retrain the corresponding shard model.

However, existing machine unlearning methods cannot be directly applied to patient similarity learning tasks, since they fail to capture the important relationships among the patient samples. Specifically, since patient similarity learning relies on the relative comparative information among the training patient samples to learn the patient similarity model, randomly partitioning the training patient samples into subsets could severely damage the resulting model utility. The primary issue concerning the traditional sampling strategies is the lack of informative patient samples for training. If we directly follow existing methods to sample the patient pairs, a large fraction of patient samples may satisfy the constraints imposed by the loss function and provide no supervision information for the training model. In addition, the aggregation methods proposed in existing unlearning methods fail to identify the optimal conditions of the local objective functions.

To address the above challenges, for the first time, we in this paper propose a novel erasable patient similarity learning framework, namely *PatEraser*, which can achieve high unlearning efficiency while keeping high model utility in patient similarity learning. Specifically, in order to keep the informative comparison relationships among patients, we first design a novel data partition strategy based on the informative comparison relationships, which can divide the training patient samples into multiple informative shards. After splitting patient data into multiple smaller informative subsets, we apply the basic discriminative patient similarity learning to each of the subsets to train the submodels. Then, we design a novel aggregation strategy based on the optimal conditions of the local objective functions of patient data shards. The separately learned submodels are summarised into the final result via the proposed aggregation procedures. The algorithm of the aggregated patient similarity learning model scales well with the data size and can be controlled by the partition. We also conduct extensive experiments to verify the desired performance properties of the proposed PatEraser.

II. Related Work

Patient similarity learning has become a hot topic in recent years, with many researchers using patient similarity as a tool to enable different healthcare tasks [6], [9], [12]–[14], [33]. For example, the authors in [17] utilize the cosine similarity metric to identify similar patients for the downstream 30-day mortality prediction based on the MIMIC-II database. [6] utilizes the Euclidean distance-based metric to select similar patients for anomaly detection and characterization on the basis of numeric laboratory data. However, existing methods fail to remove the impact the patient data in the training set had on the final patient similarity learning model.

The emergence of the right to be forgotten gave birth to a paradigm named machine unlearning, which enables data holders to proactively erase their data from a trained model [7], [28]. Specifically, machine unlearning refers to a process that aims to remove the influence of a specified subset of training data upon request from a trained model at a cheaper computational cost than fully retraining those models. Currently, many different machine unlearning methods have been proposed [1], [10], [19], [21], [25], [31], [32]. However, existing machine unlearning methods cannot be directly adopted here, since they fail to either provide the provable guarantee or capture the important relationships among patient samples.

III. Methodology

A. Problem Formulation

We are concerned with the task of patient similarity learning. Patient similarity learning aims to develop computational algorithms for defining and locating clinically similar patients to a query patient under a specific clinical context. Let $\mathcal{H} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{N}$ be the training dataset of N labeled patients with patient samples $\boldsymbol{x}_i \in \mathbb{R}^D$ and class labels $y_i \in \{1, \cdots, C\}$. For the labeled patients in the training dataset $\mathcal{H}$, we can derive the following two sets of pairwise constraints:

$$\begin{aligned} \mathcal{S} &= \{(\boldsymbol{x}_i, \boldsymbol{x}_j)|\ \boldsymbol{x}_i \text{ and } \boldsymbol{x}_j \text{ are in the same class}\} \\ \mathcal{D} &= \{(\boldsymbol{x}_i, \boldsymbol{x}_j)|\ \boldsymbol{x}_i \text{ and } \boldsymbol{x}_j \text{ are in two different classes}\} \end{aligned} \tag{1}$$

where $\mathcal{S}$ is the set of similar pairwise constraints, and $\mathcal{D}$ is the set of dissimilar pairwise constraints. In patient similarity learning, the distance between any two patients $\boldsymbol{x}_i, \boldsymbol{x}_j \in \mathbb{R}^D$ is calculated as

$$d^2_{\boldsymbol{W}}(\boldsymbol{x}_i, \boldsymbol{x}_j) = ||\boldsymbol{x}_i - \boldsymbol{x}_j||^2_{\boldsymbol{W}} = (\boldsymbol{x}_i - \boldsymbol{x}_j)^T \boldsymbol{W} (\boldsymbol{x}_i - \boldsymbol{x}_j) \tag{2}$$

where $\boldsymbol{W} \in \mathbb{R}^{D \times D}$ is the Mahalanobis metric, a symmetric matrix of size $D \times D$. Note that $\boldsymbol{W}$ is a positive semi-definite matrix (i.e., $\boldsymbol{W} \succeq 0$) to satisfy the properties of metric (e.g., non-negativity and triangle inequality). Note that the constraint $\boldsymbol{W} \succeq 0$ is implicitly satisfied because of the decomposition $\boldsymbol{W} = \boldsymbol{M}\boldsymbol{M}^T$. Patient similarity learning can be cast as an optimization problem with pairwise constraints. We focus on learning the similarity metric $\boldsymbol{W} = \boldsymbol{M}\boldsymbol{M}^T$ by leveraging the similar and dissimilar pairwise relations in $\mathcal{S}$ and $\mathcal{D}$. In patient similarity learning, we learn the similarity metric so that the distances of similar patients become smaller and the distances of dissimilar patients become larger. Specifically, given a triplet $(\boldsymbol{x}_i, \boldsymbol{x}_j, \boldsymbol{x}_k)$, patient similarity learning aims to learn a good similarity metric such that patients from the same class are closer than patients from different classes, i.e.,

$$\forall (i, j, k), \quad d^2_{\boldsymbol{W}}(\boldsymbol{x}_i, \boldsymbol{x}_k) - d^2_{\boldsymbol{W}}(\boldsymbol{x}_i, \boldsymbol{x}_j) \geq 0, \tag{3}$$

where $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are from the same class and $\boldsymbol{x}_k$ is from a different class. For a given triplet $(\boldsymbol{x}_i, \boldsymbol{x}_j, \boldsymbol{x}_k)$, $(\boldsymbol{x}_i, \boldsymbol{x}_j)$ have the same class labels and $(\boldsymbol{x}_i, \boldsymbol{x}_k)$ have different class labels.

In this work, we frame the problem of data deletion in patient similarity learning as follows. Suppose a patient similarity learning model is trained on N patient samples (i.e., $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{N}$). For example, the patient similarity learning model could be trained to perform disease diagnosis from data which are collected from N patients. To delete the data sampled from the i-th patient (i.e., $\boldsymbol{x}_i$) from the trained patient similarity learning model, we would like to update it such that it becomes independent of patient $\boldsymbol{x}_i$, and looks as if it had been trained on the remaining $N-1$ patients. Formally, the task of patient similarity unlearning is to achieve the following three general objectives:

- **Provable Guarantee.** It is the basic requirement of unlearning which demands the revoked patient data must be really unlearned and not influence model parameters.
- **High Unlearning Efficiency.** The unlearning process of forgetting the required data should be as fast as possible.
- **Comparable Performance.** The performance of the unlearned patient similarity model should be comparable to that of retraining from scratch.

B. Proposed Method

Overview. In order to address the above challenges of unlearning in patient similarity learning, we propose PatEraser, which consists of the following three phases: *data partition*, *submodel training*, and *optimal aggregation*. Specifically, the data partition part is designed to divide original patient data while preserving the comparative information among the patients. Upon partitioning the training patient data into shards, a submodel is trained for each of the data shards. All submodels share the same model architecture. The model owner can train submodels in parallel to speed up the training process. At the prediction phase, an optimal aggregation strategy is applied to obtain the aggregated patient similarity learning model while satisfying the optimal conditions of the local objective functions. For each incoming test patient, its prediction result can be derived from the aggregated model. When one patient data needs to be unlearned, only one of the submodels whose shard contains the patient data to be unlearned and the aggregation part need to be retrained, which is much more efficient than retraining the whole patient similarity learning model from scratch. Next, we will detail each phase.

Data partition with informative comparison relationships. Note that patient similarity learning aims to maximize the inter-class distance and minimize the intra-class distance by using the comparison information among the training patient data. As we have mentioned before, the training data used for patient similarity learning tasks usually contains rich comparative information. Randomly dividing the patient data can result in a lack of informative samples for training. In order to address this challenge, we propose a novel data partition mechanism for patient similarity learning. Specifically, we first randomly sample P patients of each class. We use $\mathcal{P}$ to denote the set of sampled P patient samples. In this way, since there are C classes, the total number of the local data shards is $K = PC$, where P is the number of randomly sampled patients for each class. Then, for each given patient $\boldsymbol{x}_i$ in the k-th shard, we will construct the informative shards, each of which consists of a number of challenging patients that carry discriminative information for patient $\boldsymbol{x}_i$. In order to achieve this goal, for the k-th shard, we first select N_k^+ most challenging positive patients $\mathcal{H}_k^+$, which are from the same class as patient $\boldsymbol{x}_i$. Note that for the given patient $\boldsymbol{x}_i$ in the first shard, its hardest positive patient is defined as follows

$$(\boldsymbol{x}_i^+, y_i^+) = \underset{\boldsymbol{x}_j \in \mathcal{H}/\mathcal{P}, \boldsymbol{x}_j \neq \boldsymbol{x}_i, y_j = y_i}{\arg\max} ||\boldsymbol{x}_i - \boldsymbol{x}_j||_2^2, \tag{4}$$

where $\mathcal{H}$ is the set of training patients. Then, for the k-th shard, we select N_k^- nearest neighbours $\mathcal{H}_k^-$ from different classes. For the given patient $\boldsymbol{x}_i$ in the first shard, the hardest negative patient is defined bellow

$$(\boldsymbol{x}_i^-, y_i^-) = \underset{\boldsymbol{x}_j \in \mathcal{H}/\mathcal{P}, \boldsymbol{x}_j \neq \boldsymbol{x}_i, y_j \neq y_i}{\operatorname{argmin}} ||\boldsymbol{x}_i - \boldsymbol{x}_j||_2^2. \tag{5}$$

The k-th shard, $\mathcal{H}_k$, is the joint of $\{(\boldsymbol{x}_i, y_i)\} \cup \mathcal{H}_k^+ \cup \mathcal{H}_k^-$, where $\mathcal{H}_k^+$ and $\mathcal{H}_k^-$ are of size N_k^+ and N_k^-, respectively. In this way, for the data shards, we can mine the most valuable comparable information and select pairs of patients that provide the greatest violation of the pairwise constraints.

Submodel training. Here, we aim to train the submodels by incorporating discriminative information (i.e., the informative relative comparison relationships among the patients) available in the shards. Let $\boldsymbol{W}_k = \boldsymbol{M}_k \boldsymbol{M}_k^T$ denote the submodel for the k-th shard (i.e., $\mathcal{H}_k$), which is trained using the entirety of the patient data available in k-th shard ($\mathcal{H}_k$). Note that the learned patient similarity learning model ensures that patients with the same label from physician's feedback are close while the patient with different labels is away from each other. For each shard $\mathcal{H}_k$, we use $\boldsymbol{X}_k^+$ and $\boldsymbol{X}_k^-$ denote the feature matrix for $\mathcal{H}_k^+$ and $\mathcal{H}_k^-$, respectively. Then, based on $\boldsymbol{X}_k^+$ and $\boldsymbol{X}_k^-$, we can calculate the following two matrices

$$\Sigma_{\boldsymbol{X}_k^+} = \sum_{\boldsymbol{x}_j \in \boldsymbol{X}_k^+} (\boldsymbol{x}_i - \boldsymbol{x}_j)(\boldsymbol{x}_i - \boldsymbol{x}_j)^T, \tag{6}$$

$$\Sigma_{\boldsymbol{X}_k^-} = \sum_{\boldsymbol{x}_j \in \boldsymbol{X}_k^-} (\boldsymbol{x}_i - \boldsymbol{x}_j)(\boldsymbol{x}_i - \boldsymbol{x}_j)^T. \tag{7}$$

Next, we will set up the learning objective locally such that the trained submodels can encode the discriminative information and the local geometric structure of the patient data in the shards. The basic idea here is to maximize the distance between patients if they do not belong to the same distribution and instead minimize the distance between them if they belong to the same distribution. Specifically, for the k-th submodel, it can be derived by solving the following optimization problem

$$\begin{aligned} \boldsymbol{M}_k = \arg\min_{\boldsymbol{M}_k} \ell(\boldsymbol{M}_k) = \{ &\sum_{j \in N_k^+} ||\boldsymbol{M}_k^T(\boldsymbol{x}_i - \boldsymbol{x}_j)||^2 \\ &- \sum_{j \in N_k^-} ||\boldsymbol{M}_k^T(\boldsymbol{x}_i - \boldsymbol{x}_j)||^2 \} \\ = \arg\min_{\boldsymbol{M}_k} &\{Tr(\boldsymbol{M}_k^T \Sigma_{\boldsymbol{X}_k^+} \boldsymbol{M}_k) - Tr(\boldsymbol{M}_k^T \Sigma_{\boldsymbol{X}_k^-} \boldsymbol{M}_k)\} \\ = \arg\min_{\boldsymbol{M}_k} &Tr(\boldsymbol{M}_k^T \boldsymbol{R}_k \boldsymbol{M}_k), \end{aligned} \tag{8}$$

where $Tr(\cdot)$ denotes the trace of matrix, and $\boldsymbol{R}_k = (\Sigma_{\boldsymbol{X}_k^+} - \Sigma_{\boldsymbol{X}_k^-})$ and $||\boldsymbol{M}_k^T(\boldsymbol{x}_i - \boldsymbol{x}_j)||^2 = ||\boldsymbol{M}_k^T\boldsymbol{x}_i - \boldsymbol{M}_k^T\boldsymbol{x}_j||^2 = (\boldsymbol{M}_k^T\boldsymbol{x}_i - \boldsymbol{M}_k^T\boldsymbol{x}_j)(\boldsymbol{M}_k^T\boldsymbol{x}_i - \boldsymbol{M}_k^T\boldsymbol{x}_j) = (\boldsymbol{x}_i - \boldsymbol{x}_j)^T \boldsymbol{M}_k \boldsymbol{M}_k^T (\boldsymbol{x}_i - \boldsymbol{x}_j)$. The above loss aims to minimize the distances of similar patients and maximizes the distances of dissimilar patients. The local criterion of submodels on shards is motivated by encoding discriminative information into the geometry induced by the objective metric.

Note that when the model owner receives a request to delete a new patient data, it just needs to retrain the local submodel whose shard contains this patient data. If the center patient sample in the k-th sample is deleted, we will select the new center patient sample $\boldsymbol{x}_i^k$ as follows

$$\begin{aligned} \boldsymbol{x}_i^k = \underset{\boldsymbol{x}_i \in \mathcal{H}_k^+ \cup \mathcal{H}_k^-}{\operatorname{argmin}} \{ &\sum_{\boldsymbol{x}_j \neq \boldsymbol{x}_i, y_j \neq y_i} ||(\boldsymbol{x}_i - \boldsymbol{x}_j)||^2 \\ &- \sum_{\boldsymbol{x}_j \neq \boldsymbol{x}_i, y_j = y_i} ||(\boldsymbol{x}_i - \boldsymbol{x}_j)||^2 \} \end{aligned} \tag{9}$$

where $\boldsymbol{x}_j \in \mathcal{H}_k^+ \cup \mathcal{H}_k^- / \boldsymbol{x}_i$. In the above, $\mathcal{H}_k^+$ and $\mathcal{H}_k^-$ are of size N_k^+ and N_k^-, respectively. In this way, we can ensure that a large fraction of patient samples in this shard do not have the constraints imposed by the loss function and provide plentiful supervision information for the training model.

Aggregation. The next task is to aggregate the local solutions (i.e., $\{\boldsymbol{M}_k\}_{k=1}^{K}$) into a global patient similarity metric $\boldsymbol{M}_A$. The most naive straightforward method is to linearly combine the local solutions. However, $\{\boldsymbol{M}_k\}_{k=1}^{K}$ are solutions to locally defined optimization problems over the patient data shards and the linear combination can damage their optimality and yield invalid solutions. In order to address this challenge, we design the aggregation strategy based on the optimal conditions of the objective functions. Recall that a submodel $\boldsymbol{M}_k$ is a stationary point of the corresponding objective function ($\ell_k(\boldsymbol{M}_k)$), i.e.,

$$\frac{\partial \ell_k(\boldsymbol{M}_k)}{\partial \boldsymbol{M}_k} = 2(\Sigma_{\boldsymbol{X}_k^+} - \Sigma_{\boldsymbol{X}_k^-})\boldsymbol{M}_k = 0, \forall k \in [K], \qquad (10)$$

where $[K] \in \{1, \cdots, K\}$ and $\ell_k(\boldsymbol{M}_k)$ is defined in Eqn. (8). For a global solution $\boldsymbol{M}_A$, it is ideal for it to fulfill the optimal conditions of the objective functions in all local patient data shards, which is generally impossible. Therefore, we propose to cancel out the violations among all the patient data shards by vanishing the summation of the local gradients of the submodels, which is given as follows

$$\sum_{k=1}^{K}(\Sigma_{\boldsymbol{X}_k^+} - \Sigma_{\boldsymbol{X}_k^-})\boldsymbol{M}_A = 0, \qquad (11)$$

where $\boldsymbol{W}_A = \boldsymbol{M}_A\boldsymbol{M}_A^T$ denotes the aggregated model. By combining Eqn. (10) and (11), we can derive the following

$$\sum_{k=1}^{K}(\Sigma_{\boldsymbol{X}_k^+} - \Sigma_{\boldsymbol{X}_k^-})\boldsymbol{M}_k - \sum_{k=1}^{K}(\Sigma_{\boldsymbol{X}_k^+} - \Sigma_{\boldsymbol{X}_k^-})\boldsymbol{M}_A = 0, \quad (12)$$

$$\Rightarrow \boldsymbol{M}_A = \sum_{k=1}^{K}\frac{(\Sigma_{\boldsymbol{X}_k^+} - \Sigma_{\boldsymbol{X}_k^-})}{\sum_{k=1}^{K}(\Sigma_{\boldsymbol{X}_k^+} - \Sigma_{\boldsymbol{X}_k^-})}\boldsymbol{M}_k, \qquad (13)$$

where $\boldsymbol{M}_k$ denotes the k-th submodel. From the above, we can see that the aggregated patient similarity model has the form of the weighted submodels, where the weight of the k-th submodel is $\omega_k = \frac{(\Sigma_{\boldsymbol{X}_k^+} - \Sigma_{\boldsymbol{X}_k^-})}{\sum_{k=1}^{K}(\Sigma_{\boldsymbol{X}_k^+} - \Sigma_{\boldsymbol{X}_k^-})}$. And the aggregated patient similarity learning model is the weighted sum of the local submodels, which is given as follows

$$\boldsymbol{M}_A = \sum_{k=1}^{K}\frac{(\Sigma_{\boldsymbol{X}_k^+} - \Sigma_{\boldsymbol{X}_k^-})}{\sum_{k=1}^{K}(\Sigma_{\boldsymbol{X}_k^+} - \Sigma_{\boldsymbol{X}_k^-})}\boldsymbol{M}_k. \qquad (14)$$

When the model owner receives a request to delete a new patient data, it just needs to retrain the local shard model whose shard contains this patient data, leading to a significant time gain with respect to retraining the whole model from scratch. Then we can aggregate these local patient similarity learning models based on the above equation.

IV. Experiments

A. Experimental Setup

Datasets. In experiments, we adopt three real-world patient datasets and a synthetic dataset to measure the performance of the proposed method (PatEraser). The **Diabetes** health indicators dataset consists of 70,692 survey responses to CDC's BRFSS2015 [3]. The target variable has 2 labels, where label 0 denotes no diabetes and label 1 denotes prediabetes or diabetes. The **Cardiovascular** disease dataset [11] is a collection of 69,301 patient data used to predict the presence or absence of cardiovascular disease. The input features are collected from factual information and medical examination results. The **Heart disease** health indicators dataset contains 253,680 patients, and the features in this dataset are collected from cleaned BRFSS2015 [3]. It can be primarily used for the binary classification of heart disease. The **Synthetic** dataset is a randomly distributed binary classification dataset generated using Scikit-learn [24] dataset module. We initialize the classification dataset for 10 input features with no duplicate or redundant features, for a total of 100,000 samples. The details of the adopted datasets are reported in Table I. Note that patient datasets are from the Kaggle dataset repository [1].

TABLE I: Details of the adopted datasets in experiments.

Dataset	# patients	# features
Diabetes	70,692	21
Cardiovascular	69,301	11
Heart Disease	253,680	21
Synthetic	100,000	10

Baselines. We compare the proposed approach with three state-of-the-art baselines. **Retrain** is the most straightforward machine unlearning method, which removes the revoked samples and retrains the entire model. It is treated as a base benchmark. **Average** follows the ideas of the state-of-the-art machine unlearning method [1], which randomly splits the training data into shards and then aggregates the results of all submodels by averaging to make the final prediction. The **Random** method randomly selects a model from all local submodels and treats it as the aggregated model.

Performance metrics. In experiments, we adopt the following evaluation metrics: unlearning time for unlearning efficiency, recall@1 and recall@2 for classification performance. Unlearning time measures the retraining time of models after requesting unlearning samples. Recall@1 and recall@2 measure whether the ground truth is ranked among the top-1 item or top-2 items, respectively.

Training. In experiments, we employ the Adam [16] optimizer with a learning rate of 0.01 for ParEraser and train 100 epochs on adopted datasets. The batch size setting in each submodel training is the corresponding shard size.

B. Experimental Results

Unlearning efficiency. In this section, we conduct experiments to investigate the unlearning efficiency of the proposed method (PatEraser). Specifically, we set the shard number to 20 and randomly sample 1 patient data to be forgotten from the training data. The derived experimental results are reported in Table II. We can observe that the proposed PatEraser can significantly improve the unlearning efficiency compared to the

[1] https://www.kaggle.com/datasets

TABLE II: Unlearning efficiency comparison for removing 1 data sample. For PatEraser (20 shards), when receiving an unlearning request of data, only the corresponding submodel and the aggregation part need to be retrained.

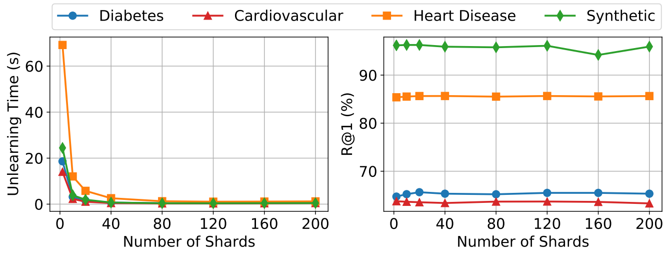
Dataset	Diabetes				Cardiovascular				Heart Disease				Synthetic			
Training Size	20%	40%	60%	80%	20%	40%	60%	80%	20%	40%	60%	80%	20%	40%	60%	80%
Retrain (s)	75	149	220	298	70	142	210	293	362	723	1090	1435	146	282	421	547
PatEraser (s)	**0.8**	**1.0**	**1.3**	**1.6**	**0.8**	**1.0**	**0.2**	**1.5**	**1.3**	**2.2**	**3.2**	**4.2**	**0.8**	**1.1**	**1.3**	**1.8**

TABLE III: Classification performance comparison of different baselines. We apply 20 shards for the Average and PatEraser methods, and report recall@K (%) for all methods.

Dataset	Diabetes		Cardiovascular		Heart Disease	
Recall	R@1	R@2	R@1	R@2	R@1	R@2
Retrain	64.19	82.11	63.35	81.28	85.35	92.68
Average	64.53	82.31	63.31	81.67	85.33	92.69
Random	64.72	82.36	62.90	81.57	85.23	92.51
PatEraser	**65.13**	**82.67**	**63.89**	**81.73**	**85.60**	**92.72**

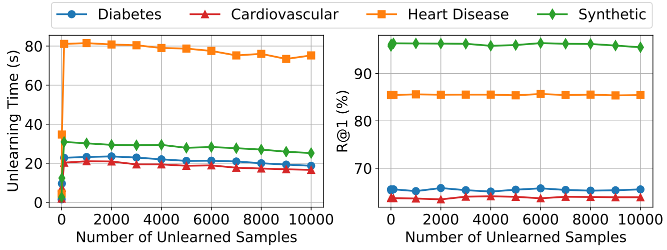
Retrain method. For example, on the Cardiovascular dataset and the Heart disease dataset with both 80% training sizes, the proposed PatEraser only needs 1.5 seconds and 4.2 seconds to achieve the optimal performance, respectively, while the Retrain method needs about 293 seconds and 1435 seconds, respectively. This acceleration is 195× for the Cardiovascular dataset and 341× for the Heart disease dataset, which is highly valuable in practice and is difficult to achieve through simple engineering effects. The main reason is that in the proposed PatEraser, only the corresponding submodel and the aggregation part need to be retrained to forget the requested data. Therefore, even for large datasets and a high training scale, the unlearning time is still incredibly fast. From the shard-based perspective, we can tolerate more shards for larger datasets to further improve the unlearning efficiency while preserving the patient similarity performance.

Classification performance. Next, we evaluate the classification performance of the proposed PatEraser. Specifically, the shard number is set to 20 for PatEraser and Average methods. The obtained classification results are shown in Table III. We can observe that on the adopted patient datasets, the proposed PatEraser can achieve a better classification performance compared to the baselines. For example, on the Diabetes dataset, the recall@1 of PatEraser is 65.13%, while the corresponding result of the Average baseline is 64.53% and the Random baseline is 64.72%. Similarly, the substantial classification improvement is due to the fact that we partition the patient data by preserving the relative comparison information among the patient samples and differentiating the importance scores of different submodels in aggregation. In addition, the Retrain method can achieve similar classification results but its unlearning efficiency is extremely low when the original dataset is large. Therefore, we can derive the conclusion that the classification performance of the proposed PatEraser is better than the baselines.



(a) Unlearning Efficiency **(b) Model Performance**

Fig. 1: Impact of the shard number on the unlearning efficiency (unlearning 1 data sample) and model performance (recall@1) on the experiment datasets.



(a) Unlearning Efficiency **(b) Model Performance**

Fig. 2: Impact of the unlearned patient samples on the unlearning efficiency (20 shards) and model performance (recall@1 after unlearning) on the experimental datasets.

Impact of the shard number. Furthermore, we conduct experiments on the adopted datasets to investigate the impact of the shard number. As shown in Figure 1 (a), the unlearning time decreases when the number of shards increases for the proposed PatEraser model on all experiments. This makes sense since a larger number of shards means a smaller shard size for each submodel, which will improve the unlearning efficiency. As shown in Figure 1 (b), the model performance may slightly decrease when the shard number is too small (e.g., 2 shards) or the shard number is too large. This is because patient similarity submodels require comparison information for model learning. A small shard means the comparison information may not diverge; a large shard but small shard size means the comparison information may not strong.

Impact of the unlearned patient samples. Lastly, we conduct experiments to study the impact of the unlearned patient samples. Figure 2 (a) illustrates the impact on the unlearning efficiency. The results suggest that before a particular number of unlearned samples (related to the number of shards), the unlearning time increases rapidly, then slowly decreases after that. The reason is that the unlearning time is determined by the number of submodels that need to be retrained. As the number of unlearned samples increases, there is a greater chance that more submodels need to be retrained, and hence the unlearning time increases. Once all submodels are required to be retrained but the shard size is reduced, the unlearning time will be decreased. Figure 2 (b) summarizes the impact on the model performance, in which we examine recall@1 for PatEraser. We observe that PatEraser is robust to patient similarity unlearning, even when the unlearned samples reach 10,000 of the training data.

V. Conclusion

In this paper, we propose a novel PatEraser framework, which is, to the best of our knowledge, the first machine unlearning method for patient similarity learning. To permit efficient unlearning while keeping the comparison information of the patient data in different data shards, we first design a novel data partition strategy to keep the informative comparison relationships among the patients. Then, based on the optimal conditions of the local objective functions, we propose an adaptive aggregation method to improve the global model utility. We also conduct extensive experiments to verify the effectiveness of the proposed method.

Acknowledgment

This work is supported in part by the US National Science Foundation under grants IIS-2212175, and IIS-1750326. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

[1] L. Bourtoule, V. Chandrasekaran, C. A. Choquette-Choo, H. Jia, A. Travers, B. Zhang, D. Lie, and N. Papernot, “Machine unlearning,” in *2021 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2021.

[2] S.-A. Brown, “Patient similarity: emerging concepts in systems and precision medicine,” *Frontiers in physiology*, vol. 7, p. 561, 2016.

[3] “Behavioral risk factor surveillance system survey data,” Centers for Disease Control and Prevention (CDC), Atlanta, Georgia: U.S. Department of Health and Human Services, Centers for Disease Control and Prevention, 2015.

[4] C. Chen, F. Sun, M. Zhang, and B. Ding, “Recommendation unlearning,” in *Proceedings of the ACM Web Conference 2022*, 2022, pp. 2768–2777.

[5] M. Chen, Z. Zhang, T. Wang, M. Backes, M. Humbert, and Y. Zhang, “Graph unlearning,” *arXiv preprint arXiv:2103.14991*, 2021.

[6] G. David, L. Bernstein, and R. R. Coifman, “Generating evidence based interpretation of hematology screens via anomaly characterization,” in *AMERICAN JOURNAL OF CLINICAL PATHOLOGY*, vol. 134, no. 4. AMER SOC CLINICAL PATHOLOGY 2100 W HARRISON ST, CHICAGO, IL 60612 USA, 2010, pp. 668–668.

[7] S. Garg, S. Goldwasser, and P. N. Vasudevan, “Formalizing data deletion in the context of the right to be forgotten,” in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*. Springer, 2020, pp. 373–402.

[8] A. Ginart, M. Guan, G. Valiant, and J. Y. Zou, “Making ai forget you: Data deletion in machine learning,” *Advances in neural information processing systems*, vol. 32, 2019.

[9] A. Gottlieb, G. Y. Stein, E. Ruppin, R. B. Altman, and R. Sharan, “A method for inferring medical diagnoses from patient similarities,” *BMC medicine*, vol. 11, no. 1, pp. 1–10, 2013.

[10] V. Gupta, C. Jung, S. Neel, A. Roth, S. Sharifi-Malvajerdi, and C. Waites, “Adaptive machine unlearning,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 16 319–16 330, 2021.

[11] R. K. Halder, “Cardiovascular disease dataset,” 2020. [Online]. Available: https://dx.doi.org/10.21227/7qm5-dz13

[12] M. Huai, C. Miao, J. Liu, D. Wang, J. Chou, and A. Zhang, “Global interpretation for patient similarity learning,” in *2020 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. IEEE, 2020, pp. 589–594.

[13] M. Huai, C. Miao, Q. Suo, Y. Li, J. Gao, and A. Zhang, “Uncorrelated patient similarity learning,” in *Proceedings of the 2018 SIAM International Conference on Data Mining*. SIAM, 2018, pp. 270–278.

[14] Y. Huang, N. Wang, H. Liu, H. Zhang, X. Fei, L. Wei, and H. Chen, “Study on patient similarity measurement based on electronic medical records,” in *MEDINFO 2019: Health and Wellbeing e-Networks for All*. IOS Press, 2019, pp. 1484–1485.

[15] M. Kayaalp, “Patient privacy in the era of big data,” *Balkan medical journal*, vol. 35, no. 1, pp. 8–17, 2018.

[16] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *CoRR*, vol. abs/1412.6980, 2015.

[17] J. Lee, D. M. Maslove, and J. A. Dubin, “Personalized mortality prediction driven by electronic medical data and a patient similarity metric,” *PloS one*, vol. 10, no. 5, p. e0127428, 2015.

[18] J. Lee, J. Sun, F. Wang, S. Wang, C.-H. Jun, X. Jiang *et al.*, “Privacy-preserving patient similarity learning in a federated environment: development and analysis,” *JMIR medical informatics*, p. e7744, 2018.

[19] R. Mehta, S. Pal, V. Singh, and S. N. Ravi, “Deep unlearning via randomized conditionally independent hessians,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 10 422–10 431.

[20] M. Mozaffari-Kermani, S. Sur-Kolay, A. Raghunathan, and N. K. Jha, “Systematic poisoning attacks on and defenses for machine learning in healthcare,” *IEEE journal of biomedical and health informatics*, vol. 19, no. 6, pp. 1893–1905, 2014.

[21] 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,” *arXiv preprint arXiv:2202.13585*, 2022.

[22] S. Pai and G. D. Bader, “Patient similarity networks for precision medicine,” *Journal of molecular biology*, pp. 2924–2938, 2018.

[23] E. Parimbelli, S. Marini, L. Sacchi, and R. Bellazzi, “Patient similarity for precision medicine: A systematic review,” *Journal of biomedical informatics*, vol. 83, pp. 87–96, 2018.

[24] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, “Scikit-learn: Machine learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[25] S. Schelter, S. Grafberger, and T. Dunning, “Hedgecut: Maintaining randomised trees for low-latency machine unlearning,” in *Proceedings of the 2021 International Conference on Management of Data*, 2021, pp. 1545–1557.

[26] Q. Suo, F. Ma, Y. Yuan, M. Huai, W. Zhong, J. Gao, and A. Zhang, “Deep patient similarity learning for personalized healthcare,” *IEEE transactions on nanobioscience*, vol. 17, no. 3, pp. 219–227, 2018.

[27] Q. Suo, F. Ma, Y. Yuan, M. Huai, W. Zhong, A. Zhang, and J. Gao, “Personalized disease prediction using a cnn-based similarity learning method,” in *Bioinformatics and Biomedicine (BIBM)*, 2017.

[28] 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.

[29] B. Wang, A. M. Mezlini, F. Demir, M. Fiume, Z. Tu, M. Brudno, B. Haibe-Kains, and A. Goldenberg, “Similarity network fusion for aggregating data types on a genomic scale,” *Nature methods*, vol. 11, no. 3, pp. 333–337, 2014.

[30] D. Wang, M. Huai, and J. Xu, “Differentially private sparse inverse covariance estimation,” in *2018 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*. IEEE, 2018, pp. 1139–1143.

[31] G. Wu, M. Hashemi, and C. Srinivasa, “Puma: Performance unchanged model augmentation for training data removal,” *arXiv preprint arXiv:2203.00846*, 2022.

[32] Y. Wu, E. Dobriban, and S. Davidson, “Deltagrad: Rapid retraining of machine learning models,” in *International Conference on Machine Learning*. PMLR, 2020, pp. 10 355–10 366.

[33] L. Yao, S. Li, Y. Li, M. Huai, J. Gao, and A. Zhang, “Ace: Adaptively similarity-preserved representation learning for individual treatment effect estimation,” in *2019 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2019.