Privacy-Preserving Edge Cloud Architecture for IoT Healthcare Systems

PAYAL GOYAL

A THESIS SUBMITTED TO THE FACULTY OF GRADUATE STUDIES IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF ARTS

GRADUATE PROGRAM IN INFORMATION SYSTEMS AND TECHNOLOGIES

YORK UNIVERSITY

TORONTO, ONTARIO

MAY 2021

Abstract

With the surging demand for Internet of Things (IoT) healthcare applications, a myriad of data privacy concerns come to light. Cloud computing inherits the risks of exposing data to re-identification vulnerabilities. A secure solution is storing and processing data locally on edge, but it lacks the provision of powerful machine learning (ML) needs. An improved computing framework is required to incorporate ML capabilities and user-data confidentiality. We perform a systematic study of IoT healthcare systems and propose a three-tier architecture that protects and enables data sharing. The edge anonymizes data using differential privacy (DP); transmits it to the cloud to train ML classifier; sent back trained classifier to edge to make inferences. Our findings show 1) XgBoost classifier performs relatively well; classifiers' accuracy trained using DP data is close to that of original data 2) Round-trip execution performance of architecture shows high average mean and variance with higher privacy budgets.

Acknowledgement

Firstly, I would like to thank my supervisor, Professor Marin Litoiu for consistently guiding my efforts towards achieving this thesis accomplishment. His expertise in the related field has helped me to hone my research skills and his vision for delivering innovative solutions is recommendable. Having worked in the Data Engineering space for three years, I am glad I got this opportunity to explore more in the Data Science and Data Privacy domain.

Thank you to Dr. Lauren Sergio for accepting to be on my supervisory committee. A sincere thanks to Dr. Sumona Mukhopadhyay for providing feedback and suggestions on improvising the thesis. I also would like to thank Dr. Mark Shtern for his continuous assistance and excellent technical expertise vital to this thesis. I thank the committee members Dr. Serban Dinca-Panaitescu and Dr. Manar Jammal, for providing valuable feedback and exciting perspectives on my research during the thesis defense.

Last but certainly not least, I would like to extend my warmest thanks to my father Naresh, my mother Kusam, my brother Kunal and my friends Rishabh, Pharmeet, and many others for their constant support throughout my Masters. It has been an incredible journey, and completing this milestone would not have been possible if it were not for your unwavering and never-ending cheers along the way.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

As forecasted by the International Data Corporation (IDC), by 2025, there will be 41.6 billion connected IoT devices worldwide, collectively generating 79.4 zettabytes (ZB) of user data. The growth and scale of these platforms opens up several opportunities for both the technology industry and researchers to use this data and information to create new knowledge aimed at improving the remote healthcare management systems. Today, IoT generated data is being used extensively in smart healthcare [2] [3] [4] [5] [6] to solve problems like fall detection, seizures prone behaviours, monitoring the elderly and personalized healthcare support that might include receiving healthcare support from a practitioner or receiving real-time alerts due to an emergency. When designing robust and socially responsible IoT systems such as these, two important principles are at play; the system's overall architecture and data privacy. The architectural design phase is one of the most critical activities in the development of IoT systems, and the decisions made here have significant implications for both economic and quality assurance goals. Examples of these IoT architectural decisions include distributing processing and analytics capabilities over the edge or cloud tier of IoT systems. Due to the increasing complexity of real-time healthcare services, full-fledged architecture must utilize the full potential of technologies like the cloud and the machine learning (ML) which is extensively being used in healthcare for different tasks such as classification, regression, and deep learning. The efficiency and the round-trip speed of these algorithms depends on data volume and placement of data and computation

tasks in the overall architecture of the application.

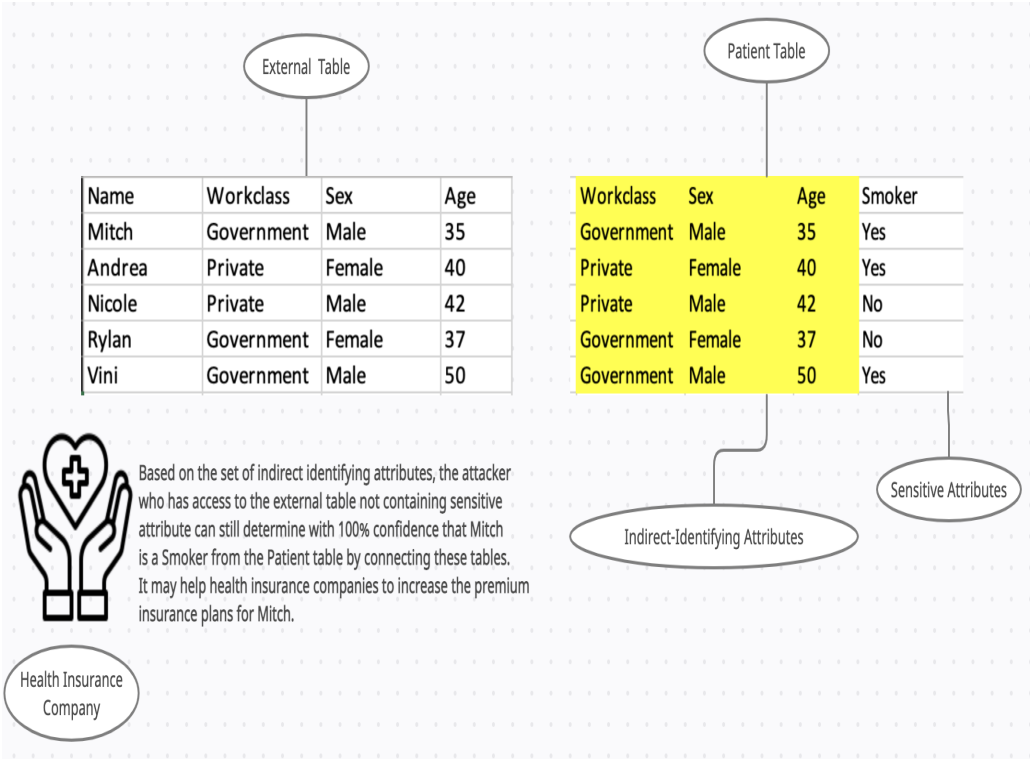

Figure 1: Existing Problem in Data

Large-scale data collection in the IoT poses significant privacy challenges and may hamper the further development and adoption of these technologies by privacy-conscious individuals and organizations [7]. Storing data in its raw form on the Cloud poses a severe threat to an individual's privacy as background knowledge can be secured and utilized by third-party vendors [8]. Figure 1 explains one problem that arises from sharing data, with a user subscribed to an application with their health insurance provider [9]. Even when some private data (such as "Name") is not present in the Patient profile table, sensitive information can be inferred by linking the patient table with one that is available publicly, marked as the External table in Figure 1. Concerns further build when the data involved is micro-data, representing an individual's entire information as a record in analytical databases. When this data is being used in data mining computations with the purpose of analyzing patients' health problems for example, the data's privacy can be compromised through these complex computing processes. Though the

possibility of data breach is present in so many IoT systems, remote healthcare management systems pose further controversial issues when sharing data as the information is critical to someone's identity and well-being. It is thus essential to enable both sharing useful information over the cloud while protecting data privacy [10].

Several studies have mentioned privacy-preserving techniques that deal with ethical concerns of data identification, or the linking of information stored in different databases about the same individual. One such technique is k-anonymity, first introduced by Latanya et al. [11] in 1998, which states that, a release of data is said to have the k-anonymity property if each person's information contained in the data release cannot be distinguished from at least k - 1 individuals whose information also appears in this same release. Despite having promising group-based anonymity, k-anonymous data is susceptible to many attacks. It becomes even worse if the adversary who is trying to unveil data already has some background knowledge at their disposal with what exists in the public domain. To overcome this issue, another technique known as (ϵ, δ)-DP was developed in 2006 by Cynthia et al. [8] which ensures that the probability of a randomized function's output on the database is equal to its probability on a neighboring database that differs by at most 1 record. (ϵ, δ)-DP is achieved by adding noise to the data, through several methods such as the Laplacian distribution while preserving statistical usefulness [9]. DP can be added at any step of the workflow, including data ingestion, data collection, data transmission, data storage, ML training and output data, allowing performing statistical analysis without compromising data. Hence any learning or analysis made from such databases is devoid of an individual's contribution to the data because it protects it against table linkage attacks, as illustrated in Figure 1.

With the recent success of ML applications, efforts are being directed towards integrating ML computing with privacy-preserving mechanisms. Hittmeir et al. [12] evaluated the performance of ML classifiers trained on synthetic data generated using different tools, and learned that the DataSynthesizer tool is suitable for classification tasks because of the similar accuracy achieved by this data as that

of original data . Vanichayavisalsakul et al. [13] also evaluated the performance of several privacy models and ensembled classification algorithms to determine any significant changes in the accuracy of ML classifiers. Though they found ensemble algorithms performing better in comparison to single classification algorithms, the privacy budget ϵ was not considered. Both of these works do not explore integrating privacy-preserving computations within IoT architecture. Though Wang, Tian, et al. [14] used edge-based DP computing by storing only partial data across all the layers of architecture, sharing only part of data over the cloud makes the ML capabilities limited. Hence, our architecture works on this gap by enabling data sharing and balancing the trade-off between data quality and privacy.

To this end, this thesis aims to bridge the gaps in this research, by providing a comparative study using a data anonymization framework integrated with sensor-edge-cloud architecture and (ϵ,δ) ML computations on DP data. The proposed privacy-preserving sensor-edge-cloud architecture ensures that any analysis from the database and ML computations is devoid of any specific individual. We will first outline the research objectives, including the research questions that this thesis answers. This is followed up by enumerating thesis contributions and then explanation of the thesis' structure.

1.1 Research Objectives

The concerns about developing IoT architecture that maintain data privacy and accuracy of ML classifiers trained on anonymized data has led to the following research questions:

RQ 1: Will the most optimal ML classifier for the original data, also be the most optimal one for the DP data?

RQ 2: Does the proposed data anonymization process brings better ML classifier accuracy as compared to previous studies?

RQ 3: Is it feasible to develop an edge-cloud architecture that preserves data privacy while maintaining ML accuracy?

1.2 Thesis Contributions

The main contributions of this thesis are:

- A systematic review of existing remote healthcare IoT systems from different aspects like architecture, processing, machine learning, and privacy.
    1. With this research, we find that most existing systems use three-tier architecture such as sensor-edge-cloud or sensor-fog-cloud. The choice between edge and fog is based on the emergency of real time analytics and how crucial the data's privacy is.
    2. From a privacy perspective, though traditional methods like verifier based password-authentication and concealment processes protect data, the data utility gets reduced and machine learning performs poorly on data with less quality. Privacy methods that preserve the data's utility, such as privacy models like K-anonymity, L-diversity, Differential Privacy, are required.
- Validated the usage of DP data by comparing performance of ML classifiers on original and anonymized data.
    1. The accuracy of the ML classifier trained on differentially private data is close to the original data's accuracy.
    2. For three out of the five datasets, the results on anonymized data were better than the previous studies as mentioned above.
    3. The XgBoost algorithm works fairly well with most of the datasets for both the original and private data.
- A novel three-tier privacy aware ML IoT framework using (ϵ, δ)-DP.
    1. We propose sensor-edge-cloud based IoT architecture incorporating (ϵ, δ)-DP at the edge (local device such as desktop or laptop) to ensure privacy and classifier training at cloud (Amazon Web Services or any cloud) for utilizing its computational power.

2. The round trip performance evaluation of proposed architecture shows the anonymization time and cloud model training time increases with increase in privacy budget and shows high variance.

1.3 Thesis Organization

The rest of the thesis is organized as follows. Chapter 2 provides background details in and then related research work in Chapter 3. Chapter 4 presents the systematic literature review of existing remote healthcare management systems (RHMS), followed by privacy-aware edge-cloud architecture in Chapter 5. We will explain the experimental validation in Chapter 6 and discuss the findings of our experimentation with the architecture from there. We will conclude with Chapter 7, that presents the future scope of our thesis.

Chapter 2

Background

This chapter provides details on the background of the concept of data anonymization by discussing privacy-preserving mechanisms that this thesis uses in Chapter 2.1. It is followed by explaining different data anonymization models in Chapter 2.2. Many studies also use the hybrid data privacy models to reap the benefits of all the models used in the process described in Chapter 2.3. Chapter 2.4 briefly explains the ARX data anonymization tool and the SafePub algorithm used by this tool to anonymize data.

2.1 Privacy-Preserving Mechanisms

Though there are several ways user privacy can be ensured, we will discuss ensuring privacy in databases that are being considered for the scope of this thesis. One of the main data privacy strategies is making data un-linkable to individuals, that is, to anonymize them. Anonymized data is no longer considered personal because they are outside the scope of GDPR, lifting legal restrictions that apply to personal data. This chapter describes state of the art data anonymization techniques and models [15].

2.1.1 Statistical Disclosure Control

Statistical disclosure control (SDC), also known as statistical disclosure limitation (SDL) or disclosure avoidance, is a technique used in data-driven research

to ensure no person is identifiable from the results of an analysis of a survey, administrative data or in the release of microdata where every record conveys information on a particular respondent [16]. Usually, a microdata set contains attributes that may be classified as identifiers, quasi-identifiers, sensitive and insensitive attributes, which allow unequivocal identification of individuals such as social security numbers or full names, which need to be removed before the publication of the microdata set onto the cloud. On the other hand, a group of quasi-identifiers may allow linkage with public information. Examples include occupation, address, age, gender, height, and weight. The insensitive attributes are the ones that do not carry any personal information and hence can be released whereas, the microdata set which contains sensitive information such as salary, religion, political affiliation, or health condition can be far more damaging. Beyond protecting against identity disclosure, SDC must prevent intruders from guessing the confidential attribute values of specific respondents (attribute disclosure) [15], because such attacks reveal new information about an individual without actually revealing their identity.

2.1.2 Non-Perturbative Masking

In SDC, masking refers to the process of obtaining an anonymized data set $X^{'}$ by modifying the original X. Masking can be done using perturbative or non-perturbative methods. With the first approach, the data values of X are perturbed to obtain $X^{'}$. In contrast, in non-perturbative masking, $X^{'}$ is obtained by removing some values and by making them more general. Yet the information in $X^{'}$ is still accurate, although less detailed; as an example, a value might be replaced by a range containing the original value [15]. Some of the standard non-perturbative methods include:

Sampling: Instead of publishing the whole data set, only a sample of it is released.

Generalisation: The values of the different attributes are recorded in new, more general categories such that the information remains the same, albeit less specific.

Top/bottom coding: In line with the previous method, values above (resp. be-

low) a certain threshold is grouped into a single category.

Local suppression: Too few records sharing a combination of quasi-identifier values may lead to re-identification. This method relies on replacing specific individual attribute values with missing values so that the number of records sharing a particular combination of quasi-identifier values becomes larger and thus mitigating de-identification of data as more data becomes indistinguishable.

2.1.3 Privacy Models

For an anonymized data set $X^{'}$ to be safe/private enough, it needs to be sufficiently anonymized. The level of anonymization can be assessed after the generation of $X^{'}$ or before generating it. Though the former method of posterior assessment requires several iterations before generating a suitable database for further analysis, the latter method of anterior assessment is implemented using privacy models that allow selecting the desired privacy level before producing $X^{'}$ [15]. Several privacy or anonymization models are explained in the following subsection.

2.2 Data Anonymization Models

Data anonymization models describe the criteria and process to set a privacy guarantee for an anonymised database. These models specify conditions that the data set must satisfy to keep disclosure risk under control. Privacy models usually depend on one or several parameters that determine how much disclosure risk is acceptable.

Table 1: Patient Table

Workclass	Sex	Age	Dementia
Federal Government	M	35	Y
Local Government	M	38	Y
State Government	M	38	Y
Self Employed	F	30	N
Private	F	30	N
Self Employed	F	30	N
Private	F	30	N

2.2.1 K-anonymity

It is one of the first privacy-preserving models introduced by Latanya Sweeney and Pierangela Samarati in a paper published in 1998 for the purpose of preventing record linkage attacks [11] [17]. For the given set of attributes in a database (D1) containing sensitive information, the same set of attributes in another database (D2) that does not contain sensitive information, can still be mapped to determine an individual to which that sensitive information belongs. Hence, the set of quasi-identifiers needs to be indistinguishable from at least k-1 records. So even if the adversary attacker gets access to the data, there will be a k number for the same query of records returning as an answer. This group of the same set of quasi-identifiers is known as equivalence classes. This model assumes that the adversary is aware of the set of attributes forming quasi-identifiers. As can be seen from Table 1, if "Workclass", "Sex", "Age" are set to be quasi-identifiers in the patient table, then the adversary who has access to an external table not containing sensitive information, can still link an individual to the possibility of having Dementia. For example, based on the accessible information "State Government", "M", 38, by an attacker, it can be deduced with 100% confidence from Table 2 that Mitch has a Dementia because the quasi-identifier information “State Government”, “M” and “38” from Table 2 is linkable with Table 1. Whereas after applying 3-anonymous (Table 3) models to the patient table, the confidence of determining if Mitch has Dementia is reduced to 50% as now there are 2 indistinguishable records that satisfies the same group of quasi-identifiers.

2.2.2 L-diversity

Even if the k-anonymity is satisfied, there is still the possibility of determining sensitive information if all the values in an equivalence group are similar. To overcome this issue, Machanavajjhala et al. in 2007 [18] introduced an L-diversity model to prevent this from attribute linkage. This model requires every quasi-identifier group to contain at least l "well-represented ” sensitive values. For example, in the in Table 1, the equivalence group "Self Employed", "F", 30 are two anonymous quasi-identifiers. However, an adversary with access to external

Table 2: External Table

Name	Workclass	Sex	Age
Nicole	Private	F	30
Rylan	Federal Government	M	35
Ana	Private	F	30
Mitch	State Government	M	38
Andrea	Self Employed	F	30
Yosua	Local Government	M	38
Angelica	Self Employed	F	30
Amine	Local Government	M	39
Vini	Self Employed	F	32

Table 2 can still guess with 100% confidence that Angelica does not have Dementia. In contrast, if we apply the 2-diversity model to Table 1, it will ensure that this equivalence group contains at least two distinct values of the sensitive attribute "Dementia" , reducing the confidence of estimation or matching the information to 50%

Table 3: 3-Diversity Patient Table

Workclass	Sex	Age	Dementia
Government	M	[35-40)	Y
Government	M	[35-40)	Y
Government	M	[35-40)	N
Government	F	[30-35)	N
Non-Government	F	[30-35)	N
Non-Government	F	[30-35)	N
Non-Government	F	[30-35)	N

2.2.3 t-Closeness

Even if the L-diversity models ensure diverse sensitive attributes corresponding to respective equivalence classes, it does not consider the global distribution of the data, and the close distance placement of similar information results in easy derivation of sensitive attributes. Hence, Li et al. in 2007 [19] proposed another privacy model known as T-closeness, which ensures that the distribution of a

Table 4: 4-Anonymous Patient Table

Postal Code	Sex	Age	Response
123**	M	[35-40)	Medium
123**	M	[35-40)	High
123**	M	[35-40)	Medium
123**	M	[35-40)	High
123**	F	[30-35)	Medium
123**	F	[30-35)	Low
123**	F	[30-35)	Medium
123**	F	[30-35)	Low

sensitive attribute in any equivalence class is close to the distribution of the attribute in the overall table (i.e., the distance between the two distributions should be no more than a threshold of t). For example, in Table 4, even though there are 4-anonymous and 2-diverse based categories with "Postal Code" and "Age", an adversary who is assumed to know that Bob's information is in the first four records could still estimate that his response time is more than average. The previous privacy models group them without considering their overall distribution in the table. However, after ensuring t-closeness in Table 5, the responses are uniformly transformed among all equivalence groups, reducing the confidence level in estimating the overall response time.

Table 5: Illustration of t-Closeness

Postal Code	Sex	Age	Response
1234*	M	[30-40)	Medium
1234*	F	[30-40)	High
1234*	M	[30-40)	Medium
1234*	F	[30-40)	Low
1235*	M	[30-40)	Low
1235*	F	[30-40)	Medium
1235*	M	[30-40)	High
1235*	F	[30-40)	Medium

2.2.4 δ-presence

Both record linkage and attribute linkage assume that the attacker already knows that the victim's record is in the table, revealing the victim's sensitive information [20]. A table linkage occurs when someone is able to estimate the presence or absence of the victim's record in the released table. For example, in Table 6, if an adversary has access to an external table and then patient Table 3, after joining them together with "Workclass", "Sex", "Age", the probability of inferring the presence of Mitch in the patient Table 3 is 3/4=0.75 as there are three matching records for the given quasi-identifiers in Table 3 and 4 in Table 6 (Government, M, [35-40)). To prevent this, Nergiz et al. in 2007 [21] proposed δ-presence model to bound the probability of inferring the presence of any potential victim's record within a specified range $\delta = (\delta_{min}, \delta_{max})$. However, this assumes that the data publisher has access to the same External Table 2 as the attacker, which may not be a reasonable assumption.

Table 6: External Anonymous Table

Name	Workclass	Sex	Age
Nicole	Non-Government	F	[30-35)
Rylan	Government	M	[35-40)
Ana	Non-Government	F	[30-35)
Mitch	Government	M	[35-40)
Andrea	Non-Government	F	[30-35)
Yosua	Government	M	[35-40)
Angelica	Non-Government	F	[30-35)
Amine	Government	M	[35-40)
Vini	Non-Government	F	[30-35)

2.2.5 Differential Privacy

This is one of the privacy models that does not focus on attacks like record linkage and attribute linkage but rather focuses on how someone's posterior belief in sensitive information changes after accessing the published data. Dwork in 2006 [9] proposed that the risk to the record owner's privacy should not substantially in-

crease due to participating in a statistical database. (ϵ, δ)-DP is thus coined on the below notion:

A mechanism K satisfies (ϵ, δ)-DP if for every $S \subseteq Range(K)$ and for every pair of neighboring datasets D1 and D2 (differing on at most one record):

$$Pr[K(D1) \in S] \leq exp(\epsilon) \times Pr[K(D2) \in S] + \delta \tag{2.1}$$

where probability is taken over the randomness used by the mechanism K. It ensures that the outcome of analysis on a database would be no different irrespective of the presence or absence of an individual's data.

Hence any learning about such databases is devoid of an individual's contribution to the data. For instance, if the probability of determining the presence of particular individual is 0 in both the datasets and considering there is no privacy leakage (δ=0) then taking log on both sides implies ϵ=0 (absolute privacy). The two key parameters used for implying DP are Epsilon ϵ and Delta δ where epsilon is the privacy budget, and delta is the maximum accidental leak of some information. Instead of defining these private parameters for the data, the application of these parameters is on the data processing method [22], that is, the process used to select data based on some randomized mechanism. Also, with increasing the privacy budget , data privacy is reduced as they are inversely proportional. Hence, less noise will be injected into the data due to decrease in data privacy. As mentioned earlier, noise entails adding the randomization to the data. This is usually used in two settings - interactive (noise is added with every answer to a query on a database and the actual database is not released) [23] and non-interactive (noise is added once for all first, and then the database is published). It is widely being adopted by some of the world's largest companies including the U.S. Census Bureau used it in 2008 for demonstrating commuting patterns [24], and in 2015, Google used DP when sharing historical traffic statistics [25]. Apple employed it to improve its intelligent personal assistant technology in 2016 [26], Microsoft for telemetry in Windows [27] in 2017, as well as LinkedIn for advertiser queries [28] in 2020.

2.3 Hybrid Data Privacy Models

This combines syntactic models such as K-Anonymity, L-Diversity, and semantic models like DP. Li et al. in [29] demonstrated that random sampling followed by attribute generalization and the suppression of every record which appears less than k times satisfies (ϵ, δ)-DP for every privacy budget $\epsilon \geq -ln(1-\beta)$ which is the probability of record randomization. This is extended by Bild et al. [30] to implement their ARX tool. Because of these hybrid models, the advantages of different privacy models can be merged into one database, and as such several studies have explored different combinations of these models. These are also often used for a non-interactive, general-purpose setting [22] [31] [23] which is why we have adopted the combination of these for this thesis.

2.4 ARX Tool & SafePub Algorithm

ARX tool is comprehensive open-source software for anonymizing sensitive personal data [32]. As it supports a wide variety of privacy and risk models, methods for transforming data and analyzing the use of output data and has been widely used and demonstrated in many research studies such as Prasser, Fabian, et al. [33], used it for anonymizing biomedical data as it preserves the truthfulness of data. They also extended this ARX tool to optimize de-identified health data by enabling the usage of statistical classifiers and a method of assessing their performance [34]. Eicher, Johanna, et al. [35] have also utilized the well-known ARX anonymization tool for biomedical data with ML techniques to support the creation of privacy-preserving prediction models. It follows the SafePub mechanism to generate the (ϵ, δ) differential private data for which the high-level algorithm can be seen as below:

It first randomly samples records with probability $\beta = 1\text{- } e^{-\epsilon_{anon}}$ using function RandomSampling () followed by attribute generalization and then suppressing every record that appears less than k times where k is derived from ϵ_{anon} and δ. Here the the total privacy budget $\epsilon = \epsilon_{anon} + \epsilon_{search}$. The transformations set in T are full domain generalization schemes that generate all attribute values to a common

Algorithm 1 SafePub Algorithm to Generate (ϵ, δ) DP Data [30]

Input: Database A, ϵ_{anon}, ϵ_{search}, δ, β, counts, evaluation metric e
Output: Anonymized database $A_{\epsilon,\delta}$
$A_s \leftarrow RandomSampling(A, \epsilon_{anon}, \beta)$
$T \leftarrow InitializeTransformationsSet()$

for $i \leftarrow 1$ `to counts` **do**
    $T \leftarrow$ Update()

    for $(t \in T)$ **do**
        $A_a \leftarrow Anonymize(\mathrm{A}_s, \mathrm{t}, \epsilon_{anon}, \delta)$
        score$\leftarrow Evaluation(\mathrm{A}_a, \mathrm{e})$
    end for
    $t_e \leftarrow$ Probabilistically select $t \in T$ based on score calculated from evaluation metric and ϵ_{search}
end for
$A_{\epsilon,\delta} \leftarrow Anonymize(\mathrm{A}_s, t_e, \epsilon_{anon}, \delta)$
return $A_{\epsilon,\delta}$

level of a certain hierarchy [36]. For instance, if one node at level 2 is generalized to a value at level 1, all the nodes at the same level 2 will also be generalized to a value at level 1. Anonymize () perform the database's anonymization based on the optimal generalization scheme t chosen from set T. Each result from Anonymize () is evaluated based on a given evaluation metric that gives a generalization scheme score. The metric used in this study was granularity [30] that penalizes values that are generalized to a higher level in the generalization hierarchy. The score is then used when a generalization scheme is randomly selected, where a better score increases the chance of choosing that scheme. The final database is anonymized based on the best generalization scheme found. From the parameters ϵ_{anon} and δ in (ϵ_{anon}, δ)-DP, the k for the k-anonymization is computed, the resulting k thus depends on the parameters set for the DP. This paper states that δ should be chosen so that $\delta < 1/\mathrm{n}$, where n is the size of the database, and at least $\delta \leq 10^{-4}$ holds. For ϵ the authors set $\epsilon_{search} = 0.1$ since ϵ_{anon} had greater impact on the performance. This value was also used in this study, but for $\epsilon \leq 0.1$,$\epsilon_{search} = \epsilon/10$.

Chapter 3

Related Work

In this chapter, we present key related work in using multi-tier architecture to develop IoT solutions in Chapter 3.1. Several studies rely on three-tier architecture because of the enormous amount of data generation and the need to avoid loading all the computational tasks like data processing, data storage and data analytics on cloud. As the sensor layer is not capable of performing heavy machine learning tasks, lightweight devices like fog and edge (close to data origination and low processing power) are being used consistently to divide the task load of the cloud (farthest from data origination and high processing power). From there, we'll briefly outline privacy-aware IoT architectures in Chapter 3.2. Several papers have been published regarding data privacy, which relies on numerous privacy law enforcement such as GDPR, HIPAA, HITECH, and its concrete implementation of these rules and laws still remain unclear at the present moment. Li, Chao, and Balaji Palanisamy [37] reviewed the state-of-the-art principles of privacy laws, the architectures for IoT, and the representative privacy-enhancing technologies (PETs). They analyzed how legal principles can be supported with careful implementation of PETs to meet the privacy requirements of the individuals interacting with IoT systems.

3.1 Multi-Tier Architecture For IoT

With the increasing demand for healthcare services and regularly monitoring patients, sensor technologies and multi-tier IoT architectures make it possible to develop more intelligent systems. Uddin et al. in [38] proposes a three-tier sensor-edge-cloud architecture to reduce pressure on the clouds by conducting activity prediction using Recurrent Neural Network (RNN) on an edge device (i.e., personal computer or laptop). Their experimental results show that the recommended approach outperforms other traditional methods. Devarajan, Malathi, et al. in [6] propose an energy-efficient fog-assisted healthcare system to maintain the blood glucose level and illustrate the improved performance of this system in terms of energy efficiency, prediction accuracy, computational complexity, and latency. Akrivopoulos, Orestis, et al. in [39] also utilizes the patient's smartphone as a Fog gateway (device farthest from the cloud in terms of proximity to the data origination) for securely sharing them to other authorized entities. Andriopoulou et al. in [40] transfer the computing intelligence from the cloud to the edge network as fog computing operates closer to the user, avoiding delay and network failures in healthcare service delivery. Mahmud et al. in [41] introduce a fog-based IoT-Healthcare solution structure and the integration of Cloud-Fog services in inter-operable Healthcare solutions extended upon the traditional Cloud-based structure. Their experimental results point towards improvements that include instance cost, network delay, and energy usage.

3.2 Privacy-Aware IoT Architectures

Numerous privacy-preserving technologies have been explained by DomingoFerrer et al. [15] such as the usage of Statistical Disclosure Control, Perturbative Masking, Non-Perturbative Masking, and Privacy Models such as K-anonymous, and Differential Privacy. Though not all of them provide full protection of data against attacks like table linkage and probabilistic linkage, DP is assumed to best ensure data privacy by making data unlinkable to an individual. Ukil et al. [42] demonstrated an automated health cardiac management system using simpler ar-

chitecture at the edge by using less number of features to keep the ML computations lightweight and DP on sensitive healthcare data. In contrast, our privacy-preserving edge-cloud architecture is not limited to choosing fewer features as we are harnessing the power of cloud to train the ML classifiers and doing predictions at the edge for ensuring data privacy. Naga Prasanthi Kundeti et al. [43] shows that K-anonymization along with generalization and suppression performed better than K-anonymization alone and evaluated its performance using two classification algorithms, namely Naive Bayes J48. However, we are evaluating (ϵ,δ) -DP as the privacy model and analyze its performance against several ML classifiers such as Logistic Regression, SVM, KNN, Random Forest, Decision Tree, Xgboost, Naive Bayes and demonstrate that XgBoost works well with most of the data. Paranthaman et al. [44] studied the effect of anonymization due to k-anonymity on the data mining classifiers using Naïve Bayes classifier and concluded that accuracy decreases with an increase in anonymity. However, this thesis advocates in-depth analysis into application of multiple classifiers on (ϵ,δ)-DP and validates its performance with original data. Abay, Nazmiye Ceren, et al. also demonstrated that deep learning models can be used to generate differentially private synthetic datasets [45] and compared the performance of existing techniques against many utility metrics. Xiao et al. used the DP framework [46] and applied it to wavelet transforms on the data before adding noise. Wang et al. used DP with regression models [47] by perturbing coefficients of the polynomial objective function, whereas here we are using DP by perturbing input data. Ji et al. [10] paper surveyed the interplay between ML and DP by demonstrating how noise can be added to the model at no cost to the utility, and cites considering generalization of the ML model, whereas we add the noise to the data anonymization process and apply Nested Cross-Validation to avoid any data bias. Many researchers have even incorporated this privacy model in IoT architectures. Xu, Chugui et al. [48] propose a local DP obfuscation framework (LPDO) and validate its performance in terms of privacy preservation level and data utility. Wang, Tian, et al. [49] propose edge-based DP computing for sensor–cloud systems where they are splitting the data differentially and storing it in three different layers. Piao, Chunhui et

al. also proposed a fog-based DP approach. However, their study was not covered in-depth, and only one database was used to evaluate the performance [50].

In summary , the importance of using data privacy models like (ϵ, δ)-DP in tandem with three-tier IoT applications, is one of the most useful data privacy algorithms that protects significant attacks like Record Linkage, Attribute Linkage, Table Linkage & Probabilistic Attacks . Researchers and industries are using it intensively, and we have used the combination of privacy-preserving techniques such as SDC, non-perturbative masking and privacy models using the ARX tool, and comprehensive open-source software for anonymizing sensitive personal data. This tool has been used and demonstrated in many research studies [33] [34] [35]. This thesis has explored the research questions that have not been studied directly to the best of our knowledge. It comprehensively brings out the performance evaluation of different ML classifiers on differentially private data by comparing it with original data and also validates it over the IoT architecture.

Chapter 4

Systematic Review of IoT Healthcare Systems

This chapter sheds light on the existing Remote Health Management Systems (RHMS) after a systematic and thorough review of nearly 80 papers. The analysis is presented through a taxonomy, categorizing the IoT systems into architecture, processing, machine learning, and privacy security. This classification was chosen based on the thesis's focus; nonetheless, all of them are based on IoT architecture. This quantitative summary shows the relative count to differentiate the purpose of survey and doesn't include the excluding count as many had overlapping focuses and themes. We first explain the research method being employed to filter papers out from the entire search query in Chapter 4.1. A taxonomy classifying the existing systems is then presented, which answers different questions that must be addressed while discussing RHMS in Chapter 4.2. Finally, some of the biggest challenges that arise while developing these IoT-based healthcare systems are listed in Chapter 4.3.

4.1 Research Method

As shown in Figure 2, the search strategy starts with collecting papers from widely accepted literature search engines including IEEE, Springer and ACM as well as the Google Scholar database. Software Publish or Perish With a focus between

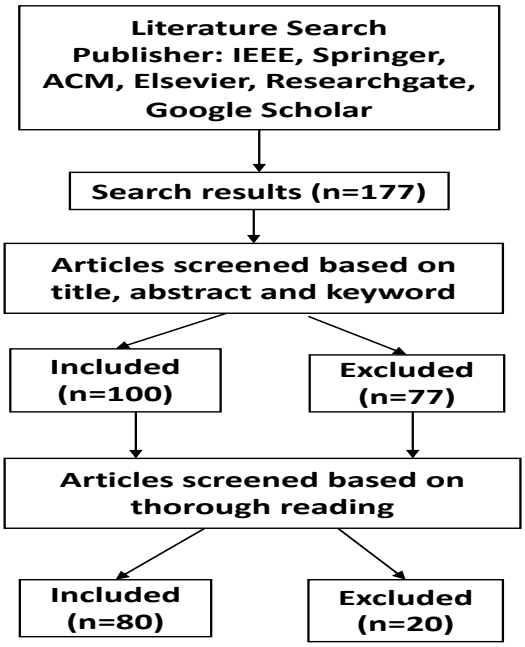

Figure 2: Survey Search Method

2011-2020, and using search words containing "Wearable health monitoring IoT Cloud". All 177 results were downloaded to a Microsoft Excel sheet. From here, the papers were skimmed through the abstract to narrow the research by categorizing them into five group: "Architecture (52) ","ML (11)", " Privacy Security (17)"," "Generic (20)", "Not Relevant (77)". Research papers categorized as "Not Relevant" were discarded as some were not accessible, while others had a small number of citations. The final usable research material contained 100 papers, all with at least ten citations. A further iteration of these winners included combining the abstracts and conclusions to get a handle on the subject matter. Choosing 80 papers as the scope of our review, we left the other 20 categorized as Generic, as those papers focus solely on theoretical knowledge which has little relevance to our thesis's objective.

4.2 Taxonomy Classifying Existing RHMS

A literature review's quality highly depends on the selected taxonomy scheme as this influences the depth of knowledge recorded about each studied approach. This

article has employed an iterative coding process to identify the taxonomy categories which answer the first research question (RQ1) of this thesis. The resulting taxonomy hierarchy is depicted in Figure 3. The first level of our taxonomy hierarchy structures the existing work according to four fundamental questions which are: (1) What is the type of architecture being used in the current RHMS? (2) Under what circumstances, particular type of data processing is preferred? (3) What kinds of cognitive algorithms are applied to the current RHMS? (4) What are the different privacy and security mechanisms applied to protect the data? We will discuss each of these questions in detail and define the implied taxonomy scheme as such. For each, we derived the sub-categories of the taxonomy related to the assigned question.

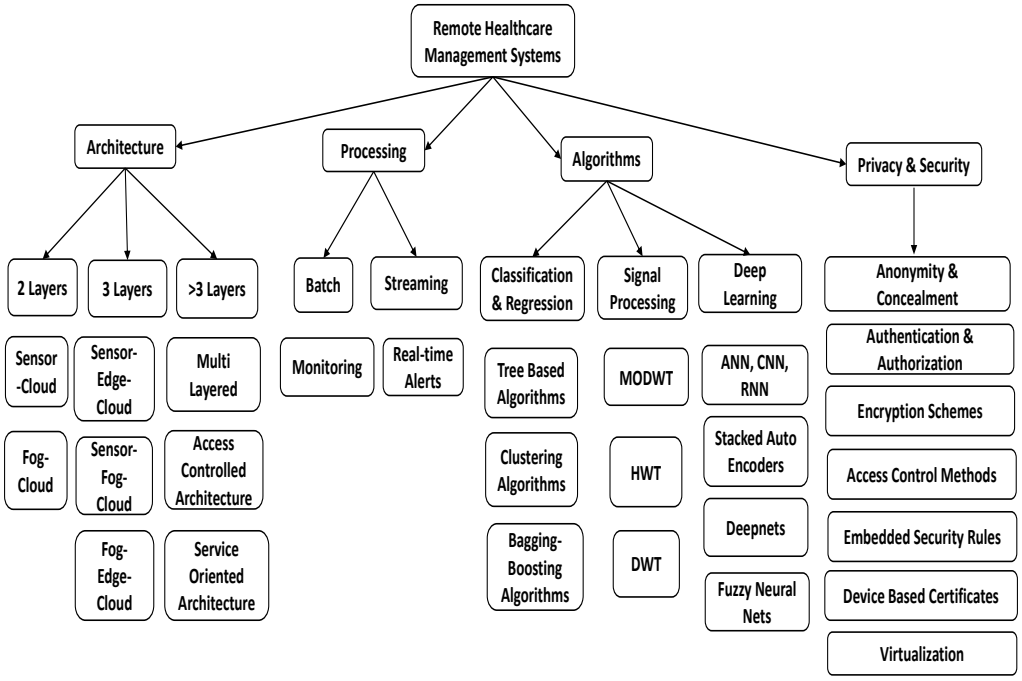

Figure 3: Classification of RHMS

4.2.1 Architecture

The first category relates to the architecture which is the building block of any IoT system. This classification aims to divide the papers based on the number of tiers involved in the system's architecture, the tiers represent the layers such as sensor, edge, fog or cloud. These are simplified into three groups: two-tier, three-tier, and

more than three-tier. Most of the available research focuses on three layers, where the first aims to gather data in sensors, the second aims to get that data from sensors to gateways, and the third layer analyzes, processes and stores.

1. Two-Tier architecture

   Table 7 represent the classification of papers based on the number of tiers involved. 2 tier architectures can be further divided into Sensor-Cloud and Fog-Cloud. In Sensor-Cloud architectures, sensors do data collection, analytic, and processing; storage is done by cloud. Wan et el. [51] presented wearable IoT-cloud based health monitoring system which embeds several sensors including the heartbeat and blood pressure sensors to capture data to be transmitted to cloud directly for analytical purpose. Sara at el. [52] propose an IoT-based service-oriented framework integrated with wireless body area network (WBAN) that outperforms baseline WBANs based on sensor life, existing cost and energy consumption. In Fog-Cloud architectures, fog device has been used for data collection and processing data temporarily for emergency alerts; the cloud is used for storing and monitoring data. Barik et al. [53] developed and evaluated a Fog-based spatial data infrastructure (SDI) framework and showed the efficacy of proposed system for enhanced analysis of geo-health big data generated from a variety of sensing frameworks. Dubey et al. [54] implemented fog computing on various types of physiological data and shows that the proposed Fog architecture could be used for signal enhancement, processing and analysis of various types of bio-signals.

2. Three-Tier architecture

   These architectures include three layers and are the most widely implemented architecture as per the statistics shown in Table 7. Several possible arrangements of different layers are Sensor-Edge-Cloud, Sensor-Fog-Cloud or Edge-Fog-Cloud. In Sensor-Edge-Cloud architectures, Wu et al. [55] present a hybrid wearable sensor network system with edge computing to improve the safety of working environments and reduce health risks in the construction industry. Greco et al. propose an edge-stream based computing infrastruc-

ture for real-time analysis of wearable sensor data [56]. Uddin et al. in [38] suggest a wearable sensor-based activity prediction system to facilitate edge computing in an intelligent healthcare system. Here we see several studies that rely on this three-tier architecture where the sensor layer is being used to gather data, the edge layer is used for data transmission or analytics, and the cloud is utilized for storage and computations. In Sensor-Fog-Cloud architectures, Devarajan et al. [6] proposed an energy-efficient fog-assisted healthcare system to maintain the blood glucose level and combat computational complexity, high latency, and mobility problems. Mahmud et al. in [41] analyze Cloud-Fog Interoperability in IoT-enabled Healthcare Solutions and evaluate it with simulations using the iFogSim simulator, and the results analyzed with distributed computing, reduction of latency, optimization of data communication, and power consumption. Yaseen et al. in [57] introduce a model based on Fog Computing infrastructure to keep track of IoT devices and detect collusion as mobility of IoT devices increases the difficulty of detecting such types of attacks. Similarly, the same kind of architecture has been adopted in [58] [59] [40] [60] [61] [62]. Akrivopoulos et al. outline Fog-Edge-Cloud [63] where the first layer is responsible for gathering data, the second layer for transmitting data from sensors to gateways, and the third layer analyzes and stores the data, and the second and third layer do interchange their operations based on the suggested approaches. Deploying such an end-to-end healthcare application includes IoT layers and cloud computing back-end services, which leverage the Fog computing approach's benefits, alleviating a series of security issues, scalability, and scalability bandwidth consumption reduction, latency decrease, and seamless operation.

3. More than Three-Tier architecture

   This is the architecture group with more than three layers. One such multi layered architecture [64] selects the communication medium based on data vitality and summarizes the patient data smartly using medically accepted severity levels resulting in reduction of data size. Bhatt et al. developed

access-based controlled architecture [65] based on interactions between different layers and illustrated it on remote health and fitness monitoring use cases. In contrast, Javdani et al. propose service-oriented architecture [66] which supports modular design, interoperation and software reuse. However, there have been open research problems that these multi layer architectures need to address by adopting measures like User-Based Device Authentication, User-Centric Data Security and Privacy, Edge Computing in wireless IoT, and Multi-Cloud Architecture [65] [67] [66].

Table 7: Quantitative Summary of Architecture Category

Architectures	Number Of Papers
Two-Tier	7
Three-Tier	33
>Three-Tier	5

4.2.2 Processing

This category was chosen to classify the existing RHMS based on the required type of processing related to the patient's health outcomes or needs.

1. Streamline Processing

   Firstly, streamline processing is required in the scenarios where the situation is life critical (for instance, getting a heart attack), and an emergency alert is required to be generated (calling for immediate help). For instance, Yacchirema, Diana, et al. [4] employ a wearable device that measures the acceleration of older people's body movements and analyzes the received data to rapidly detect falls. From there health care professionals would act accordingly, being alerted with messages in real-time. Verma et al. predicts a student's potential disease with its severity level by temporally mining the health measurements (frequent changes occuring numerously during a time interval) collected from IoT devices [68]. Wu et al. uses wearable sensors to measure the environmental conditions around the subject, and monitors vital signs and physiological data and triggers an alert if any emergency circumstance is detected [55]. Swaroop et al. present the design of a real-time

health monitoring system that facilitates multiple modes of connectivity between patients and clinicians [69]. Bhatia et al. also proposed architecture which is designed to monitor different activities (physical and ambient environment) inside an office that are relevant to the health of a person directly or indirectly for health severity assessment [62].

2. Batch Processing

   Secondly, there is batch time processing where a patient doesn't require constant monitoring, as health conditions are not considered critical. For instance, Uddin et al. used a database that consolidates vital signs and body motion recordings for ten volunteers from diverse profiles while also performing 12 physical activities to predict the underlying activity [38]. Hassan et al. [70] used the cloud-based framework that facilitates storing and processing the big data generated by ambient assisted living systems used to monitor patients suffering from chronic diseases in their homes, particularly the elderly. Batch or streamline processing can thus be used based on the severity of the underlying problem. As reflected in the statistics from Table 8, most papers focus on processing real-time data rather than batch data due to the need for the architecture to be responsive enough when dealing with healthcare data.

Table 8: Quantitative Summary of Processing Category

Processing	Number Of Papers
Streamline	38
Batch	11

4.2.3 Machine Learning

This category aims to classify the existing papers based on ML algorithms' applications of different tasks to improve RHMS. We first group them into three subcategories. First, "Classification & Regression Tasks" aims to diagnose diseases such as Seizure Detection, Chronic Diseases and Diabetes based on the present conditions, using linear, tree based, clustering, and bagging boosting algorithms.

Second, "Signal Processing Tasks" are used to analyze Electrocardiogram signals and detect a change in R waves' peaks, and monitoring patient health using morlet wavelets (MODWT), discrete wavelets (DWT) and haar wavelets (HWT). Finally, "Reinforcement Learning Tasks" aim to develop self-learning systems in the presence of an interactive environment using deep learning algorithms like artificial neural network (ANN), convolutional neural network (CNN), stacked encoders and deepnets. Think of this as methods based on learning representations from a wide variety of data types (numerical, categorical, text, images). Table 9 represents the quantitative summary of papers that focus on implementing these varieties of ML algorithms. Most of these papers use classification algorithms in different areas like efficient fog device placement for task offloading [71], or classifying imbalanced ECG beats while others used ML in fall detection systems for ambient assisted living [4], an intelligent system for indoor environments. Kumar et al. used fuzzy rule-based neural classifiers [72] whereas Verma et al. used classification algorithms for the prediction of potential disease severity [68]. Likewise, Castro et al. employed these algorithms for Human activity recognition [73] whereas Rajesh et al. used resampling techniques and ensemble classifiers for imbalanced ECG beats [74]. Signal processing algorithms such as maximum overlap discrete wavelet transform (MODWT) have been used [75] to identify R peaks in ECG signals. Numerous deep learning algorithms have been implemented including Deepnets to differentiate active daily living tasks effectively, and detect falls for elderly people [4], neural classifiers in [72], the artificial neural network in [68] with cloud centric IoT based disease diagnosis. Finally, CNN with the stacked encoder in [3] with a cognitive cloud-based intelligent healthcare framework for seizure detection and K means for discovering patterns in physiological data [76] are used.

Table 9: Quantitative Summary of Algorithms Category

Algorithms	Number Of Papers
Classification & Regression	6
Signal Processing	1
Deep Learning	5

4.2.4 Privacy & Security

This category classifies the existing papers based on security and privacy measures being taken to ensure data protection. Table 10 depicts that most papers focus on the security of RHMS and there is an obvious need to address the privacy aspect of these technologies. From security aspect, Elmisery et al. in [77] used a two-stage concealment process to preserve the privacy of users' health profiles using three mechanisms: trust-based concealment, a distributed paillier threshold cryptosystem, and attribute-based encryption. The first stage includes a local concealing process at the end-user's gateways, used to disguise the recorded health data before submission to external parties. The second stage is a global concealing process used to encrypt patient profiles before submitting them to the cloud healthcare recommendation services, all taking place on the fog layer side. Jia et al. in [78] also focused on authentication and key agreement using a verifier-based password-authenticated key exchange protocol; meaning server only preserves a verifier instead of an image of the password. Liu et al. [79] propose an EHR access control scheme that allows access policies encoded in linear secret sharing schemes by moving encryption computation offline. Sharma et al. [80] however, discusses the impact of privacy and the potential tradeoff among privacy, efficiency, and model quality. Their study suggests using privacy primitives such as homomorphic encryption schemes, data perturbation, differential privacy, and parallelizable methods including ensemble learning, to generate reliable privacy models. Hence, more studies are required that use privacy-preserving techniques such as private communications, privacy in databases and privacy-preserving computations based on the minimal research currently present [15].

Table 10: Quantitative Summary of Privacy & Security Category

Protection	Number Of Papers
Privacy	4
Security	14

4.2.5 Overall Analysis

Table 11 shows that most of the published papers were authored over the last five years, whereas there were few papers which fall in the first five years of our given range, likely due to the early evolution of 4G. The number of studies on telehealth continues to increase, however, they have limitations including privacy and security that remain unaddressed [81].

Table 11: Year-wise Quantitative Summary of Papers

Year Range	Number Of Papers
2011-2015	8
2016-2020	92

Table 12 shows that most of the gathered papers for systematic review aim to focus on the RHMS architecture. Many possible architectures are identified either by the number of layers involved (two-tier, three-tier, and more than three), or by the basis of architecture (contextual, distributed and hybrid). In contrast, few papers worked upon introducing cognitive abilities to the system and incorporating privacy security and based on this gap in knowledge, this thesis focuses on RHMS equipped with ML intelligence and Data privacy.

Table 12: Quantitative Summary of All Papers

Category	Number Of Papers
Architectures	52
Machine Learning	12
Privacy & Security	17
Generic	19

4.3 Challenges

When pertaining to IoT data, a majority of the time is spent cleaning and processing data to a stage where it can be useful for data analytics. There is a need to validate proposed systems with real data to see the practical implications of rapid generation of IoT data. To deal with how fast-paced data has become, IoT architectures, ML computation, and Data privacy all need to be acted upon sys-

tematically with state-of-the-art technologies. We will briefly define some of the challenges as below:

4.3.1 Using big data & semantic technologies for interoperability

As IoT is an interconnection of a range of devices over the Internet, each solution provides its own IoT infrastructure, devices, APIs, and data formats leading to interoperability issues. With the complexity of all this, dealing with a massive amount of big data generated from healthcare wearable devices, open-source big data technologies like Apache Flink, Kafka, and Cassandra and semantic technologies recommendations by W3C can be used to represent semantic data streams and convert data [56]. Heterogeneous smart devices such as laptops and smartphones can be used for gathering multimedia data [3] to deal with the same issue of integrating data from multiple sources. However, there are only a few papers that implement the RHMS using the technologies mentioned above, to deal with the issue of scalability and interoperability. Hence, future research should focus on exploring the architecture of RHMS with the perspective of using big data technologies to handle streaming IoT data.

4.3.2 Systematic implementation of proposed system

Based on the literature review, some papers have experimented their proposed systems on synthetic databases and have improved results to a level of very high accuracy. However, many papers leave the scope of validating their proposed system by executing them in real life scenarios [82] [81] [83] [70]. Some studies also focus on experimenting their system with an individual instead of a representative population [55] [56] [38] [6] [4] and as such there is a need for systematic implementation of proposed systems in real case scenarios similar to this.

4.3.3 Exploration of intelligence based systems

As per Table 12, there are fewer papers aimed at improving or bringing advanced technologies like ML abilities of the RHMS. Self-learning by the different architecture components is required to take proactive steps to provide necessary healthcare

services on time or in real time. Existing few cognitive equipped systems are working well in diagnosing diseases with a precise accuracy and sensitivity of 99.2%, and 93.5% respectively [3] Here, signals are classified as seizure or non-seizure with a probability score along with many other comparisons between different classification algorithms. All of these have developed the best algorithms that have mapped different types of diseases to date [68]. Hence, there is an utter need to research more about these topics, and with it develop a cognitive healthcare system.

4.3.4 Systematic guidelines for selecting the architecture

When integrating IoT & cloud with healthcare, systematic guidelines for selection of processing at local or cloud levels are still lacking. This depends upon the emergency of the information shared with the patient and on the network infrastructure settings. In the case of sending emergency alerts, local processing can be used to remove latency and delays and provide updates in real-time by storing essential data near devices. If the system requires monitoring patients data over the time and there are no critical conditions involved, then cloud can be used. Similarly, real-time and batch time processing can be done considering the importance of the information analyzed based on the underlying criteria.

4.3.5 Usefulness of container based virtualization technologies

Since an enormous amount of data is generated every second, more processing is required to handle it, bringing with it the need for more computation and resources. Hence, docker and virtualization technologies can be used, removing the burden of waiting for resources to execute tasks and facilitate seamless access. Some papers did mention the need to introduce Virtual Machines (VM) resources to deal with the healthcare data [70] [84]. The proposed architecture is scalable, cost-effective, and supports interoperability and lightweight access. However, there is no significant evidence as they have not been implemented in real-time, thus indicating the need to bring in more evidence.

In summary, this chapter categorizes the existing RHMS into the type of ar-

chitectures employed, types of ML computations that are being used, and privacy measures used with IoT applications. Most studies employed three-tier architecture because of the increase in the amount of data being generated and the need to divide the data storage, processing, and analytics over the layers. As healthcare studies involve real time requirements, most studies deal with streaming data. IoT is also used with ML in healthcare in different problems like classifying patients if they have cancer or not, regression tasks, signal processing to analyze EEG brain waves, and deep learning tasks. Likewise, privacy measures involving three-step-concealment, threshold cryptosystem, and attribute-based encryption, and more are applied based on a trade-off between privacy and data quality. Overall, most papers are published in the last five years, probably because of the advancements in 4G technologies. This chapter also identifies the most prominent challenges while developing IoT solutions such as interoperability, systematic implementation of the proposed system, exploration of intelligence-based systems and the usefulness of contained-based virtualization technologies. Additionally, this chapter highlights the need to integrate privacy and cognitive abilities with RHMS as only few studies focused on these aspects, considering most of the focus in category "Privacy & Security" was based out on Security measures.

Chapter 5

Privacy Aware Edge-Cloud IoT Architecture

This chapter proposes a privacy-preserving edge-cloud architecture based on Differential Privacy. We first explain the high level components and data flow using swim lanes in Chapter 5.1. This is followed up by explaining the data anonymization process in detail in Chapter 5.2, where we provide suitable examples to give a clear description of how anonymization is applied during the training phase. Then we explain the data transformation process in Chapter 5.3, which must be performed on testing data (unseen data) to bring it into the same feature space as that of training data.

5.1 High Level Components and Data Flow

Data storage and processing capabilities provided by cloud service providers (CSPs) might worry users about privacy leakage or data integrity on the cloud [79]. The time taken to provide the inference from the cloud takes more time than doing it over the edge. Since the edge layer is closer to the data's owner and the local devices, it provides low delay and better real-time operation [14]. Our proposed privacy-preserving framework assumes the data curator on the edge to be trustworthy, whereas the data recipient present at the cloud may or may not. Considering the low risk of privacy leakage at the edge over cloud computing [8], we will adopt

a data anonymization process at the edge before sending to the cloud, as shown in Figure 4. The fundamental principle is that the personal data is anonymized locally at the edge layer using DP before sending it to the cloud layer. This means we are using the cloud layer to do machine learning computations only, and the analytical data is DP anonymized. DP is immune to post-processing, meaning that once applied, an adversary cannot increase the privacy loss [85] or cannot learn anything new about it in the future.

As shown in the swim lane process map in Figure 4, the testing pipeline imitates the production as we validated this framework using testing data in the development phase. After gathering training data, data engineering is performed which includes missing information imputation, feature engineering, encoding of categorical or non-numeric features, splitting databases into training and testing sets. It is followed by the data anonymization process, which includes applying generalization and suppression techniques, privacy models, and utility models. After the anonymization is done, two outputs are provided: differentially private data and the optimal transformation that is chosen by the algorithm to be applied on testing data. The DP data is then transmitted to the core layer to train the classifier. The edge layer gets back the trained classifier once it is ready on the cloud. The testing phase which occurs at the edge layer takes the testing data, applies the required transformation, and sends the data to the trained classifier for predictions that would be evaluated based on achieved accuracy. If we achieve the required accuracy, we send the classifier to be deployed in production, otherwise we retrain the model. Following the same testing phase process during the production phase, actual data is first transformed using the optimal transformation chosen in the development phase and then fed to the deployed model to make predictions. As seen in Figure 4, only model training happens on the cloud and the rest of the process takes place on the edge (let say hospital computer systems), ensuring data privacy in this IoT architecture.

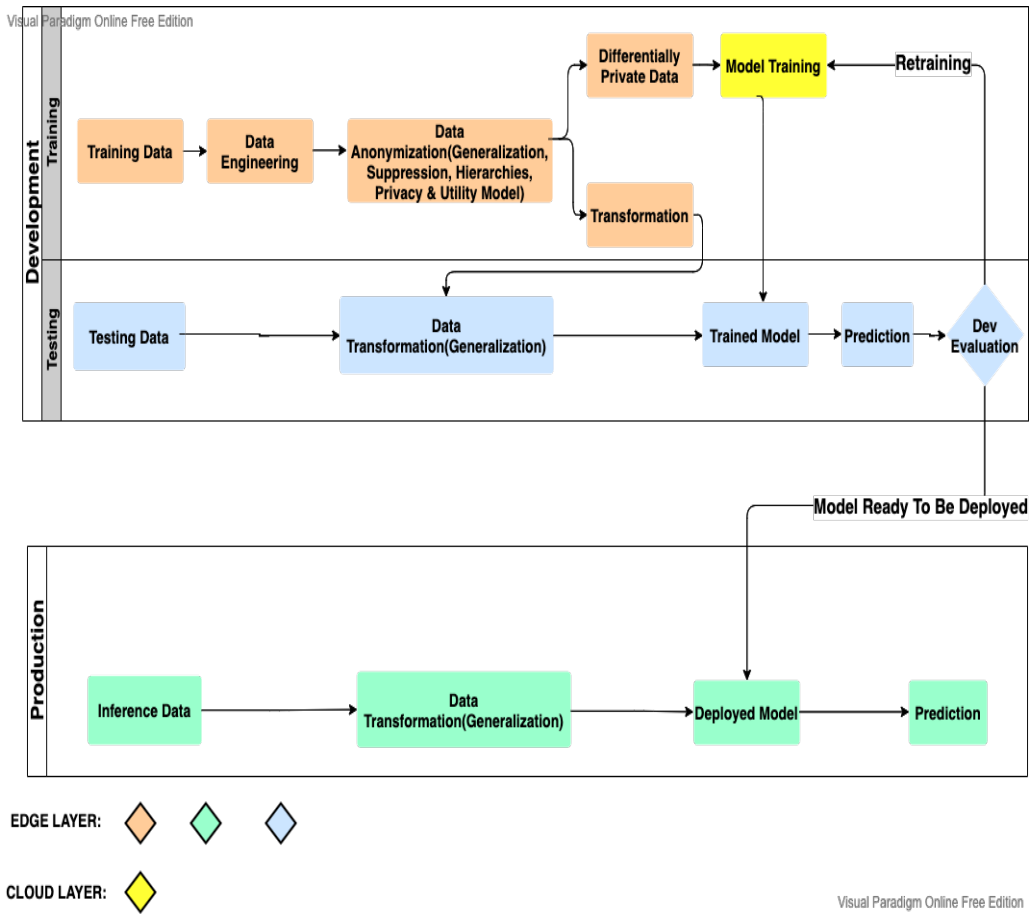

Figure 4: Swim Lane Process Map Representing Proposed Framework

5.2 Data Anonymization

We apply three different privacy-preserving techniques to the data: Statistical Disclosure Control, Non-Perturbative Masking, and Privacy Models. Statistical Disclosure Control [86] is a privacy technique that deals with the inherent trade-off between protecting the respondents' privacy and ensuring that the disseminated information is still useful to researchers. To implement this, we first classify the data into identifying attributes, sensitive attributes, insensitive attributes, and quasi-identifying attributes. Secondly, several non perturbative methods such as sampling, generalization, top/bottom encoding, and suppression are applied to anonymize the chosen attributes as part of the previous step. It either removes some values or makes them less specific. The output information is still accurate, although less detailed, and as an example, it replaces a value by a range contain-

ing the original value [15]. We apply generalization hierarchies that modify the original values of linkable attributes to more generic values semantically, as shown in Figure 5. However, we will maintain a depth of no more than five while creating hierarchies to avoid making the data over-generic. However, this can be increased depending upon the size of data and variance in values.

In Figure 5, we have created intervals at a difference of 5, 10, 20, all at respective levels, which may vary based on how generic values are needed. For example, level 1 would generalize values lying between 1 and 4 to [1,5), so the reader would see it as [1,5) and not the actual value, which might be 1, 2, 3, or 4. As shown in Figure 5, Level 0 contains original values followed by more general values as we level up and end with '*', which denotes suppression (hiding values). Thirdly, we choose privacy models that specify conditions that the data set must satisfy to keep disclosure risk under control. We will use (ϵ,δ)-DP and L-diversity models for this analysis, ensuring the data is not linkable to other databases even if accessed by a third-party and protects the sensitive attributes respectively. Lastly, we define utility models to determine the quality of data. We choose Loss measure, which summarizes the degree to which transformed attribute values cover an attribute's original domain. Based on the data quality of the best transformation chosen by the specified utility model, the anonymization tool chooses a particular level of data transformation. For any record that is not satisfying the specified criteria, we apply a suppression technique to remove it. Anonymized data thus obtained, and can be used to transmit over the cloud for further computations. Detailed anonymization steps can be seen in Figure 6.

5.3 Data Transformation

As the training data is semantically modified after applying data anonymization techniques, as shown in Table 13, we transform test data into the same feature space before using them for predictions. By transforming testing data, we mean applying the same generalizations that were being applied on training data excluding and re-applying all the privacy models again as the testing occurs at edge nodes only. The anonymization process chooses the best transformation (inter-

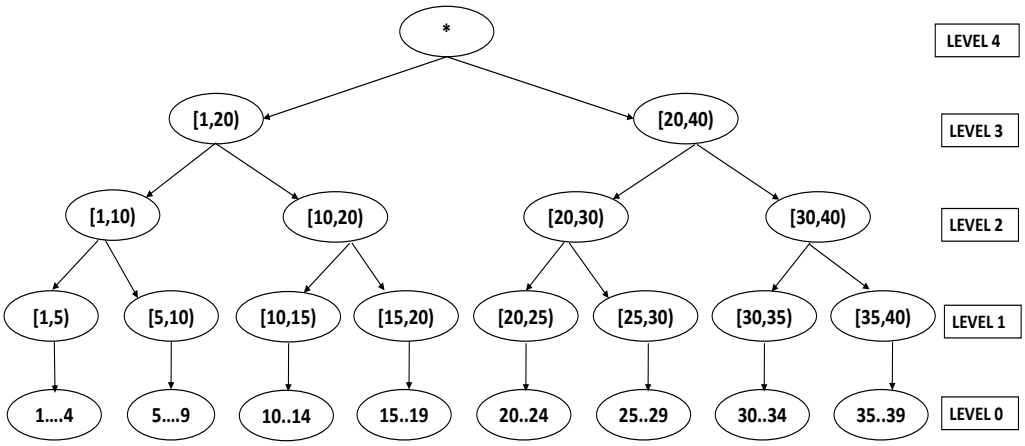

Figure 5: Generalization & Suppression Hierarchies for Age Column

val gap), which is applied to the testing data to bring it into the same feature space. For instance, looking at Figure 5, let's estimate that if the Level 2 hierarchy is chosen as a transformation after applying the data anonymization process on training data, we will apply the same interval of difference 10 to testing data as shown in Table 13. We can also see from the example in Table 1, attributes "Workclass", "Sex", "Age" are quasi-identifiers. The k-anonymization process's output includes generalization, such as grouping the data based on intervals, suppression (*), which happens if the underlying data is not distinguishable from at least k-1 records formed by quasi-identifiers. Four groups are made based on the three quasi identifiers and hence are not distinguishable. Since the SafePub algorithm [30] uses (ϵ, δ)-DP that makes use of the k operator along with ϵ and δ, Figure 5 and Table 13, which shows the "Age" attribute, represents the difference in the data anonymization and data transformation process. Hence, in transformation, there is no suppression of records based on any privacy or utility model. Since we are doing prediction at the edge, assuming it to be a trusted party, there is no need to anonymize testing data again, effectively saving on cost and time of the execution process. Thus, inferring on edge decreases the response times for predictions as compared to the cloud.

In summary, this chapter explains the proposed privacy-preserving architecture by first detailing the need for three-tier architecture for IoT systems and then the challenges of sharing raw data over the cloud. To counter these issues, we explain

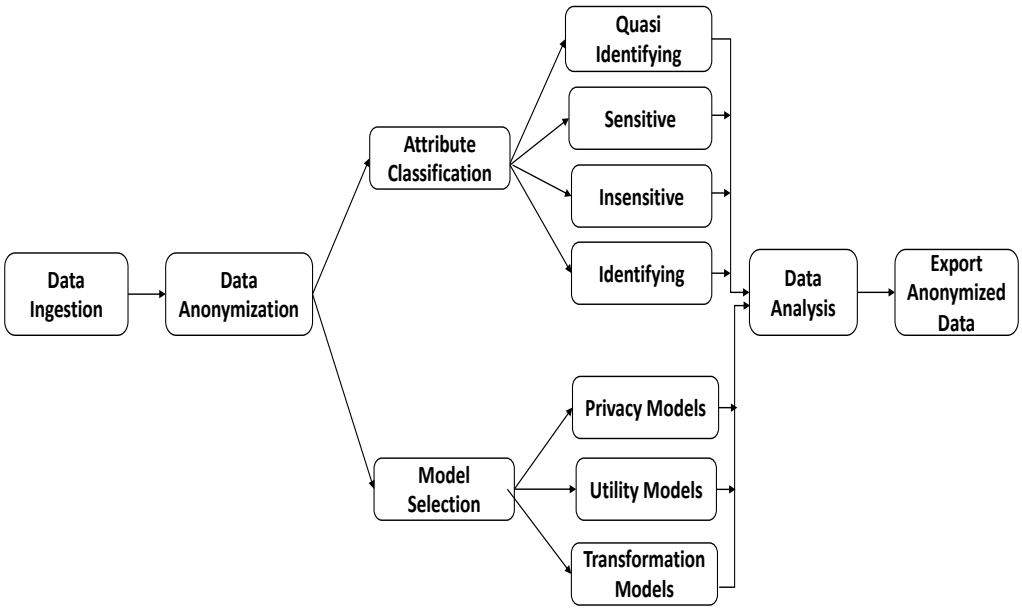

Figure 6: Detailed Data Anonymization Process

Table 13: Example of Data Transformation

Age Before Generalization	Age After Generalization
45	[40,50]
66	[60-70]
37	[30-40]
80	[70-80]
23	[20-30]
49	[40-50]

how we will use DP to anonymize the data without compromising user data(RQ3). We explain the data anonymization process and detailed steps of generalizing the data to less specific value so that the semantics of data is still preserved. In a ML context, we anonymize the training data and observe the anonymization impact on ML performance. Then while using testing data for predictions, we do not want to increase the cost overhead of doing data anonymization again. Also, since we are making predictions at the edge, we only applied data transformation on the testing data to bring it into the same feature space as that of training data.

Chapter 6

Experimental Validation

This chapter presents the five main experiments and other related experimental setups as represented in Table 14. We first explain the five databases from the UCI ML repository [87] in Chapter 6.1, which are being used to validate the performance of ML classifiers on private data as compared to the original data. We have used the Python programming language and ARX data anonymization tool [33]. For each task, we label the database containing categorical data using Label Encoder. We explain the default configuration and the hierarchies that we apply to generalize the data in Chapter 6.2. It is followed up by Chapter 6.3, where we briefly explain all the classification algorithms, including the hyperparameters range used to tune the classifiers. We used a balanced database as, in the healthcare domain, accurately detecting minority class observations is equally important [88]. Farrand et al. in [89] note that when we move to high levels of data imbalance, both the fairness metrics worsen across all levels of privacy. Since the data is balanced, we use accuracy metric to evaluate classifiers. Thus, we applied the SMOTE algorithm to get a fair accuracy and account for this. To measure the performance of the classifier, data quality, execution time of the anonymization process and classifier training on cloud, we used evaluation metrics as described in Chapter 6.4. In Chapter 6.5 we evaluate the performance of the ML classifier on differentially private data using proposed privacy-preserving edge-core architecture, with the core being the AWS cloud. Finally, we discuss the findings of results from our experimentation in Chapter 6.6.

6.1 Database

The databases from the UCI ML repository [87] are widely used by researchers for statistical analysis. These databases differ from each other on various aspects such as classification tasks (binary, multi-class), size of the data (699-48842), number of attributes (6-14), number of quasi-identifier (1-7), and the type of database (medical, non-medical). We used non-medical databases to show that the proposed anonymization process over the IoT architecture is not limited to the healthcare domain but is applicable to other domains as well who are looking to incorporate data privacy in their IoT systems.

6.1.1 Adult Database

The Adult database [87] contains 48842 number of records and 14 attributes. It is a binary classification task that predicts whether the individual's income exceeds fifty thousand US dollars.

6.1.2 Mammography Mass Cancer Database

The Mammography mass (Mammo) database [87] contains 961 number of records and 6 attributes. It is a binary classification task that predicts whether the patient has breast cancer based on mammographic mass.

6.1.3 Breast Cancer Database

The Breast cancer (Bcw) database [87] contains 699 number of records and 10 attributes. It is a binary classification task that predicts if the patient has breast cancer or not.

6.1.4 Contraceptive Method Choice Database

The Contraceptive method choice (Cmc) database [87] contains 1473 number of records and 10 attributes. It is a multi-class classification task that predicts the current contraceptive method choice.

6.1.5 Car Evaluation Database

The Car evaluation database [87] contains 1728 number of records and 7 attributes. It is a multi-class classification task that predicts the acceptability of a car based on different conditions.

Table 14: Summary of Datasets Properties and Privacy Parameters

Database	Records	Attributes	QIs	Classification	δ	ϵ
Adult	30162	9	8	Binary	10^{-6}	1,2,3, ∞
Bcw	579	10	1	Binary	10^{-6}	1,2,3, ∞
Cmc	1473	10	2	Multi-Class	10^{-6}	1,2,3, ∞
Car	1728	7	1	Multi-Class	10^{-6}	1,2,3, ∞
Mammo	830	5	1	Binary	10^{-6}	1,2,3, ∞

6.2 Data Anonymization

ARX tool is a comprehensive open-source software used for anonymizing sensitive personal data [32]. Below subsections represent the configuration chosen for the experimentation and the anonymization hierarchies applied to different columns to generalize the data.

6.2.1 Configuration

For anonymizing the data, we first specify the type of attributes as Insensitive (no modification), Identifying (will be removed), Sensitive (no modification but diversity check based on privacy model) and Quasi-Identifying (modified) on a case by case basis. With larger databases that include many dimensions, it is okay to take many attributes as quasi-identifiers because of the significant available number of records that are revealed after suppression. However, with smaller databases and fewer attributes, keeping more quasi-identifiers could reduce the quality of data because of the removal of several records by suppression. For our experimentation, we are using DP model for quasi-identifiers with privacy leakage $\delta = 0.000001$, privacy budget as $\epsilon = 1,2,3$ and ∞ (no privacy), medium

generalization, and distinct 2-diversity model for sensitive attributes. As explained in Chapter 2.2.2, if earlier using the K-anonymity model, Angelica's probability of having income <90K was 100%, now after implementing the L-diversity model, the same probability has been reduced to 50%. Hence, this model ensures there is diversity in the sensitive attributes (generally target features) based on the quasi-identifiers group. We choose Granularity/Loss to quantify information loss as explained in Chapter 6.4.

6.2.2 Hierarchies

For the Adult database, we have applied generalization hierarchies as suggested in [90]. We removed five columns as they were continuous and set all the attributes as a quasi-identifier before applying anonymization. Columns "sex" and "race" were applied to 2 level hierarchy with the second level as suppression since they can not be generalized more. Attribute "age" is the linkable attribute, and as such, we applied five levels of generalization and suppression, with level-0 being the specific values and subsequent levels being created at an interval difference of 5, 10, and 20, as shown in Figure 5. Similarly, column "workclass" containing values Private, Local-gov, Self-emp-not-inc, Federal-gov, State-gov, Self-empinc, Without-pay is generalized to Government, Non-Government. Also, column "education" is linkable with another database; we apply generalization and suppression at a depth of 3. Likewise, the columns "marital-status", "native-country," and "occupation" are the key attributes in the census database that must be generalized, maintaining its data utility semantically. Hence, all of these columns are generalized at a depth of 2. The sample original and anonymized database can be seen from Appendix C.

We kept the target class variable as sensitive for the remaining databases depending upon the underlying problem. In the Mammography database, we assumed the attribute "age" to be quasi-identifiers with three levels of generalization and suppression techniques applied at an interval of 20. However, any number of levels and intervals can be chosen, keeping in mind not to make it too generic. For example, generalizing age 25 to [20,30] would be better than [20-60] as the latter

over generalize the value, and hence the results may become less accurate. With the Breast Cancer database, column "clump thickness" is assumed to be linkable and hence is generalized at a depth of 3, keeping the interval difference of 5. With the Car evaluation database, for this experiment, we have set "number of doors" as quasi-identifying but based on the use case, we can set more quasi-identifiers. Lastly, in the Contraceptive method choice database, we assumed columns "wife age" and "number of children born" to be the attributes are linked. We applied the generalization followed by suppression at a depth of 3, which can be less or more depending on the quality of data achieved at the end of the anonymization process.

6.3 Classification Algorithms

We used seven different classification algorithms to comprehensively evaluate their performance with the original and differentially private data. As different algorithms work differently based on the underlying principle, we wanted to see how these perform compared to each other. Below we elaborate on the classifiers that we have used along with the chosen hyperparameters. We tuned the hyperparameters using Nested cross-validation and Grid search methods. More details on the chosen parameters can be seen in Table 15.

6.3.1 Gaussian Naive Bayes

It is a variant of Naive Bayes that follows Gaussian normal distribution and is a supervised ML classification algorithm. It is a generative probability model based on the Bayes theorem that assumes underlying features to be independent and takes the joint probability. Since it has no hyperparameters to tune, we used the default parameters.

6.3.2 Logistic Regression

It is a binary classification algorithm where either the event happens (1) or the event does not happen (0). It is a deterministic probability model that does not assume any independent relationship with underlying features. Penalty strength

C was varied exponentially from 0.1 to 100, regularization penalty l1 and l2 were used since not all of them work for different optimization solvers such as newton-cg, lbfgs, liblinear, saga.

6.3.3 K nearest neighbors

It is a simple algorithm that stores all available cases and classifies new cases based on a similarity measure [91]. It is a non-parametric technique and is a lazy learning model with local approximation. The number of neighbors varied oddly from 3 to 15. Different weights such as uniform and distance and varied distances Manhattan and Euclidean were passed as hyperparameters to compute the nearest neighbor using algorithmic parameters like auto , ball_tree, kd_tree, and brute to tune the model.

6.3.4 Support Vector Machine

It is a supervised ML algorithm capable of performing classification and regression problems. Penalty strength C varied exponentially from 0.1 to 100, and we chose gamma parameters as auto and scale. We use kernels poly, rbf, and sigmoid interchangeably depending upon the database and computation time.

Random Forest

It is a classification algorithm consisting of many decision trees. It uses bagging and features randomness when building each tree. The number of depth, split, and leaf parameters varied evenly from 2 to 6. We choose several estimators from a range of 100 to 300. Both criterion Gini and Entropy were passed along with max_features parameters log2 and sqrt to tune the model.

6.3.5 Decision Trees

It is a type of Supervised ML where data is continuously split according to a specific parameter. The number of depth, split, and leaf parameters were varied evenly from 2 to 6. Both criterion Gini and Entropy were passed along with max_features parameters log2 and sqrt to tune the model.

6.3.6 XGBoost

It is a widespread and efficient open-source implementation of the gradient boosted trees algorithm. It uses a more regularized model formalization to control overfitting, giving it better performance. We chose the number of estimators from a range of 100 to 300, learning_rate was varied exponentially from 0.001 to 0.1. The number of sub-samples varied oddly from 0.5 to 1.0, whereas we choose max_depth from 2 to 6 as hyperparameters.

Table 15: Chosen Algorithms and Hyperparameters

Algorithm	Hyperparameters
Gaussian NB	**Parameters:** default
Logistic Regression	**C:** 0.1, 10, 100; **Regularization:** l1, l2; **Optimization:** newton-cg, lbfgs, liblinear, saga
KNN	**Neighbors:** 3, 7, 9, 15; **Weights** uniform, distance; **Distance:** manhattan, euclidean; **Algorithm:** ball_tree, kd_tree, brute, auto
SVM	**C:** 0.1, 10, 100; **Gamma:** 0.001, auto, scale; **Kernels:** poly, rbf, sigmoid;
Random Forest	**Estimators:** 100, 200, 300; **Max Depth:** 2, 4, 6 ; **Max Split:** 2, 4, 6; **Max Depth:** 2, 4, 6; **Max Leaf:** 2, 4, 6; **Criterion:** Gini, Entropy; **Max Features:** log2, sqrt
Decision Trees	**Max Split:** 2, 4, 6; **Max Depth:** 2, 4, 6; **Max Leaf:** 2, 4, 6; **Criterion:** Gini, Entropy; **Max Features:** log2, sqrt
XgBoost	**Estimators:** 100, 200, 300; **Learning Rate:** 0.001, 0.01, 0.1; **Sub samples:** 0.5, 0.7, 1; **Max Depth:** 2, 4, 6

6.4 Metrics for Evaluation

We used Nested cross-validation to avoid overfitting the training database and choose the best classifier trained with the hyper parameterization technique. Many metrics are used to evaluate ML Models like average accuracy, precision, recall, but in this case, we used average accuracy score because this data is balanced meaning it contains a fair percentage of target classes. We used the Scikit learn [92]

library of python to evaluate the classifiers. When evaluating the proposed privacy-preserving architecture, we examined the time taken to anonymize the data, train classifiers locally and train it on cloud. We will briefly define them in the below subsections.

Algorithm 2 Nested K-Fold Cross Validation with Grid Search

Require: K folds, Database D, Parameters P, Models M, Accuracy List A
for $i = 1$ `to` K_o `splits` **do**
    Split D into $train_o$ & val_o data for ith split.
    for $j = 1$ `to` K_i `splits` **do**
        Split $train_o$ into $train_i$ & val_i data for the jth split.
        for each p in P and m in M **do**
            Train m on $train_i$ with set of hyper parameters p
            Test m on val_i and save the best M_i
        end for
        Choose best model from list of M_i for training $train_o$ & test using val_o.
    end for
end for
Choose the best model from the list of M_o and test it on $test_X$,$test_y$

6.4.1 Accuracy Score

It is another metric that can be derived out of confusion metrics. It checks the number of correct predictions made by the classifier against the total number of predictions. However, it should not be used in the database is imbalanced. Hence, we ensured that all the datasets were balanced before using this metric to evaluate the performance. In this case, we averaged the accuracies from Nested Cross Validations outer loop where the classifier has been trained using the best parameters from the inner Grid Search CV.

$$accuracy(y\_t, y\_p) = \frac{1}{n}\sum_{i=0}^{n-1} 1(y\_p_i = y\_t_i) \tag{6.1}$$

In the above equation, y_t is the true value, and y_p is the predicted value. Also, n refers to the number of samples, whereas i is used to iterating over the samples.

6.4.2 Data Anonymization Time

Here, we focus on the time taken by execution of the anonymization process at the edge. We enter required privacy parameters such as ϵ and δ into the UI. It is followed by choosing the type of attributes as quasi-identifying, sensitive, and non-sensitive, assuming that identifying attributes are removed already from the database. Then two files are input into the system, the first being the file to anonymize and the second being the hierarchy listing to be applied. Currently, the UI accepts one quasi-identifier and one sensitive attribute. It is sent to the ARX through its API, which then applies the passed privacy parameters and the default (ϵ, δ)-DP model for quasi-identifiers along with the l-severity model for sensitive attributes. Once the process is done, it outputs the anonymized file.

6.4.3 Model Training Time

Here, we focus on the time taken by the cloud to train the classifier. The training classifier on the cloud needs an anonymized file uploaded to the S3 bucket after receiving an anonymized output from the first scenario. The classifier takes the input from the S3 bucket, trains the classifier, and tests the unseen data uploaded as a test file on the S3 bucket. Once the classifier is ready, the classifier is saved as a tar file in the S3 bucket, which can then be downloaded to this application to use the inference classifier back at the edge.

6.4.4 Granularity/Loss

This metric is used to determine data quality of the anonymized data. It summarizes the degree to which transformed attribute values cover the original domain of an attribute. For instance, in the below Table 16, generalizing value Doctors to the interval {Doctors, Masters} will have an information loss given by the below formula:

IL = ((U_i - L_i) /(U - L))*100 = $((3-1)/16-1)*100 = (2/15)*100 = 13.33\%$

Table 16: Mapping Values for Education Column [1]

1. Doctorate	9.12th
2. Professional school	10.11th
3. Masters	11.10th
4. Bachelors	12. 9th
5. Associate (vocational)	13. 7th-8th
6. Associate (academic)	14.5th-6th
7. Some college	15.1st-4th
8. High School grad	16. Preschool

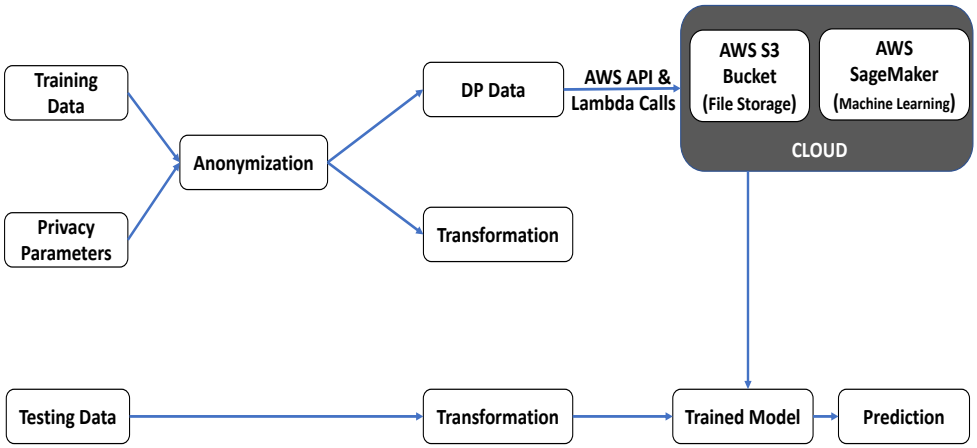

Figure 7: Implementation on Edge-Cloud Architecture

6.5 Implementation

In this chapter, we explain the implementation of the suggested framework as shown in Figure 7. The user will first enter the training data and privacy parameters as per their budget. This involves entering privacy budget, maximum offset leakage, quasi-identifying attributes, insensitive attributes, and sensitive attributes. Following this, it will be sent for an anonymization process which will output two items - differentially private data and the chosen optimal transformation. Once the anonymized data is ready, it will be sent to the Amazon S3 bucket through API calls to AWS Lambda function, a serverless computing method. Then AWS Sagemaker will take this data as an input to train a classifier. Once the classifier is ready, it is sent back to the edge layer. Now the trained classifier can be

used for making an inference. During inference, testing data is inputted, which will be first transformed as per the optimal transformation chosen before sending it to the deployed model for generating a prediction. The UI for this implementation method can be referred to in Appendix B.

In the context of our experiments, we deployed this proposed anonymization-based framework on Amazon Web Services (AWS), an on-demand cloud computing platform. AWS perfectly fits our case studies with the availability of building an end-to-end ML pipeline, given that it provides a huge stack of services to deal with enormous data from disparate sources. AWS's S3 buckets provide unlimited storage service for data objects favorable to the high volume of IoT data. Our case study's main idea is to automate the creation of S3 buckets and classifier training using AWS Sagemaker service. Our architecture was built using an Apple laptop with a configuration of 128GB memory and 8GB ram and a 1.4 GHz Quad-Core Intel Core i5 processor, and acts as the edge where data anonymization is performed using ARX API. For the cloud, we used AWS encompassed with AWS Sagemaker and S3 services; the "ml.m5.4xlarge" general purpose EC2 instance type with no GPUs for training AWS Sagemaker algorithms as recommended by Amazon. However, bigger instances with optimized compute or accelerated computing can be used as per the requirement. We apply the XgBoost classifier by downloading the containerized image from the respective region name and kept the parameters to default such as "train_instance_count as 1", "train_instance_type" as 'ml.m5.4xlarge', "train_volume_size" as 2, "train_use_spot_instances" as True, "train_max_run" as 300, "train_max_wait" as 600 where we are using managed spot training to reduce overall costs. By simply changing values such as instance_count and instance_type, we can change the size and number of instances we want to run on, which scales and distributes the training. The entire framework is illustrated using the python flask application.

6.6 Results & Discussion

This section will address our research questions and the corresponding analysis based on different classification algorithms' performance, accuracy comparison

between original and differentially private data, and comparison of differentially private accuracy with previous studies. We discuss the results of performance evaluation of different classification algorithms in response to RQ1 in Subsection 6.6.1 and determine if the most optimal ML classifier for optimal data will also be optimal for DP data. Subsection 6.6.2 compares ML accuracy achieved by our study with the previous studies for the same datasets. Finally, Subsection 6.6.3 discusses cloud performance results for two of the use cases in response to our research question RQ3 that analyzes the feasibility of edge-cloud architecture that preserves privacy while maintaining ML accuracy.

6.6.1 Classification algorithms performance

Let's inspect the research question, RQ1, by analyzing different classification algorithm's performance on multiple datasets against privacy budgets ϵ= 1,2,3 and ∞. The infinity privacy budget implies no privacy applied to the data. To our best knowledge, we are the first to bring out this comprehensive comparison among numerous classification tasks performed on data anonymized using (ϵ, δ)-DP and the original data. We analyze this by comparing if particular classification algorithms giving the best accuracy with original data also perform similarly with differentially private data as shown in Table 17. We achieve the listed test accuracies using the best models found using Nested Cross-Validation and Grid Search methods.

The listed accuracy for the original data is what we could achieve and may not necessarily be more than the benchmark accuracy. Thus, the comparison of DP data's accuracy is relevant to our accuracy and not the benchmark accuracy. With the Adult and Car evaluation database, we got less accuracy with differentially private data than original data. For the Mammography database, we could achieve similar accuracy with differential private data as that of original data and slightly higher for the Breast Cancer and Contraceptive Method Choice database. The difference between both accuracies is not more than 5%, but may vary based on the chosen privacy parameters and hyper-parameters. Hence, in this experiment, we could effectively train this classifier on differentially private data to reach an

accuracy relatively closer to that of original data. Further results on both the training and testing accuracy for all the use cases can be seen in Appendix A.

Database	Original Acc.	Original Alg.	DP Acc.	DP Alg.
Adult	82.15	XgBoost	77.66	XgBoost
Mammo	81.92	XgBoost	81.92	XgBoost
Bcw	98.53	XgBoost	99.62	KNN
Cmc	56.89	XgBoost	57.42	XgBoost
Car	99.44	Logistic Regression	98.58	XgBoost

Table 17: Best Testing Accuracy Comparison Respective Algorithms

As seen in Table 17, the Xgboost algorithm gives the best accuracy with original and differentially private data with most of the scenarios. It uses the best parameters found from nested grid search and hyper parameterization. Most of the datasets performed well with 100-200 ensembling trees, 0.01-0.1 as the learning rate and maximum depth of 2-6 across varied privacy budgets. However, as shown in Figure 8, Decision trees out of all give the worst performance with original and differentially private data in most cases. However, this might not apply to every type of data as the data distribution may vary based on the chosen parameters. Though, in 60% of the cases, the same algorithm performs best with both the datasets, we cannot ignore the remaining 40%. The XgBoost classifier gives the best accuracy 80% of the time with both the original and differentially private data.

6.6.2 Comparison with previous studies

This section compares the best accuracy that we could achieve and the previous studies. As shown in Figure 9, the green bars represent our study results, whereas the blue bars show the previous study results. For three out of the five databases, namely Breast Cancer, Car Evaluation, and Mammography Mass databases, our accuracies are slightly better than the previous studies. Table 18 shows the same comparison in tabular format for better understanding, where we highlight our accuracy that was higher than previous studies. We could not achieve the best

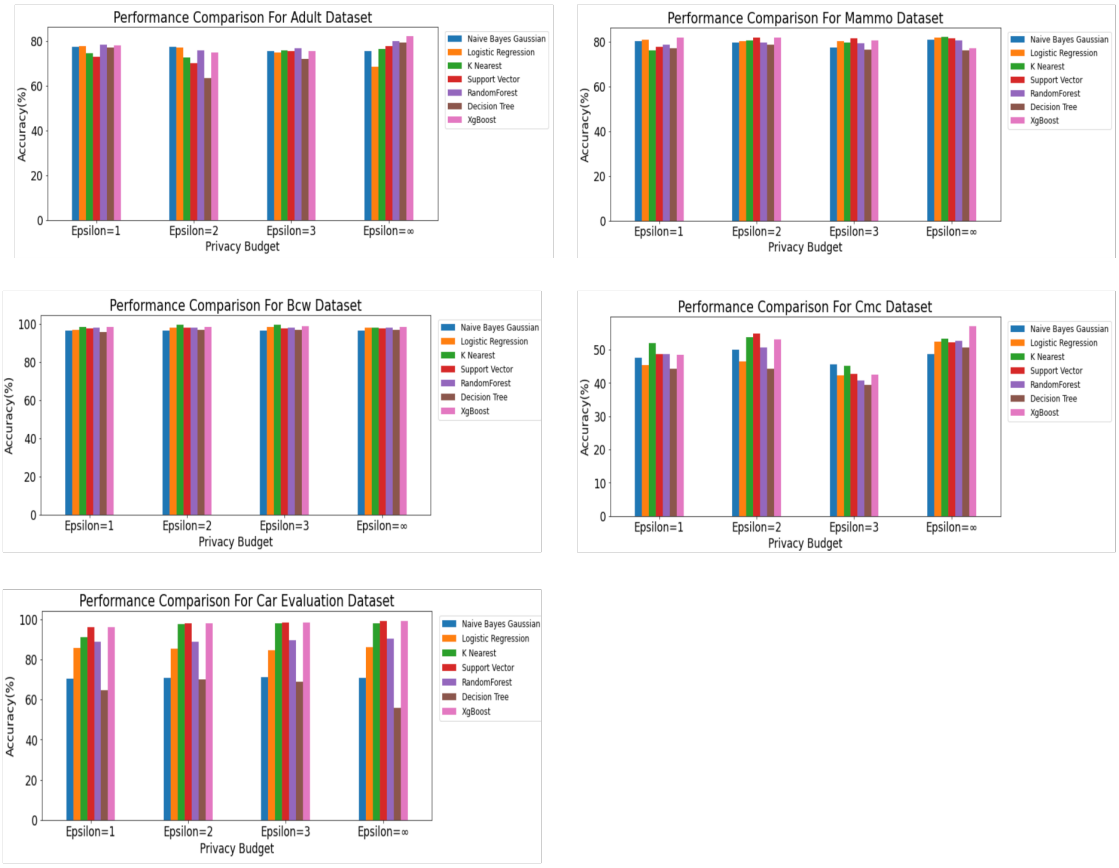

Figure 8: Classification Algorithms Comparison for Different Databases

accuracy in our study for the Adult database, with only 77.66% using the Xgboost algorithm on DP data which is less than 81% achieved in [93] using Bayesian DP. The CMC database gave an accuracy of 57.42% which is less than the previous study [45] that achieved a maximum accuracy of 63.67% through DP synthetic data generation using deep learning. However, with the Breast cancer database, our experiment showed slightly better accuracy of 99.62% using KNN and $\epsilon = 2$ in comparison to 97% achieved in [94] using $\epsilon = 1$ through feature selection using correlated DP. The Mammography database achieved an accuracy of 81.92% when compared to [45] which got maximum accuracy of 79.25% through synthetic data generation using auto-encoders. Overall, we achieved a good accuracy of 98.58% with the Car Evaluation database using the Xgboost algorithm compared to 94.04% achieved by previous study [95] using differentially private random forest. Collectively, this methodology works reasonably well compared with previous

studies.

Table 18: Accuracy Comparison Between Previous And Our Study

Database	Previous Study	Our Study
Adult	81	77.66
Mammo	79.25	**81.92**
Bcw	97	**99.62**
Cmc	63.67	57.42
Car	94.04	**98.58**

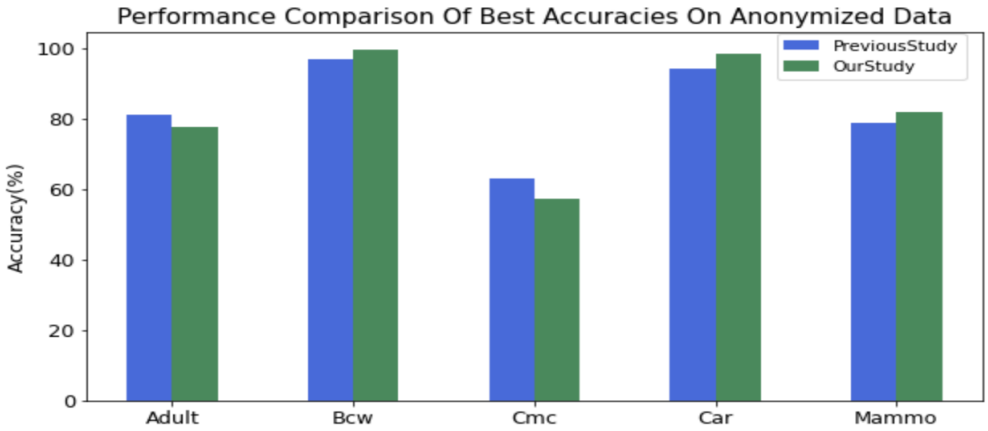

Figure 9: Accuracy Comparison Between Previous & Our Study

6.6.3 Architecture performance

Here we present the results for the experiments described in Chapter 6.5. The experiment was conducted ten times each for both the Mammography mass cancer & Car evaluation database. We reported anonymization time ($Anon_M$, $Anon_C$) and cloud model training time (Cmt_M, Cmt_C) in Table 19 for respective databases to show architecture performance in terms of execution time. We observed the execution time values against three different privacy budgets from ϵ= 1-3 and noted mean of the observations along with standard deviations. On observing the mean of anonymization execution times, we find that time taken to anonymize data increases with an increase in ϵ, because of the increased number of transformations built with the different combinations of hierarchies. With ϵ= 3, the mean of execution time is relatively high compared to lower privacy budgets, suggesting

not to prefer a high privacy budget. While training the classifier on the cloud, execution time to train the classifier keeps increasing with ϵ , because of the increasing size of the file as with increasing ϵ, less noise is introduced, and fewer records are suppressed. Hence, there is a trade-off between the quality of data and the privacy achieved. The time taken by cloud accounts for downloading the data from S3 buckets, downloading the containerized XgBoost classifier's image and training a model. If we observe the standard deviation for anonymization time and classifier training at cloud, the highest was using ϵ =3 compared to lower privacy budgets, which means high variance with high privacy budgets. Lower privacy budgets shows more stability in terms of their average performance across all the experiment runs.

Epsilon	Measure	Anon_M	Cmt_M	Anon_C	Cmt_C
ϵ =1	Mean	2.33	40.6	2.70	40.6
ϵ =2	Mean	4.77	41.2	4.53	47.3
ϵ =3	Mean	30.04	50.8	29.45	70.9
ϵ =1	SD	0.35	5.36	0.87	6.60
ϵ =2	SD	0.46	8.11	0.36	7.88
ϵ =3	SD	2.00	14.54	2.34	29.47

Table 19: Execution Performance For Mammo & Car Database (Seconds)

In summary, this chapter first explains the databases that we used to validate our framework. Then we explain the data anonymization process by listing out the default configurations to be used and the required hierarchies. It is followed by different classification algorithms that we used to determine the best classifier and the metrics used to evaluate classifier performance, cloud performance and data quality. Subsequently, we explain the implementation of edge-cloud architecture that facilitates privacy-preserving machine learning computation. Finally, we answer all the stated research questions through the results obtained after experimentation. With nested cross-validation grid search and hyper parameterization, we could take the ML classifier's accuracy to slightly better than original data for few cases and otherwise achieved accuracy with a difference of no more than 5% between both the data. Though different algorithms work differently with data,

however, XgBoost algorithm gave better results in many use cases on both original and differentially private data(RQ1). Also, the proposed data anonymization process involving statistical disclosure control, non-perturbative masking and privacy models did work well with 3 out of the 5 databases as compared to previous studies (RQ2) which were not that extensive in their scope. Finally, we implemented the experiments over the proposed edge-cloud architecture and achieved similar ML accuracies over the cloud and the end-to-end execution of the architecture shows the feasibility of the proposed privacy-preserving architecture (RQ3).

Chapter 7

Conclusion

The Internet of Things brought revolution to the technology industry with its ability to intelligently leverage the devices being used heavily around us. It is widely adopted in various domains like intelligent healthcare, smart cities and intelligent buildings and such adoption requires choosing suitable architecture for data storage, data processing, and data analytics. Every entity involved in the IoT process is vulnerable to different attacks and privacy threats and before building any IoT systems, one must understand the need to ensure data privacy and make it available for researchers to use and produce tools to combat this. Although several past studies focus on the use of three-tier architecture, there is no concrete analysis of how privacy is or can be ensured and at which tier of the architecture. We first present the summary of this thesis in Chapter 7.1 by answering key research questions. Finally, we list the future work that can be done to extend our work in Chapter 7.2.

7.1 Summary

In this subsection, we explain the contribution of this thesis and the final results of the experiments.

For our first contribution, we first presented a systematic literature review on existing RHMS from different aspects like architectures, ML tasks, and privacy measures. We found that three-tier architecture is used mainly with sensor-edge

cloud layers or sensor-fog-cloud layers. Based on the critical level of analytics, data processing is either done in batches or streamlined into real-time data. For instance, if the purpose is to monitor patients, then real-time analytics is employed. However, if the purpose is to analyze the patient history, then batch processing should suffice. IoT architectures are used in conjunction with ML intelligence because big data helps derive insights. Though traditional methods conceal the data from the privacy aspect, the quality of data is reduced, which does not facilitate using this data for ML or analytics purposes. Thus, semantic models like DP or hybrid models that take advantage of all the combining models are being encouraged for us, based on the data quality that is achieved for use for further processing.

For our second contribution, we validated our proposed framework that uses (ϵ, δ)-DP by implementing experiments on multiple case studies to compare the performance of ML classifiers trained on data with no privacy and anonymized data using varied privacy budgets in DP. After comparing their performance, we found that the accuracy of classifiers trained on differentially private data can achieve the accuracy as that of original data and may surpass some scenarios based on the underlying data. We have compared our results with the previous studies and observed that our methodology works reasonably well with most of the databases. Hence, we can effectively use (ϵ, δ)-DP and other privacy techniques such as statistical disclosure control and non-perturbative masking to make the data unlinkable to an individual without affecting its data quality. Therefore, differentially private data can be transmitted securely over the cloud without compromising an individual's privacy. As the accuracy of ML classifiers with differentially private data is similar to that of original data, this privacy model could be adopted in integration with the edge-cloud framework.

For our final contribution, we designed the three-tier edge-based data anonymization framework using (ϵ, δ)-DP for RHMS. Since the edge is closer in location to the data owner, there is less chance of adversary attacks on the private data. As cloud computing provides superior and advanced processing technological capabilities, we are doing data anonymization at the edge and ML training on the cloud.

The data is anonymized before sending it to cloud after applying (ϵ, δ)-DP which can then be shared over the cloud without worrying about re-identification risks. We evaluated the performance of the architecture by deploying it on edge-core architecture, with the core being the AWS cloud and edge being the local machine (computer). The anonymization process took longer due to the increase in the value of the privacy budget, and number of transformations to choose an optimal solution. Similarly, cloud model training increases with increase in the privacy budget because more records are chosen and hence the size of the file gets larger.

7.2 Future Work

For future work, as the current analysis includes classification tasks, it can be extended to apply the proposed architecture incorporating (ϵ, δ)-DP for regression tasks. With problems like regression tasks, the data would be mostly numerical, applying hierarchies would involve modifying these values using some distance calculations or replacing them with the mean or median of the numerical values being studied. Inan et al. [96] in their studies show how numerical attributes can be categorized into ranges or respective lower and upper bounds. The proposed anonymization process can be extended to accommodate the other uncertainties like how anonymization modifies the data by increasing quasi-identifiers and how the selection of privacy budgets is automated based on the data. Currently, the data is taken from the UCI ML repository and is processed batch-wise. The proposed architecture can thus be validated using real-time data to observe the performance of edge-cloud architecture in terms of other essential aspects such as overhead costs, change in execution times of anonymization and classifier training with increase in the size of databases, data engineering pipeline to combat streaming data. As the model's performance changes over time as part of the model drift, it can be automated to retrain the model when the drift crosses a certain threshold, allowing it to become self-adaptive. Also, deep learning models could be utilized with differential private data as well to determine further improvement on the achieved machine learning accuracies.

Bibliography

[1] V. S. Iyengar, "Transforming data to satisfy privacy constraints," in *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2002, pp. 279–288.

[2] S. B. Baker, W. Xiang, and I. Atkinson, "Internet of things for smart healthcare: Technologies, challenges, and opportunities," *IEEE Access*, vol. 5, pp. 26 521–26 544, 2017.

[3] M. Alhussein, G. Muhammad, M. S. Hossain, and S. U. Amin, "Cognitive iot-cloud integration for smart healthcare: case study for epileptic seizure detection and monitoring," *Mobile Networks and Applications*, vol. 23, no. 6, pp. 1624–1635, 2018.

[4] D. Yacchirema, J. S. de Puga, C. Palau, and M. Esteve, "Fall detection system for elderly people using iot and ensemble machine learning algorithm," *Personal and Ubiquitous Computing*, vol. 23, no. 5, pp. 801–817, 2019.

[5] M. Al-Khafajiy, T. Baker, C. Chalmers, M. Asim, H. Kolivand, M. Fahim, and A. Waraich, "Remote health monitoring of elderly through wearable sensors," *Multimedia Tools and Applications*, vol. 78, no. 17, pp. 24 681–24 706, 2019.

[6] M. Devarajan, V. Subramaniyaswamy, V. Vijayakumar, and L. Ravi, "Fog-assisted personalized healthcare-support system for remote patients with diabetes," *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, no. 10, pp. 3747–3760, 2019.

[7] J. H. Ziegeldorf, O. G. Morchon, and K. Wehrle, "Privacy in the internet of things: threats and challenges," *Security and Communication Networks*, vol. 7, no. 12, pp. 2728–2742, 2014.

[8] J. Sen, "Security and privacy issues in cloud computing," in *Cloud Technology: Concepts, Methodologies, Tools, and Applications*. IGI Global, 2015, pp. 1585–1630.

[9] C. Dwork, A. Roth *et al.*, "The algorithmic foundations of differential privacy." *Foundations and Trends in Theoretical Computer Science*, vol. 9, no. 3-4, pp. 211–407, 2014.

[10] Z. Ji, Z. C. Lipton, and C. Elkan, "Differential privacy and machine learning: a survey and review," *arXiv preprint arXiv:1412.7584*, 2014.

[11] P. Samarati and L. Sweeney, "Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression," 1998.

[12] M. Hittmeir, A. Ekelhart, and R. Mayer, "On the utility of synthetic data: An empirical evaluation on machine learning tasks," in *Proceedings of the 14th International Conference on Availability, Reliability and Security*, 2019, pp. 1–6.

[13] P. Vanichayavisalsakul and K. Piromsopa, "An evaluation of anonymized models and ensemble classifiers," in *Proceedings of the 2018 2nd International Conference on Big Data and Internet of Things*, 2018, pp. 18–22.

[14] T. Wang, Y. Mei, W. Jia, X. Zheng, G. Wang, and M. Xie, "Edge-based differential privacy computing for sensor–cloud systems," *Journal of Parallel and Distributed computing*, vol. 136, pp. 75–85, 2020.

[15] J. Domingo-Ferrer and A. Blanco-Justicia, *Privacy-Preserving Technologies*, M. Christen, B. Gordijn, and M. Loi, Eds. Springer International Publishing, 2020.

[16] C. Skinner, "Statistical disclosure control for survey data," in *Handbook of statistics*. Elsevier, 2009, vol. 29, pp. 381–396.

[17] L. Sweeney, "k-anonymity: A model for protecting privacy," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 05, pp. 557–570, 2002.

[18] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, "l-diversity: Privacy beyond k-anonymity," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 1, no. 1, pp. 3–es, 2007.

[19] N. Li, T. Li, and S. Venkatasubramanian, "t-closeness: Privacy beyond k-anonymity and l-diversity," in *2007 IEEE 23rd International Conference on Data Engineering*. IEEE, 2007, pp. 106–115.

[20] B. C. Fung, K. Wang, R. Chen, and P. S. Yu, "Privacy-preserving data publishing: A survey of recent developments," *ACM Computing Surveys (Csur)*, vol. 42, no. 4, pp. 1–53, 2010.

[21] M. E. Nergiz, M. Atzori, and C. Clifton, "Hiding the presence of individuals from shared databases," in *Proceedings of the 2007 ACM SIGMOD international conference on Management of data*, 2007, pp. 665–676.

[22] J. Bromark, "Privacy-preserving sharing of health data using hybrid anonymisation techniques: A comparison," 2019.

[23] N. Mohammed, R. Chen, B. C. Fung, and P. S. Yu, "Differentially private data release for data mining," in *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2011, pp. 493–501.

[24] A. Machanavajjhala, D. Kifer, J. Abowd, J. Gehrke, and L. Vilhuber, "Privacy: Theory meets practice on the map," in *2008 IEEE 24th international conference on data engineering*. IEEE, 2008, pp. 277–286.

[25] A. Eland, "Tackling urban mobility with technology," *Google Europe Blog, November*, vol. 18, 2015.

[26] A. P. Info, "Apple previews ios 10, the biggest ios release ever, 2016," *URL https://www. apple. com/pr/library/2016/06/13Apple-Previews-iOS-10-The-Biggest-iOS-Release-Ever. html*.

[27] B. Ding, J. Kulkarni, and S. Yekhanin, "Collecting telemetry data privately," *Advances in Neural Information Processing Systems*, vol. 30, pp. 3571–3580, 2017.

[28] R. Rogers, S. Subramaniam, S. Peng, D. Durfee, S. Lee, S. K. Kancha, S. Sahay, and P. Ahammad, "Linkedin's audience engagements api: A privacy preserving data analytics system at scale," *arXiv preprint arXiv:2002.05839*, 2020.

[29] N. Li, W. Qardaji, and D. Su, "On sampling, anonymization, and differential privacy or, k-anonymization meets differential privacy," in *Proceedings of the 7th ACM Symposium on Information, Computer and Communications Security*, 2012, pp. 32–33.

[30] R. Bild, K. A. Kuhn, and F. Prasser, "Safepub: A truthful data anonymization algorithm with strong privacy guarantees," *Proceedings on Privacy Enhancing Technologies*, vol. 2018, no. 1, pp. 67–87, 2018.

[31] J. Li, M. M. Baig, A. S. Sattar, X. Ding, J. Liu, and M. W. Vincent, "A hybrid approach to prevent composition attacks for independent data releases," *Information Sciences*, vol. 367, pp. 324–336, 2016.

[32] A. . A. Tool, "A comprehensive software for privacy-preserving microdata publishing." [Online]. Available: https://arx.deidentifier.org/anonymization-tool/analysis/

[33] F. Prasser, F. Kohlmayer, R. Lautenschlaeger, and K. A. Kuhn, "Arx-a comprehensive tool for anonymizing biomedical data," in *AMIA Annual Symposium Proceedings*, vol. 2014. American Medical Informatics Association, 2014, p. 984.

[34] F. Prasser, J. Eicher, R. Bild, H. Spengler, and K. A. Kuhn, "A tool for optimizing de-identified health data for use in statistical classification," in *2017 IEEE 30th International Symposium on Computer-Based Medical Systems (CBMS)*. IEEE, 2017, pp. 169–174.

[35] J. Eicher, R. Bild, H. Spengler, K. A. Kuhn, and F. Prasser, "A comprehensive tool for creating and evaluating privacy-preserving biomedical prediction models," *BMC Medical Informatics and Decision Making*, vol. 20, no. 1, pp. 1–14, 2020.

[36] D. Narula, P. Kumar, and S. Upadhyaya, "Privacy preservation using various anonymity models," in *Cyber Security*. Springer, 2018, pp. 119–130.

[37] C. Li and B. Palanisamy, "Privacy in internet of things: from principles to technologies," *IEEE Internet of Things Journal*, vol. 6, no. 1, pp. 488–505, 2018.

[38] M. Z. Uddin, "A wearable sensor-based activity prediction system to facilitate edge computing in smart healthcare system," *Journal of Parallel and Distributed Computing*, vol. 123, pp. 46–53, 2019.

[39] O. Akrivopoulos, I. Chatzigiannakis, C. Tselios, and A. Antoniou, "On the deployment of healthcare applications over fog computing infrastructure," in *2017 IEEE 41st Annual Computer Software and Applications Conference (COMPSAC)*, vol. 2. IEEE, 2017, pp. 288–293.

[40] F. Andriopoulou, T. Dagiuklas, and T. Orphanoudakis, "Integrating iot and fog computing for healthcare service delivery," in *Components and services for IoT platforms*. Springer, 2017, pp. 213–232.

[41] R. Mahmud, F. L. Koch, and R. Buyya, "Cloud-fog interoperability in iot-enabled healthcare solutions," in *Proceedings of the 19th international conference on distributed computing and networking*, 2018, pp. 1–10.

[42] A. Ukil, A. J. Jara, and L. Marin, "Data-driven automated cardiac health management with robust edge analytics and de-risking," *Sensors*, vol. 19, no. 12, p. 2733, 2019.

[43] N. P. Kundeti and C. S. R. MVP, "Accuracy and utility balanced privacy preserving classification mining by improving k-anonymization," *International Journal of Simulation–Systems, Science & Technology*, vol. 19, no. 6, 2018.

[44] J. Paranthaman and T. A. A. Victoire, "Performance evaluation of k-anonymized data," *Global Journal of Computer Science and Technology*, 2013.

[45] N. C. Abay, Y. Zhou, M. Kantarcioglu, B. Thuraisingham, and L. Sweeney, "Privacy preserving synthetic data release using deep learning," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2018, pp. 510–526.

[46] X. Xiao, G. Wang, and J. Gehrke, "Differential privacy via wavelet transforms," *IEEE Transactions on knowledge and data engineering*, vol. 23, no. 8, pp. 1200–1214, 2010.

[47] Y. Wang, C. Si, and X. Wu, "Regression model fitting under differential privacy and model inversion attack." in *IJCAI*, 2015, pp. 1003–1009.

[48] C. Xu, J. Ren, D. Zhang, and Y. Zhang, "Distilling at the edge: A local differential privacy obfuscation framework for iot data analytics," *IEEE Communications Magazine*, vol. 56, no. 8, pp. 20–25, 2018.

[49] T. Wang, Y. Mei, W. Jia, X. Zheng, G. Wang, and M. Xie, "Edge-based differential privacy computing for sensor–cloud systems," *Journal of Parallel and Distributed computing*, vol. 136, pp. 75–85, 2020.

[50] C. Piao, Y. Shi, J. Yan, C. Zhang, and L. Liu, "Privacy-preserving governmental data publishing: A fog-computing-based differential privacy approach," *Future Generation Computer Systems*, vol. 90, pp. 158–174, 2019.

[51] J. Wan, M. A. Al-awlaqi, M. Li, M. O'Grady, X. Gu, J. Wang, and N. Cao, "Wearable iot enabled real-time health monitoring system," *EURASIP Journal on Wireless Communications and Networking*, vol. 2018, no. 1, pp. 1–10, 2018.

[52] S. Ghanavati, J. H. Abawajy, D. Izadi, and A. A. Alelaiwi, "Cloud-assisted iot-based health status monitoring framework," *Cluster Computing*, vol. 20, no. 2, pp. 1843–1853, 2017.

[53] R. K. Barik, H. Dubey, K. Mankodiya, S. A. Sasane, and C. Misra, "Geofog4health: a fog-based sdi framework for geospatial health big data analysis," *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, no. 2, pp. 551–567, 2019.

[54] H. Dubey, A. Monteiro, N. Constant, M. Abtahi, D. Borthakur, L. Mahler, Y. Sun, Q. Yang, U. Akbar, and K. Mankodiya, "Fog computing in medical internet-of-things: architecture, implementation, and applications," in *Handbook of Large-Scale Distributed Computing in Smart Healthcare*. Springer, 2017, pp. 281–321.

[55] F. Wu, T. Wu, and M. R. Yuce, "An internet-of-things (iot) network system for connected safety and health monitoring applications," *Sensors*, vol. 19, no. 1, p. 21, 2019.

[56] L. Greco, P. Ritrovato, and F. Xhafa, "An edge-stream computing infrastructure for real-time analysis of wearable sensors data," *Future Generation Computer Systems*, vol. 93, pp. 515–528, 2019.

[57] Q. Yaseen, M. Aldwairi, Y. Jararweh, M. Al-Ayyoub, and B. Gupta, "Collusion attacks mitigation in internet of things: a fog based model," *Multimedia Tools and Applications*, vol. 77, no. 14, pp. 18 249–18 268, 2018.

[58] G. L. Santos, P. T. Endo, M. F. F. da Silva Lisboa, L. G. F. da Silva, D. Sadok, J. Kelner, T. Lynn *et al.*, "Analyzing the availability and performance of an e-health system integrated with edge, fog and cloud infrastructures," *Journal of Cloud Computing*, vol. 7, no. 1, pp. 1–22, 2018.

[59] S. K. Sood and I. Mahajan, "Wearable iot sensor based healthcare system for identifying and controlling chikungunya virus," *Computers in Industry*, vol. 91, pp. 33–44, 2017.

[60] B. Farahani, F. Firouzi, V. Chang, M. Badaroglu, N. Constant, and K. Mankodiya, "Towards fog-driven iot ehealth: Promises and challenges of iot

in medicine and healthcare," *Future Generation Computer Systems*, vol. 78, pp. 659–676, 2018.

[61] C. S. Nandyala and H.-K. Kim, "From cloud to fog and iot-based real-time u-healthcare monitoring for smart homes and hospitals," *International Journal of Smart Home*, vol. 10, no. 2, pp. 187–196, 2016.

[62] M. Bhatia and S. K. Sood, "Exploring temporal analytics in fog-cloud architecture for smart office healthcare," *Mobile Networks and Applications*, vol. 24, no. 4, pp. 1392–1410, 2019.

[63] O. Akrivopoulos, I. Chatzigiannakis, C. Tselios, and A. Antoniou, "On the deployment of healthcare applications over fog computing infrastructure," in *2017 IEEE 41st Annual Computer Software and Applications Conference (COMPSAC)*, vol. 2. IEEE, 2017, pp. 288–293.

[64] R. K. Pathinarupothi, M. V. Ramesh, and E. Rangan, "Multi-layer architectures for remote health monitoring," in *2016 IEEE 18th International Conference on e-Health Networking, Applications and Services (Healthcom)*. IEEE, 2016, pp. 1–6.

[65] S. Bhatt, F. Patwa, and R. Sandhu, "An access control framework for cloud-enabled wearable internet of things," in *2017 IEEE 3rd International Conference on Collaboration and Internet Computing (CIC)*. IEEE, 2017, pp. 328–338.

[66] H. Javdani and H. Kashanian, "Internet of things in medical applications with a service-oriented and security approach: a survey," *Health and Technology*, vol. 8, no. 1, pp. 39–50, 2018.

[67] S. Bhatt, F. Patwa, and R. Sandhu, "Access control model for aws internet of things," in *International Conference on Network and System Security*. Springer, 2017, pp. 721–736.

[68] P. Verma and S. K. Sood, "Cloud-centric iot based disease diagnosis healthcare framework," *Journal of Parallel and Distributed Computing*, vol. 116, pp. 27–38, 2018.

[69] K. N. Swaroop, K. Chandu, R. Gorrepotu, and S. Deb, "A health monitoring system for vital signs using iot," *Internet of Things*, vol. 5, pp. 116–129, 2019.

[70] M. K. Hassan, A. I. El Desouky, S. M. Elghamrawy, and A. M. Sarhan, "Intelligent hybrid remote patient-monitoring model with cloud-based framework for knowledge discovery," *Computers & Electrical Engineering*, vol. 70, pp. 1034–1048, 2018.

[71] D. Rahbari and M. Nickray, "Task offloading in mobile fog computing by classification and regression tree," *Peer-to-Peer Networking and Applications*, vol. 13, no. 1, pp. 104–122, 2020.

[72] P. M. Kumar, S. Lokesh, R. Varatharajan, G. C. Babu, and P. Parthasarathy, "Cloud and iot based disease prediction and diagnosis system for healthcare using fuzzy neural classifier," *Future Generation Computer Systems*, vol. 86, pp. 527–534, 2018.

[73] D. Castro, W. Coral, C. Rodriguez, J. Cabra, and J. Colorado, "Wearable-based human activity recognition using an iot approach," *Journal of Sensor and Actuator Networks*, vol. 6, no. 4, p. 28, 2017.

[74] K. N. Rajesh and R. Dhuli, "Classification of imbalanced ecg beats using re-sampling techniques and adaboost ensemble classifier," *Biomedical Signal Processing and Control*, vol. 41, pp. 242–254, 2018.

[75] R. Sundarasekar, M. Thanjaivadivel, G. Manogaran, P. M. Kumar, R. Varatharajan, N. Chilamkurti, and C.-H. Hsu, "Internet of things with maximal overlap discrete wavelet transform for remote health monitoring of abnormal ecg signals," *Journal of medical systems*, vol. 42, no. 11, pp. 1–13, 2018.

[76] D. Borthakur, H. Dubey, N. Constant, L. Mahler, and K. Mankodiya, "Smart fog: Fog computing framework for unsupervised clustering analytics in wearable internet of things," in *2017 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*. IEEE, 2017, pp. 472–476.

[77] A. M. Elmisery, S. Rho, and M. Aborizka, "A new computing environment for collective privacy protection from constrained healthcare devices to iot cloud services," *Cluster Computing*, vol. 22, no. 1, pp. 1611–1638, 2019.

[78] X. Jia, D. He, N. Kumar, and K.-K. R. Choo, "Authenticated key agreement scheme for fog-driven iot healthcare system," *Wireless Networks*, vol. 25, no. 8, pp. 4737–4750, 2019.

[79] Y. Liu, Y. Zhang, J. Ling, and Z. Liu, "Secure and fine-grained access control on e-healthcare records in mobile cloud computing," *Future Generation Computer Systems*, vol. 78, pp. 1020–1026, 2018.

[80] S. Sharma, K. Chen, and A. Sheth, "Toward practical privacy-preserving analytics for iot and cloud-based healthcare systems," *IEEE Internet Computing*, vol. 22, no. 2, pp. 42–51, 2018.

[81] O. Albahri, A. Albahri, K. Mohammed, A. Zaidan, B. Zaidan, M. Hashim, and O. H. Salman, "Systematic review of real-time remote health monitoring system in triage and priority-based sensor technology: Taxonomy, open challenges, motivation and recommendations," *Journal of medical systems*, vol. 42, no. 5, pp. 1–27, 2018.

[82] K. Mohammed, A. Zaidan, B. Zaidan, O. Albahri, M. Alsalem, A. Albahri, A. Hadi, and M. Hashim, "Real-time remote-health monitoring systems: a review on patients prioritisation for multiple-chronic diseases, taxonomy analysis, concerns and solution procedure," *Journal of medical systems*, vol. 43, no. 7, pp. 1–21, 2019.

[83] H. Mora, D. Gil, R. M. Terol, J. Azorín, and J. Szymanski, "An iot-based computational framework for healthcare monitoring in mobile environments," *Sensors*, vol. 17, no. 10, p. 2302, 2017.

[84] K. Jaiswal, S. Sobhanayak, A. K. Turuk, S. L. Bibhudatta, B. K. Mohanta, and D. Jena, "An iot-cloud based smart healthcare monitoring system using container based virtual environment in edge device," in *2018 International Conference on Emerging Trends and Innovations In Engineering And Technological Research (ICETIETR)*. IEEE, 2018, pp. 1–7.

[85] M. Al-Rubaie and J. M. Chang, "Privacy-preserving machine learning: Threats and solutions," *IEEE Security & Privacy*, vol. 17, no. 2, pp. 49–58, 2019.

[86] A. Hundepool, J. Domingo-Ferrer, L. Franconi, S. Giessing, E. S. Nordholt, K. Spicer, and P.-P. De Wolf, *Statistical disclosure control*. John Wiley & Sons, 2012.

[87] D. Dua and C. Graff, "Uci machine learning repository," 2017. [Online]. Available: http://archive.ics.uci.edu/ml

[88] Y. Zhao, Z. S.-Y. Wong, and K. L. Tsui, "A framework of rebalancing imbalanced healthcare data for rare events' classification: a case of look-alike sound-alike mix-up incident detection," *Journal of healthcare engineering*, vol. 2018, 2018.

[89] T. Farrand, F. Mireshghallah, S. Singh, and A. Trask, "Neither private nor fair: Impact of data imbalance on utility and fairness in differential privacy," in *Proceedings of the 2020 Workshop on Privacy-Preserving Machine Learning in Practice*, 2020, pp. 15–19.

[90] F. Prasser, "arx-deidentifier," *URL https : / / github . com / arx-deidentifier/arx*.

[91] D. Subramanian, "A simple introduction to k-nearest neighbors algorithm," *URL https://towardsdatascience.com/a-simple-introduction-to-k-nearest-neighbors-algorithm-b3519ed98e*.

[92] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg *et al.*, "Scikit-learn: Machine learning in python," *the Journal of machine Learning research*, vol. 12, pp. 2825–2830, 2011.

[93] A. Triastcyn and B. Faltings, "Bayesian differential privacy for machine learning," in *International Conference on Machine Learning*. PMLR, 2020, pp. 9583–9592.

[94] T. Zhang, T. Zhu, P. Xiong, H. Huo, Z. Tari, and W. Zhou, "Correlated differential privacy: feature selection in machine learning," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 3, pp. 2115–2124, 2019.

[95] A. Patil and S. Singh, "Differential private random forest," in *2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*. IEEE, 2014, pp. 2623–2630.

[96] A. Inan, M. Kantarcioglu, and E. Bertino, "Using anonymized data for classification," in *2009 IEEE 25th International Conference on Data Engineering*. IEEE, 2009, pp. 429–440.

Appendices

A Training & Testing Accuracy for Datasets

Below tables can be referred for detailed accuracy comparison of different datasets against privacy budgets.

Adult Database

Table 20: Comparison of Training Accuracies for Adult Database

Algorithms	ϵ=1	ϵ=2	ϵ=3	ϵ= ∞
Naive Bayes Gaussian	84.52	83.94	98.37	74.89
Logistic Regression	83.15	85.55	98.37	68.79
K Nearest	72.18	83.94	90.94	82.87
Support Vector	86.71	88.6	98.37	77.06
RandomForest	87.72	85.45	98.37	78.89
Decision Tree	77.05	74.14	94.8	75.01
XgBoost	84.86	89.85	98.37	82.64

Table 21: Comparison of Testing Accuracies for Adult Database

Algorithms	ϵ=1	ϵ=2	ϵ=3	$\epsilon = \infty$
Naive Bayes Gaussian	77	77.29	75.48	75.43
Logistic Regression	77.5	77.03	74.71	77.45
K Nearest	74.45	72.69	75.7	74.29
Support Vector	72.96	70.05	75.46	77.45
Random Forest	77.27	75.92	76.82	79.83
Decision Tree	77	63.45	72.13	79.12
XgBoost	77.66	74.77	75.48	82.15

Breast Cancer Database

Table 22: Comparison of Training Accuracies for Breast Cancer Database

Algorithms	ϵ=1	ϵ=2	ϵ=3	$\epsilon = \infty$
Naive Bayes Gaussian	95.38	94.87	95.43	96.61
Logistic Regression	95.56	95.14	96.4	96.72
K Nearest	97.68	96.9	97.95	97.29
Support Vector	95.73	96.76	96.75	96.72
Random Forest	96.97	96.48	96.99	97.4
Decision Tree	93.6	94.6	96.15	95.93
XgBoost	95.91	96.35	97	95.82

Table 23: Comparison of Testing Accuracies for Breast Cancer Database

Algorithms	ϵ=1	ϵ=2	ϵ=3	$\epsilon = \infty$
Naive Bayes Gaussian	96.58	96.58	96.58	96.58
Logistic Regression	97.07	98.04	98.53	98.04
K Nearest	98.53	99.62	99.51	98.04
Support Vector	97.56	98.04	97.56	97.56
Random Forest	98.04	98.04	98.04	98.04
Decision Tree	95.6	97.07	97.07	97.07
XgBoost	98.53	98.53	99.02	98.53

Contraceptive Method Database

Table 24: Comparison of Training Accuracies for Contraceptive Method Database

Algorithms	ϵ=1	ϵ=2	ϵ=3	$\epsilon = \infty$
Naive Bayes Gaussian	46.62	47.3	43.82	50.53
Logistic Regression	50.58	48.04	46.05	54.25
K Nearest	47.55	49.16	45.97	54.4
Support Vector	51.68	50.28	45.89	57.58
Random Forest	50.92	49.16	45.49	56.76
Decision Tree	44.35	45.49	44.62	49.84
XgBoost	51.26	49.41	45.49	58.87

Table 25: Comparison of Testing Accuracies for Contraceptive Method Database

Algorithms	ϵ=1	ϵ=2	ϵ=3	$\epsilon = \infty$
Naive Bayes Gaussian	47.52	49.29	45.58	48.58
Logistic Regression	45.4	45.75	42.22	53.88
K Nearest	51.94	56	45.22	56.36
Support Vector	48.76	55.83	42.75	55.12
Random Forest	48.76	49.64	40.63	51.76
Decision Tree	44.34	43.99	39.39	46.64
XgBoost	48.4	57.42	42.57	56.89

Mammography Mass Cancer Database

Table 26: Comparison of Training Accuracies for Mammography Database

Algorithms	ϵ=1	ϵ=2	ϵ=3	$\epsilon = \infty$
Naive Bayes Gaussian	79.58	79.31	79.14	79.97
Logistic Regression	80.14	79.6	79.77	80.7
K Nearest	77.36	77.31	79.52	78.88
Support Vector	80.14	80.88	81.54	79.85
Random Forest	80.89	81.02	79.39	79.37
Decision Tree	78.48	76.6	79.27	77.8
XgBoost	79.96	81.03	80.78	79.49

Table 27: Comparison of Testing Accuracies for Mammography Database

Algorithms	ϵ=1	ϵ=2	ϵ=3	$\epsilon = \infty$
Naive Bayes Gaussian	80.32	79.91	77.51	81.12
Logistic Regression	81.12	80.32	80.32	81.92
K Nearest	76.3	80.72	79.91	81.52
Support Vector	77.91	81.92	81.52	81.52
Random Forest	78.71	79.91	79.51	78.31
Decision Tree	77.1	78.71	76.7	79.11
XgBoost	81.92	81.92	80.72	79.91

Car Evaluation Database

Table 28: Comparison of Training Accuracies for Car Evaluation Database

Algorithms	ϵ=1	ϵ=2	ϵ=3	$\epsilon = \infty$
Naive Bayes Gaussian	69.59	67.11	68.77	69.03
Logistic Regression	83.66	84.3	84.34	86.05
K Nearest	89.91	93.77	93.68	96.86
Support Vector	96.19	98.2	98.49	98.84
Random Forest	87.15	87.35	88.57	87.41
Decision Tree	73.06	72.25	73.79	61.65
XgBoost	95.32	98.29	98.19	98.87

Table 29: Comparison of Testing Accuracies for Car Evaluation Database

Algorithms	ϵ=1	ϵ=2	ϵ=3	$\epsilon = \infty$
Naive Bayes Gaussian	70.57	70.77	71.26	70.84
Logistic Regression	85.73	85.38	84.76	86.21
K Nearest	91.24	97.72	98.07	98.13
Support Vector	96.07	98.2	98.34	99.24
Random Forest	88.69	88.69	89.45	90.48
Decision Tree	64.71	70.02	75.6	69.33
XgBoost	96.2	98	98.58	99.44

B UI Implementation of Suggested Framework

Figure 10: Dev Initial Phase

Figure 11: Dev Phase

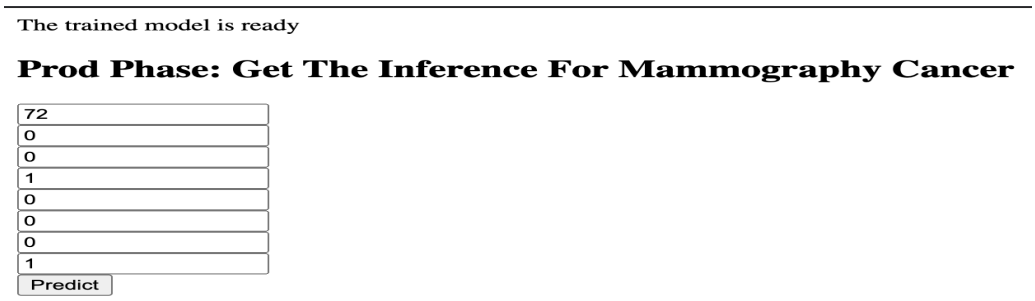

Figure 12: Prod Phase

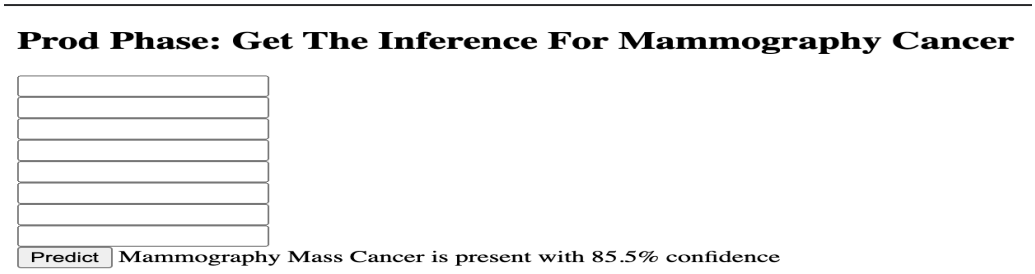

Figure 13: Prod Inference

C Sample Original and Anonymized Database

age	workclass	education	marital-status	occupation	relationship	race	sex	native-country	class
50	Self-emp-not-inc	Bachelors	Married-civ-spouse	Exec-managerial	Husband	White	Male	United-States	<=50K
38	Private	HS-grad	Divorced	Handlers-cleaners	Not-in-family	White	Male	United-States	<=50K
53	Private	11th	Married-civ-spouse	Handlers-cleaners	Husband	Black	Male	United-States	<=50K
28	Private	Bachelors	Married-civ-spouse	Prof-specialty	Wife	Black	Female	Cuba	<=50K
37	Private	Masters	Married-civ-spouse	Exec-managerial	Wife	White	Female	United-States	<=50K
49	Private	9th	Married-spouse-absent	Other-service	Not-in-family	Black	Female	Jamaica	<=50K
52	Self-emp-not-inc	HS-grad	Married-civ-spouse	Exec-managerial	Husband	White	Male	United-States	>50K
31	Private	Masters	Never-married	Prof-specialty	Not-in-family	White	Female	United-States	>50K
42	Private	Bachelors	Married-civ-spouse	Exec-managerial	Husband	White	Male	United-States	>50K
37	Private	Some-college	Married-civ-spouse	Exec-managerial	Husband	Black	Male	United-States	>50K
30	State-gov	Bachelors	Married-civ-spouse	Prof-specialty	Husband	Asian-Pac-Islander	Male	India	>50K
23	Private	Bachelors	Never-married	Adm-clerical	Own-child	White	Female	United-States	<=50K
32	Private	Assoc-acdm	Never-married	Sales	Not-in-family	Black	Male	United-States	<=50K
34	Private	7th-8th	Married-civ-spouse	Transport-moving	Husband	Amer-Indian-Eskimo	Male	Mexico	<=50K
25	Self-emp-not-inc	HS-grad	Never-married	Farming-fishing	Own-child	White	Male	United-States	<=50K
32	Private	HS-grad	Never-married	Machine-op-inspct	Unmarried	White	Male	United-States	<=50K
38	Private	11th	Married-civ-spouse	Sales	Husband	White	Male	United-States	<=50K
43	Self-emp-not-inc	Masters	Divorced	Exec-managerial	Unmarried	White	Female	United-States	>50K
40	Private	Doctorate	Married-civ-spouse	Prof-specialty	Husband	White	Male	United-States	>50K
54	Private	HS-grad	Separated	Other-service	Unmarried	Black	Female	United-States	<=50K
35	Federal-gov	9th	Married-civ-spouse	Farming-fishing	Husband	Black	Male	United-States	<=50K
43	Private	11th	Married-civ-spouse	Transport-moving	Husband	White	Male	United-States	<=50K
59	Private	HS-grad	Divorced	Tech-support	Unmarried	White	Female	United-States	<=50K
56	Local-gov	Bachelors	Married-civ-spouse	Tech-support	Husband	White	Male	United-States	>50K

Figure 14: Adult Original Database

age	workclass	education	marital-status	occupation	race	sex	native-country	salary-class
[21, 30]	Government	Higher education	Spouse not present	Other	*	*	North America	<=50K
[21, 30]	Government	Higher education	Spouse not present	Other	*	*	North America	<=50K
[21, 30]	Government	Higher education	Spouse not present	Other	*	*	North America	<=50K
[21, 30]	Government	Higher education	Spouse not present	Other	*	*	North America	<=50K
[21, 30]	Government	Higher education	Spouse not present	Other	*	*	North America	<=50K
[21, 30]	Government	Higher education	Spouse not present	Other	*	*	North America	<=50K
[21, 30]	Government	Higher education	Spouse not present	Other	*	*	North America	<=50K
[21, 30]	Government	Higher education	Spouse not present	Other	*	*	North America	<=50K
[21, 30]	Government	Higher education	Spouse not present	Other	*	*	North America	<=50K
[21, 30]	Government	Higher education	Spouse not present	Other	*	*	North America	<=50K
[21, 30]	Government	Higher education	Spouse not present	Other	*	*	North America	<=50K
[21, 30]	Government	Higher education	Spouse not present	Other	*	*	North America	<=50K
[21, 30]	Government	Higher education	Spouse not present	Other	*	*	North America	<=50K
[21, 30]	Government	Higher education	Spouse not present	Other	*	*	North America	<=50K
[21, 30]	Government	Higher education	Spouse not present	Other	*	*	North America	<=50K
[21, 30]	Government	Higher education	Spouse not present	Other	*	*	North America	<=50K
[21, 30]	Government	Higher education	Spouse not present	Other	*	*	North America	<=50K
[21, 30]	Government	Higher education	Spouse not present	Other	*	*	North America	<=50K
[21, 30]	Government	Higher education	Spouse not present	Other	*	*	North America	<=50K
[21, 30]	Government	Higher education	Spouse not present	Other	*	*	North America	<=50K
[21, 30]	Government	Higher education	Spouse not present	Other	*	*	North America	<=50K
[21, 30]	Government	Higher education	Spouse not present	Other	*	*	North America	<=50K
[21, 30]	Government	Higher education	Spouse not present	Other	*	*	North America	<=50K
[21, 30]	Government	Higher education	Spouse not present	Other	*	*	North America	<=50K

Figure 15: Adult Anonymized Database with ϵ =1