UNIVERSITY OF CALIFORNIA SAN DIEGO

Strengthening Health Research Workflow for Research-Oriented Sharing, Predictive Modeling, and Cross-Institutional Collaboration

A Dissertation submitted in partial satisfaction of the requirements
for the degree Doctor of Philosophy

in

Bioinformatics and Systems Biology with a Specialization in Biomedical Informatics

by

Anh Pham

Committee in charge:

Professor Tsung-Ting Kuo, Chair
Professor Lucila Ohno-Machado, Co-Chair
Professor Ery Arias-Castro
Professor Robert El-Kareh
Professor Rodney Gabriel

2024

The Dissertation of Anh Pham is approved, and it is acceptable in quality and form for publication on microfilm and electronically.

University of California San Diego

2024

DEDICATION

Cháu xin cảm ơn đại gia đình họ Phạm và họ Hoàng đã giúp đỡ gia đình cháu trong những ngày khó khăn. Cháu có được ngày hôm nay là do công sức của bố mẹ cháu và gia đình hai bên.

Em kính gửi lời cảm tạ công ơn giáo dục của các thầy cô giáo của em ở Việt Nam.

I would also like to dedicate this thesis to my advisor Dr. Tsung-Ting Kuo, an exceptional mentor without whom I cannot get to the end of this PhD journey.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I would like to acknowledge Professor Tsung-Ting Kuo for his unwavering support as the chair of my committee. There is no word to express my gratitude for his guidance. I am grateful for the support from BISB/DBMI faculty, staff, and fellow students since the day of my admission, especially from Dr. Ohno-Machado, Dr. Sitapati, among many others. I am also grateful for Dr. Karen Stocks and Mr. Steve Diggs at Scripps Institution of Oceanography, who first introduced me to research. I appreciate those who have taken their time to endorse a reference on my behalf. With regards to non-academic support, I am thankful to receive the help from Ms. Alicia Magallanes (Basic Needs Hub), Dr. Walters and Dr. Behymer (Student Health Services), Dr. Gross (UCSD Health), and more.

Chapter 3, in part, has been submitted for publication of the material as it may appear in Computers in Biology and Medicine, 2024. Pham, Anh;* Edelson, Maxim;* Nouri, Armin; Kuo, Tsung-Ting, Science Direct, 2024. The dissertation author was one of the primary researchers and co-first authors of this paper. (* These authors contributed equally to this work)

Chapter 4, in full, is a reprint of the material as it appears in Journal of the American Medical Informatics Association, 2023. Kuo, Tsung-Ting;* Pham, Anh;* Edelson, Maxim;* Kim, Jihoon; Chan, Jason; Gupta, Yash, Ohno-Machado, Lucila; The R2D2 Consortium, Oxford University Press, 2023. The dissertation author was one of the primary investigators and co-first authors of this paper. (* These authors contributed equally to this work)

Chapter 5, in part, has been submitted for publication of the material as it may appear in Heliyon, 2024. Pham, Anh; El-Kareh, Robert; Myers, Frank; Ohno-Machado, Lucila; Kuo, Tsung-Ting, Elsevier, 2024. The dissertation author was the primary researcher and author of this paper.

Chapter 6, in part, is currently being prepared for submission for publication of the material. Yu, Yufei;* Edelson, Maxim;* Pham, Anh;* Pekar, Jonathan; Johnson, Brian; Post, Kai; Kuo, Tsung-Ting. The dissertation author was one of the primary investigators and co-first authors of this paper. (* These authors contributed equally to this work)

Chapter 7, in part, has been submitted for publication of the material as it may appear in Informatics in Medicine Unlocked, 2024. Shao, Xiyan;* Pham, Anh;* Kuo, Tsung-Ting, Elsevier, 2024. The dissertation author was one of the primary researchers and co-first authors of this paper. (* These authors contributed equally to this work)

VITA

2017 Bachelor of Science in Joint Mathematics-Economics, University of California San Diego

2020 Master of Science in Computer Science, University of California San Diego

2024 Doctor of Philosophy in Bioinformatics and Systems Biology with a Specialization in Biomedical Informatics, University of California San Diego

PUBLICATIONS

Kuo, T-T.,* Pham, A.,* Edelson,* M., Kim, J., Chan, J., Gupta, Y., Ohno-Machado, L., R2D2 Consortium. "Blockchain-enabled immutable, distributed, and highly available clinical research activity logging system for federated COVID-19 data analysis from multiple institutions." Journal of American Medical Informatics Association, 2023. (* contributed equally)

Kuo, T-T., Pham, A. "Quorum-based model learning on a blockchain hierarchical clinical research network using smart contracts." International Journal of Medical Informatics, 2023.

Pham, A., Minihan, C., Augustine, A., Kuo, T-T. "Corda Blockchain for Interoperable Insurance Billing." AMIA Informatics Summit, 2023.

Li, MM., Pham, A., Kuo, T-T. "Predicting COVID-19 county-level case number trend by combining demographic characteristics and social distancing policies." JAMIA Open, 2022.

Kuo, T-T., Pham, A. "Detecting Model Misconducts in Decentralized Healthcare Federated Learning." International Journal of Medical Informatics, 2022.

Pham, A., El-Kareh R., Ohno-Machado, L., Kuo, T-T. "Early Prediction of Positive Clostridioides Difficile Test Results." AMIA Annual Symposium, 2021.

Pham, A., Samragh, M., Wagh, S., & Wenger, E. "Private Movie Recommendations for Children." in Protecting privacy through homomorphic encryption, Springer Nature, 2021.

Kim, J., Kim, H., Bell, E., Bath, T., Paul, P., Pham, A., Jiang, X., Zheng, K., & Ohno-Machado, L. "Patient perspectives about decisions to share medical data and biospecimens for research." JAMA Network Open, 2(8), 2019.

Stocks, K., Diggs, S., Olson, C., Pham, A., Arko, R., Kinkade D., Shepherd A. "SeaView: Bringing together an ocean of data." Oceanography 31(1):71, 2018.

FIELD OF STUDY

Major Field: Bioinformatics and System Biology
Studies in Biomedical Informatics
Professor Tsung-Ting Kuo

ABSTRACT OF THE DISSERTATION

Strengthening Health Research Workflow for Research-Oriented Sharing, Predictive Modeling, and Cross-Institutional Collaboration

by

Anh Pham

Doctor of Philosophy in Bioinformatics and Systems Biology
with a Specialization in Biomedical Informatics

University of California San Diego, 2024

Professor Tsung-Ting Kuo, Chair
Professor Lucila Ohno-Machado, Co-Chair

Machine learning and artificial intelligence (AI) hold the promise to innovate clinical practices and to improve quality of care. Naturally, the health research workflow may require modern adaptations to best capitalize the fast growth of AI. One challenge along the pipeline from health data to AI products is the balance between privacy-focused patients and access-focused

researchers. It is thus crucial for scientists to overcome sharing barriers of health-research activities without doing harm to data security and privacy, so as to enable the meaningful use of electronic health records (EHR) in research. In particular, the sharing of research-oriented activities to benefit both patients and researchers can be facilitated through establishing a secured, blockchain-based informed consent infrastructure with which patients can grant data access to researchers. Other processes such as the management of clinical research activities in data consortia may also take advantage of such platforms. Following this feasibility of a robust, prolific data pipeline, the next step in health AI is the use of EHR for predictive analysis. Specifically, the COVID-19 pandemic has cast light on the critical value of predictive modeling in the fight against infectious diseases. A meaningful demonstration of EHR use in AI may include the construction of models to predict the risk of *Clostridioides difficile* infection. Last but not least, the efficiency and usability of cross-institutional research collaboration may also stand to benefit from workflow improvement, as it may enable better use of data from multiple sources. For instance, the credential-verification procedure to which researchers must abide to access external datasets may enhance efficiency through secured automation. Similarly, the use of privacy-preserving algorithms can be promoted by providing non-technical users with intuitive tools to participate in federated learning schemas. Together, it is seen that the health research workflow can be bolstered through fortifying the sharing framework of research-oriented activities, meaningfully utilizing EHR in predictive analysis, and improving the efficiency and usability of cross-institutional research. Such developments may boost scientific growth by keeping the right equilibrium among data quantity, data privacy, and research efficiency and usability, permitting the simultaneous expansion of patients' autonomy and researchers' innovation.

Chapter 1 THREE AIMS TO STRENGTHEN HEALTH RESEARCH WORKFLOW

1.1 Introduction

Health-related policy and regulations have actively promoted the meaningful use of electronic health records (EHR).[1] This push towards digitalization of health data can lead to a surge in the quantity of data available for research purposes.[2,3] In the field of biomedical informatics, researchers are looking to use this data source to innovate clinical practices and to improve quality of care.[4-7] Among such data-driven innovations, Artificial Intelligence (AI)/machine learning (ML) tools to inform care decisions (clinical decision support – CDS) can be valuable.[6,8-10] In particular, ML can further suggest actionable clinical factors, such as projecting patients' health risks of certain diseases or conditions based on analysis of past data.[6,8-10] As ML applications proliferate, it is expected that the AI health market will save the U.S. healthcare economy $150 billion in estimated annual savings by 2026.[11] European healthcare can also expect significant savings with AI.[12] Naturally, the health research workflow may require innovations to best capitalize along this fast growth of AI as it accelerates meaningful health data uses.

Throughout the pipeline from health data to AI products, there are nonetheless challenges for both *privacy-focused* patients and *access-focused* researchers. Although patients may wish to help advance clinical care, privacy and security breaches can dampen patient support towards data-sharing for research purposes.[13] Meanwhile, researchers want an efficient and seamless pipeline from data acquisition, data preparation, data distribution, to the construction of predictive models either within or across institutions.[14] This desire is often tested by multiple obstacles; for example, delayed communication by third-party data concierge service and/or lack of sufficient sharing channels among health institutions. The 2018 Report to Congress by the Dept. of Health & Human Services (HHS) confirms that restricted data access would prevent physicians and patients from

enjoying the benefit of ML and AI's computing power,[15] while the 2019 Future Health Index report highlights the need to **overcome sharing barriers of health research-oriented activities** without doing harm to data security and privacy.[16]

Granted the feasibility of a robust, prolific data pipeline, the next step in health AI is **the use of such data for predictive analysis**. Particularly, recent unprecedented developments in the COVID-19 pandemic have cast light on the critical value of predictive modeling in the fight against infectious diseases; for instance, it can affect how authorities decide on mitigation method and how much precaution should be applied at either a social or individual level,[17,18] a lesson applicable to other major infections that are similarly challenging to contain after outbreaks. The effective use of real-life health data for risk prediction is an important tool to keep infections under control.[19]

Along this pipeline from health data acquisition to data products, the **efficiency and usability of cross-institutional research collaboration** may be another topic of interest. Take for example the credential-verification procedure to which researchers must abide in order to gain access to external datasets. To show observance of data-protection requirements, a researcher would have to manually send their compliance certificates to some authority that oversees the access request. Upon the manual review of such certificates, they may then receive a Data Use Agreement (DUA) to sign and return before getting approved for data release, all through manual means. Should they need to access another dataset with the exact same data-handling requirements, they would still need to go through the same manual steps. This non-automatic channel may be a significant source of delay that can stretch to days, even months due to unresponsive parties. Naturally, a new way to automate the data-requesting process may help increase overall research productivity. Another process that may benefit from workflow improvement is federated learning (FL), which is a privacy-preserving schema where multiple clients share the task of model

computation without direct exchange of raw data, and is a great adaptation technique as data sources get more diverse and privacy protection become more complicated.[20] It allows for the dissemination of partially trained predictive models instead of observation-level patient data, and can reduce risks to patient privacy. However, from the viewpoint of practical implementation, these systems may require complicated setup steps involving specialized languages, knowledge of tech-heavy fields, and interactions with non-graphical, non-visual tools. The burden that comes with their current configurations may discourage researchers from employing algorithmic designs already proven to comply with stringent privacy standards. It is seen that efforts are needed to enhance the *usability* of FL systems to enable wider adaptation of developed methods, which in turn directly contributes to the growth of cross-institutional collaborative ML.

Efforts to boost the health research workflow are thus of critical value to science, whether in constructing a robust infrastructural framework to support the sharing of research-oriented activities, in the meaningful use of EHR for predictive modeling, or in facilitating cross-institutional research collaboration. It may boost scientific growth by keeping the right equilibrium among data quantity, data privacy, and research efficiency and usability, permitting the simultaneous expansion of patients' autonomy and researchers' innovation.

1.2 Hypothesis and Specific Aims

Hypothesis: The health research workflow can be bolstered through fortifying the sharing framework of research-oriented activities, efficiently utilizing patient EHR in predictive analysis, and improving the efficiency and usability of cross-institutional research collaboration.

Aim 1. Support the sharing of research-oriented activities to benefit both patients and researchers (**Figure 1.1.A**). In particular, the informed consent through which patients grant data access to researchers may improve with a sharing platform that can enhance patient trust and researcher productivity (Chapter 3). Similarly, other health research processes such as the management of clinical research activities in data consortia may take advantage of such platforms (Chapter 4).

Aim 2. Examine the use of EHR data in predictive data analysis to help combat impacts of major infectious disease (**Figure 1.1.B**). Specifically, EHR may be used to build models to predict the risk of *Clostridioides difficile* infection (Chapter 5).

Aim 3. Facilitate the efficiency and usability of cross-institutional research collaboration (**Figure 1.1.C**) either through accelerating researchers' access to cross-institutional datasets (Chapter 6), or through providing non-technical users with intuitive tools to participate in privacy-preserving federated learning systems (Chapter 7).

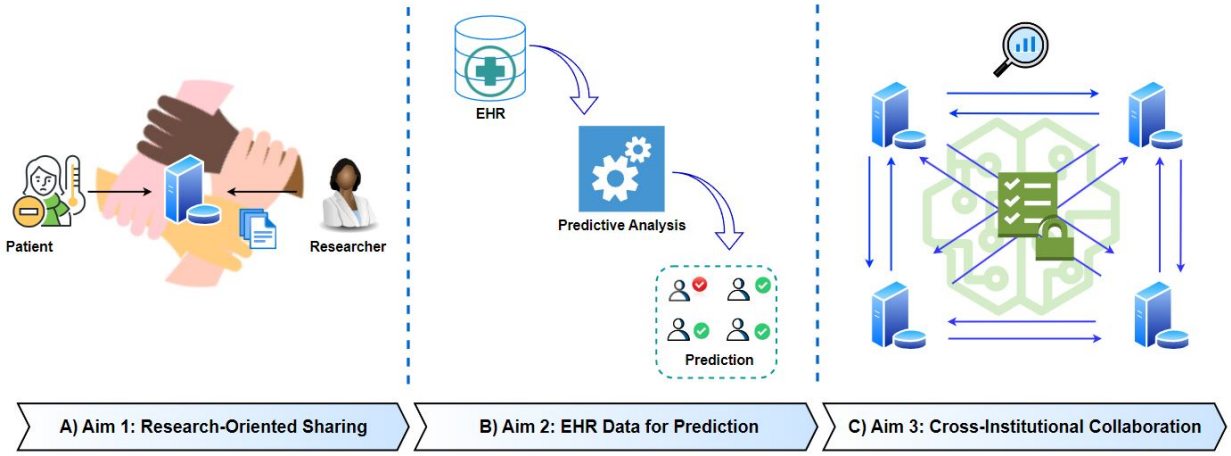


Figure 1.1: Three aims to strengthen the health research workflow.

Chapter 2 BLOCKCHAIN TECHNOLOGY TO SUPPORT HEALTH RESEARCH

2.1 How Blockchain Technology Can Bolster the Research Pipeline

In the era of modern computation, biomedical research starts with sharing due to the pragmatically data-intensive nature of big data. While providers, patients and policymakers agree that better access to health records will improve care and spur innovations,[15,16] sharing across multiple parties remains a challenge, especially when privacy concerns are in question.[21] This is a critical problem given the compulsory need to increase study cohort size, bolster research statistical power, and/or increase generalizability.

An intuitive remedy to this problem involves depositing data collected from different institutions in a single, central repository – the centralized model (**Figure 2.1.A.1**). In this conventional framework, all resources and administrative powers of the network reside with the central node, and the administrators of the central repository have the ultimate privilege to alter stored data and/or hide the trail of such actions. Existing real-life events have proven how data manipulation can occur in the shadow due to either disgruntled staff[22,23] or exploitation of internal system vulnerability.[24,25] Moreover, the central repository is a potential *Single-Point-of-Failure* (*SPoF*), which is an inherent drawback of centralized databases.[26] *SPoF* happens when data becomes inaccessible as this central server fails to function for any reasons (**Figure 2.1.A.2**), whether hostile takeovers or routine maintenance, causing damage risks even to systems with redundancy in place.[27,28]

A *decentralized* approach may be better suited to solve this challenge of privacy-versus-access in health research. **Blockchain technology**[29] is a decentralized, distributed ledger technology (DLT) that originated from the financial technology field and is now being proposed for various healthcare applications.[30-33] The DLT composition of blockchain means each

participating node would maintain a copy of all transactions within the network, eliminating the possibility of ambiguity should some bad-faith agent choose to manipulate their internal data.[34] From the perspective of patients, the *immutability* of blockchain can be an appealing feature; once information is broadcast to the network in the form of a "block," it is practically impossible to falsify records.[35,36] The data access history is also preserved with an *transparent* audit trail and *tamper-proof timestamp* mechanism,[37,38] offering *resistance against unauthorized access* and acting to *mitigate future disputes*. Therefore, blockchain may solve the pragmatic issue of disrupted workflow due to the *Single-Point-of-Failure* (*SPoF*) risk. As a decentralized network, blockchain avoids *SPoF* threats and is a *highly available* system since (i) no central authority is needed to facilitate data provenance, (ii) network communication is peer-to-peer, and (iii) each participating node always has the most up-to-date copy of records (**Figure 2.1.B.1**).[39,40] These built-in features allow a blockchain network to remain in operation when some individual node is taken offline (**Figure 2.1.B.2**). Furthermore, blockchain systems capable of supporting "smart contracts,"[41] or programming languages to execute on-chain logics, can immutably automate activities which may otherwise require manual efforts, and thus potentially cut down unnecessary delay, prevent potential manipulations, and *increase productivity*.

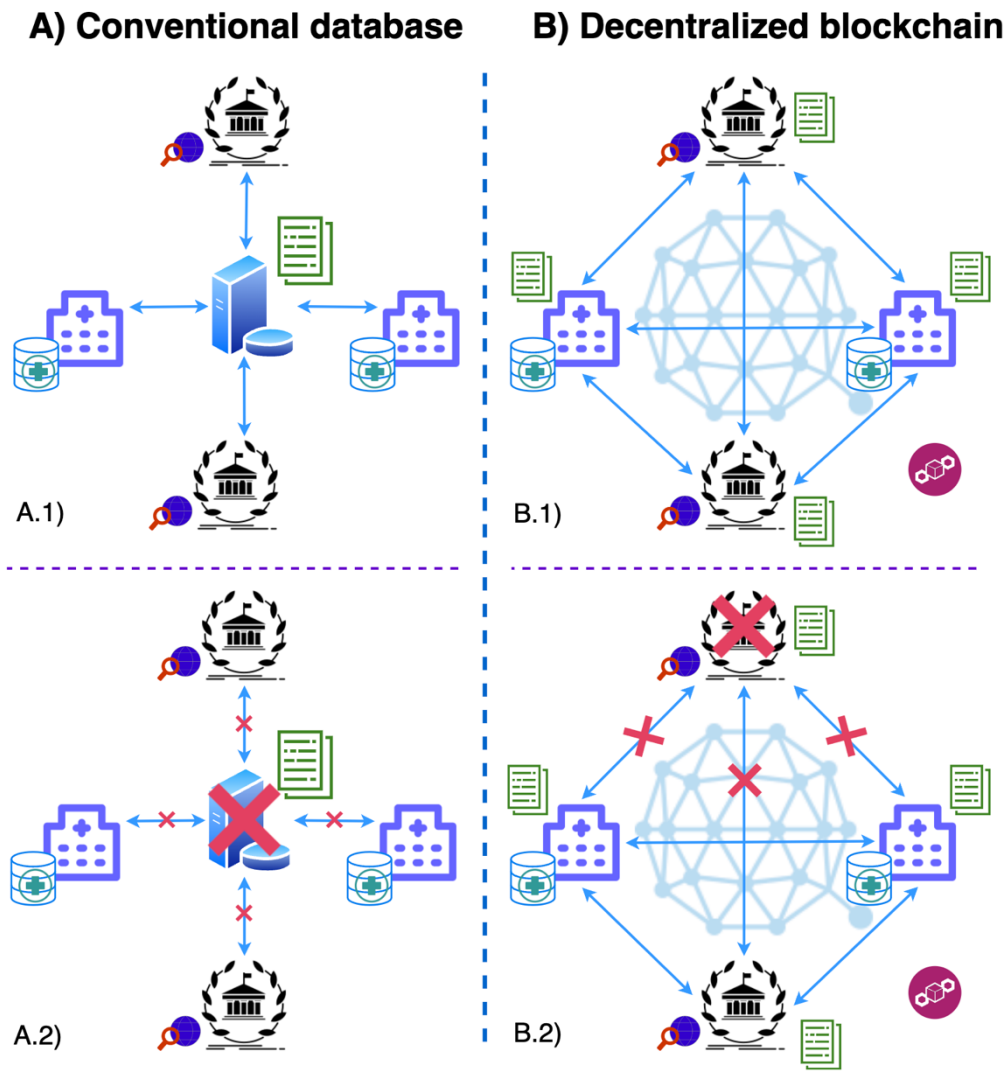


Figure 2.1: The impact of the Single-Point-of-Failure (*SPoF*) threat.

A) **Conventional database**: **A.1)** Network members connect to the central server, depositing to and receiving data solely from this central server; **A.2)** When the central server is taken down due to either scheduled maintenance or hostile activities, the whole network cannot function.

B) **Decentralized blockchain**: **B.1)** Network members communicate pair-wise, and each has a copy of the most up-to-dated data. **B.2)** If a member is taken offline, the rest of the network can still function.

Chapter 3 INFORMED CONSENT PROTOCOL

3.1 Abstract

Background. The consent protocol is now a critical part in the overall orchestration of clinical research. We aimed to demonstrate the feasibility of an Ethereum-based informed consent system, which includes an immutable and automated channel of consent matching, to simultaneously assure patient privacy and increase the efficiency of researchers' data access.

Method. We simulated a multi-site scenario, each assigned 10000 consent records. A consent record contained one patient's data-sharing preference with regards to seven data categories. We developed a blockchain-based infrastructure with a smart contract to record consents on-chain, and to query consenting patients corresponding to specific criteria. We measured our system's recording efficiency against a baseline design and verified accuracy by testing an exhaustive list of possible queries.

Results. Our method achieved ~3-4% lead with an average insertion speed of ~2 seconds per record per node on either a 3-, 4- or 5-node network, and 100% accuracy. It also outperformed other solutions in external validation.

Discussion. The speed we achieved is reasonable in a real-world system under the realistic assumption that patients may not change their minds too frequently, with the added benefit of immutability. Furthermore, the per-insertion time did improve slightly as the number of network nodes increased, attesting to the benefit of node parallelism as it suggests no attrition of insertion efficiency due to scale of nodes.

Conclusions. Our work confirms the technical feasibility of a blockchain-based consent mechanism, assuring patients with an immutable audit trail, and providing researchers with an efficient way to reach their cohorts.

3.2 Background and Motivation

3.2.1 The Need for Electronic Tiered/Dynamic Consent

Biomedical data, especially patient data collected in the course of care, are important resources that drive health sciences forwards [42]. With the rise of electronic health records (EHR) [43], more patient data can be made available for research purposes. Such secondary use of clinical data poses questions concerning transparency and security [44], and has led to the need of patients' informed consent, which may include the process for the patient to grant or deny a party the access to their protected health information (PHI). For example, the U.S National Institutes of Health (NIH) has mandated the implementation of informed consents even in the use of de-identified cell lines and biospecimens, which traditionally has not been the case [45]. It is thus both legally and ethically restricted to embark on the clinical research without first obtaining informed consents, and the consent protocol is now a critical part in the overall orchestration of health research. Throughout the process of PHI data access, there are nonetheless challenges for both privacy-focused patients and access-focused researchers. Although patients may wish to help the advance of clinical care, the presence of privacy breaches could reduce their enthusiasm [13]. Specifically, there could be reservations regarding the sharing of sensitive health information, such as one's mental status [46] or past difficult health events [47]. Tiered consents have thus been suggested, that is, instead of patients having only one option to either share all (or decline all) of their health history, they can select specific data categories to share [48]. In addition, the motivation behind a specific study may affect patients' sharing decision. For instance, patients may feel more inclined to share their health data with non-profit works over commercial ones, requiring another sharing choice that should be customizable [48].

3.2.2 Current Challenges in Electronic Consent Platforms

When implementing such a tiered consent mechanism, one important consideration is that patients are entitled to changing their minds at any time. This requires instant and continuous access to the consent system, as well as the assurance that such changes are recorded in real time. This points to dynamic consent, or consents embedded on an electronic-based platform so that the practice of storing, retrieving and amending personal consents can be executed conveniently at will [49]. In fact, dynamic consents are seen as potential tools to facilitate and encourage patient engagement in research activities over time [50].

While convenient, embedding the informed consent protocol on a digital platform leads to other challenges. The conventional model of centralized databases carries innate risks [51-53], with recent high-profile cybersecurity incidents highlighting issues of unauthorized administrative access and inherent system vulnerabilities [54-57], consequently causing distrust among patients who might then hesitate to use the digital health tools due to the possibility of disputes **(Figure 3.1.A)** [58]. In reality, healthcare is among the industries most affected by privacy and security breaches, with incidents caused by either external attackers or internal staff being the main offenders [57]. This in effect creates a dual demand from patients for both easy modification and rigorous protection: consents should be modifiable, but by no authority other than the patients/their legal guardians themselves. This is even more critical in the case of vulnerable minorities whose harms due to unwarranted data sharing have been documented [59,60], even litigated [61].

Researchers, on the other hand, desire undisrupted access to patient data. The current procedure to extract patient data involves third parties (such as data concierge services) that may slow down research, creating unnecessary obstacles in the search for new, meaningful medical understandings (**Figure 3.1.B**). In addition to such workflow bottlenecks, researchers can also lose

productivity due to technical failures of the centralized database system. By design, a centralized database brings with it the security risk of single-point-of-failure (SPoF); a SPoF is when data becomes inaccessible as the central server fails to function for any number of reasons, whether due to hostile takeovers or routine maintenance [53,62], causing damage risks even for systems with redundancy in place [27,28]. These workflow disruptions caused by unavailable databases and/or unresponsive "middle-man" data agents can, unfortunately, delay or even derail research progress.

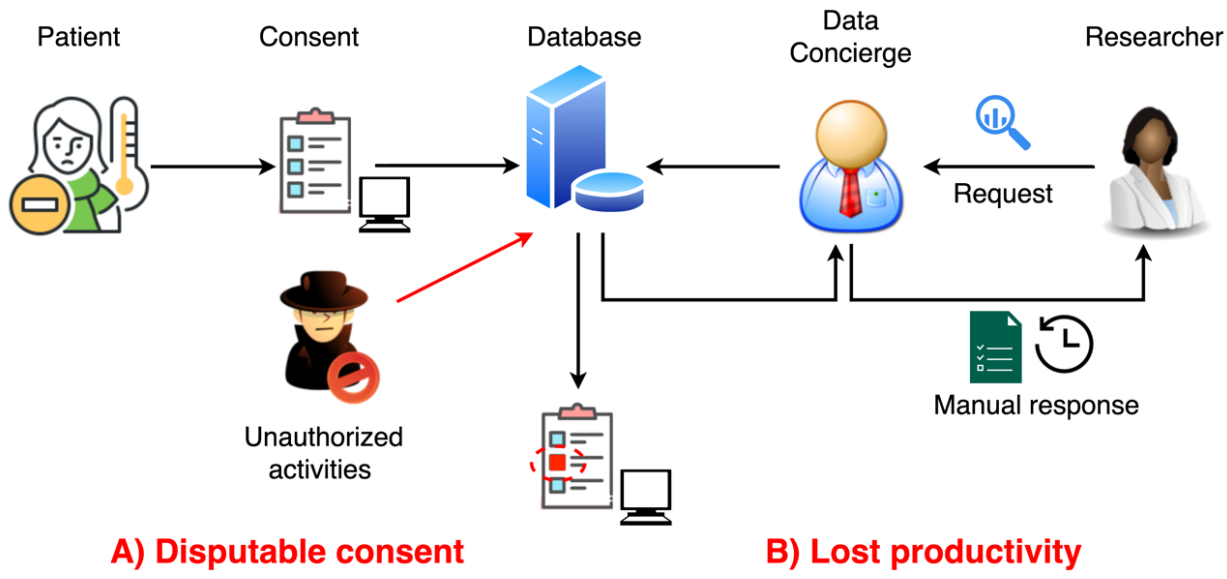


Figure 3.1: Challenges with current consent process.

A) **Disputable consent**: Patients may have doubts about the integrity of their stored consent records and how to handle disputes, given infamous hacking events in the healthcare industry.

B) **Lost productivity**: Researchers may have to wait for delayed responses from third-party data concierge services before getting information about their study cohorts.

3.2.3 Blockchain Technology for Informed Consent

To address these shortcomings, a decentralized system with native security features may be useful [39]. Blockchain is an emerging decentralized technology that has the potential to transform data management [32]. While it was originally envisioned for financial purposes [29], studies have shown the meaningful values of blockchain in healthcare [30,31], which suggest its potential use in the consent protocol. In particular, the blockchain network Ethereum [63] has been widely advocated to serve as the underlying infrastructure for data exchange [64-66], because of its ability to offer private blockchains that explicitly require permission to join [65], thereby reducing the chance of unauthorized engagement, as well as the beneficial fact that the Ethereum network itself is popular, well-tested, and actively maintained by the developer community. From the perspective of patients, the immutability of blockchain data provenance can be an appealing reason to entrust their consents; once information is broadcast to the network in the form of a "block," it is practically impossible to falsify records [39]. The data access history (instead of the sensitive data themselves, to protect patients' privacy) is also honored with an immutable audit trail and tamper-proof timestamp mechanism [31,37,53], offering resistance against unauthorized access and acting to mitigate future disputes. Coupled this built-in technical characteristic of blockchain with the fact that it is an electronic-based system, patients may find a solution to their dual interests in amendable and indisputable consents.

For researchers, blockchain can help prevent productivity loss due to server unavailability as caused by SPoF. As a decentralized network, blockchain avoids SPoF threats and is a highly available system since i) no central authority is needed to facilitate data provenance, ii) network communication is peer-to-peer, and iii) each participating node always has the most up-to-date copy of records [29,67]. In contrast to the centralized scenario, when the whole network ceases to

function should there be failure at the central node, with blockchain, the rest of the network can still operate when a node is taken offline (more details in **Figure 2.1**).

Furthermore, blockchain systems capable of supporting "smart contracts," or programming languages to execute on-chain logics (e.g., data management) [65], can immutably automate data storage and retrieval which may otherwise require intermediate manual efforts, and thus potentially cut down unnecessary delay, prevent potential manipulations, and increase productivity. In short, blockchain can serve as a new form of immutable and automated database infrastructure for the consent process: patients can be assured that their given consents cannot be modified by unauthorized parties (**Figure 3.2.A**), and researchers can access data with ease when smart contracts are embedded to verifiably execute consent retrieval from granted records (**Figure 3.2.B**).

Investigating the adoption of blockchain technology in the informed consent protocol is thus of significant intellectual merit, and of high technological and social impact. For patients, enhanced security may ease privacy concerns and boost overall data-sharing rates among populations with sharing inhibitions due to sensitive and/or vulnerable status. Physicians and researchers, on the other hand, are allowed to pursue scientific advances with confidence in the immutability of their cohort-matching process, making data-backed healthcare research more efficient and with verifiable, enforceable respect for patient privacy.

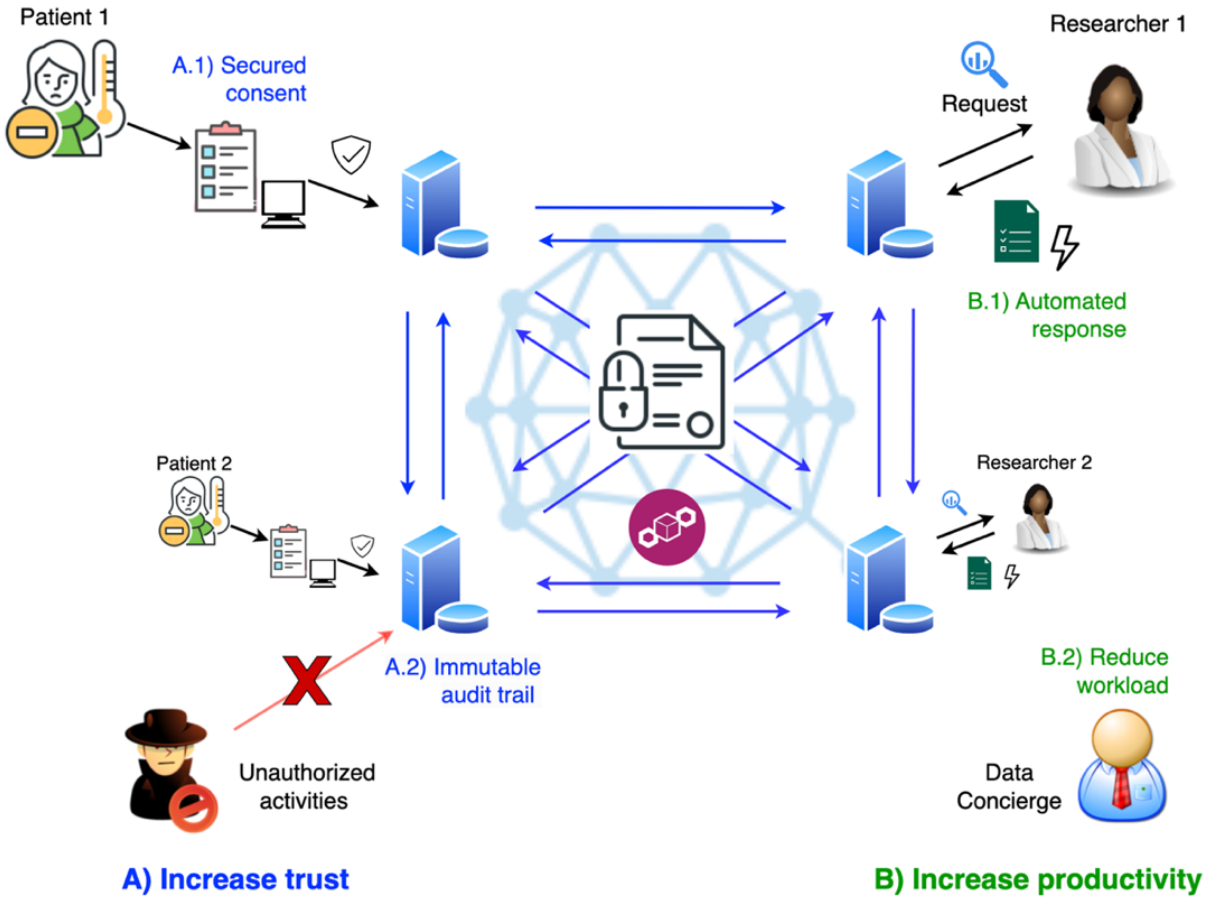


Figure 3.2: Using blockchain and smart contracts to help mitigate challenges along the informed consent process.

A) **Increase trust**: **A.1)** Patient 1 feels encouraged and chooses to share their data with researchers, as the tamper-resistance timestamp mechanism of blockchain may help in case of disputes. **A.2)** Unauthorized activities may be strongly discouraged due to the existence of the immutable audit trail.

B) **Increase productivity**: **B.1)** For researchers, smart contracts that automate consent retrieval can reduce time wasted due to communication backlog or *SPoF*. **B.2)** Moreover, such an immutable automation mechanism can free up human labor.

3.2.4 Related Work

The technological value of blockchain for data-focused activities has been acknowledged in both non-healthcare and healthcare sectors. For example, researchers have advocated for the use of blockchain within the electric grid, such as for the transfer of hashed data among interconnected grids of a smart city [68], for peer-to-peer energy management where users' private information is offered to a management algorithm [69], and for secured data coordination among reconfigurable microgrid components to prevent cyberattacks [70].

In the field of healthcare, blockchain has been enthusiastically recommended for the informed consent protocol. For instance, a study confirmed that blockchain addresses three bioethical principles of consent: patient autonomy over data control, beneficence to facilitate research efficiency, and justice when patients with rare conditions can pseudo-anonymously aggregate data for more robust analysis [71]. Another study offered a conceptual framework to allow patients in assisted living facilities, whose health data may be frequently collected through diverse Internet-of-Thing devices, to assert consent with regards to how their data points may be circulated [72]. Yet another one agreed that research integrity may be bolstered with trial subjects' consents being recorded on the blockchain immutable log [33]. However, to the best of our knowledge, there is a lack of concrete prototypes with demonstrable performance that closely mirror the practical principle of a live, two-way consent infrastructure on which both patients and researchers may carry out frequent activities; instead, previous studies emphasized on high-level descriptions [73] and conceptual system designs with qualitative evaluations [74].

Among those with efficiency measurements, some reported the average computational cost of patient-generated consent activities on a test blockchain network [75], which may not be of most practical concerns to users who may care more about the actual response time of the system as a

whole. Some other works investigated blockchain with regards to cost and speed; however, the actual consent activity was limited to a "share all" approach [76]; or patients could only either actively look up a clinic to grant full access, or revoke access after the fact [77]. Similarly, a study considered consent matching latency in time metrics, yet the classification of consent tiers into only broad categories (e.g., "open", "restrictive", and "very restrictive") might limit true patient preferences over exactly which data category to circulate [78]. In brief, a novel management system that can address these gaps in functionality and practicality, where both patient and researcher-oriented tiered consent activities are facilitated, and whose performance metrics are in time measurements may be of use to support patients and clinical researchers during the consenting process.

3.2.5 Objective

In this study, we aim to demonstrate the feasibility of a private Ethereum-based informed consent system to meet the needs of both health data parties. For patients, their data-sharing consents are securely recorded on-chain with increased trust. For researchers, research productivity can improve when consent records are made available through an immutable and automated channel as they seek consenting subjects for their respective cohorts.

3.3 Materials and Methods

3.3.1 Workflow Overview

The main steps of our method include i) ensuring efficient on-chain storing of patient consents (insertion), and ii) allowing researchers to accurately obtain consenting patients that satisfy specific criteria (querying). For this purpose, we used simulated study-specific consent data that encompass patients' choices over seven health categories (more details in **3.3.2.1**). We chose Ethereum blockchain network [63] as the underlying infrastructure based on prior reviews [64-66], because of Ethereum's ability to offer private blockchains that explicitly require permission to join [65], thereby reducing the chance of unauthorized engagement, as well as the beneficial fact that the Ethereum network itself is popular, well-tested, and actively maintained by the developer community. We designed our Ethereum-based system with two parts: an on-chain Consent Management smart contract, and an off-chain Connection Bridge to support the interactions between participating nodes and the smart contract. We developed the Consent Management smart contract that focused on insertion runtime efficiency because the act of insertion (writing data to a blockchain) by design would involve changing the state of the blockchain and thus is substantially more computational and time-intensive than reading its data (querying) [79]. We measured its performance against a baseline model across different network settings (i.e., number of nodes in the blockchain network), verified the querying results using an off-chain implementation, and submitted our method to the international 2021 iDASH blockchain competition [80] as an external validation.

3.3.2 Data and Smart Contract Design

3.3.2.1 Data

We used the synthetic datasets from the 2021 iDASH blockchain competition [81], which challenged participants to improve the efficiency of a blockchain-based consent infrastructure, in which patients' data-sharing consents (*consent records*) can be stored (*inserted*) on an Ethereum network via Solidity [63] smart contracts, with no off-chain buffering mechanism allowed. There were four datasets, each consisted of 10000 consent records. A consent record contained one patient's data-sharing preference with regards to seven data categories (*Demographics*, *Mental Health*, *Biospecimen*, *Family History*, *Genetic*, *General Clinical Information*, and *Sexual & Reproductive Health*). Details about what constitutes a consent record are in **Table 3.1**. . This structure of the synthetic data simulates the scenario in which patients have the liberty to specify exactly which research study (via a specific Study ID) may obtain their health information, and exactly which information among the seven categories (via a Boolean vector with a size of 7). Furthermore, the patient can change their mind at any given time, leading to two or more entries with the same Patient ID and same Study ID, but different consent records. In such cases, the newer selection will take precedence.

Table 3.1: Detailed information about a consent record.

Data Field	Data Description	Data Type	Sample value	Number of distinct values
Patient ID	The unique identification number assigned to a patient.	Integer	1675	4000
Study ID	The unique identification number assigned to a research study.	Integer	10	60
Timestamp	The timestamp as the patient made their data-sharing choices.	Unix format	1620315008	39991
Consent Record	The seven data categories that a patient may choose for or against sharing with researchers.	Boolean vector of size 7	[“true”, “false”, “false”, “false”, “true”, “true”, “false”]	128

3.3.2.2 Smart contract Design

3.3.2.2.1 The Insertion Algorithm

As our main goal was to increase insertion runtime efficiency, our method focused on minimizing the on-chain computational load during the insertion phase. We examined the inner structure of the datasets to devise a nested mapping design that best served our goal (**Figure 3.3**). This nested Insertion mapping used Patient ID as the main key, and Study ID as the sub key. The values of the inner mapping are data structs created with two attributes, one being the Timestamp when a patient indicated their consent, and the other being the consent record encompassing the seven Boolean choices, one for each of the health data categories. In the case that a patient amended their original consent for a specific study, the nested mapping will update the patient's previous selection with the new consent record; specially, the later choice with a larger Timestamp would override the former to reflect the most recent consent state (**Figure 3.3.A**). At the same time, we utilized another mapping to store Patient IDs as records were pushed on the blockchain at each insert of the Insertion algorithm. For this Consented Patients mapping (**Figure 3.3.B**), the key was a Study ID, and the value was a dynamically-sized vector of Patient IDs who had indicated data-sharing consents (either grant or decline any/all data categories) with regards to that particular Study ID, accepting duplicate values (a patient who had changed their choices for the same study would have their Patient ID appear multiple times in the dynamic vector). Based on a realistic assumption that patients might change minds in relatively low frequency, this was a step to reduce on-chain conditional checking during the insertion phase while still limiting the search space for the Querying algorithm (detailed in **3.3.2.2.2**)

Figure 3.3: Insertion algorithm.

A) **Insertion mapping**: the nested mapping that stores a patient's most recent data-sharing consent with regards to a specific study by overriding previous states. Here, Patient 1001 initially made their sharing choice for Study 10 on 02/22/2021 08:03:01 PST (timestamp: 1614009781), then changed their mind on 09/06/2021 18:28:49 PST (timestamp: 1630978129); whereas Patient 1002 has only made a single consent for Study 10. The updated mapping (the lower box) reflects both Patient 1001's update and Patient 1002's original consent concerning Study 10. The Consent Array is ordered as per ["Demographics," "Mental Health," "Biospecimen," "Family History," "Genetic," "General Clinical Information," "Sexual & Reproductive Health"].

B) **Consented Patients mapping**: another mapping is utilized to minimize membership checking during insertion while still limiting the search space for each query. In this example, the Consented Patients array for Study 10 initially has one record each for Patient 1001 and Patient 1002. As Patient 1001 changed their mind, their Patient ID is again added to the Study 10's Consented Patients (the lower box).

3.3.2.2.2 The Querying Algorithm

In this component, a researcher could seek a list of consenting patients by passing into the Querying algorithm (**Algorithm 1**) i) their specific Study ID, and ii) a subset of the seven health categories, indicating which specific piece of health data the researcher is expecting to get from the patient. Note that this subset can have from one up to all seven categories (the case of an empty subset or "consent to share no category" was excluded). The expected return would be a list of unique Patient IDs who i) had indicated consent for this study, and ii) had agreed to share at least all of the researcher-selected categories in their most up-to-date states. For example, a query combination can seek patients whose last consents have dictated that they would share with Study 10 at least 4 categories of "*Demographics*," "*Mental Health,*" "*Genetic*," and "*Biospecimen.*" Upon querying, the algorithm would traverse the consent history of those specific patients who had ever indicated any consent records for the specific study (instead of looking up all patients). Using these Patient IDs and the Study ID, it would then check against the nested Insertion mapping as per whether the recorded consent matches the requested categories. Since there could be duplications of Patient IDs in the vector, another check was in place to ensure the uniqueness of returned values. In the previous example, a query looking for patients to share with Study 10 at least 4 categories of "*Demographics*," "*Mental Health,*" "*Genetic*," and "*Biospecimen*" would see that the Consented Patients of Study 10 includes [1001, 1002, 1001] (**Figure 3.3.B**, lower box) thus only patients 1001 and 1002 need to be verified. Next, it would find that only patient 1001 satisfied the search criteria (**Figure 3.3.A**, lower box), and that only one instance of Patient ID 1001 should be returned.

Algorithm 1: The high-level Querying algorithm.

Input: The identifier of research *Study ID*, and the array of data categories *Requested Categories* which includes up to 7 data items that researchers expect patients to share with them.

Output: A list of patient IDs whose last consent dictates "true" to at least each of the data items in the *Requested Categories* array for the *Study ID*.

Step 1. Convert *Requested Categories* to a Boolean array *Bool Requested Categories* of fixed size 7.

Step 2. Initialize the array *Consenting Patient IDs*.

Step 3. Loop through the array of *Patient IDs* (including duplicate values) that is associated with the *Study ID* in the mapping *Consented Patients*. For each *Patient ID*, if it is not in array *Consenting Patient IDs*:

- **3.1** Initialize a Boolean flag *Qualified* to *true*.
- **3.2** Locate their latest *Consent Record* for that *Study ID* in the nested mapping *Insertion*.
- **3.3** For each data category, if the value in the array *Bool Requested Categories* is *true* and its corresponding value in the array *Consent Record* is *false*, the flag *Qualified* is set to *false*.
- **3.4** If *Qualified* is *true*, add *Patient ID* to *Consenting Patient IDs*.

Step 4. Return the *Consenting Patient IDs* array.

3.3.3 System Design

Our overall system design is illustrated in **Figure 3.4**. The on-chain Consent Management smart contract was developed in the Solidity [63] language and consisted of the Insertion and Querying algorithms. It can be reached by all nodes in the consent network through the off-chain Connection Bridge component (via the Web3j [82] library). Each participating node would have the same set of the Connection Bridge component to pass insertion input data to, and receive querying output data from, the on-chain contract. The blockchain network itself is an Ethereum [63] network. We adopted the Proof-of-Authority (PoA) consensus [83] (a consensus protocol for private blockchain [84]) and the Go-Ethereum (Geth) implementation [85], on which the Solidity smart contract was launched. The full technological stack is illustrated in **Figure 3.4.A**. Based on these technologies, each network node used the Connection Bridge to insert such data in parallel to the Consent Management Contract (**Figure 3.4.B**).

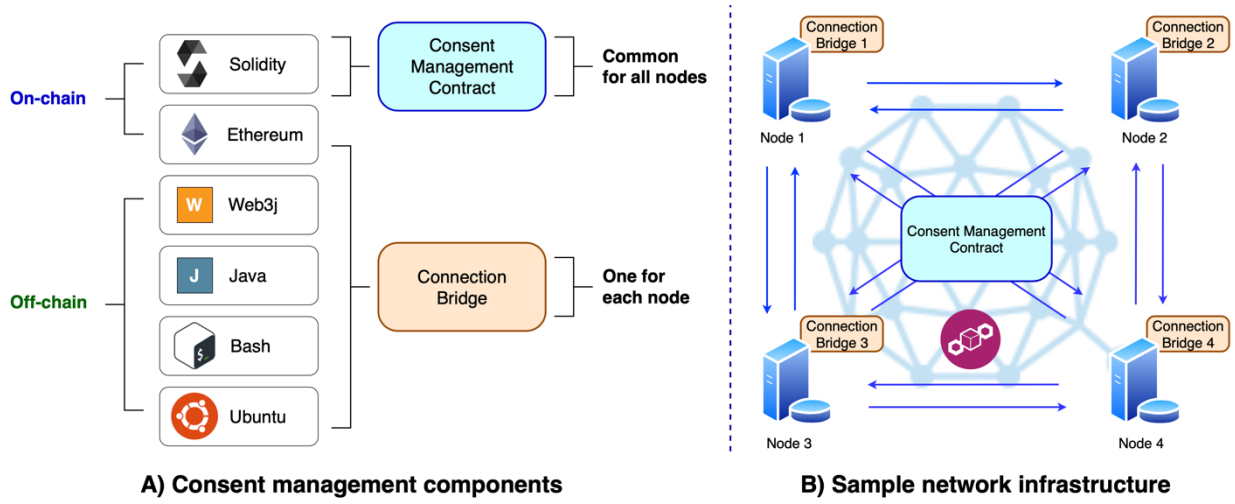


Figure 3.4: Consent system design.

A) **Consent management components**: The application part of the system and its technological stack includes two components: on-chain and off-chain. The on-chain element consists of the Consent Management Contract (i.e., Insertion and Querying algorithms) and the blockchain network. The off-chain part encompasses the Connection Bridge which parses input data and then connects to run the algorithms.

B) **Sample network architecture**: The Consent Management Contract is launched on a blockchain network and is accessible to all four network member nodes/computers. Each node has the exact same set of the Connection Bridge component to interact with the Consent Management Contract.

3.3.4 Development and Evaluation Workflow

We developed the Consent Management contract and evaluated its algorithms against a baseline. In our Ethereum network, each of the nodes was an Amazon Web Service (AWS) Ubuntu virtual machine (VM) (2 vCPUs, 8G RAM, 100GB HD). On a 3-node network, we conducted four trials to get an estimation of their performance consistencies, in which each of their number of consent records successfully stored on the blockchain within each one-hour trial was noted. In each of the trials, after an hour of parallel insertion from all nodes, we would run the Querying algorithm to verify insertion accuracy. We created an exhaustive list of 127 queries ($2^7 - 1 = 127$, since each query consists of 7 Boolean criteria, excluding the "consent to share no category" option, due to the assumption that a patient who preferred not to share their data would also not want their Patient ID exposed), ensuring that all possible consent states were covered. We invoked the Querying algorithm 127 times accordingly and compared the returned results with those of a Python-based, off-chain gold-standard implementation to ensure the accuracy of our Query method. We repeated the process above with the network configuration changed to 5-node. We also submitted our method to the international iDASH blockchain competition to validate our method externally.

3.4 Results

3.4.1 Insertion Efficiency and Querying Accuracy Evaluation

The results of our insertion algorithm are presented in **Figure 3.5**. With regards to insertion efficiency, our method consistently outperformed the baseline design. Moreover, this did not come at the cost of querying accuracy, since the on-chain querying results always yielded a 100% match with its off-chain querying counterpart in any trial, over any network configuration. When launched over a 3-node network, the number of successful consent insertions per hour was 5133, versus the 4957 of baseline as averaged over 4 trials (**Figure 3.5.A.1**). This corresponded to an average of 1711 per hour per node, whereas the baseline's per-hour per-node average was 1652.33 (**Figure 3.5.A.2**). It took 2.10 seconds on average to insert one consent record on-chain (**Figure 3.5.A.3**) when parallelism is considered. For the network setting with 5 nodes, both our method and the baseline improved in performance, while still maintaining their relative efficiency rankings. In particular, our method's average number of successful insertions per hour was 8906.25, outpacing the 8551.25 of baseline (**Figure 3.5.B.1**). The per-hour per-node numbers of insertions were 1781.25 versus 1710.25, respectively (**Figure 3.5.B.2**). The average time per insertion was 2.02 seconds (**Figure 3.5.B.3**).

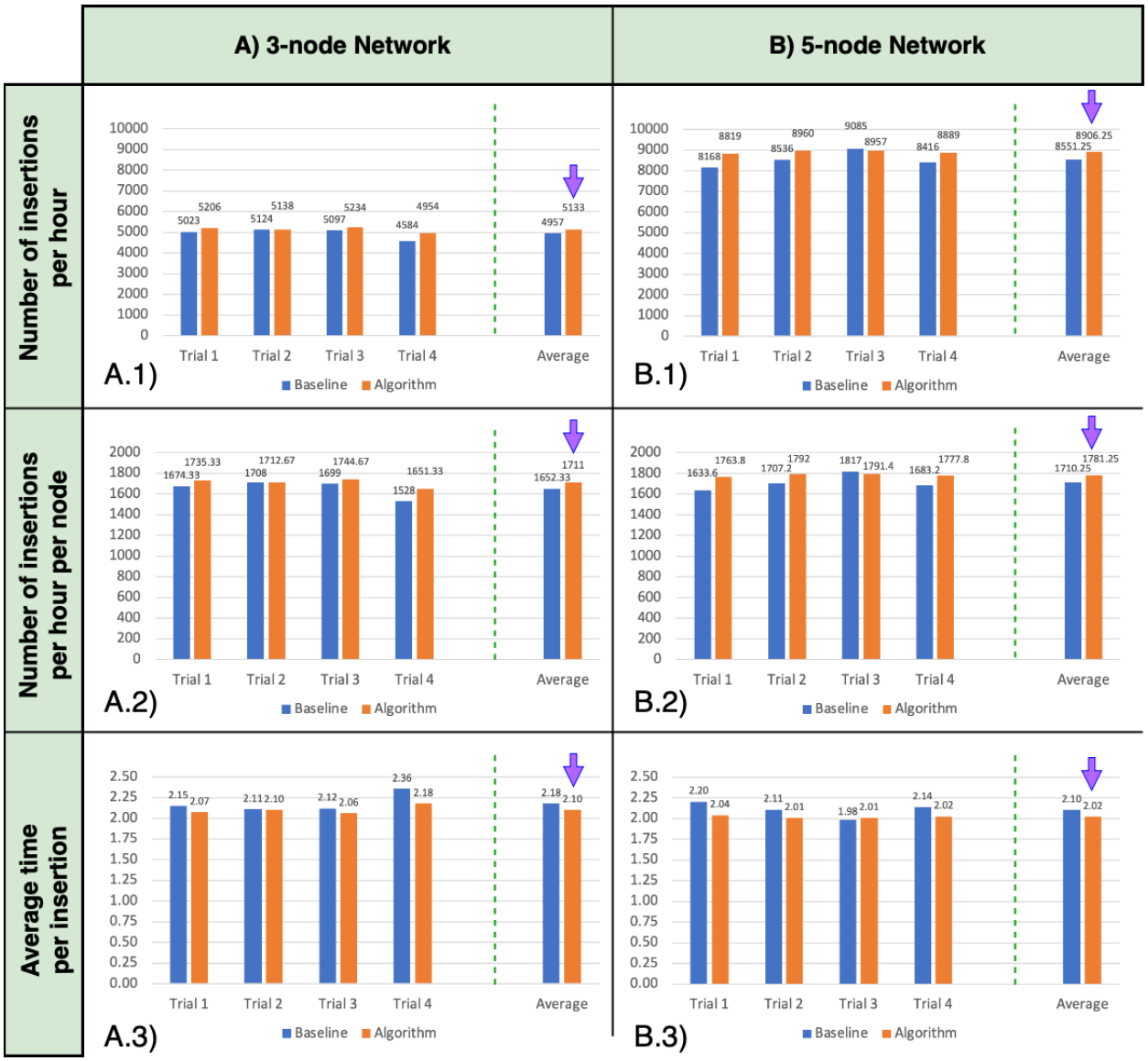


Figure 3.5: Consent insertion results.

A) **On a 3-node network**: **A.1)** The number of total successful insertions per hour; **A.2)** The number of total successful insertions per hour was averaged over 3 nodes; and **A.3)** The number of seconds it took to successfully insert a consent record from a node.

B) **On a 5-node network**: **B.1)** The number of total successful insertions per hour; **B.2)** The number of total successful insertions per hour averaged over 5 nodes; and **B.3)** The number of seconds it took to successfully insert a consent record from a node. In each panel, an arrow points to the better performed method.

3.4.2 External Validation via International Competition

Among all teams that successfully completed the international 2021 iDASH blockchain competition [80], we achieved the best insertion speeds and our querying performance also achieved 100% accuracy. Specifically, the competition adopted a 4-node network and used two settings: "one-at-a-time" and "two-at-a-time". "One-at-a-time" means one consent record is submitted to the chain before waiting for a confirmation receipt and is similar to our synchronous design. For this setup, our algorithm could store 6891.2 consent records in an hour as averaged over 30 trials (with a standard deviation of 135 records), obtaining an average speed of 2.09 seconds per record per node. When the number of concurrent submissions per transaction receipt was increased to two, our method stored 13443.3 consents records per hour on average (standard deviation = 177.8 records), with the average speed of 2.14 seconds per record per node. Our solution performed the best in both "one-at-a-time" and "two-at-a-time" settings. Overall, the performance of our method as validated externally by the international competition was consistent with our evaluation results.

3.5 Discussion

Our algorithm to insert and query study-specific patient consents demonstrated the feasibility of a blockchain-based informed consent protocol, on which tiered and dynamic consents regarding how patients may want to share their health data can be stored and retrieved in an immutable manner. From our results, our method achieved ~3-4% runtime improvement over the baseline for the average time per insertion (as shown in **Figure 3.5.A.3** and **Figure 3.5.B.3**), demonstrating that our insertion algorithm consistently excelled when compared to the baseline on both 3- and 5-node networks. It is possible that researchers may be interested in recruiting patients from specific geographical regions, whose numbers of large medical centers may not substantially exceed five, giving assurance to the empirical merit of our solution. Regarding the external validation, we achieved the best performance among competing teams in the international iDASH blockchain competition 2021, with an average insertion speed of about 2 seconds per consent record per node, and 100% querying accuracy on a 4-node network. It is noted that when the "one-at-a-time" synchronous setting was applied, the external evaluation yielded results comparable to our internal performance, affirming performance stability of our solution. This speed is reasonable in a live system, especially considering the added benefit of immutable audit trail. It is also reasonable under the realistic assumption that patients may not change their minds too frequently. Furthermore, this per-insertion time did improve slightly as the number of network nodes increased, attesting to the benefit of node parallelism, as it offers an insight into how scale of nodes might not degrade insertion efficiency. Moreover, the 100% accuracy in querying ascertains that our method strictly respects patients' sharing preferences, ensuring full adherence to privacy choices.

With regards to limitations, we have not examined the scalability of our algorithms as the number of patients grows, nor have we developed a graphical user interface (GUI) to help test our consent system with real users. An expansion in the granularity of health data categories may affect our system, as it may require adjustment of function parameters and data structure designs within the constraint of the smart contract language [86]. Therefore, a more generalizable design that allows flexible health data categories is yet to be studied. Another limitation is that we have yet to consider other blockchain platforms such as Quorum [87] or Hyperledger Fabric [88] as the architectural infrastructure for our method for a comparative study. Finally, it is also possible that future changes in the underlying technology of blockchain and the smart contract language themselves may alter performance, and thus further investigations may be warranted.

3.6 Chapter Conclusion

In conclusion, we have demonstrated the feasibility of a blockchain-based informed consent protocol, which may help remedy the current security shortcomings of conventional centralized databases. Our methodology achieved a writing speed improvement of about 3-4% as shown in concrete evaluation metrics, an applicable technique future researchers can adopt when designing their own decentralized applications in healthcare as well as in other fields. We considered tiered dynamic consent that enables both autonomic patient control over their own data and enhanced data flow for research purposes; sensitive data may be sequestered for patients' comfort while researchers may still include the remaining "sharable" patients' data for their clinical studies. Such a tiered/dynamic consent matching mechanism confirms complete adherence to the underlying consenting principle that no data may be shared against a patient' wish. Our contribution is that we showcased the feasibility of a blockchain-based consent system using smart contract, which is conducive to increasing patient trust. Our patient consent platform might afford patients the flexibility to amend their consents at will and with strong immutability assurance, while allowing researchers to fast approach their prospective study cohort through unalterable automation of consent matching without third-party's manual processing.

3.7 Acknowledgements

Chapter 3, in part, has been submitted for publication of the material as it may appear in Computers in Biology and Medicine, 2024. Pham, Anh;* Edelson, Maxim;* Nouri, Armin; Kuo, Tsung-Ting, Science Direct, 2024. The dissertation author was one of the primary researchers and co-first authors of this paper. (* These authors contributed equally to this work)

Chapter 4 MANAGEMENT OF DATA CONSORTIUM RESEARCH ACTIVITIES

4.1 Abstract

Objective. We aimed to develop a distributed, immutable, and highly available cross-cloud blockchain system to facilitate federated data analysis activities among multiple institutions.

Materials and Methods. We pre-processed 9,166 COVID-19 Structured Query Language (SQL) code, summary statistics, and user activity logs, from the GitHub repository of the Reliable Response Data Discovery for COVID-19 (R2D2) Consortium. The repository collected local summary statistics from participating institutions and aggregated the global result to a COVID-19-related clinical query, previously posted by clinicians on a website. We developed both on-chain and off-chain components to store/query these activity logs and their associated queries/results on a blockchain for immutability, transparency, and high availability of research communication. We measured run-time efficiency of contract deployment, network transactions, and confirmed the accuracy of recorded logs compared to a centralized baseline solution.

Results. The smart contract deployment took 4.5 seconds on average. The time to record an activity log on blockchain was slightly over 2 seconds, versus 5-9 seconds for baseline. For querying, each query took on average less than 0.4 seconds on blockchain, versus around 2.1 seconds for baseline.

Discussion. The low deployment, recording, and querying times confirm the feasibility of our cross-cloud, blockchain-based federated data analysis system. We have yet to evaluate the system on a larger network with multiple nodes per cloud, to consider how to accommodate a surge in activities, and to investigate methods to lower querying time as the blockchain grows.

Conclusion. Blockchain technology can be used to support federated data analysis among multiple institutions.

4.2 Background and Motivation

4.2.1 Federated COVID-19 Data Analysis

The global COVID-19 pandemic has changed the field of healthcare in profound ways,[89] including how healthcare professionals gather and communicate information given the exponential increase of available data and research works related to the topic.[90] A task that relies on aggregated information from a large quantity of data is federated data analytics, which facilitates ongoing research by combining a larger sample size from multiple sources, thereby potentially increasing statistical power and generalizability of findings.[91,92] Specifically in the context of COVID-19, a rapidly spreading infectious disease with vast socioeconomic and geographic impact, the practice of gathering information across institutions and regions in a timely manner is desired,[93,94] thus leading to the formation of data consortia for federated data analysis. These consortia facilitate achieving a larger sample size from multiple sources, thereby potentially increasing statistical power and generalizability of findings without sharing individual level data in each participating institution.[95,96] Meanwhile, rapid communication among participating members of data consortia is critical to enhance clinical and public health efforts, as they help analyze broader observational data when compared to registries, on which large-scale analyses can be carried out to interrogate clinical progression, therapeutic efficacy, and outcomes.[18,97,98]

4.2.2 The Consortium for COVID-19 Federated Data Analysis

The Reliable Response Data Discovery for COVID-19 (R2D2) Consortium[99] was established in 2020 and funded by the Gordon and Betty Moore Foundation, consisting of 13 US health institutions – Cedars Sinai (CS), Emory University (EMORY), Georgetown University (GT), San Mateo Medical Center (SMMC), University of California Davis (UCD), University of California Irvine (UCI), University of California Los Angeles (UCLA), University of California San Diego (UCSD), University of California San Francisco (UCSF), University of Colorado Anschutz Medical Campus (CUA), University of South California (USC), the University of Texas (UT) Health Science Center at Houston, Veterans Affairs (VA) – and Ludwig Maximilian University (LMU) in Germany. The R2D2 Consortium allowed prompt exchange of summary statistics such as percentage among participating institutions, taking advantage of the substantially large patient pool to empower research while still maintaining patient privacy, since patient-level data never left their original sites.[99] The Consortium workflow started from a researcher who had a clinical question about COVID-19 (e.g., What percentage of patients needed dialysis?) submitted text input to a web portal (https://covid19question.org), and ended with the researcher receiving email notification about a response (e.g., 15% receive dialysis at a particular institution). Specifically, R2D2 consortium data have been used to facilitate ongoing research and patient care analytics in a privacy-preserving manner. Some of the research questions include: (1) Among adult COVID-19 patients who were hospitalized excluding the Intensive Care Unit and discharged alive, how many returned to the hospital within a week, either to the Emergency Room or for another hospital stay? The answer was 8.64%, or 279 patients out of 3230 ones from 10 health systems. (2) Among adults hospitalized with COVID-19, how does the in-hospital mortality rate compare per subgroup, such as age, ethnicity, sex, and race? The result showed that the mortality rate was

higher in older age groups (26% in age group 81 years or older vs 1% below age 40 years), Non-Hispanic ethnicity (12.32%), male sex (13.32%), and white race (12.25%).

In the backend, the whole process entails several steps. First, the received question would be assigned to one of the Consortium members' sites (Lead Site) and reviewed by clinicians who would work with the database analyst to translate the clinical question from text format to Structured Query Language (SQL) code, run the code, verify the resulting local summary statistics, and release the SQL code to the Consortium Hub. At the other site (Responding Site), the database analyst would download the SQL code written by Lead Site, review, and make custom changes to the code including Export-Translate-Load (ETL) process of local Electronic Health Record (EHR) systems, local data harmonization, and local data quality assurance in an iterative process of communication with the Lead Site and local clinicians. The full details of the workflow are described by Kim et al.[99]

An example of how R2D2 and the consortium model might enable original research is shown in **Figure 4.1.A**.[37] Upon receiving a COVID-19 research query in SQL format, each participating hospital would use this SQL code to search their local database for the corresponding numerical result, then submit their local statistics to a central repository that aggregated individual numbers to yield a global statistics result.

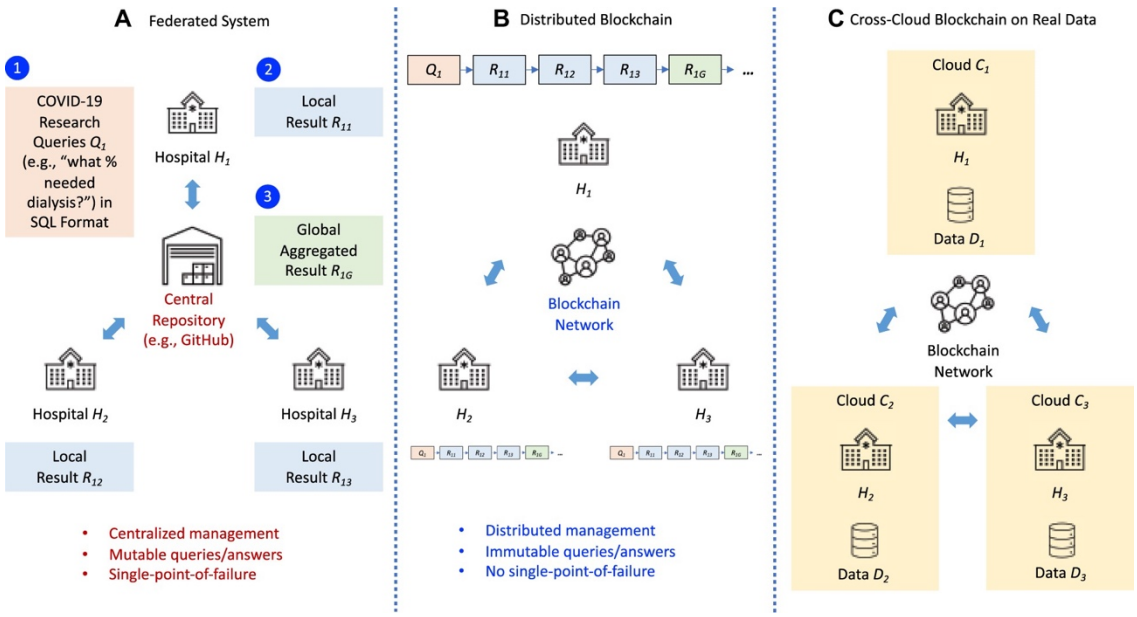


Figure 4.1: Comparison of different COVID-19 data analysis systems.

A) **Federated system**. This example system consists of 3 hospitals (H_1, H_2, and H_3) and a central repository (e.g., GitHub). In Step (1), a researcher in a hospital (H_1) proposed a COVID-19 research query (Q_1). Next in Step (2), each hospital views and submits results (R_{11}, R_{12}, and R_{13}) based on local Electronic Health Record (EHR) data. Finally in Step (3), the results are aggregated to formulate the global results (R_{1G}) to the original query. No patient-level data are disseminated; thus, patient privacy is protected. However, all queries, local results, and global results are stored in the central repository, which presents a centralized management scheme, a single "root" administrative user (who may alter the research queries and results without being noticed), and a single-point-of-failure.

B) **Distributed Blockchain**. Using the peer-to-peer blockchain technology, no central repository is required, and everyone can see all "transactions (TXs)" (i.e., queries, their local results, and global results) on the blockchain. This way, the system is fully distributed (no centralized management), immutable (no single user can change the transactions), and highly available (no single-point-of-failure), while it still protects patient privacy by only sharing aggregated data.

C) **Cross-Cloud Blockchain on Real Data**. Since each hospital may host their real-world COVID-19 data (D_1, D_2, and D_3) in different types of cloud computing environment, the design of the system should also take this into account.

4.2.3 Challenges of Using GitHub to Share SQL Code and Summary Statistics

R2D2 Consortium used a website-based GitHub,[100] a widely popular platform for a distributed version control system in both academia and industry, to share SQL codes and query results of summary statistics.[101] As the role of Lead Site was a resource-intensive task, each institution took a turn based on clinical expertise and subject matter proficiency. This required any consortium member to be able to submit initial SQL code, modify it, and merge with other codes at any given time. Thus, restricting privileges to specific members was not in the consortium's best interest, as workflow bottlenecks could easily happen should one have to defer their work and wait for someone else with higher access privilege. Even for well-adapted services such as GitHub, it is not trivial to customize access control permissions and/or establish proper topology without defaulting to a hierarchical setting in which some members bear more control than others.[102] This led the consortium to decide early on to democratize repository read/write permissions to all members, given the urgency of obtaining clinical knowledge about rapidly spreading COVID-19, the unprecedented nature of the pandemic, and the ultimate goal of the consortium to facilitate rapid research progress across institutions. However, the use of GitHub to share SQL code and summary statistics faced several limitations.

First, it is not possible to ensure the *preservation* of the recorded SQL code and summary statistics results. When all members have read/write permission to the GitHub repository, where SQL codes and their corresponding local summary statistics are gathered, there is a chance some institution may accidentally delete files without being timely noticed, causing the loss of intermediate and/or final results of the federated data analyses.

Second, even when the preservation issue of SQL code and query results of summary statistics might be mitigated by the utilization of automatic backup and the nature of version

control software, the *"root" administrative user* of the central repository still has the privilege to alter the submitted queries/results. While the standard practice is to avoid using "root" access, existing real-life events have proven that exploitative "root" overreach can still occur due to either non-compliant employees [103,104] or malicious attacks on inherent system vulnerability,[24,25] posing a genuine threat to network security.

Finally, the central repository (i.e., GitHub) is a potential *single-point-of-failure*, as the whole system will stop working if this central repository becomes unavailable, as evidenced by previous incidents of severe disruptions and damages when a central system turns non-functional,[105-108] including redundancy failures in major databases.[27,28,109] In the specific instance of GitHub, server issues causing service disruptions have recently occurred.[110,111]

4.2.4 Blockchain Technology to Improve Immutability, Distributedness, and Availability

To overcome some of the challenges outlined above, we explored blockchain, a practically immutable distributed technology with desirable technological features that may ease the time-sensitive pressure on data consortiums, by providing immutable SQL codes and query results of summary statistics, decentralized management without needing a central server, and highly available ledgering.

Blockchain technology[112], which originated from the financial technology field and is now being proposed for various healthcare applications,[30,113] can be a solution to the centralized hub approach used to integrate data from various sites.[30,113] As shown in **Figure 4.1.B**, in our prototype each COVID-19 research SQL code, local query results of summary statistics, and global (i.e., aggregated) summary statistics result were stored on a blockchain, which is *immutable* because once data is recorded on chain it is very difficult, if not impossible, to change the ledgered data.[113] Beyond SQL code and summary statistics, the immutable timestamping mechanism of blockchain also ensures the chronological originality of SQL code posed, allowing for both in-network collaborations and protection of research idea ownerships. This feature may have value when we take into consideration that it is technically possible to amend GitHub history,[114] or timestamps of other file-transferring systems such as ext4 without leaving traces that can easily be noticed in time.[115]

Next, as no centralized server controls all resources, no single site can change the stored SQL code and summary statistics. If any institution leaves the consortium, the whole system could still function, as opposed to the scenario in which the institution that is in charge of integrating data leaves and the whole consortium halts operations until a new "main governing" role is established. This *decentralized* management capability is essential to support the consortium.

Also, based on the blockchain mechanism, each site had a full copy of the blockchain ledger. This in effect makes *high availability* a built-in part of blockchain without the need for intentional human designs, as compared to conventional database systems where redundancy requires conscious efforts yet may still fail.[27,28,109]

In short, the blockchain system is distributed, immutable, and remains highly available, and thus is suitable for record clinical research activities in the consortium. Meanwhile, the protection of patient privacy is still maintained in this design, given that only aggregated data, and not individual-level information, are shared. This makes blockchain as strong a candidate for privacy protection as any other federated systems. Additionally, proposed or proven cases of blockchain technology for clinical and genomic applications such as EHR access control,[116,117] privacy-preserving modeling,[118-123] genomic access logging,[124] gene-drug interaction data sharing,[125] clinical image sharing,[126] training certificates,[127] and patient data sharing consents [128] suggest that the use of blockchain networks may support consortium communications, adding its inherent advantages of immutability, distributedness, and high availability to current solutions.

4.2.5 Objective

We aimed to develop an immutable, distributed, and highly available blockchain system prototype on which cross-cloud, real-world COVID-19 clinical research communications among data-contributing members of a health consortium were logged and made searchable.

4.2.6 Contributions

We showcased the technical feasibility of a blockchain-based system, which could store and query the activities of the federated clinical research data analysis in an immutable, distributed, and highly available way. In particular, our work provided a robust architecture on which federated analysis using consortium data can be carried out efficiently and securely, allowing for secured collaboration on a larger aggregated sample size without subjecting patient privacy to external risks, and ultimately leading to meaningful research and/or quality care improvement in public health.

4.3 Materials and Methods

To demonstrate the feasibility of a blockchain-based system for health data consortiums, we constructed a blockchain-based platform in which users could disseminate their research queries (in SQL format) to participating sites, provide corresponding results, and track activities through a search function that allows location of specific files meeting the search criteria.

4.3.1 Data

During the internal workflow of the R2D2 Consortium, we stored the SQL codes in .sql format (instead of natural language texts) and site-level summary statistics in .csv and .txt format files in the internal private GitHub central repository.[100] To prepare for the blockchain implementation, we pre-processed these SQL codes and summary statistics files collected between May 28, 2020 and June 7, 2021, along with 14,178 user activity logs and 3,596 relevant files (i.e., SQL codes and data results). The dataset included 27 researcher-provided questions that have been translated into SQL code by the consortium Lead Sites. The institutional-level summary statistics files were encrypted with a hashed value to prevent privacy leaks, while keeping SQL code as a plaintext. Multiple activity logs were associated with the same file when the file was modified, renamed, or deleted after it was first added to the repository. In this case, there were two activity logs for one file. The details of data fields captured from each file are summarized and described in **Table 4.1**, with several fields being parsed from other fields. Specifically, *Institution* was extracted from the *User* field (35 users in total) and then mapped to the abbreviation names of the 14 institutions (e.g., "*UCSD*"). Also, the Source/Target *Query Number* and the *File Type* fields were parsed from the Source/Target *File Name* fields.

Table 4.1: Information captured from the COVID-19 consortium private GitHub repository. The fields with asterisks ("*") were derived from other fields. In the original GitHub log, there is a "similarity score" after the "R" operation (e.g., "R98", indicating that the target file, after renaming, has 98% similarity with the source file). In our study we only focused on the operation and considered "R98" as "R".

Type	Field	Description	Example(s)
Log	Timestamp	Unix timestamp of the operation	*1613693792* (Friday, February 19, 2021, 12:16:32 AM)
	User	User who performed the operation and is identified by their email address	*tskuo@ucsd.edu*
	Institution *	User's institution, derived and mapped from *User*	*UCSD*
	Operation	Operation conducted	*A* (Add), *D* (Delete), *M* (Modify), *R** (Rename)
	Source File Name	Full name of the source file ("*N/A,*" i.e., "not applicable", for the "*A*" operation, and the same as the target file name for the "*M*" operation)	*Query_0026/results/Site01_results_new.csv*
	Source File Type *	Type of file, derived from *Source File Name*	*MD* (description of the query), *SQL* (query details of the query), *CSV* (local/aggregated results)
	Source Query Number *	Numerical reference for the COVID-19 queries, derived from *Source File Name*	*0, 1, 2, …, 26*
	Target File Name	The full name of the target file ("*N/A*" for the "*D*" operation, and the same as the source file name for the "*M*" operation)	*Query_0026/results/Site01_results.csv*
	Target File Type *	Type of file, derived from Target File Name	*MD*, *SQL*, or *CSV*
	Target Query Number *	Numerical reference for COVID-19 queries, derived from *Target File Name*	*0, 1, 2, …, 26*
File	Compressed File Content	Content of the operated file after compression	*Site01_results.csv*

4.3.2 Method Overview

An overview of our method is depicted in **Figure 4.2**. There are four components in our method: (1) the data pre-processing step, which filters, parses, and/or cleans up the activity logs and their files; (2) an off-chain log management component that submits the parsed logs/files to the on-chain program of our blockchain network; (3) a smart contract (i.e., programs that are stored and executed on the blockchain) and is responsible for the actual on-chain storing/querying of logs and files; and (4) an activity querying component, which allows on-chain searching of stored logs/files. Together, the components facilitate a workflow in which pre-processed activity logs and files would be recorded, and also made available for querying such as, "*Which institutions responded to Query 26 in May 2021, and what are the result files they sent?*" The outputs are logs and files that satisfy the search. The details of these four components are introduced in the following four sections.

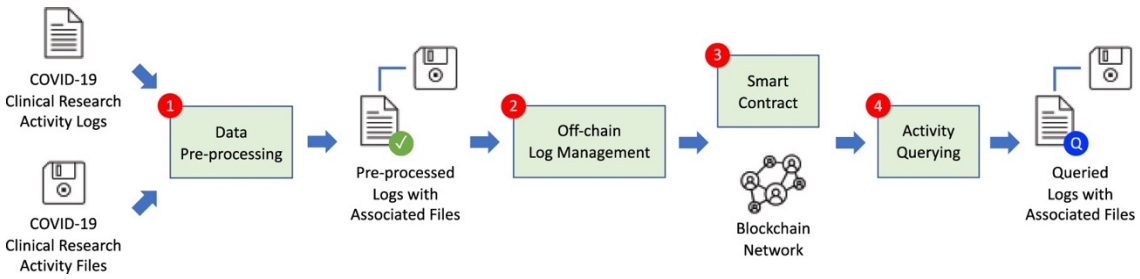


Figure 4.2: Experiment Design. The inputs include COVID-19 queries/results logs and files, and the outputs are the queried logs with associated files. There are 4 components: (1) data preprocessing, (2) off-chain log management, (3) smart contract, and (4) activity querying.[37]

4.3.3 Data Processing

The 4 data preprocessing steps are as follows:

1. Filtering and splitting of logs: we removed administrative logs that were not directly related to COVID-19 clinical research activities (e.g., dashboard statistics). This step reduced our dataset size from 14178 to 9433 logs.
2. Filtering and deidentifying of files: similarly, we removed administrative files to only keep the 3 file types of MD (general description of a particular research query), SQL (the actual code that entails programming language-specific details), and CSV (SQL result of institutional-level summary statistics). An example of excluded file types is a human-readable research question in PDF format. We also “deidentified” any personal information in the files. Specifically, we removed the "requester" information in each MD file and replaced the actual content of a CSV result with a random Secure Hash Algorithm 1 (SHA-1) value with a length of 20 bytes.49 This was for the purpose of synthetic testing; in a real-world deployment, authentic data can be transmitted among authorized network members securely via the SHA-1 algorithm. This step reduced our dataset size from 3596 to 2143 files.
3. Linking logs and files: the file names mentioned in the activity logs were used to identify the associated files. For logs with operation "R" (Rename), we used the target file names for the linkage, because after renaming the source file names would have been outdated. This step did not change the size of our dataset.
4. Cleaning-up and parsing logs/files: in this final step, we removed duplicated "typo" logs such as those with simple upper/lower case issues. Then, we also removed the files that were not associated with any activity logs (e.g., because of the removal of the duplicated

typo logs). Next, we parsed the fields of the logs to derive other necessary information (i.e., those with "*" as shown in **Table 4.1**[37]). After this step, we reduced the dataset size to 9166 logs and 2098 files, to be recorded and made available for searching on the blockchain. Among the 9166 logs, only 4250 logs were associated with files; this was due to the fact that one file could be linked to multiple activity logs. For example, a "Rename" operation would result in at least 2 activity logs for one single file.

As shown in **Figure 4.3**,[37] the filtering process mainly removed nonclinical research activity logs/files, thus the impact to the effectiveness of our dataset is minimal.

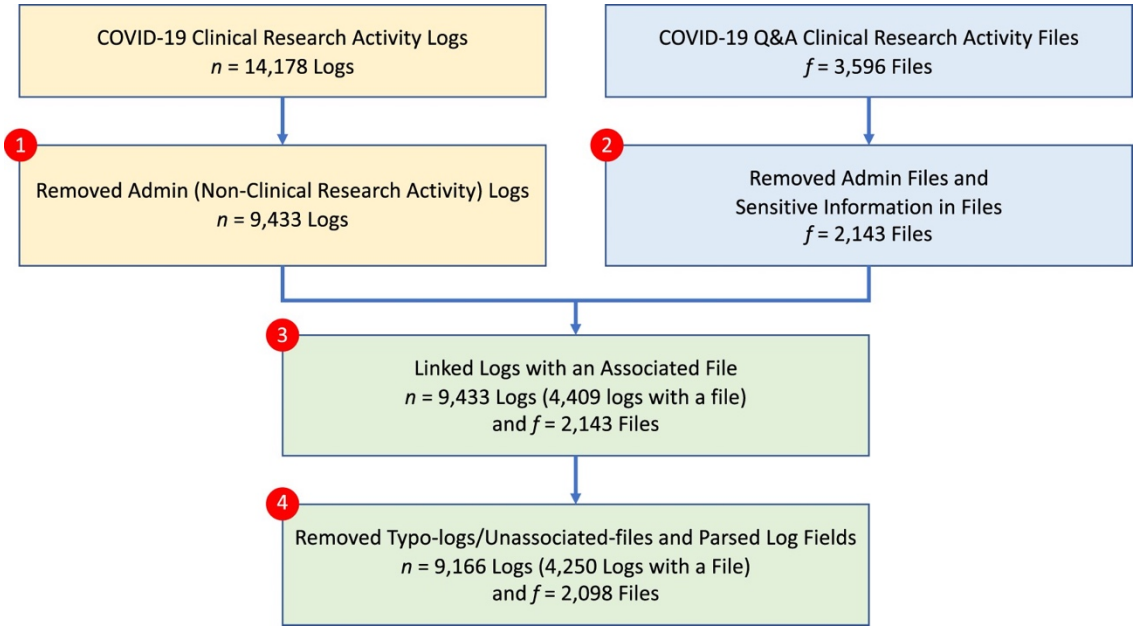


Figure 4.3: Data preprocessing flowchart. The preprocessing steps include: (1) filtering and splitting logs, (2) filtering and deidentifying files, (3) linking logs and files, and (4) cleaning-up and parsing logs/files.[37]

4.3.4 Off-chain Log Management.

After activity logs and associated files had been pre-processed, we developed an off-chain log management component to automate the submissions of logs/files to the blockchain network. To make the input data compatible with the smart contract design (detailed in Subsection 0**)**, we parsed out necessary information from the activity logs such as the original timestamp of the research activity, the user email address, the institution to which the user belonged, the operation committed ("add," "modify," "delete," or "rename"), the specific research query that the activity log pertained, and the content of the associated file (if any). Although the COVID-19 clinical research activity logs are relatively small, the files associated with such activities were up to ~330 KB. As a key consideration of storing data on blockchain is scalability,[113] we compressed each file before submitting it to the blockchain network to improve the storage speed. The parsed fields and compressed files were then ready to be passed onto the next component (i.e., the on-chain smart contract, to be described in the next section). In addition to automating the submission of logs/files to the blockchain for storage purposes, the off-chain log management component also automated the querying task by submitting search criteria to the blockchain network, i.e., it would "call" or "invoke" the on-chain querying function and receive the returned matches (more details in **4.3.6**).

4.3.5 Smart Contract.

We developed a smart contract to store and query the COVID-19 clinical research query/result activity logs with their corresponding files on-chain with two main functions, storage and querying, to be invoked by the off-chain log management component (described in **4.3.4**). For the storage function, each activity log as well as its associated compressed file content (if any) is recorded on the blockchain in the form of a "data struct," which is a set of smart contract variables, to store the parsed-out activity information and the compressed file (when appropriate). For the querying function, our smart contract design also allowed user-specified search criteria to filter the variables within the data structs. The high-level data structure and functions are demonstrated in **Figure 4.4**, and more specific details about the querying capacity are provided in the next section.

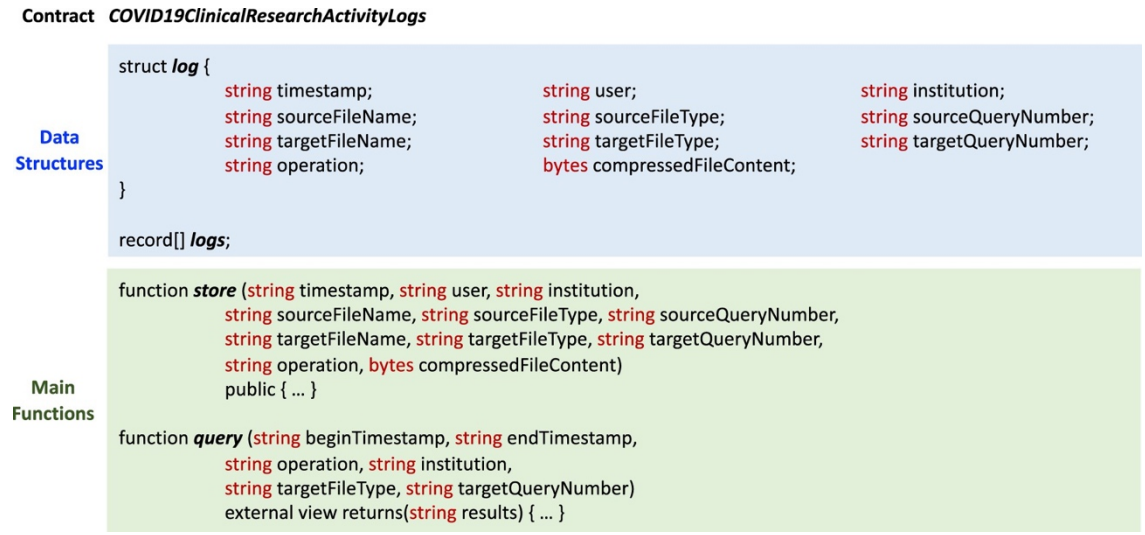


Figure 4.4: The high-level smart contract developed to store/query COVID-19 clinical research activity logs. The data structure mainly contains an array of "line of log," and each line of log consists of all fields listed in **Table 4.1**[37] (including the Compressed File Content if a file is associated with this line). The store/query main functions are designed for recording and searching the logs on the blockchain.[37]

4.3.6 Activity Querying.

To allow researchers and network auditors to search the COVID-19 clinical research data stored on-chain, we developed the query component with the following six search criteria: *Begin Date*, *End Date*, *Operation*, *Institution*, *Query Number*, and *File Type*. Since the target query number and file type are more up-to-date than their source counterparts, we only focused on target fields (e.g., Query Number and File Type). The *Begin Date* and the *End Date* were converted to Unix timestamps to be compared with the logs and files' Timestamp fields listed in **Table 4.1**. After querying, logs as well as their associated files (if any) that met the search criteria were returned to the researchers/auditors. Both the search criteria and the querying matches were passed between the off-chain log management component and the blockchain smart contract.

4.3.7 GitHub Repository Baseline.

To understand the performance of our proposed solution versus a centralized one, we developed a GitHub baseline where users initiate SQL code to the consortium's GitHub repository and await member results via file commits. Since GitHub is designed to store files (instead of data entries), fields of information regarding timestamp, user email, institution, etc., were concatenated into a string, so as the string could be pushed to the repository as a file commit (with compressed file content if the specific activity log included an actual file, empty otherwise). Taking into consideration the inner working mechanism of GitHub, in which a file needs to be sequentially added, committed, and pushed from a local repository that is fully up-to-date with its remote counterpart, we first created multiple branches/sub-repositories (one for each node), to which the particular node would push a file without the need for frequent pulling. After all nodes had finished pushing to their individual branches, a coordination script merged the branches into a single, final repository. Then, we queried the file commits using the search criteria (described in **4.3.6)** to find matches on a local repository that had pulled all contents from the remote repository (i.e., a "pull" action) to ensure the local repository is always up-to-date with the remote , making the "pull" action an inherent part of the query.

4.3.8 Cross-Cloud System Implementation.

The implementation architecture of our system is shown in Figure 4.5. We simultaneously deployed our system on three major cloud providers: Microsoft Azure (MA), Google Cloud Platform (GCP), and Amazon Web Services (AWS). Based on our prior reviews,[129,130] we selected the community-supported, open-source Ethereum [131] blockchain platform. Specifically, we adopted the Go-Ethereum (Geth) implementation,[132] and used a Proof-of-Authority (PoA) [133] consensus protocol Clique [134] which is designed for private blockchain (i.e., only invited institutional nodes/computers may join the network). We developed the smart contract using the Solidity [135] programming language, utilized Java and Bash to develop the off-chain log management and log querying components, and leveraged Web3j [136] as the bridging library between our off-chain Java application and the blockchain network. The baseline methods were implemented using Python and Bash. We deployed our system on Ubuntu Virtual Machines (VMs) with 2 vCPUs, 8GB RAM, and 100GB Disk. This same specification was adopted for all three cloud providers. Considering the fact that cloud platforms may be susceptible to privacy leakage, we only uploaded the de-identified files (i.e., Step 2 in **4.3.3**) to the cloud platforms to conduct the experiments to protect privacy.

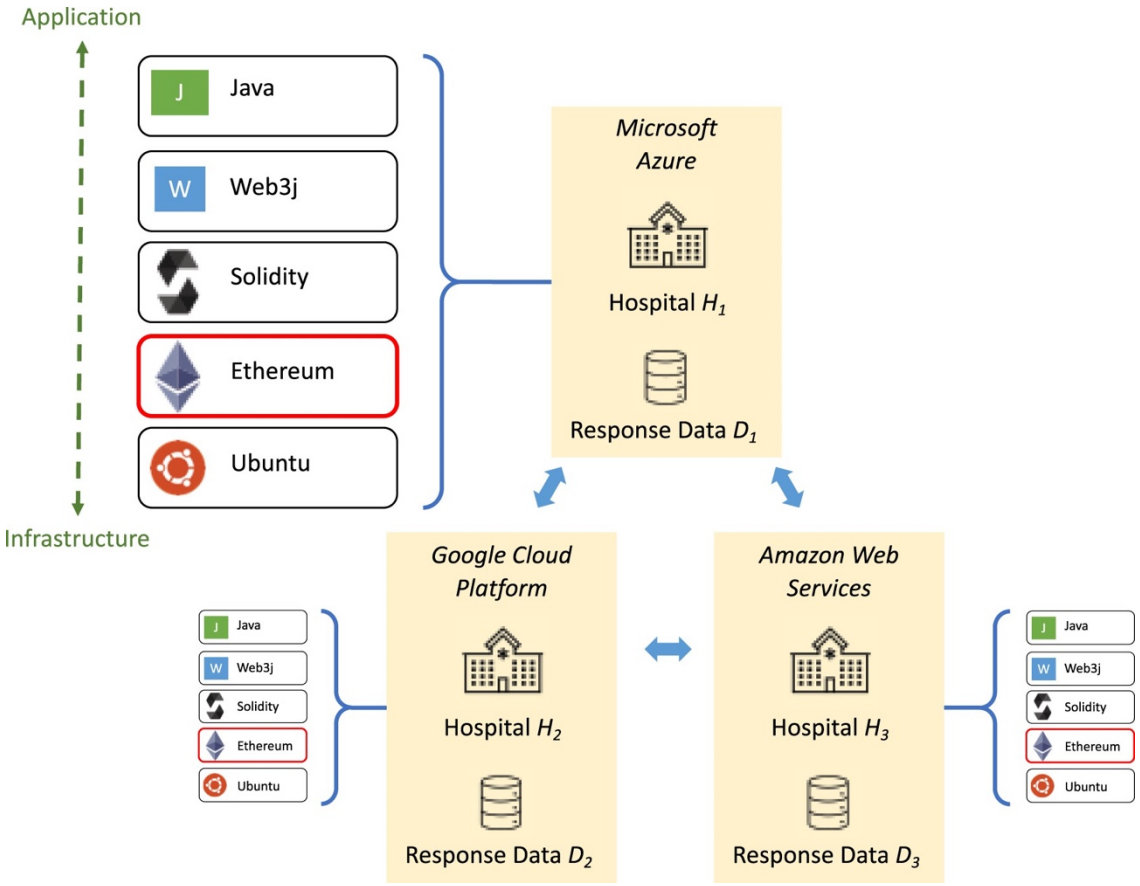


Figure 4.5: Implementation architecture. We simulate a 3-node blockchain network, with each node hosted in a different cloud provider. The technology stack on each hospital is the same (with upper being application-level and lower infrastructure-level). Ethereum is the only connection point to other nodes.[37]

4.3.9 Experiment Settings

To demonstrate its practical feasibility, we evaluated the initialization, data recording, and search time performance of the system, and compared it with the baseline. We simulated two scenarios in our experiments, referring to different cases of how the data load could have been distributed: *log-level,* and *institution-level* data splitting. For *log-level* splitting, we randomly split the 9,166 logs (and any associated files) into approximately three equal parts, which were stored on the three VMs, each within a different cloud (i.e., MA, GCP, or AWS). This was to simulate the ideal situation in which each blockchain node has approximately an equal number of logs/files in their data inputs. For *institutional-level* splitting, we first split the 9,166 logs/files to 14 bins based on their institutions. We then randomly grouped the 14 bins of logs/files into three parts to be stored again on three VMs within a different cloud. For example, a bin might have five institutions, and the other two bins would have four institutions each. This scenario simulated the more realistic situation where each blockchain node may include a cluster of institutions with vastly different activity loads, and thus each contains a vastly different number of logs/files as inputs.

For both scenarios, we measured the time required for initialization (i.e., deploying the smart contract on the blockchain). Next, we measured the average recording time per log, which was measured by the duration between the time the first recording transaction was initiated and the time when the last transaction finished among all VMs. This was particularly important in the *institutional-level* data splitting scenario, as a node might have significantly more input logs than other nodes, and thus would require more time to finish all transactions. We repeated this process 30 times and conducted a paired t-test with alpha = 0.05 to understand whether there are statistically significant differences ($p < 0.05$) between our proposed method and the baseline.

After the logs and files were recorded on-chain, we conducted 18,144 queries using the exhaustive combinations of six query criteria (details in **Table 4.2**) and measured the average search time per query. An example of a query combination could be "*Find all activity logs and files within 9/1/2020 and 3/1/2021 that are of operation type 'Modify' ('M'), were initiated by UCSD for Query 9, and were CSV extensions.*" We also compared the results with the ones from the baseline methods.

Table 4.2: Possible values per query criterion. For Begin and End timestamps, we selected two candidate dates each.

Field	Values	Number of Combinations
Begin Timestamp	6/1/2020, 9/1/2020	*2*
End Timestamp	3/1/2021, 6/1/2021	*2*
Operation	A, D, M, R	*4*
Institution	CS, CUA, EMORY, GT, LMU, SMMC, UCD, UCI, UCLA, UCSD, UCSF, USC, UTH, VA	*14*
Query Number	00 – 26	*27*
File Type	MD, SQL, CSV	*3*
Total Number of Combinations		***18,144***

4.4 Results

4.4.1 COVID-19 Clinical Research Data Analysis

At the institutional level, the log statistics are shown in **Figure 4.6.A**. The log count and the compressed file size both follow a long-tailed distribution (i.e., most logs/files were from the more actively participating institutions, and most files after compression were of size up to 30 KB).[137] Some institutions may store larger files per log, indicated by the relatively higher compressed size value when compared with its activity log count. The required number of transactions (TXs) per log is shown in **Figure 4.6.B**; for example, after compression, the largest file became ~90 KB, which required three blockchain TXs with a size limit of ~30 KB each for the Ethereum blockchain platform used in our experiment. The logs without an associated file still required one TX to be recorded on-chain. The compressed file sizes ranged from 0 – 90 KB, and the corresponding numbers of required blockchain TXs ranged from 1 – 3, both following a long-tail distribution. For log-level data splitting, 9,166 logs were evenly split into three nodes, and the maximum number of logs per node was $\lceil 9{,}166 / 3 \rceil = 3{,}056$ during any of the 30 trials. For the institution-level data splitting, the number of logs on each node may have differed due to the randomization of institutional combinations; on average, a node might carry up to 6,783 logs per trial, while the other nodes had fewer input logs.

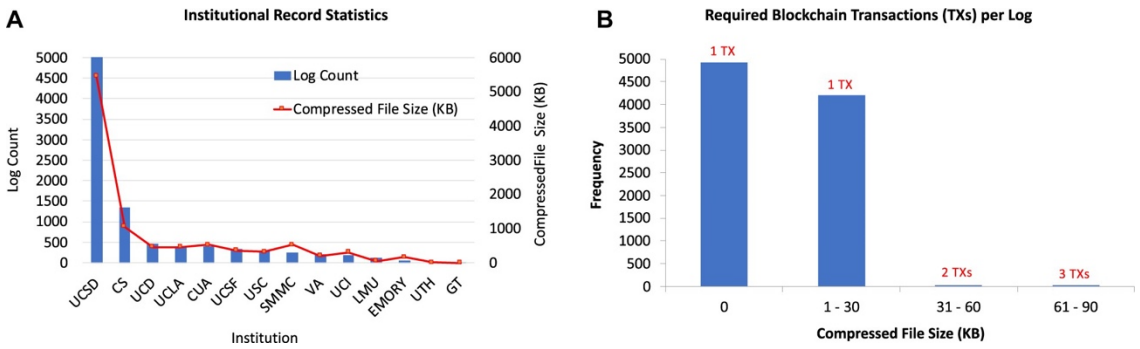


Figure 4.6: Statistics of records and compressed file sizes.

A) **Institutional record statistics**. The X axis shows the 14 participating institutions, ordered by the record count (left Y axis, blue bars). The total compressed file size for each institution is depicted in the right Y axis.

B) **Blockchain transactions (TXs) per record**. Each blockchain transaction has a size limit of ~30 KB for the Ethereum blockchain platform. Also, the logs without an associated file (i.e., compressed file size=0 KB) still required 1 TX to be stored on the blockchain. Therefore, the X axis shows the range of the compressed file sizes for 1, 2, and 3 TXs, and the Y axis demonstrates the frequency of the records within the corresponding size range.

4.4.2 Smart Contract Deployment Times

The average smart contract deployment time for the log-level data splitting was 4.557 seconds, which is a one-time cost. The average deployment time for the institution-level splitting was 4.430 seconds. The standard deviations of log- and institution-level splitting were 0.354 and 0.203 seconds, respectively.

4.4.3 Storage Times

The results for storage times are shown in **Table 4.3**. In general, the total time for the baseline and the proposed blockchain method depended on the distribution of input logs among nodes and was in direct proportion with the largest number of input logs among the three nodes. However, the storage speeds between the two comparing methods showed statistical differences for both data splitting scenarios of the 9166 input logs: For the *log-level* splitting scenario, each node had either 3055 or 3056 input logs, indicating that each VM should finish pushing to the blockchain or GitHub around the same time. The GitHub baseline took ~8 hours with a relatively high standard deviation (~43 minutes) over 30 trials. The proposed blockchain network achieved consistent performance (~1.8 hours) with a relatively low standard deviation (~1.5 minutes) across 30 trials. For the *institution-level* splitting, the GitHub baseline took an average of ~10 hours, versus the proposed blockchain's ~3.7 hours, averaged over their relative 30 trials each. Their corresponding standard deviations showed a similar trend (i.e., the one for the blockchain method is smaller than that of the baseline) to the log-level splitting case.

The differences in total time and consistency between the two methods translated to their speed differences when averaged over all 9166 input logs or the largest number of input logs assigned among the three VMs during a particular trial. When averaged over the largest number of input logs, the GitHub baseline took ~9.4 seconds at log-level and ~5.7 seconds at the institution-level on average. Meanwhile, whether log-level or institution-level, the blockchain took ~ 2.1 seconds on average.

Table 4.3: Storage time results. The time was measured in seconds. The values were averaged over 30 trials, with standard deviation shown in parentheses.

Splitting	Measurements	Baseline	Proposed	P-value
Log-level	Total Time	28811.467 (2571.143)	6405.482 (88.000)	$p < 10^{-28}$
	Average Time over 9,166 Records	3.143 (0.281)	0.699 (0.010)	$p < 10^{-28}$
	Average Time over the Largest Number of Input Logs	9.428 (0.841)	2.096 (0.029)	$p < 10^{-28}$
Institution-level	Total Time	35769.300 (7864.710)	13401.609 (1882.121)	$p < 10^{-14}$
	Average Time over 9,166 Records	3.902 (0.858)	1.462 (0.205)	$p < 10^{-14}$
	Average Time over the Largest Number of Input Logs	5.713 (1.183)	2.130 (0.201)	$p < 10^{-15}$

4.4.4 Query Times

For correctness, the on-chain query result from the proposed blockchain mechanism agreed 100% with the returned matches as yielded locally by running a Python querying script on a fully updated GitHub local repository (a "pull" command was run as part of each query). Each local GitHub query took ~2.4 seconds for log-level splitting and ~1.8 seconds for institution-level splitting. Meanwhile, the blockchain query would take ~0.5 seconds for either splitting. Detailed information on querying speed of the proposed blockchain method is illustrated in **Table 4.4**.

Table 4.4: Query time results. Time is measured in seconds, and per-query values are averaged over 18,144 combinations and 30 trials, with standard deviation shown in parentheses.

Splitting	Measurements	Proposed
Log-level	Total Time	10670.472 (4730.111)
	Per-Query Time	0.588 (0.261)
Institution-level	Total Time	7537.861 (361.600)
	Per-Query Time	0.415 (0.020)

4.5 Discussion

4.5.1 Findings

In general, the deployment time (~4 seconds) was negligible since the smart contract only needs to be deployed once when the blockchain network is being set up. Compared to the GitHub baseline, our proposed blockchain mechanism was more effective at storing federated data analysis activities among consortium members ($p < 10^{-28}$ in the log-level and $p < 10^{-14}$ for the institution-level splitting scenarios). The difference in speed between the two systems might be attributed to the fact that GitHub does require frequent reconciliation of states between local and remote repositories before each commit can be made, and that the sequence of actions entailing "add/commit/push" might also need more time to complete. There might also be external factors that affected GitHub performance such as server traffic.

Similarly, the proposed blockchain was also faster in querying (~4 times as fast), which demonstrated the feasibility of log auditing. It took advantage of the fact that a blockchain network node always has the most up-to-date data record due to the distributed ledger infrastructure, whereas the GitHub local/remote setting mandates a thorough "pull" as part of each query. The feasibility of our cross-cloud blockchain-based system for recording clinical research query/result activities was thus demonstrated by the low deployment time, activity recording time, and querying time, showing that from a technical perspective, blockchain with its native security edges can function as the supporting infrastructure for communications among members of clinical data consortiums.

We also implemented and tested a *round-robin* design of the GitHub storage method, which in effect most closely resembled the blockchain mechanism by letting all nodes/computers publish to the same remote repository. Specifically, before any node could push a file commit from

local to remote, it would need to successfully pull down the most current state of the remote repository (i.e., wait for the completion of the previous node, thus “round-robin”). The average storage time of this design is ~42.4 seconds for log-level and ~21.8 seconds for institution-level splitting, which is slower than our GitHub baseline (~9.4 seconds and ~5.7 seconds, respectively) as well as our proposed blockchain method (~ 2.1 seconds for both splitting scenarios).

4.5.2 Contribution of the blockchain-based solution.

Although GitHub was not specifically designed for federated data analysis nor team communication, we used it as our baseline in this study to be consistent with the R2D2 Consortium's mechanism.[99] By design, GitHub is vulnerable to potential server outages (and thus carries the single-point-of-failure threat). In comparison, our proposed blockchain approach offers both faster storage and querying speeds, and the benefit of being highly available without *Single-Point-of-Failure*; given that each network node always has a full copy of the blockchain, our mechanism will not be impacted if any of the participating nodes fails. This technical feature emphasizes the benefit that our blockchain-based solution may have over other conventional centralized methods such as GitHub.

Based on results that demonstrated the feasibility of using blockchain for clinical research activity logging, we anticipate our system can be adapted for other collaborative research activities beyond COVID-19. For example, it may be of use in modern times when global transportation increases the risk of spreading diseases (e.g., monkeypox, Zika virus, etc.), and the data consortium model is an ideal candidate to gather data from simultaneous, disconnected outbursts. We believe that blockchain-based solutions, while not being "one-size-fits-all," would bring the most value when multiple institutions would like to collaborate for a time-critical mission, and where issues such as single-point-of-failure would impede such collaborations. Nevertheless, it is imperative to weigh the costs and benefits of adopting this new distributed technology in lieu of a mature and centralized one. Moreover, an experienced team that understands how to design and develop blockchain-based solutions in an effective and efficient way will be an important consideration when generalizing this study for different application fields.

4.5.3 Limitations

There are limitations to our study:

1. The format of our queries remained in the SQL format as described in the Data subsection, and thus is different from NLP-based question-answering methods such as CoQUAD[138] that take natural language texts as their input. We are yet to investigate and adopt NLP methods[139,140] to automatically parse the COVID-19 questions from their natural texts.
2. We only evaluated our system on a blockchain network with 3 nodes (one from each cloud provider), while more nodes unevenly distributed across 3 clouds might impact system performance.
3. We assumed that the requests to store/query the COVID-19 research queries and results were spread evenly at any time. However, since multiple users may simultaneously work on one blockchain node, chances are people may suddenly have interest in the same research topics (e.g., a new virus variant appears) in a short time.
4. We are yet to redesign our system to accommodate such potential surges of storing/querying requests and compare our system with its centralized counterpart.
5. With regards to the size of files that can be recorded on-chain, we have yet to investigate the limit of the size.
6. Blockchain is a linear linked-list data structure by design. Therefore, the query time may increase as the blockchain grows. Methods to improve querying scalability such as ones reported in recent benchmarking studies[124,125] are yet to be investigated.
7. We are yet to evaluate our method in real-world settings such as when pandemic data may increase exponentially. While the direct inputs for our system are SQL queries and statistical results, which may not scale in synchronization with the abundance of patient

data at each institution, the pandemic may still impact the frequency of federated data analysis activities and/or the number of data consortium members. Further work would be needed to investigate both the scalability of the system and the use cases for other pandemics such as monkeypox.

4.6 Chapter Conclusion

Our results show that it is feasible to store and query COVID-19 queries and results on blockchain in the way similar to conventional platforms such as GitHub, to take advantage of blockchain's intrinsic desirable technical features such as decentralization, immutability, and high availability. Given that our experiment utilized three different cloud environments, they may reflect a real-world setting, where a data-contributing consortium consists of multiple institutions whose computational architectures are placed on different cloud providers. This study can help support the recording/management of future clinical federated data analysis activity logging and file transferring systems, where data from as many sources as possible are required to yield a better picture of disease progression to inform public health, and especially to better combat pandemics.

4.7 Aim 1 Statement of Impact

Our works in Chapter 3 and Chapter 4 showed that it is strategically practical and technologically feasible to embed the sharing of research-oriented activities on a blockchain cyber-infrastructure. From the perspective of patients, knowing that their given consents are free of compromises might raise trust towards secondary reuse, giving them true control and a realistic understanding of the data economy in an era when infamous privacy breaches have made the general public weary. This in turn might promote the expansion of the health data pipeline, allowing for a healthy flow of patient-level data in research. For researchers, our work may help improve productivity and boost innovations, as evidenced in the use cases of automatic consent-matching through the informed consent process, and management of federated data-consortium research activities. In both instances, the decentralized approach obtained good efficiency with regards to their time-wise performance, while still adding security benefits. This affirms the potential of blockchain in supporting the health research workflow.

4.8 Acknowledgements

Chapter 4, in full, is a reprint of the material as it appears in Journal of the American Medical Informatics Association, 2023. Kuo, Tsung-Ting;* Pham, Anh;* Edelson, Maxim;* Kim, Jihoon; Chan, Jason; Gupta, Yash, Ohno-Machado, Lucila; The R2D2 Consortium, Oxford University Press, 2023. The dissertation author was one of the primary investigators and co-first authors of this paper. (* These authors contributed equally to this work)

Chapter 5 HEALTH DATA USE FOR PREDICTIVE ANALYSIS IN INFECTION

5.1 Abstract

Objective. *Clostridioides difficile* infection is a major health threat. Healthcare institutions have strong medical and financial incentives to keep infections under control. Blanket testing at admission is in general not recommended, and current predictive models either used moderate sample sizes, over-inflated the number of covariates, or chose non-interpretable algorithms. We aim to develop models using patient data to predict positive *Clostridioides difficile* test results with discrimination performance, interpretable results, and a reasonable number of covariates that reflect health over a long-time span.

Methods. We processed records from 157,493 University of California San Diego Health patients seen between 01/01/2016 – 03/07/2019 with at least 6 months of medication history, excluding pregnant women, patients under 18, and prisoners. Three models (Logistic Regression, Random Forest, and Ensemble) were constructed using hyper-parameters selected through 10-fold cross-validation. Model performance was measured by the Area Under the Receiver Operating Characteristic Curve (AUROC). The model coefficients' odds ratios and p-values were calculated for the Logistic Regression model, as were Gini indices for Random Forest. Decision boundary analysis was conducted using pair-wise false positive and false negative cases each model would predict at a specific threshold.

Results. Logistic Regression, Random Forest, and Ensemble models yielded test AUROCs of 0.839, 0.851, and 0.866, respectively. Significant covariates that may affect risk include age, immuno-compromised treatments, past antibiotic uses, and some medications for the gastrointestinal tract.

Conclusions. The models achieve high discrimination performance (AUROC > 0.83). There is a general consensus among different analysis approaches regarding predictors that impact patients' chances of having a positive test, which may influence Clostridioides difficile risk, including features clinically proven to increase susceptibility. These human-interpretable models can help distinguish significant predictors that affect a patient's chance of testing positive, which may influence their Clostridioides difficile risk.

5.2 Introduction

5.2.1 *Clostridioides Difficile* Infection and Difficulty in Diagnosis

Clostridioides difficile infection (CDI) is caused by the *Clostridioides difficile* (*C. diff*) bacterium and may lead to colitis, pseudomembranous colitis, life-threatening diarrhea, and sepsis,[141,142] especially in older, antibiotic-treated patients in long-term care.[143] The Centers for Disease Control and Prevention (CDC) deems CDI a major health threat;[144] in 2017, there were an estimated 223900 cases among US hospitals, with 12800 deaths and an healthcare-associated (HA) cost of $1billion.[145] In addition to critical public health concerns, failure to meet the infection-specific standardized infection ratio (SIR)[146] for HA-CDI as set by CDC can damage the reputations of healthcare facilities and bring about consequential financial implications.[147] For example, the 2015-2017 HA-CDI rate at University of California San Diego (UCSD) Health was significantly worse than the 2015 national baseline SIR.[148] It is a crucial goal for institutions to reduce HA-CDI so as to assure quality of care, avoid financial penalties and provide superior patient outcomes.

As *C. diff* bacteria are highly resistant to standard hospital decontamination chemicals, highly capable to survive on surfaces disinfected with standard agents, and can form transmissible spores,[149-151] it is appropriate to consider a proactive approach that is still respectful of current guidelines. One organization's experience was that admission screening on almost all patients prevented up to 62% of expected cases, and resulting a gradual decrease of infections over time.[152] From a purely administrative standpoint, early screening may help distinguish true HA-CDI from community-onset incidents, thus improving the accuracy of CDC surveillance efforts, and easing performance stress on healthcare facilities. However, current guidelines do not recommend blanket testing at admission due to fear of overdiagnosis, and subsequently unnecessary antibiotic

treatments that may lead to resistance;[153] rather, testing is administered at symptom onset.[154] This leads to tension between accurate diagnosis and timely intervention, given that carriers (those with colonization) who are asymptomatic at admission have higher risks of progressing to CDI,[155] and that they may shed resistant spores and contribute to outbreaks.[156,157] This practical need calls for innovative, data-driven screening methods that do not require upfront bio-sample testing, while still offering physicians with discriminative insights.

5.2.2 Current Machine Learning Models to Predict CDI and Their Drawbacks

To aid screening efforts, researchers have built stratification models to classify potential *C. diff* positive cases. Most models were built on data with (i) a modest number of patients (about 8,000 to 36,000 patients[158,159]), (ii) several thousand covariates (about 1,800 to 5,000 features, among those are binary encoding of variables not necessarily common among all patients[160]) leading to difficulty in human interpretation of artificially-inflated, high-dimensional vectors, and/or (iii) predictors that bias towards a narrow window of near-current, or current admission data (e.g., antibiotic use within 30 days prior to testing[161]) that may fail to capture the accumulated effect of microbiota-alteration medications over time.[162]

Due to these limitations, we sought to construct CDI predict models that address (i) large-scale sample size, by using a significantly large number of patients (i.e., 157,493 patients) from a real-world institution such as UCSD Health, (ii) high discrimination and interpretable results, employing a reasonable number of core demographics and medical history covariates (i.e., 104 features) that (iii) better reflect each patient's health history over a longitudinal time span (i.e., 3 years). By predicting positive test results which may differ from a true CDI diagnosis (i.e., it may present colonization), allowing for expert discretion from the care staff, our goal is to add to the physicians' decision-support toolbox to use with other testing strategies, finding a compromise between universal testing for asymptomatic carriers and waiting to test until the presence of severe symptoms.

5.2.3 Objective

Specifically, our predictive task is to predict the first event of a positive *C. diff* lab test given demographic and medication history up to one calendar day before the ordering date of the first positive test (**Figure 5.1**). In most cases, this meant at least two calendar days before the result would be known, allowing physicians to proactively care for potential cases before their actual test confirmations. As adult carriers of toxicogenic *C. diff* are six times more likely to develop infections,[155] this early prediction of colonization may make significant contributions to the fight against CDI.

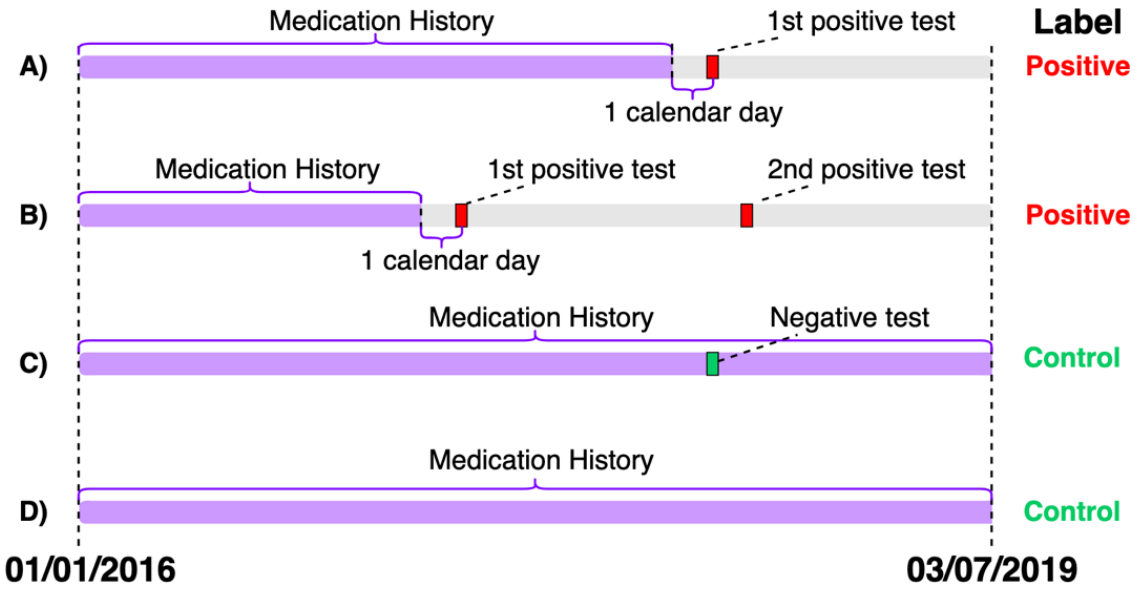


Figure 5.1: Prediction task, illustrating examples of *Clostridioides difficile* Infection (CDI) Positive test result labels versus Control ones used in our study. The four cases here are:

A) A patient has a single positive test during the study window, in which case they are labeled as a "Positive" case.

B) A patient has multiple positive tests, the first of which is also used as the prediction target of "Positive".

C) A patient has a negative *C. diff* test and serves as a "Control" label.

D) A patient never has any *C. diff* test and is a "Control" as well.

5.3 Materials and Methods

The overall workflow of our study is shown in **Figure 5.2**. There are three main parts in our workflow: Data, Model Construction, and Evaluation. Each part will be introduced in the following subsections accordingly.

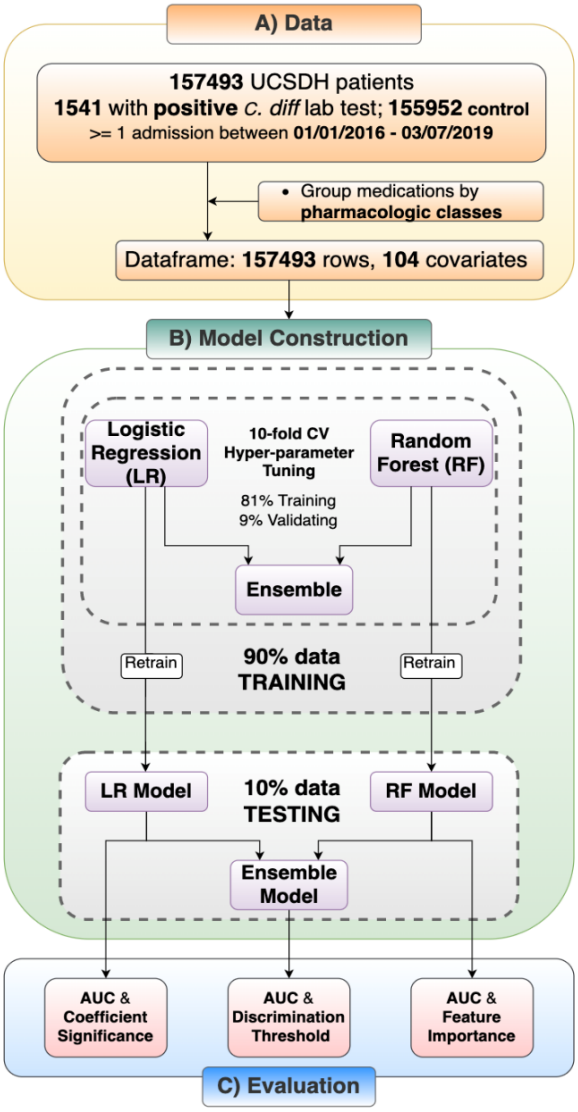


Figure 5.2: Study workflow.

A) **Data**: Demographics and medication history of 157493 patients with at least one admission to UCSD Health between 01/01/2016-03/07/2019 were vectorized.

B) **Model Construction**: Three algorithms (Logistic Regression, Random Forest, and Ensemble) were constructed.

C) **Evaluation**: The corresponding predictive performances of the models were recorded, and feature analysis carried out.

5.3.1 Data

The overall data preparing process is shown in **Figure 5.2.A**. We obtained data of 157,493 patients admitted between 01/01/2016 and 03/07/2019 (date approved by IRB# 190457CX) from the UCSD Health System. Our inclusion criteria included patients with at least one admission to UCSD Health facilities within the study window, whose data included at least six months of medication history. Our exclusion criteria included vulnerable subjects such as patients under the age of 18, pregnant women, and prisoners, as well as those who did not have age information. Among included patients, there were 1,541 (1%) CDI Positive test result labels defined in **Figure 5.1**.

We chose a core number of covariates that encompass demographic information and medication history, which naturally reflect a patient's pathophysiology and health development over time. The dataset contained 104 covariates, including (a) demographic details of race, age, and gender (10 demographic predictors), and (b) past medication history in the form of tabulated instances used within the time frame (94 medication predictors). Specifically, medications were grouped from 4,420 individual names to 94 respective pharmacologic classes, to ensure computational feasibility, reduce hyper-dimensional imputation, and prioritize weight interpretation of features while still maintaining crucial information about a patient's health conditions and progress. The pharmacologic class of a drug refers to the grouping of medications with active ingredients that are similar based on a combination of any three attributes: their mechanism of action, their physiologic effect, and their chemical structure.[163]

5.3.2 Model Construction

The overall modeling and hyper-parameter tuning process is illustrated in **Figure 5.2.B**. We adopted two machine learning algorithms, multivariate Logistic Regression (LR) and Random Forest (RF), to build the models. The two algorithms were chosen to prioritize human interpretability, helping physicians identify features of interest in a straightforward manner. Additionally, we constructed an ensemble model (Ensemble) of LR and RF, to increase the discrimination capacity.[164,165] That is, for each patient, the ensemble model averaged the two prediction scores yielded from the LR and RF models and used this new average as the patient's final predicted score for CDI positive test result. To tune the hyper-parameters of the LR and RF models, we shuffled and held out 10% of the dataset for testing purposes (**Supplemental Figure 5.1.A**). Tuning of hyper-parameters for both algorithms was done via grid search using the 10-fold cross-validation procedure on the remaining 90% data; at each fold, we trained on 81% and validated on 9% of data (**Supplemental Figure 5.1.B**). With the selected hyper-parameters (details in **Supplemental Table 5.1**), two models (LR and RF) were retrained on 90% of data (excluding the hold-out test set), and the performance of our models are evaluated on the hold-out 10% of data (**Supplemental Figure 5.1.C**).

For implementation, we employed the Scikit-learn framework[166] for computations. We also adopted libraries for visualization (Matplotlib)[167] and statistical analysis (SpiCy).[168] We used the Python programming language. The coding task was done exclusively within a secured, HIPAA-compliant virtual environment.

5.3.3 Evaluation

The overall evaluation process is shown in **Figure 5.2.C**. For model performance, we used the Area Under the Receiver Operating Characteristic Curve (AUROC) [169] as our evaluation metric. We reported two sets of AUROCs for each of the three algorithms (LR, RF and Ensemble). (1) *Cross-Validation AUROC*. We averaged the AUROCs of each algorithm's respective 10 models that were constructed in the cross-validation step (that is, models that were built on 81% of training data using the selected hyper-parameters, then tested on their respective folds of 9% data). (2) *Test AUROC*. We recorded the AUROC of each algorithm's *final model* that was built on 90% of data and tested on the 10% hold-out set.

Additionally, we conducted exploratory data analysis to understand the demographic characteristics of the study population. Afterwards, on the LR final model, we performed coefficient significance analysis to record the magnitudes and corresponding odds ratios (which were mathematically derived from and thus representative of the coefficient magnitudes), and p-values. For this coefficient significance assessment, we first conducted a univariate analysis, then a multivariate analysis. The initial step of univariate analysis was simply to confirm the overlapping of magnitude and statistical significance of each predictor with the results seen in the main multivariate approach, if any, other than to exclude predictors with high *P-value*s from being considered in the multivariate analysis step. In addition, as it has been shown that correlations among classes of drugs may contribute to certain changes in a patient's gut health[170,171] and enhance toxin production in some strains,[172] and thus may affect the chance of CDI in that patient, we did not test for covariate independence. On the RF model, we recorded the Gini importance index [173] of each feature, to capture variables that may hold more weights in affecting the risk of CDI.[174] We also conducted a decision boundary analysis to reevaluate the conventional

discrimination threshold of 0.5, taking into consideration that physicians may want to be more proactive and assertive when it comes to weighing the cost of a false positive versus that of a false negative prediction. For the three final models, we used the shift in predictions as this threshold changes between 0 and 1 at intervals of 0.01, by recording the pairwise numbers of false positive and false negative cases predicted at each threshold. This was for easy visualization of such trade-offs as compared to the ROC (Receiver Operating Characteristic) curve.

5.4 Results

5.4.1 Demographic Characteristics

The distributions of gender (female versus non-female, which may include unidentified genders) between the Positive and Control groups did not seem to have a stark difference. Meanwhile, the proportion of patients in the Positive group that were White was statistically higher than that of the Control group, and so was the mean age of the Positive group (58.43) versus the Control group (53.59). With regards to their medical histories, on average, patients in the Positive group had more medication units per patient. **Table 5.1** summarizes the patient characteristics of our dataset.

Table 5.1: Demographic backgrounds of the two Positive and Control groups. Asterisks ("*") indicate Statistically different at $p < 0.001$.

Patient Characteristics	Positive	Control
Number of cases (N)	1,541	155,952
Gender female vs non-female (n)	734 (47.63%)	77,463 (49.67%)
Race White vs non-White (n)*	923 (59.9%)	87,130 (55.9%)
Mean age (mean, standard deviation)*	58.43 (17.23)	53.59 (18.99)
Median age	60.2	54.93
Number of medication units (mean, standard deviation)*	84.95 (107.3)	28.77(57.84)

5.4.2 Model Performance

For the *Cross-Validation AUROC*, the LR algorithm yielded an average AUROC of 0.793 (95% Confidence Interval (CI) = (0.763, 0.823)); the RF algorithm's average AUROC was 0.833 (95% CI = (0.805, 0.861)); and the Ensemble' average AUROC was 0.828 (95% CI = (0.802, 0.854)). For the *Test AUROC* at the step of testing on the hold-out set, the final LR and RF models achieved AUROCs of 0.839 and 0.851, respectively. The final Ensemble model that combined the decisions of LR and RF algorithms yielded an AUROC of 0.866 (**Figure 5.3**). In short, all our three constructed models demonstrated strong predictive powers, as shown by their AUROC scores along either construction step.

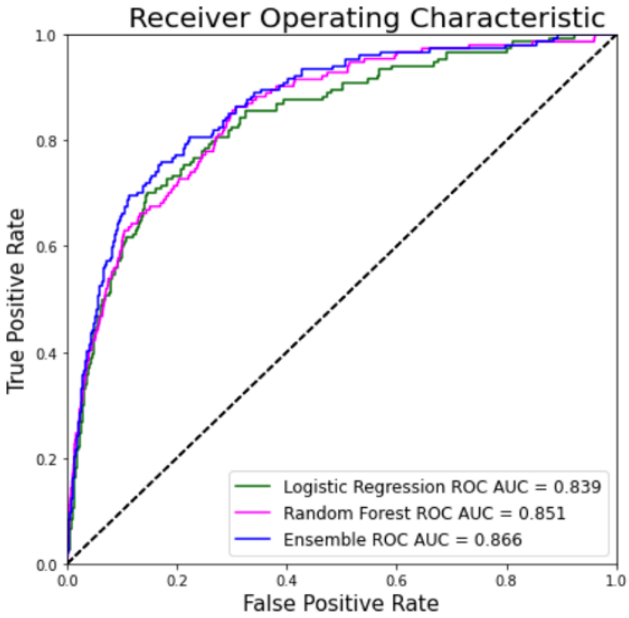


Figure 5.3: AUROC curves of three final models: Logistic Regression, Random Forest, and Ensemble.

5.4.3 Feature Analysis

For feature analysis, **Table 5.2** shows a list of top 20 features as ranked by the lowest p-values ($p < 0.0001$), and their corresponding LR odds ratios in the multivariate analysis step (**Table 5.2.A**). Similarly, **Table 5.2.B** shows 20 features as ranked by the highest Gini indices. See **Supplemental Table 5.2** for the complete list of features, which entails their corresponding AUROCs, p-values and odds ratios in the LR univariate analysis step; their p-values and odds ratios in the LR multivariate analysis step; and the ranking of their corresponding Gini indices.

Table 5.2: Results of feature analysis on the Logistic Regression and Random Forest models.

A) Top 20 features with $p < 0.001$ in LR multivariate analysis ranked by the lowest *P-value*s.

B) Top 20 features as ranked by their RF Gini indices.

A) Logistic Regression Multivariable Analysis		**B) Random Forest Feature Importance**	
Feature	Odds Ratio	Feature	Gini Index
Age	1.0144	Minerals & electrolytes	0.1507
Misc. anti-infectives	1.2558	Misc. anti-infectives	0.1493
Ophthalmic	0.8889	Unassigned group	0.0686
Anticoagulants	0.9377	Antiemetics	0.0396
Misc. GI	1.1577	Analgesics - opioids	0.0369
Fluoroquinolones	1.2250	Diuretics	0.0319
Tetracyclines	0.7735	Age	0.0295
Analgesics - opioids	1.0242	Anticoagulants	0.0261
Local anesthetics - parenteral	1.0816	Local anesthetics - parenteral	0.0256
Minerals & electrolytes	1.0265	Fluoroquinolones	0.0190
Toxoids	0.5740	Assorted classes	0.0190
Laxatives	0.9268	Misc. GI	0.0179
Antidiarrheals	2.3529	Antihistamines	0.0172
Anti-rheumatic	0.9299	Penicillins	0.0171
Gout	1.2108	Corticosteroids	0.0164
Antineoplastics	0.9684	Ulcer drugs	0.0158
Unassigned group	1.0391	Hematopoietic agents	0.0149
Penicillins	1.1491	Misc. Hematological	0.0147
Diagnostic products	0.9326	Antineoplastics	0.0144
Other or mixed races	0.8294	Analgesics - non-opioids	0.0142

5.4.4 Decision Boundary Analysis

The trade-off curves (**Figure 5.4**) show the effects as the threshold changes between 0.4 and 0.6 at intervals of 0.01, for each of the models LR (**Figure 5.4.A**), RF (**Figure 5.4.B**) and Ensemble (**Figure 5.4.C**). Each point on the curve illustrates the paired numbers of false negatives and false positives that the Ensemble model would have predicted on the hold-out set at that threshold. See **Supplemental Figure 5.2** for the full trade-off curve with decision boundaries ranging from 0 to 1.

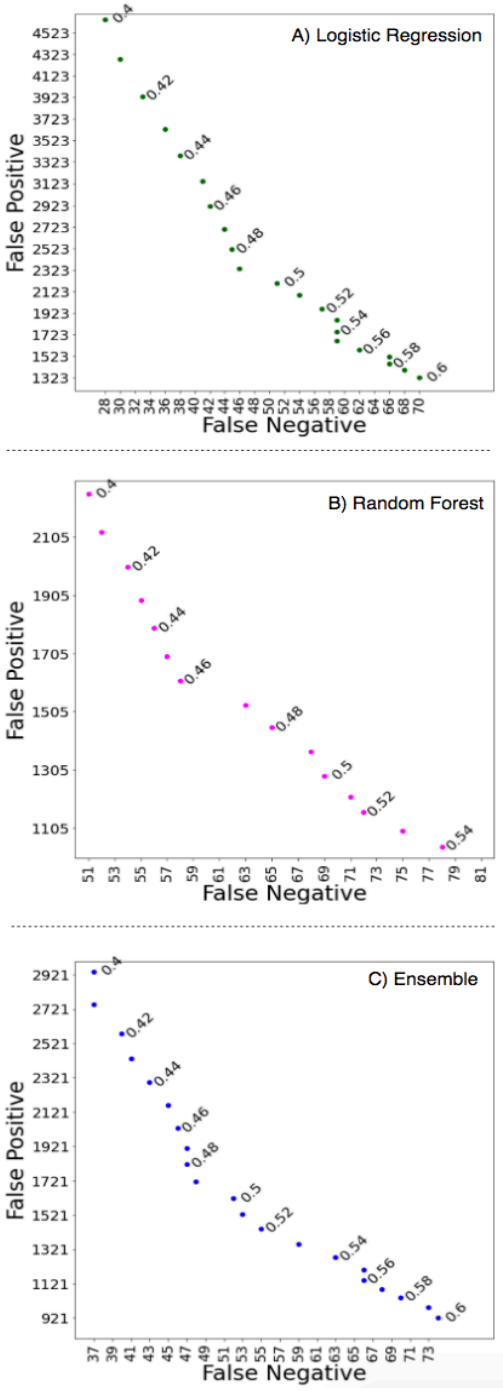


Figure 5.4: The trade-off between the number of false negatives and false positive cases as the discrimination threshold changes between 0.4 and 0.6 for each algorithm.

A) Logistic Regression.

B) Random Forest.

C) Ensemble.

5.5 Discussion

5.5.1 Findings

In general, the high AUROC scores of three models that persisted in both the cross-validation and the testing phase imply that they can be adopted to support the prediction of positive CDI test results. During cross-validation, the average AUROC of the RF algorithm was highest among the three models, while during testing the Ensemble algorithm outperformed the other two. In general, they all show relatively high discrimination results (AUROC between 0.793 and 0.866). These AUROC scores are higher than those obtained from models built on smaller sample sizes and/or with more covariates (0.75 – 0.82 of AUROC in existing studies[158,160]).

Among covariates with a low p-value in the LR final model, there are clinically seen factors that may affect a patient's CDI risk such as age,[175] historic use of drugs in the Anti-infective class (Penicillins, Fluoroquinolones),[176-178] implied immune-compromise conditions through cancer medications and their subsequent side effects of diarrhea resulting in more testing,[179] and indications of past gastrointestinal (GI) issues (Misc. GI). This agreement is again repeated in the RF model, shown through the relatively similar ranking of their Gini indices. The consistency with regards to which feature may have a higher predictive power among such approaches suggests that the models are clinically useful for physicians. Moreover, the high impact of antibiotic use is again confirmed, which may help to caution physicians against over-prescribing of antibiotics for susceptible patients. A study finds that reducing patient's susceptibility to CDI, with which antibiotic treatments can increase risk of colonization and subsequent progression to disease,[180] is more effective in combating CDI as compared to lowering the transmission rate.[181]

The discrimination threshold analysis curves, on the other hand, render a visually intuitive tool to aid physicians in deciding their own level of precaution. There are points that can yield the

same number of false negatives while reducing false positive cases as compared to their neighbors, such as the threshold of 0.55 for Logistic Regression, or 0.48 and 0.56 for the Ensemble model. These trade-off curves may help physicians in deciding the level of precaution needed in diagnosing and treating potentially at-risk patients, a lesson transferable to other serious infections that are similarly difficult to contain after outbreaks, and where any false negative case may cause great, unintended harm. It is known that in terms of healthcare cost, a CDI-positive inpatient with health plan incurs about $21,000 more in treatment charge than an inpatient without CDI.[182] This number is even higher when the CDI is recurrent.[183] At the same time, the cost of a CDI stool test ranges from $15 to $128 (pricing current as of 2021).[184] Therefore, a threshold-adjusted model may provide physicians with discretionary insights to decide between false negatives and false positives, and thus balance between over-treatment and under-diagnosing, especially when coupled with experiences and real-life knowledge of previous cases in their practices.

From a practical viewpoint, the predictions may help guide clinical workflow decisions, such as the utilization of preemptive decontamination methods on both organic and inanimate surfaces, and better tracing of patient movement within the care facility, should a patient be considered of higher risk. For example, a recent study traced CDI back to a single CT scanner in a large academic hospital,[185] highlighting the need for preventative measures even without obvious signs of spread.

5.5.2 Limitations

The limitations of our study include:

1. *Selection of algorithms and model calibration.* To encourage ease of intuitive human interpretability, we used LR, RF and Ensemble which are well-known highly interpretable predictive models. We are yet to utilize other newer and more complicated ML algorithms such as deep learning, nor to calibrate the models.
2. *Potential feature codependency*. We did not investigate possible intercorrelations among pharmacologic class predictors in our models. This may need further analysis to see how specific interactions affect CDI risk.
3. *In-practice deployment*. We did use real patient data to build our models, while deploying our models in a true clinical workflow requires more investigation. Future work may be needed to validate whether our predictions would apply to high-risk patients who would conventionally not be tested by the clinical teams.
4. *Distinction between colonization and infection*. The models predict positive test results, both true infection and colonization with *C. diff*. While colonization does have a strong positive association with true infection, this relationship is not perfect.
5. *Single-center development and evaluation of the predictive models*. The performance of the models in other institutions, or other countries, was not evaluated.

5.6 Chapter Conclusion

Our machine learning models using large-scale, longitudinal patients' medication history and demographic backgrounds can predict the first positive CDI test results with high performance. Our models are built on a large dataset of 157493 real-world UCSD Health patients and capture the underlying demographic and medical information of each patient's health over a 3-year span time window, by a reasonable number of 104 covariates. The models achieve strong AUROCs between 0.839-0.866, and their face validity is highlighted as they find clinically-attested factors that may increase a patient's risk for CDI, such as age, antibiotic uses, cancer and/or GI-related conditions. Their discrimination threshold analysis aids physicians in deciding their own level of precaution. This is a transferable lesson to other infectious diseases such as Coronavirus Disease 2019 (COVID-19), as it considers the starkly different weights of false negatives and false positives. Specifically, as both CDI and COVID-19 tend to affect people with pre-existing conditions and/or comorbidity, early screening using our developed predictive models may help prevent disease outbreaks and mitigate severe consequences beyond the initial infections. This in turn could improve patient care, limit healthcare-associated cost, and boost hospital performance with regards to meeting CDC goals.

5.7 Aim 2 Statement of Impact

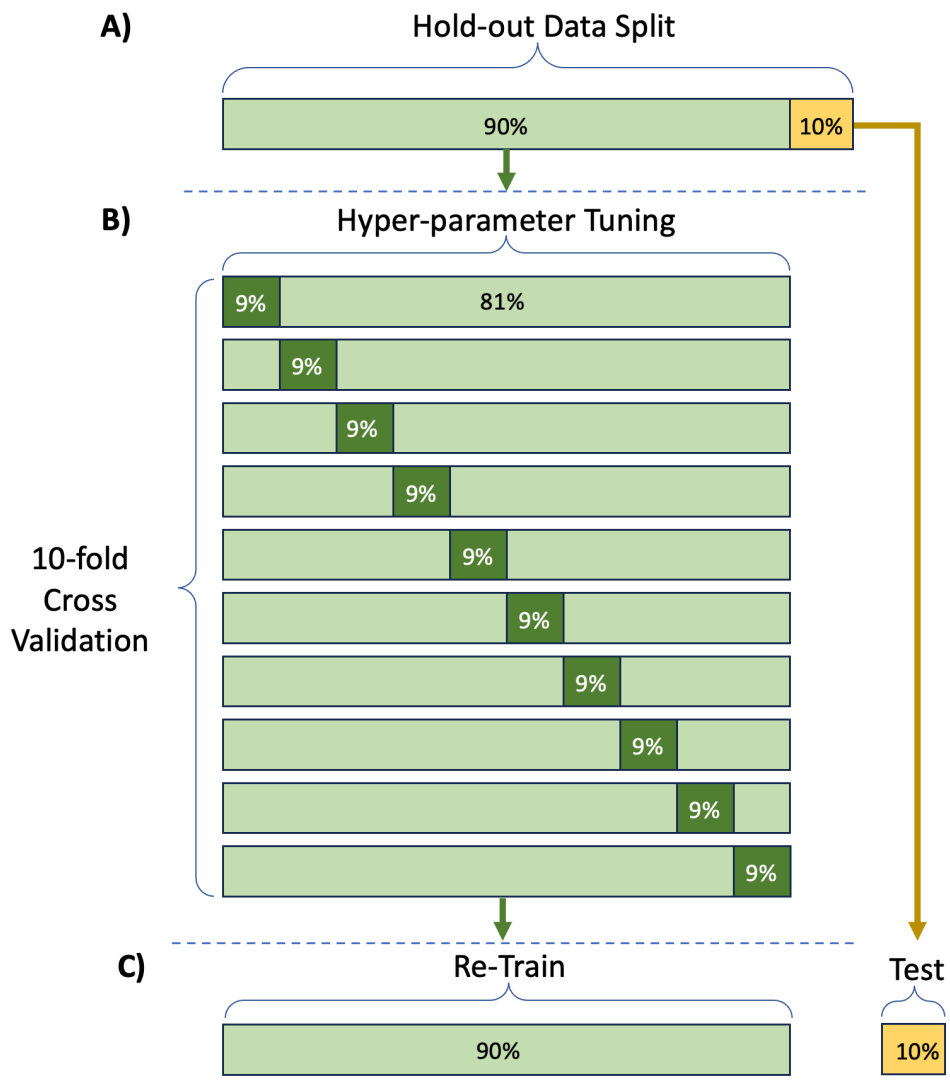
As infections may affect a community larger than the infected individuals, and that the infection may be brought on by factors beyond one's personal control emphasize the need for proactive prevention efforts, among which ML predictive analytics can be a useful tool. In the specific case of CDI, we have successfully used a selective part of real-life patients' EHR data to construct ML models to predict CDI risks with high predictive power and human-friendly interpretability, showcasing the concrete, meaningful use of EHR in supporting the advance of clinical processes.

5.8 Acknowledgements

Chapter 5, in part, has been submitted for publication of the material as it may appear in Heliyon, 2024. Pham, Anh; El-Kareh, Robert; Myers, Frank; Ohno-Machado, Lucila; Kuo, Tsung-Ting, Elsevier, 2024. The dissertation author was the primary researcher and author of this paper.

Supplemental Table 5.1: The range of values considered for each algorithm-specific parameters, and their chosen values to build the final Logistic Regression and Random Forest models after 10-fold cross validation.

Algorithm	Parameter	Values Considered	Value Chosen
Logistic Regression	Solver	lbfgs, liblinear, newton-cholesky	liblinear
	Maximum number of iterations	10, 100, 1000	100
	Regularization term	1, 10, 100, 1000	1
Random Forest	Number of trees	10, 100, 1000	100
	Tree depth	5, 10	10



Supplemental Figure 5.1: The model construction process for Logistic Regression (LR) or Random Forest (RF).

A) **Hold-out Data Split**. The original dataset was divided into a 90-10 split, and the 10% subset was reserved as the hold-out set for testing purposes. This step was done once and was common for both algorithms.

B) **Hyper-parameter Tuning**. Hyper-parameter selection for each algorithm was done through the 10-fold Cross Validation procedure. At each fold, a different subset containing 9% of original data was used as the validation set, and the remaining 81% data became the training material.

C) **Re-train/Test**. After hyper-parameters were selected, each algorithm was re-trained using the full 90% training data. These final models would then get tested on the hold-out set.

Supplemental List of Predictive Features

In total, there are 104 predictors, with 10 demographic attributes and 94 pharmacological classes. The classification of drugs into pharmacological classes was provided by the Electronic Health Record (EHR) platform of the UCSD Health System. In the case that a medication unit for a particular patient did not come with a pre-assigned pharmacological class, imputation of missing data was done with a 60-40 rule: if the active ingredient in the medication unit had been identified as pharmacologic class *C* for 60% of the time for other patients, that unit would be counted as a pharmacologic class *C*. This was common among drugs that were dose-specified, for example, "Neomycin-Polymyxin irrigant 1 ML in 1000 ML." If the 60-40 rule could not be satisfied, the unit would be classified as an "Unassigned group." No imputation was needed for demographic covariates, which were processed with one-hot-coding and drop-one techniques to prevent perfect collinearity among binary variables.

Supplemental Table 5.2: The complete list of 104 predictors and their respective AUROC metrics, *P* values, and odds ratios in the Logistic Regression (LR) univariate analysis step; *P* values and odds ratios in the LR multivariate analysis step; ranking and magnitude of the Random Forest (RF) Gini importance indices. Features are sorted in order of increasing *P* values in the multivariate coefficient analysis.

Feature	Logistic Regression					Random Forest	
	Univariate Model			Multivariate Model		GINI Importance Index	
	ROC AUC	Coefficient		Coefficient			
		P value	Odds Ratio	*P* value	Odds Ratio	Ranking	Magnitude
Age	0.6020	<.001	1.0141	<.001	1.0144	7	0.0295
Misc. antiinfectives	0.7517	<.001	1.5639	<.001	1.2558	2	0.1493
Ophthalmic	0.5169	<.001	0.9326	<.001	0.8889	35	0.0071
Anticoagulants	0.6254	<.001	1.0680	<.001	0.9377	8	0.0261
Misc. gi	0.6241	<.001	1.3513	<.001	1.1577	12	0.0179
Fluoroquinolones	0.6079	<.001	1.6653	<.001	1.2250	10	0.0190
Tetracyclines	0.5053	<.001	1.1611	<.001	0.7735	49	0.0033
Analgesics - opioids	0.6844	<.001	1.0720	<.001	1.0242	5	0.0369
Local anesthetics-parenteral	0.6457	<.001	1.2320	<.001	1.0816	9	0.0256
Minerals & electrolytes	0.7969	<.001	1.0666	<.001	1.0265	1	0.1507
Toxoids	0.5090	<.001	0.7473	<.001	0.5740	72	0.0010
Laxatives	0.5854	<.001	1.1683	<.001	0.9268	23	0.0106
Antidiarrheals	0.5427	<.001	1.8608	<.001	1.3529	21	0.0138
Anti-rheumatic	0.5132	<.001	1.0162	<.001	0.9299	22	0.0107
Gout	0.5260	<.001	1.3046	<.001	1.2108	57	0.0023
Antineoplastics	0.5641	<.001	1.0425	<.001	0.9684	19	0.0144
Unassigned group	0.6498	<.001	1.1359	<.001	1.0391	3	0.0686
Penicillins	0.6259	<.001	1.6828	<.001	1.1491	14	0.0171
Diagnostic products	0.5494	<.001	1.1577	<.001	0.9326	27	0.0093
Other Race or Mixed Race	0.5177	<.001	0.9066	<.001	0.8294	55	0.0027
Contraceptives	0.5027	<.001	0.8300	<.001	0.8119	77	0.0008
Vaccines	0.5399	<.001	1.4252	<.001	0.8462	47	0.0036
Cardiovascular	0.4944	<.001	0.9422	<.001	0.8657	63	0.0018
Diuretics	0.6472	<.001	1.1489	<.001	1.0403	6	0.0319
Hypnotics	0.5586	<.001	1.1937	<.001	1.0498	40	0.0051

Supplemental Table 5.2: The complete list of 104 predictors

Feature	Logistic Regression					Random Forest	
	Univariate Model			Multivariate Model		GINI Importance Index	
	ROC AUC	Coefficient		Coefficient			
		P value	Odds Ratio	*P* value	Odds Ratio	Ranking	Magnitude
Age	0.6020	<.001	1.0141	<.001	1.0144	7	0.0295
Undeclared	0.5130	<.001	0.4891	<.001	0.6676	84	0.0006
Androgen-anabolic	0.5036	<.001	0.9803	<.001	0.8881	88	0.0004
Black or African American	0.5036	0.13	1.0309	<.001	0.8145	66	0.0014
Neuromuscular blockers	0.5086	<.001	1.5221	<.001	1.1469	58	0.0021
Hemostatics	0.5043	0.69	1.0140	<.001	0.6070	94	0.0002
Antiparkinsonian	0.4926	0.01	0.9827	<.001	0.9131	75	0.0008
Vitamins	0.5995	<.001	1.4609	<.001	1.0758	26	0.0100
Antidepressants	0.5586	<.001	1.0381	<.001	0.9755	30	0.0084
Antiasthmatic	0.5731	<.001	1.0431	<.001	0.9758	39	0.0055
Antipsychotics	0.5635	<.001	1.1269	<.001	1.0278	36	0.0070
Antianginal agents	0.5119	<.001	1.2053	<.001	1.0937	67	0.0014
Antihyperlipidemic	0.5398	<.001	1.0388	<.001	0.9616	29	0.0085
Antimycobacterial agents	0.4986	0.04	0.9695	<.001	0.8417	95	0.0001
Antidotes	0.5175	<.001	1.3022	<.001	0.8570	70	0.0010
Oxytocics	0.5025	<.001	0.3885	<.001	0.5814	97	0.0001
Misc. Hematological	0.6225	<.001	1.3670	<.001	1.0473	18	0.0147
Cardiotonics	0.5096	<.001	1.2817	<.001	0.8468	81	0.0006
Antiemetics	0.6810	<.001	1.2110	<.001	1.0259	4	0.0396
Progestins	0.5060	0.38	1.0174	<.001	0.8150	91	0.0003
Analgesics - non-opioids	0.6936	<.001	1.1967	<.001	0.9746	20	0.0142
Decongestants	0.5156	<.001	1.0790	<.001	1.0526	48	0.0036
Antihistamines	0.6754	<.001	1.1411	<.001	1.0231	13	0.0172
Cephalosporins	0.5603	<.001	1.4056	<.001	1.0435	38	0.0055
Antianxiety agents	0.5763	<.001	1.1259	<.001	1.0171	25	0.0101
Antimalarial	0.5142	<.001	1.0883	<.001	1.0977	73	0.0010
Antacids	0.5702	<.001	1.7130	<.001	1.0698	28	0.0085
Antihypertensive	0.5481	<.001	1.0781	<.001	0.9783	34	0.0071
Multivitamins	0.5686	<.001	1.6282	<.001	0.9274	46	0.0036

Supplemental Table 5.2: The complete list of 104 predictors

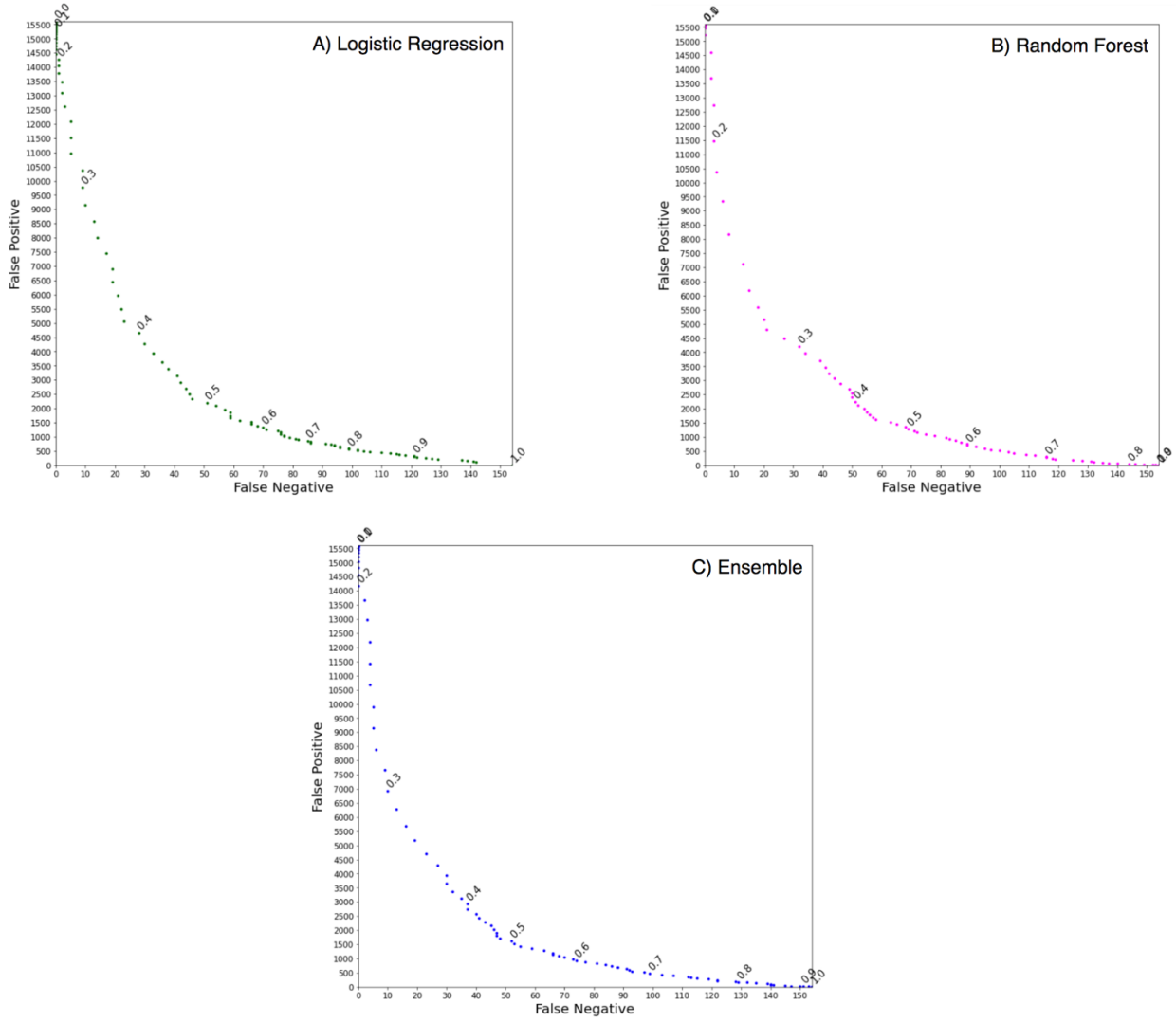
Feature	Logistic Regression					Random Forest	
	Univariate Model			Multivariate Model		GINI Importance Index	
	ROC AUC	Coefficient		Coefficient			
		P value	Odds Ratio	*P* value	Odds Ratio	Ranking	Magnitude
Thyroid	0.4938	<.001	1.0314	<.001	0.9760	43	0.0041
Assorted Classes	0.5861	<.001	1.1524	<.001	1.0188	11	0.0190
Asian	0.5127	<.001	0.8869	<.001	0.9037	69	0.0013
Chemicals	0.5045	<.001	1.8394	<.001	1.3075	87	0.0004
isFemale	0.5340	<.001	0.9413	<.001	0.9468	52	0.0029
Antiseptics & Disinfectants	0.5048	<.001	2.0331	<.001	1.4074	85	0.0005
Digestive aids	0.5193	<.001	1.7320	<.001	1.1365	65	0.0016
Native Hawaiian or Other Pacific Islander	0.5023	0.11	0.8765	<.001	0.6529	96	0.0001
Stimulants	0.5005	<.001	1.0137	<.001	1.0202	82	0.0006
Misc. respiratory	0.4989	<.001	1.1750	<.001	0.9592	89	0.0004
Nutrients	0.5503	<.001	1.1953	<.001	0.9730	24	0.0106
Misc. genitourinary products	0.5416	<.001	1.0995	<.001	0.9716	60	0.0019
Macrolide antibiotics	0.5253	<.001	1.1332	<.001	0.9531	44	0.0040
Hematopoietic agents	0.6117	<.001	1.2281	<.001	1.0146	17	0.0149
Aminoglycosides	0.5115	<.001	1.5101	<.001	0.8762	78	0.0008
Antimyasthenic agents	0.4996	<.001	1.2988	<.001	1.2752	99	0.0001
Antisera	0.5073	<.001	1.0445	<.001	0.9599	93	0.0002
Antiviral	0.5771	<.001	1.0544	<.001	0.9883	45	0.0039
Estrogens	0.4955	<.001	0.9128	<.001	0.9437	86	0.0005
Cough/Cold	0.5322	<.001	1.1386	<.001	0.9646	51	0.0030
Anticonvulsant	0.5651	<.001	1.0663	<.001	0.9900	37	0.0067
Antidiabetic	0.5582	<.001	1.0917	0.003	1.0109	32	0.0081
Corticosteroids	0.6265	<.001	1.1284	0.003	0.9905	15	0.0164
Beta blockers	0.5764	<.001	1.1575	0.01	1.0125	33	0.0072
Anorectal	0.5156	<.001	1.3164	0.02	1.0587	80	0.0007
Dietary products	0.5044	<.001	1.5932	0.04	1.1083	90	0.0003
Misc. endocrine	0.5537	<.001	1.1159	0.06	0.9907	42	0.0041
Mouth & throat (local)	0.5391	<.001	1.5618	0.06	0.9728	61	0.0019

Supplemental Table 5.2: The complete list of 104 predictors

Feature	Logistic Regression					Random Forest	
	Univariate Model			Multivariate Model		GINI Importance Index	
	ROC AUC	Coefficient		Coefficient			
		P value	Odds Ratio	*P* value	Odds Ratio	Ranking	Magnitude
Dietary products	0.5044	<.001	1.5932	0.04	1.1083	90	0.0003
Misc. endocrine	0.5537	<.001	1.1159	0.06	0.9907	42	0.0041
Mouth & throat (local)	0.5391	<.001	1.5618	0.06	0.9728	61	0.0019
Medi-Span Reserved or Unknown(95)	0.5114	<.001	1.2878	0.07	1.0443	74	0.0009
Pressors	0.5532	<.001	1.6672	0.07	1.0349	68	0.0013
Ulcer drugs	0.5995	<.001	1.1808	0.10	0.9933	16	0.0158
Medical devices	0.5286	<.001	1.1540	0.11	0.9881	54	0.0027
Calcium blockers	0.5483	<.001	1.1653	0.15	1.0082	41	0.0045
Migraine products	0.5109	<.001	1.0594	0.16	0.9857	83	0.0006
Skeletal muscle relaxants	0.5232	<.001	1.0742	0.18	1.0080	56	0.0026
Gender Unknown	0.5001	0.33	0.0004	0.20	0.0010	102	0.0000
Urinary antiinfectives	0.5108	<.001	1.2718	0.20	1.0188	64	0.0018
Antiarrhythmic	0.5182	<.001	1.2694	0.24	1.0138	71	0.0010
Biologicals misc	0.5001	0.41	0.0009	0.32	0.0042	101	0.0000
Urinary antispasmodics	0.5027	<.001	1.1182	0.36	1.0127	79	0.0007
Sulfonamides	0.5000	0.36	16.0271	0.37	0.6880	103	0.0000
Anthelmintic	0.5019	<.001	1.4260	0.39	1.0572	92	0.0003
Antifungals	0.5459	<.001	1.4511	0.42	0.9909	53	0.0028
Misc. psychotherapeutic	0.5086	<.001	1.0444	0.46	0.9950	59	0.0019
Dermatological	0.5798	<.001	1.1034	0.48	1.0022	31	0.0083
Amebicides	0.5000	0.65	0.0093	0.57	0.0426	104	0.0000
Pharmaceutical adjuvants	0.4995	<.001	1.5921	0.67	0.9396	100	0.0000
Otic	0.4988	0.13	1.0322	0.68	0.9897	76	0.0008
General anesthetics	0.5140	<.001	1.7210	0.70	0.9921	50	0.0030
American Indian or Alaska Native	0.5018	<.001	1.2849	0.80	1.0256	98	0.0001
Vaginal products	0.4955	<.001	1.0547	0.95	1.0007	62	0.0018

Supplemental Decision Boundary Analysis

The trade-off curves as the threshold changes between 0 and 1 at intervals of 0.01, for each of the models Logistic Regression, Random Forest, and Ensemble.



Supplemental Figure 5.2: The full trade-off between the number of false negatives and false positive cases as the discrimination threshold changes at 0.01 interval between 0 and 1 for each algorithm.

A) Logistic Regression

B) Random Forest

C) Ensemble

Chapter 6 THE DATA PROCESSING REQUEST IN BIOMEDICAL RESEARCH

6.1 Abstract

Objective. Our study aimed to expedite data sharing requests of Limited Data Sets (LDS) through the development of a streamlined platform that may allow distributed, immutable management of the system, provide transparent and intuitive auditing of data access history, and systematically evaluate it on a multi-capacity network setting for meaningful efficiency metrics.

Materials and Methods. We developed a blockchain-based system with six types of smart contracts to automate the LDS sharing process among major stakeholders. Our workflow included metadata initialization, access request processing, and audit-log querying. We evaluated our system on three machines with varying specifications and synthetic data to emulate a real-world scenario. The data employed included ~1,000 researcher requests and ~360,000 log queries.

Results. On average, it took ~2.5 seconds to register and respond to a researcher access request. The average runtime for an audit-log query with non-empty output was ~3 milliseconds. The runtime metrics at each institution showed general trends affiliated with their computational capacity.

Discussion. Our system can reduce the LDS sharing request time from potentially hours to seconds, while enhancing data access transparency in a multi-institutional setting. There were variations in performance across three sites that could be attributed to differences in hardware specifications. However, the performance gains became marginal beyond certain hardware thresholds, pointing to the influence of external factors such as network speeds.

Conclusion. Our blockchain-based system can potentially accelerate clinical research by strengthening the data access process, expediting access and delivery of data links, increasing transparency with clear audit trails, and reinforcing trust in medical data management.

6.2 Introduction

6.2.1 The Cross-institutional Data Requesting and Data Sharing Landscape

In the modern healthcare landscape, clinics and institutions routinely generate vast datasets throughout patients' courses of care.[186,187] As these datasets include direct biometrics of patients, they serve as a rich resource for researchers, especially when studies require insights into the timing and location of health events. This leads to the use of the Limited Data Set (LDS),[188] a category of datasets that includes up to two types of these Protected Health Information (PHI) [189] or Personal Identifiable Information (PII),[190] along with other non-identifiable data. Such constraints on LDS enables researchers to study specific populations without majorly compromising individual patient privacy.[191] Proper management of LDS access contributes to safeguarding individual privacy accordant with the Health Insurance Portability and Accountability Act (HIPAA) Privacy Rules [192] and reinforcing public trust in healthcare research and practices, thereby empowering the robust growth of data-driven biomedical research. The LDS sharing protocol often encompasses proof of signed data use agreements (DUAs),[193] as well as user demonstration of valid data-access credentials, such as possession of relevant biomedical training certificates from the Collaborative Institutional Training Initiative (CITI) program.[194] As mandated by HIPAA, a DUA defines authorized users, delineates permissible actions, restrictions, and exemptions, includes legal provisions against re-identifying or directly contacting data subjects, and may also describe breach reporting protocol.[195] Meanwhile, training certificates ensure that researchers are well-versed in the necessary regulatory standards governing the specific dataset.[194] Given the need of researchers requesting data from organizations under different regulatory regimes,[196,197] establishing an effective, institution-spanning process to verify these credentials is crucial for the continuity and efficiency of biomedical/clinical research.

6.2.2 Manual Data Access Processes

Nonetheless, as depicted in **Figure 6.1.A**, the current data access and delivery process is hindered by manual procedures involving various stakeholders, from researchers who seek data, to the data concierge offices which are responsible for verifying access credentials and facilitating dataset distribution. After gathering information on the access requirements, researchers must first obtain the necessary compliance training certificates and sign the DUA specific to their desired dataset. The documents should then be sent to the concierge office who manually verifies their validity and checks that they have not expired. The verification of these prerequisites at the data concierge office can often be a time-consuming task, especially when the information provided is incorrect or incomplete. This will not only prolong the researcher's wait time but also introduce variability in the verification pace due to human factors, thereby adding non-trivial time costs to the overall research timeline from days to weeks. From the perspective of data concierge staff, the human labor required scales linearly with the number of inquiries and the variety of documents related to each dataset. Moreover, this process must be repeated each time the researcher wishes to access another dataset, even if the required credentials such as CITI training certificates have been previously submitted and are still valid.

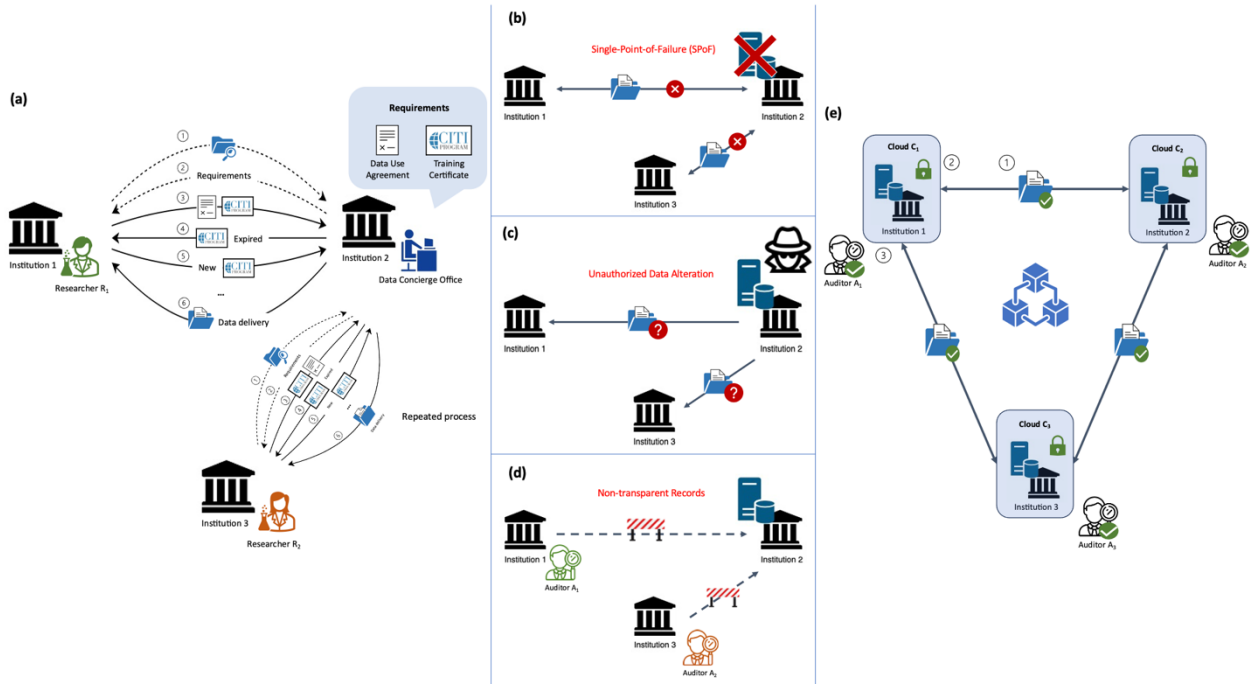


Figure 6.1: Comparison of limited dataset request process between three institutions under different systems.

A) **Traditional email process**. *Step 1.* Researcher R_1 from Institution 1 requests a dataset from Institution 2's data concierge office. *Step 2.* Office informs of requirements: training certificates, data use agreement (DUA). Steps 1 and 2 are skipped if requirements are online (e.g., on an official website). *Step 3.* Researcher R_1 submits certificates and DUA. *Step 4.* Office notes expired certificates. *Step 5.* Researcher R_1 obtains and resubmits new certificates. *Step 6.* Office approves and delivers dataset. Process repeats for subsequent researchers.

B) **Centralized system**. In this example, Institution 2 uses an automated system and serves as the centralized server; upon fulfilling dataset requirements, data is instantly sent to researchers. However, Institution 2 can present a single-point failure leading to data inaccessibility during server downtime.

C) **Another centralized server risk** of unnoticed, unauthorized data changes by administrators, resulting in different datasets for identical requests.

D) **The third centralized server difficulty** for auditors at other institutions to access request history without database permissions since the audit logs are not transparent to all parties.

E) **Decentralized blockchain system**. Each institution has a cloud environment with database storage, offering multiple benefits including: 1) no single-point failure, ensuring constant data accessibility; 2) immutable data and request records on the blockchain, preventing unauthorized changes; and 3) transparent cross-institutional request history queries for auditors.

6.2.3 Centralized Solution for Data Request Management

To address these shortcomings and enhance the overall efficiency of data sharing, it is essential to adopt an automated system to confirm access credentials [198] and consequently approve or deny access requests. A potential and intuitive solution is to employ the conventional model of an electronic, centralized database, which allows all relevant information of datasets, compliance certificates, and signed DUAs to be submitted and stored for future matching, thereby preventing repeated submitting and verification efforts. However, such centralization may pose innate workflow and security challenges: 1) by design, a centralized system is susceptible to the Single-Point-of-Failure (SPoF) risk, where information becomes inaccessible during server downtime (**Figure 6.1.B**).[199-201] 2) It is also vulnerable to unauthorized access or alteration under compromised "admin" privileges, a less prevalent event, yet with detrimental effects that may remain undetected for a long time.[200,202,203] (**Figure 6.1.C**) 3) In an architecture shared among multiple institutions, this is even more challenging as it would require additional efforts to establish transparency with central administrators, so that auditors can independently confirm access history for auditing purposes (**Figure 6.1.D**).[204]

6.2.4 Needs for a Decentralized Data Requesting System

Given these inherent issues of the conventional centralized database, a decentralized system that can function without a centralized repository emerges as the more technically fit solution to prevent: 1) central server outages, 2) data mutability, and 3) process opacity. In particular, the decentralized ledger blockchain has been advocated for use in various healthcare domains [205] such as clinical data management,[206-208] clinical research data logging,[209] secured and unified health records access,[210] genomic data access and gene-drug interaction recording,[211,212] health data compliance and abuse detection,[213] medical image sharing,[214] and credential exchange for data sharing among verified entities in healthcare.[215] Compared to the limitations of a centralized server, blockchain is known for 1) robustness against SPoF; information is broadcast across the network in a peer-to-peer fashion, enabling continuation of interaction even if a network participant (node) ceases to function, thereby eliminating the SPoF risk inherent in a centrally-coordinated schema. Furthermore, blockchain enhances security through 2) its resistance to unauthorized data modifications with the intentional duplication of data at each node, ensuring easy detection of record modification. Lastly, blockchain 3) improves widespread transparency in data sharing; by ensuring that each node holds a copy of the entire data ledger, all information becomes accessible and verifiable by any participant.[200,216] Another advantage of blockchain infrastructure is the integration of smart contracts,[216] which are customizable programs stored and executed on the blockchain that automate credential verification and data distribution. Smart contracts inherit blockchain's core features: decentralization, immutability, and transparency for the *code* (in addition to the *data*). These characteristics of blockchain and smart contracts make them particularly well-suited to enhance automation, accuracy, and decentralization in LDS sharing. **Figure 6.1.E** summarizes the benefits of a decentralized blockchain system.

6.2.5 Related Works

Several blockchain- and smart contract-based proposals have aimed to refine the data request workflow, each with varied focuses and degrees of complexity.[217-221] For example, a past study has demonstrated blockchain's capability to store and retrieve training certificates, but it has yet to address the automated verification of access credentials and the release of access links upon approval.[220] Another study focused on storing dataset metadata in the approval protocol and negotiating DUA terms multilaterally, but has yet to incorporate the credential verification process.[217] Additionally, a different approach used blockchain as an honest service to facilitate access based on user trust and reputation, granting or denying access based on predicted noncompliance risk, and has yet to verify regulatory compliance through formal training certificates and/or DUA.[219] Regarding the traceability of data request history, several studies leveraged blockchain's transparency for "built-in" auditing, displaying all data requesting activities. However, the recorded information was presented as either identifiers [218] or hashed strings,[221] which requires further maneuvers to concretely locate actors and actions within a specific timeframe. Therefore, a domain-specific metadata storage design that allows intuitive filtering and review has yet to be developed. From an architectural standpoint, previous proposals focused on single-site blockchain evaluations [219] or multi-site blockchain implementations where each site had identical hardware specifications.[218] Although one study used computers with varying capacities, it has yet to conclusively determine the impact of this design on system performance.[221] Additionally, while several studies have measured system efficiency,[217] more comprehensive evaluations have yet to be conducted. Thus, there is a need for a non-homogeneous architecture that accommodates configuration heterogeneity and is systematically evaluated for time efficiency, providing a metric that can be more statistically significant.

Therefore, there has yet to be a data-sharing request management system that can: 1) automatically handle key aspects of the LDS sharing mechanism, including verification of access credentials and delivery of data links; 2) allow users to query access activities with transparent ease; and 3) provide comprehensive evaluations in cross-site environments with varied computational configurations.

6.2.6 Objective

We aimed to expedite the sharing request of LDS through 1) the development of a streamlined platform that may allow distributed, immutable management of the system, 2) provide transparent and intuitive auditing of data access history, and 3) systematically evaluate it on a multi-capacity network setting for meaningful efficiency metrics.

6.3 Materials and Methods

6.3.1 Method Overview

A schematic representation of our proposed workflow is illustrated in **Figure 6.2**, with a sample network of three institutions, each deployed within a cloud environment with varying computational settings. From researchers' standpoint, they can request datasets from any institution within the system, and they can expect to receive fast responses to their requests. Specifically, successful requests yield data sharing links, while unsuccessful ones result in the reasons of denial (e.g., absence of a certificate, expired certificate, or lack of a signed DUA). Instead of directly sharing the entire dataset files on-chain, our system employs data sharing links to ensure data privacy, and the protected content of LDS datasets are not directly stored on chain. Additionally, auditors from each institution have the capability to query the data request history across the network (i.e. auditor at Institution 3 can query requests not only from Institution 3 but also from Institution 1 and Institution 2). The following sections will delve deeper into the specifics of our system: **Section 6.3.2** outlines the smart contract architecture; **Section 6.3.3** discusses the system workflows; **Section 6.3.4** details our test data; **Section 6.3.5** describes the system implementation; and **Section 6.3.6** covers the evaluation setting.

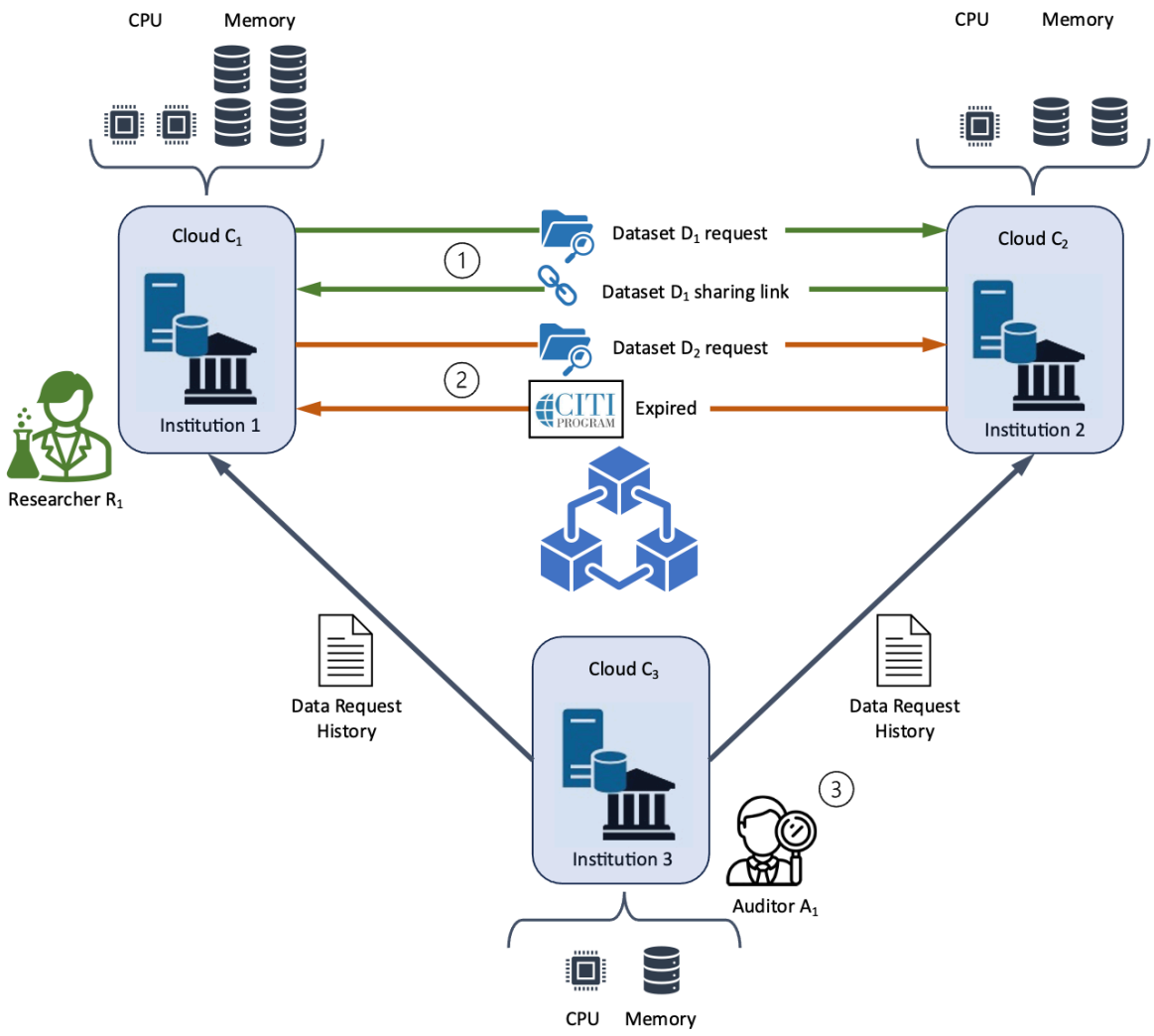


Figure 6.2: Overview of method workflow between three institutions with different computational environments.

1) **Successful Data Request**. Researcher R_1 requests dataset D_1 from Institution 2 and promptly receives a data sharing link.

2) **Unsuccessful Data Request**. Researcher R_1 from Institution 1 requests dataset D_2 from Institution 2 but is immediately notified of expired certificates.

3) **Auditing Process**. An auditor from Institution 3 who wishes to look up request history of datasets from Institution 1 and 2 can query the history at Institution 3.

6.3.2 Smart Contract Architecture

Our framework includes two categories of smart contracts: *stakeholders* and *utility*. In particular, the *stakeholder* contracts include Institution, Researcher Management, and Data Concierge, to represent each critical entity involved in the data access request process. These smart contracts can manage, monitor, and record metadata within the system according to their specific roles. Next, *utility* contracts consist of Log, Connector, and Date Time, which act as supportive utility components within the system. The main architecture governing the interplay of these smart contracts is illustrated in **Figure 6.3**. Further details of each smart contract are provided in **Table 6.1**.

Table 6.1: Detailed description of individual smart contracts.

Category	Smart Contract	Description
Stakeholder	*Institution*	• Manage other types of stakeholder contract • Communicate with Connector and other Institution contracts
	Researcher Management	• Store and update researcher's certificate metadata (email, completed certifications, etc.) • Let researchers submit data access requests
	Data Concierge	• Store and update metadata of datasets and their corresponding DUA • Receive and process access requests
Utility	*Log*	• Store timestamped records of data access requests • Allow querying of all records
	Connector	• Store and pass along addresses of each member contract
	Date Time	• Convert between calendar and Unix time units

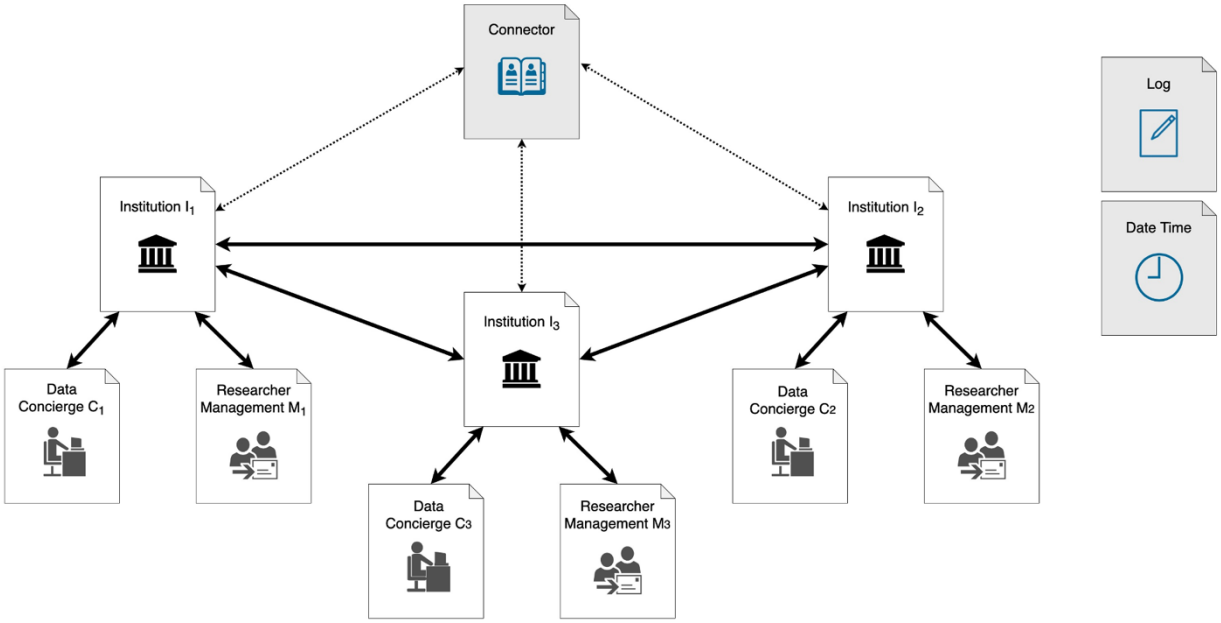


Figure 6.3: Smart contract architecture of three institutions. The system incorporates six types of smart contracts. Each 'Institution' smart contract supervises a 'Data Concierge' and a 'Researcher Management' smart contract. 'Connector', 'Log', and 'Date Time' function as utility smart contracts within this architecture, highlighted in gray.

6.3.3 Workflows

Our system streamlines LDS access requests with three sub-workflows:

1. *Metadata initialization.* Building on the capabilities demonstrated in previous studies where the metadata and PDF files of credential certificates can be stored on-chain, our initialization step extends the data request pipeline by transmitting metadata regarding the dataset into our system through the invocation of the Data Concierge smart contracts (**Table 6.2.A**). All submitted dataset metadata are coupled with their respective DUA. Records of researchers who have signed the DUAs are maintained in the Data Concierge smart contract. Similarly, the Researcher Management contracts are invoked to transmit metadata of certificate types and expiration dates (**Table 6.2.B**). Each certificate is individually imported, mirroring the practice of a researcher individually submitting their newly acquired certificate onto the system.
2. *Researcher access request processing.* Detailed backend interactions involved in the access request and delivery process between any two institutions are shown in **Figure 6.4**. First, a researcher initiates an access request through their institution's Researcher Management smart contract, which then forwards the request to the destination institution's Data Concierge smart contract. The Data Concierge smart contract will automatically verify the validity of training certificates and DUA metadata. Upon successful verification, a data sharing link is provided to the requester. In cases of denial, the system provides specific reasons for the rejection. Additionally, the Data Concierge smart contract would help record every request, along with its outcome and timestamp, to the Log contract, with which auditors can subsequently query. Descriptions of the access request fields are in **Table 6.2**.

3. *Audit-log querying.* Auditors from any institution can retrieve the request history via the log query function of the Log smart contract. For example, an auditor can find records of instances where *Researcher A* asked for *Dataset D* from *Institution 1* during the period between *Time Point M* and *N*. The details of the audit-log query are described in **Table 6.2.D**

Table 6.2: Details of data: **(a)** dataset metadata, **(b)** researcher's certificate metadata, **(c)** researcher access request, and **(d)** audit-log query. "*" is a possible input for any field in the audit-log query.

Type	Field	Description	Example(s)
(a) Dataset Metadata	Dataset Name	Dataset's full name, which is the unique identifier of the dataset	Data Set 21
	Dataset Link	Dataset's data delivery link	https://dataset_repository.com/data_set_21
	Owner Researcher	Dataset owner's email	janedoe@inst2.edu
	Required Certificate	List of required certificates for dataset access	Biomedical Data Only Research, Biomedical Informatics Research
	DUA Number	Unique identifier links to the relevant DUA	DUADS21
	Signed Researchers	List of researchers who had signed the DUA	tskuo@inst1.edu, johndoe@inst3.edu
(b) Researcher's Certificate Metadata	Researcher Name	Researcher's full name	Tsung-Ting Kuo
	Researcher Email	Researcher's email address, which is the unique identifier of the researcher	tskuo@inst1.edu
	Certificate Name	Certificate's full program name	Biomedical Data Only Research
	Certificate Expiration Date	Certificate's expiration date	12/19/2026
(c) Researcher Access Request	Researcher Email	Email address of the researcher who performed the request	tskuo@inst1.edu
	Institution of Dataset	Data-generating healthcare site	Institution 2
	Dataset	Full name of the requested dataset	Data Set 21
	Request Time	Unix timestamp of the request time	1703126751 (December 20, 2023, 18:45:51 PST)
(d) Audit-Log Query	Researcher Email	Email address of the researcher who performed the request	tskuo@inst1.edu
	Institution of Dataset	Data-generating healthcare site	Institution 2
	Dataset	Full name of the requested dataset	Data Set 21
	Start Time	Unix timestamp of query start time	1483257600 (January 01, 2017, 00:00:00 PST)
	End Time	Unix timestamp of query end time	1704009600 (December 31, 2023, 00:00:00 PST)
	Response Type	Type of response of the requests	N (denied request)

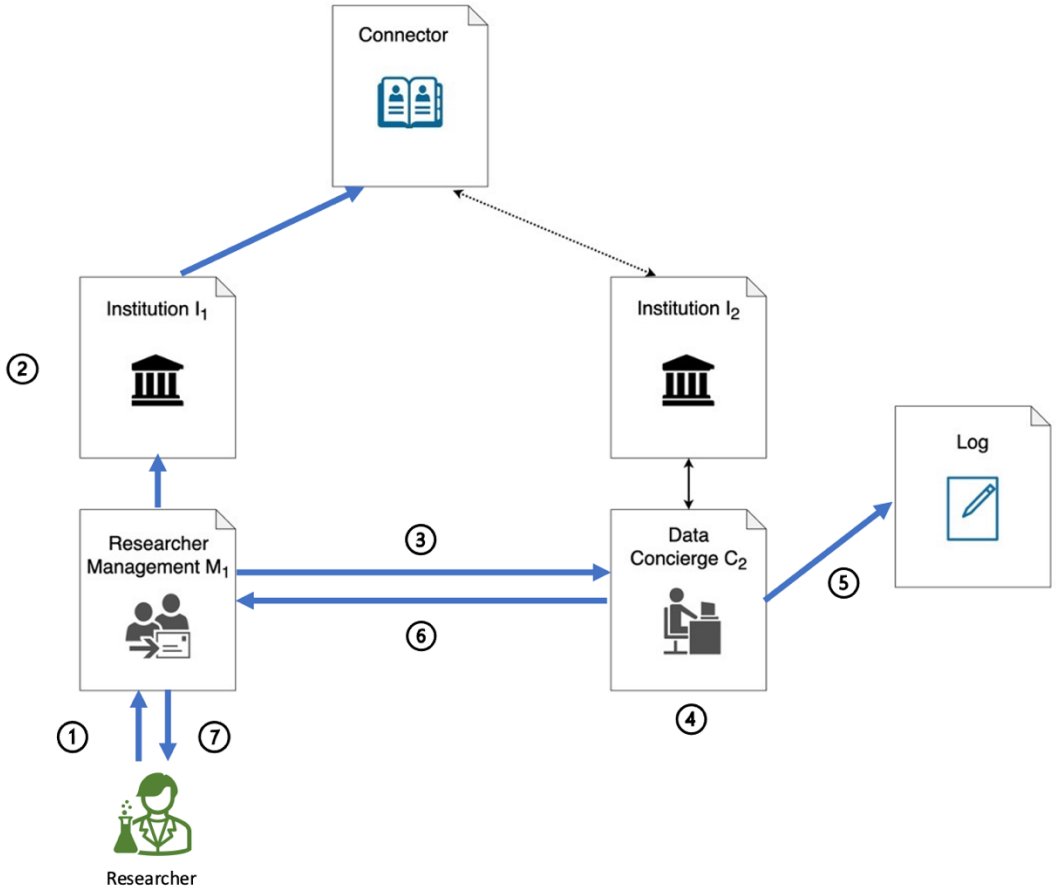


Figure 6.4: Workflow of researcher data request and smart contract interactions between two institutions. A researcher from Institution 1 requests a dataset from Institution 2's data concierge office.

Step 1. Researcher submits a request via Institution 1's Researcher Management 1 smart contract.

Step 2. This contract retrieves addresses of Connector and Institution 2's Data Concierge 2 smart contract.

Step 3. Data request is sent to Data Concierge 2 smart contract.

Step 4. Data Concierge 2 smart contract verifies training certificate and DUA information.

Step 5. Data Concierge 2 smart contract records the request history in the Log smart contract.

Step 6. Decision is relayed back to Researcher Management 1 smart contract.

Step 7. Researcher receives response.

6.3.4 Data

To evaluate our prototype, we generated synthetic data designed to emulate real-life scenarios of researchers requesting data across three institutions; each institution consisted of 20 researchers and 10 datasets, accompanied by the corresponding DUA per dataset. Each dataset may require recipients to possess up to three distinct types of training certificates, and correspondingly, each researcher may hold up to three specific training certificates. Researcher access requests were random matched among all researchers and datasets, regardless of their institutional affiliations. With regards to the auditing functionality, we generated an exhaustive list of all possible query combinations to facilitate thorough testing. An audit-log query includes six inputs: Researcher Email, Institution of Dataset, Dataset, Start Time, End Time, and Response Type (details described in **Table 6.2.D**). Additionally, the use of "*" is a "wild card" parameter in audit-log queries representing all possible inputs for that parameter. We examined every possible combination of log queries, including those that produced results and those that would return an empty value. The specific numbers of the generated data in our experiments are summarized in **Table 6.3**.

Table 6.3: Statistics of the synthetic data.

Institution	**Metadata**		**Researcher Data Request**	**Audit Log Query**
	Number of Researcher	Number of Dataset	Number of Researcher Data Requests	Number of Log Queries
1	20	10	350	363,072
2	20	10	266	
3	20	10	411	
Total	60	30	1,027	

6.3.5 Implementation

Based on prior surveys,[222,223] we chose Ethereum, an open-source blockchain platform with smart contract support, to be the underlying blockchain infrastructure for our prototype due to its robust community support. We selected the Proof-of-Authority (PoA) consensus protocol,[224] designed for private blockchains where only authorized parties or nodes may join the network. The choice of private blockchain aligns with our goal of modeling a multi-node system in an environment with semi-trusted relationships among participants, unlike public blockchains which would expose data to all, including unwarranted parties. For the development of the smart contract, we employed Solidity version 0.8.4, Go Ethereum (Geth) version 1.12.7. and used the Remix Integrated Development Environment (IDE) to develop and test the Solidity smart contract codes. Furthermore, Web3j version 4.5.0,[225] a Java library for Ethereum smart contract interactions, was leveraged for our off-chain Java programming along with Bash scripts. Summary of the implementation is illustrated in **Figure 6.5**.

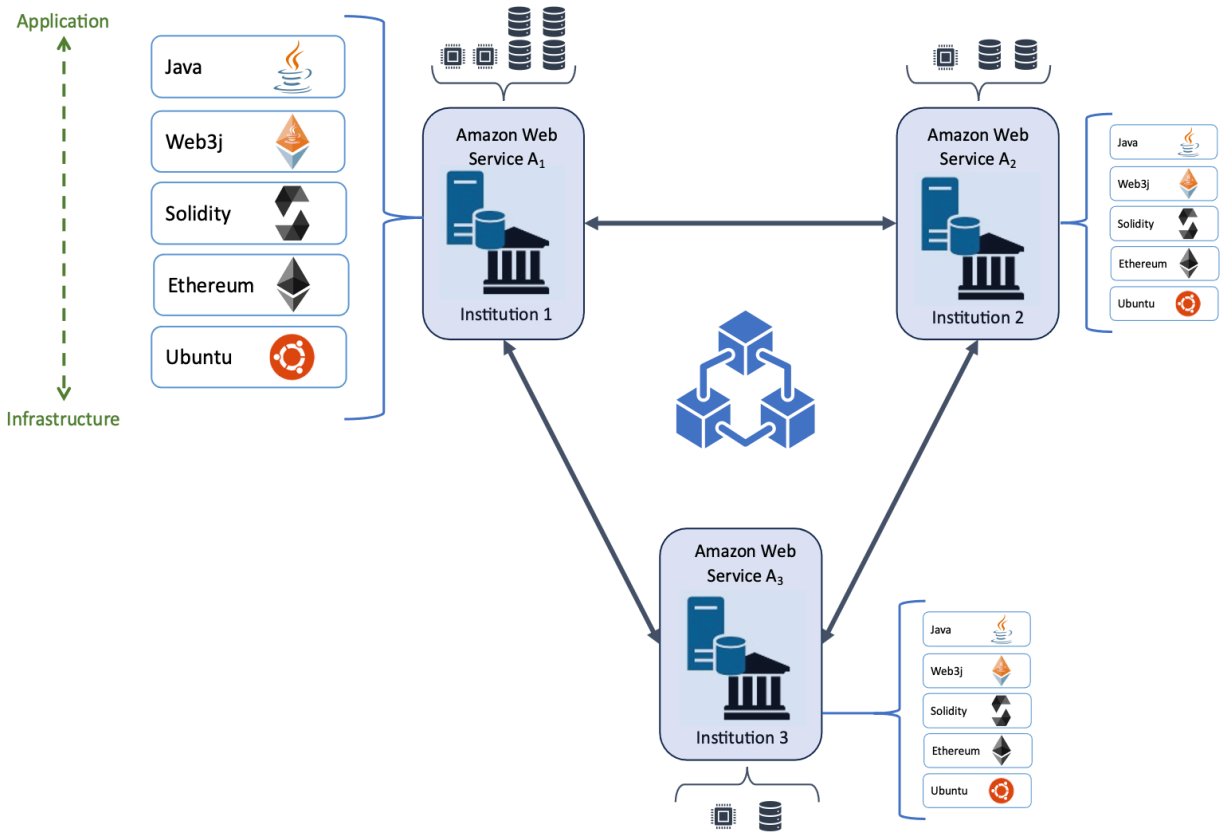


Figure 6.5: Implementation architecture of the system. The system is hosted on a 3-node (i.e., three emulated institutions) blockchain network with three Amazon Web Service (AWS) virtual machines with different computational resources. Same implementation technology is used at every node.

6.3.6 Evaluation Settings

To test the system's feasibility under real-world scenarios when each site may be equipped with varying computational capacity, we evaluated the system on three Amazon Web Service (AWS) Virtual Machines (VMs) with different hardware specifications. The system's performance may be influenced not only by the inherent speed threshold of the blockchain, which establishes a fixed upper limit,[200] but also by the specifications of the system's hardware and network capacities, including factors such as CPU and RAM. The configurations of the VMs were as follows: the large VM (Institution 1) had 4 vCPUs, 16 GB of RAM, and 200 GB of storage; the medium VM (Institution 2) had 2 vCPUs, 8 GB of RAM, and 100 GB of storage; and the small VM (Institution 3) had 2 vCPUs, 4 GB of RAM, and 50 GB of storage. Our performance evaluation metric was time-based, a concrete metric most relevant to clinicians and researchers and focused on several key operations: smart contract deployment, metadata initialization (including researcher's certificate metadata and dataset metadata, as described in **Table 6.2.A** and **Table 6.2.B**, respectively), researcher access request processing (**Table 6.2.C**), and audit-log querying (**Table 6.2.D**). We specifically measured the time of each access request and each audit-log query. To ensure the robustness and statistical significance of our findings, experiments were repeated 30 times. Additionally, we applied paired t-tests to all inter-institutional comparisons, with a significance threshold at 0.05.

6.4 Results

6.4.1 Smart Contract Deployment

The initial deployment of smart contracts was launched from Institution 1, which then shared the addresses of these smart contracts with all participating sites, representing a one-time setup cost for each experiment. The entire duration to deploy 1 Connector, 3 Institution, 3 Researcher Management, 3 Data Concierge, 1 Log, and 1 Date Time contracts, along with the branching of their interconnections, took on average 36.035 seconds with a standard deviation of 2.032 seconds over 30 experiments. On a per-contract basis, the average smart contract deployment time was 2.117 seconds with a standard deviation of 0.123 seconds.

6.4.2 Metadata Initialization

Figure 6.6.A presents the overall metadata import times for the three institutions. The system took a maximum import time of 146.620 seconds at Institution 3 for both dataset and researcher's certificate metadata. Conversely, Institution 1 took the minimum time of 135.265 seconds. Comprehensive details of the data import times for each of the three sites are demonstrated in **Table 6.4.A**.

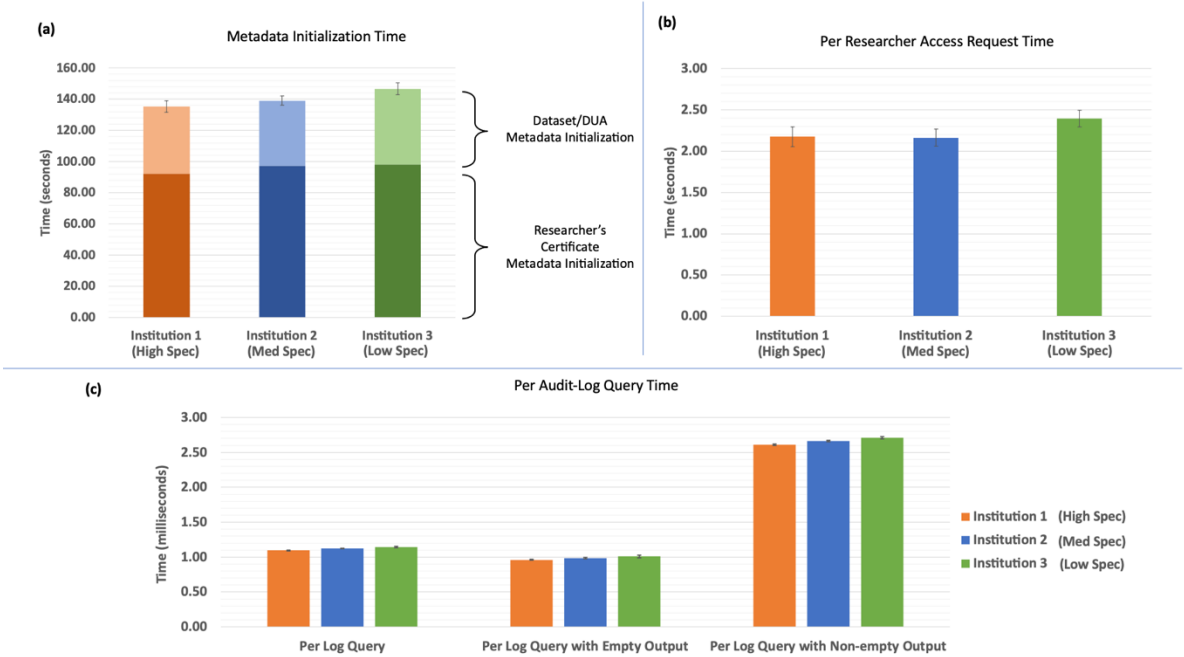


Figure 6.6: Evaluation of system performance across three institutions. The error bars represent the standard deviation for each measured value.

A) Overall data import time.

B) Per researcher data request time.

C) Per log query time.

Table 6.4: Detailed running time results of

A) Metadata Initialization,

B) Researcher Access Requests,

C) Audit-Log Queries

at three institutions. All times (except per query results) are measured in seconds, with standard deviations provided in parentheses.

Statistics		Institution 1	Institution 2	Institution 3
(a) Metadata Initialization	Overall	135.265 (6.393)	139.044 (5.400)	146.620 (5.897)
	Overall researcher's certificate metadata import	92.250 (4.806)	97.161 (4.378)	98.095 (4.719)
	# of certificates	29	34	27
	Per certificate	3.181 (0.166)	2.858 (0.129)	3.633 (0.175)
	Overall dataset metadata import	43.015 (2.342)	41.883 (1.635)	48.525 (2.933)
	# of datasets	10	10	10
	Per dataset	4.301 (0.234)	4.188 (0.163)	4.853 (0.293)
(b) Researcher Access Request Processing	Overall	760.895 (41.522)	574.783 (27.401)	984.385 (41.057)
	# of researchers' requests	350	266	411
	Per request	2.174 (0.119)	2.161 (0.103)	2.395 (0.100)
	Per successful request	2.183 (0.141)	2.148 (0.103)	2.399 (0.122)
	Per denied request	2.172 (0.116)	2.164 (0.107)	2.394 (0.100)
(c) Audit-Log Querying	Overall	399.223 (1.842)	409.094 (3.307)	417.933 (3.558)
	# of log queries	363,072	363,072	363,072
	Per query (milliseconds)	1.096 (0.005)	1.123 (0.009)	1.147 (0.010)
	Per query, with empty output (milliseconds)	0.962 (0.005)	0.987 (0.008)	1.009 (0.009)
	Per query, with non-empty output (milliseconds)	2.613 (0.010)	2.661 (0.017)	2.708 (0.018)

6.4.3 Researcher Request Processing

The total runtime for all researcher requests varied across institutions, with Institution 3 experiencing the longest duration of 984.385 seconds and Institution 2 the shortest at 574.783 seconds. Per institution, the average time for a researcher request ranged approximately from 2.161 to 2.395 seconds. Detailed results of the researcher requests for all three sites are presented in **Table 6.4.B**. These variations are further depicted in **Figure 6.6.B**, which illustrates the time taken per researcher request at each of the three institutions.

6.4.4 Audit-Log Querying

A total of 363,072 audit-log queries were performed at all three institutions. The overall runtime for these queries spanned a range from 399.223 seconds to 417.933 seconds per site. This range is further explored in **Figure 6.6.C**, which compares the per query time across the three sites. Given our exhaustive testing approach of all combinations, many of these queries did not yield any output. For queries with non-empty output (queries that yielded at least one record), the average return time varied from 2.613 to 2.708 milliseconds. For queries with empty output (queries that did not yield any returned records), the average return time varied from 0.962 to 1.009 milliseconds. Detailed results of the audit-log queries for each site are detailed in **Table 6.4.C**.

6.5 Discussion

6.5.1 Findings in Each Workflow

In general, variations were observed in time measurements of metadata initialization, researcher request processing, and audit-log record querying across the three institutions, which may be attributed to the difference in specifications of their respective hardware. The overall system performance is determined by Institution 3 (the VM with the lowest specifications).

For metadata initialization, Institution 2 has the fastest per-metadata initialization time for both researcher's certificate metadata and dataset metadata. Specifically, with regards to per certificate import speed, Institution 2 (2.858 seconds) demonstrated a 10.16% faster performance compared to Institution 1 (p-value of 7.75×10^{-11}) and a 21.34% faster performance than Institution 3 (p-value of 1.11×10^{-18}). Similarly, in the case of per dataset metadata import, Institution 2 (4.188 seconds) was marginally faster than Institution 1 at 2.63% (p-value of 0.043) and 13.69% faster than Institution 3 (p-value of 3.27×10^{-13}).

For researcher access request processing, given the variation in the total number of requests per institution, we utilized the time metric of per researcher request for comparative analysis. On this basis, Institution 1 processed requests 9.23% faster than Institution 3 (p-value of 3.90×10^{-7}). However, the difference in request times between Institution 1 and 2 was minimal (p-value of 0.117), which might be due to the fact that the ratios of successful and denied requests might differ at each site. The paired t-test analysis of per successful request (p-value of 0.049) and per denied request (p-value of 0.407) between Institution 1 and 2 confirmed that there was no significant time difference in the per researcher request.

For audit-log querying, the variation observed in overall log query time follows the differences in computational specification of the three institutions: Institution 1 had the fastest

performance, followed by Institution 2 and 3. It was also observed that the majority of the time consumed in this auditing function might be spent on constructing the string value of the output, which may explain why queries without empty output (i.e., denied) were faster than those yielding non-empty output (i.e., approved). The results showed that Institution 1 was approximately 2.41% faster than Institution 2 (p-value of 1.43×10^{-17}) and 4.48% faster than Institution 3 (p-value of 2.24×10^{-22}). Notably, queries resulting in empty output were observed to be significantly faster than those with non-empty output (p-value of 3.63×10^{-170}), which also has greater standard deviation. Despite the differences between queries with empty and non-empty output, we were able to achieve sufficiently fast query times at the millisecond range.

6.5.2 Findings in Comparative Analysis

The comparative analysis among the three institutions reveals that the hardware capacity and computational power can influence the time a VM takes to complete a given task. This was particularly evident in audit-log querying which was executed locally on each VM. The results mirror the computational capacity hierarchy of high (Institution 1), medium (Institution 2), and low (Institution 3).

Nevertheless, it is possible that once a VM's specifications reach a certain threshold ("medium" or higher in computational power), further improvements in performance become marginal, at which point other factors such as network speed begin to exert a more pronounced effect on results. In our experiments, both metadata initialization and researcher access request processing involve writing to the blockchain and possibly interacting with other smart contracts. Consequently, network speed may become a crucial factor in these processes, potentially accounting for the better performance of Institution 2 in these specific areas compared to Institution 1.

Our choice of a private blockchain implementation utilizing a PoA protocol reduces the impact of computational capacity, which is less critical than on a public blockchain where extensive computation is essential for maintaining speed. Thus, even with limited computational power as seen in Institution 3, the system managed to maintain a relatively quick average response time of 2.395 seconds per researcher request. This efficiency marks an advancement from the traditional manual email method, demonstrating the system's ability to reduce response times from potentially hours or days to mere seconds, even under varied computational capacities.

6.5.3 Limitations

There are several limitations with our current system design and experiments as follows.

1. Human-based processes may serve as a direct comparing baseline for our system, and such comparisons could provide valuable insights into the quantitative improvement and efficiency gained through our blockchain-based method. Since our study focuses on the computational aspect, comparative studies between our system and human-based solutions have yet to be conducted.
2. Currently, we evaluated our method based on a fixed number of institutions, to demonstrate the feasibility of a blockchain-based LDS management system. We have yet to investigate the scalability of as the number of participating sites changes. The impact of more institutions on the system's speed and overall performance is yet to be examined.
3. While our system has considered real-world scenarios where institutions vary in computational capacity, it has yet to be tested with actual data requests. The use of real data requests could introduce additional metadata elements that were yet to be accounted for in our process, potentially impacting the system's performance.
4. While the integration with the InterPlanetary File System (IPFS) for distributed storage has been mentioned as a potential for system enhancement,[221] we focused on facilitating secure and efficient data access within the existing framework, in which the direct sharing of full data content is discouraged. Therefore, the potential to integrate direct access to data files other than just data links is yet to be investigated.

6.6 Chapter Conclusion

Our system demonstrates a viable prototype leveraging blockchain networks and smart contracts to streamline the LDS sharing process with approximately 2 seconds of processing and immutably storing each request on chain. It addresses challenges such as the elimination of SPoF and unauthorized data modifications, while ensuring the intuitive auditability and transparency of the data access trail. The system has been evaluated across multiple sites in a pragmatic computing environment, featuring varying computational capacities across sites. The evaluation demonstrates the system's efficiency; as even the computational setup with the lowest specifications was reasonably sufficient, the system has shown its definite edge over traditional manual email-based methods.

Our blockchain-based automated data sharing system may also revolutionize the way clinical research data is shared among researchers, thereby facilitating stronger cross-institutional collaboration and expediting advancements in biomedical studies. Specifically, clinical researchers are provided with quicker and more reliable access to diverse datasets, leading to potential improvement in diagnostic accuracy and more personalized patient treatment plans. Meanwhile, patients may benefit from the immutable access history of their health information, as supported by transparent audit logs to assure that their data are only accessed by granted parties. Moreover, auditors can review the access audit trail to ensure the whole data request process is conforming to policies and regulations. This innovative approach can both accelerate the speed of biomedical research and clinical studies and establish a stronger foundation for how medical data is managed and utilized.

6.7 Acknowledgements

Chapter 6, in part, is currently being prepared for submission for publication of the material. Yu, Yufei;* Edelson, Maxim;* Pham, Anh;* Pekar, Jonathan; Johnson, Brian; Post, Kai; Kuo, Tsung-Ting. The dissertation author was one of the primary investigators and co-first authors of this paper. (* These authors contributed equally to this work)

Chapter 7 USABILITY OF FEDERATED LEARNING SCHEMA

7.1 Abstract

Background. Institutions who wish to engage in collaborative machine learning projects to improve healthcare outcomes may be deterred by privacy concerns. Decentralized federated learning is a privacy-preserving and security-robust tool to promote cross-institutional learning. Such frameworks often require complicated setups and user expertise in highly technical fields. We aimed to improve their utilization by offering an intuitive, user-friendly and secured system that integrates both front-end and back-end functionalities.

Method. We developed WebQuorumChain, an integrated system built upon the QuorumChain schema that supplies users with visual tools to set up the learning network, manage training sessions, and inspect the learning process. We tested the system on a 2-site network using two publicly-available health datasets. We measured the average vertical and horizontal-ensemble AUCs per dataset across 30 trials, as well as the average execution time of the system.

Results. Our system achieved consistently high AUCs for each dataset (0.94-0.96), and in reasonable total execution times ranging from 5-20 minutes, inclusive of modeling and all other system overheads. The event logs emitted from back-end layers to be displayed on the front-end were synchronized with the progress of the underlying algorithm.

Conclusions. WebQuorumChain helps schedule and monitor low-level processes without violating the fundamental security promises of cross-institutional decentralized machine learning. It also maintains predictive accuracy and runtime efficiency in the presence of additional layers. This may be useful to promote meaningful collaboration among healthcare institutions, who can retain full control of their data privacy while contributing to data-driven discoveries.

7.2 Background and Significance

7.2.1 Federated Learning in Healthcare

The recent rise of machine learning (ML) applications in medicine and healthcare research has led to a wide range of successful tools, including clinical diagnosis,[226-230] radiography analysis,[231-234] and drug discovery.[235-237] It is projected that the healthcare sector may be among the industries most affected by ML-backed innovations,[16,238-240] the majority of which rely on frequent access to data. However, the unique nature of health information dictates that data are wanted both in large quantity and with stringent privacy protection.[241] This dual desire to facilitate big data analysis within the constraint of restricted privacy leads to the use of federated learning (FL) techniques.[242,243] In contrast to the conventional model of centralized learning, where pooled data are deposited and analyzed at one central database and thus exposed to privacy and security drawbacks,[244] FL does not require local data to be transferred externally.[53,245] With FL, each site participant (agent/learner) retains full control of their collected data and stays in charge of local model construction; the only information being exchanged for global model aggregation is derived statistics and covariates, which naturally reduces the risk of privacy leak.[165,246] However, as FL relies on a central server for the process of model aggregation and distribution, it still retains the Single-Point-of-Failure (SPoF) security vulnerability from conventional ML.[51,247] That is, should the central authority entrusted with learning coordination stop working due to either routine maintenance or hostile attack, the entire collaboration will cease. Such disruptive events concerning the central node of a system have been observed in recent times,[28,108,109] highlighting the need for proactive measures to ensure the integrity and robustness of collaborative learning.

7.2.2 Decentralized Federated Learning for Immutability, Transparency and High-Availability

Decentralized federated learning (DFL) in which no central server is needed can incorporate both the privacy-preserving nature of FL and the security robustness against SPoF.[248,249] A decentralized, distributed ledger technology, blockchain can facilitate DFL as it can help with the transmission of model statistics among network learners in the absence of a central authority. Moreover, the innate design of blockchain offers the added benefits of immutability, transparency, and high-availability;[37] making it more resistant to SPoF and data tampering threats.[121-123] Several state-of-the-art DFL algorithms such ModelChain,[53] HierarchicalChain,[165] and QuorumChain[250] employ blockchain technology as their infrastructural frameworks thanks to these technological advantages. In particular, ModelChain lays out the technical design to enable dissemination of model updates on a "flattened", or non-hierarchical topology, where each individual node of the blockchain represents a single agent.[53] When the network topology is more hierarchically complex, a common scenario in healthcare as a research entity may comprise of several subnetworks and subsites, HierarchicalChain[165] is one architectural adaptation for network-of-networks collaborative learning. Next, to handle learning disruption when some agent leaves the network due to maintenance/attacks, a challenge that can hinder the practical implementations of distributed learning whether temporarily or permanently, QuorumChain[250] algorithmically allows learning to progress under agent unavailability scenario. The decision to continue learning rounds rests on the formation of "quorum," or a subset of learners who remains in the collaboration and whose cumulative amount of data passes a certain threshold.[250] This approach has been shown to significantly improve predictive correctness in site-unavailability scenarios.[250]

7.2.3 Usability Issues in Decentralized Federated Learning

While functional, such decentralized solutions often require complicated network/client setups, and a vast knowledge of cross-domain expertise in machine learning, distributed systems, and cryptography.[251] They also require extensive training to familiarize users with non-interactive, non-visual frameworks. Some efforts have been made to ease the adoption of federated learning through easy-to-use interfaces and GUI-driven platforms,[252-254] none of which provided blockchain implementations and their added technical requirements. Although the idea of an "application layer" atop the blockchain layer for user-level interaction and data aggregation has been proposed,[255] a prototype has yet to be developed. The lack of user-friendly and intuitive toolkits may discourage users from utilizing DFL algorithms and render the whole process opaque to the general users, despite their proven benefits of immutability and transparency in data provenance. Given that the primary concern of users is the final predictive performance other than the technical steps to expertly deploy and administer a decentralized infrastructure, it is critical to simplify the entry point for DFL systems. In the particular case of QuorumChain, while the configuration of the “quorum” mechanism is entrusted to a network of immutable, source-verifiable smart contracts, their setup still required users to launch multiple blockchain nodes, deploy the smart contracts, and interact with a command line interface. These language-specific steps may hinder the observability of the end-to-end system, due to the risks of misconfiguration and a lack of intuitive, visually transparent tools that non-technical users may desire. To enable the wider adoption and functionality of DFL schemas such as QuorumChain, it is crucial to allow users to easily establish and monitor the learning process while providing transparent access to critical information and metrics, ensuring a seamless and efficient distributed learning experience.

7.2.4 Objective

In this study, we aim to improve the utilization of decentralized federated learning for cross-institutional, privacy-preserving collaborative learning, and specifically the QuorumChain schema, by developing a web-based integrated system that allows healthcare researchers to set up the framework at ease, administer training sessions with visual, interactive tools, and automatically inspect learning/model transactions beyond manual inspection.

7.3 Materials and Methods

7.3.1 Methodology Overview

We chose QuorumChain[250] as the underlying DFL algorithm upon which our system would be constructed. It inherits the privacy-preserving nature of previous DFL schemas, as well as the ability to coordinate the transmission of local and global model statistics across a complex hierarchical topology.[250] WebQuorumChain seeks to enhance the usability of QuorumChain by introducing a web-based application layer that abstracts the system's implementation details into "clickable" actions, and presenting users with an interactive GUI to bypass complex system training while still engaging directly with the learning and networking backend. It is designed to be delivered as a local executable that can be conveniently loaded on a per-site basis. Three key features are offered (**Figure 7.1**): setup tools, learning dashboard, and inspection tools. Specifically, the setup tools are designed to aid users in initiating the federated learning rounds, such as starting the blockchain daemon process and connecting to the permissioned network. These tools also allow for the customization of learning parameters such as learning rate, regularization, number of epochs, and enable users to link to their local training and testing data (**Figure 7.1.A**). Next, the learning dashboard abstracts the implementation details of synchronization among multiple sites, enabling users to focus on the iteration-by-iteration model updates and eventual convergence without having to manage complex inter-site synchronization tasks (**Figure 7.1.B**). Meanwhile, the inspection tools provide a means for users to review training logs automatically emitted by the learners and network agents, in addition to blockchain transaction logs. This allows users to monitor and review the training progress in real-time as needed, enhancing transparency and control over the system's operations (**Figure 7.1.C**).

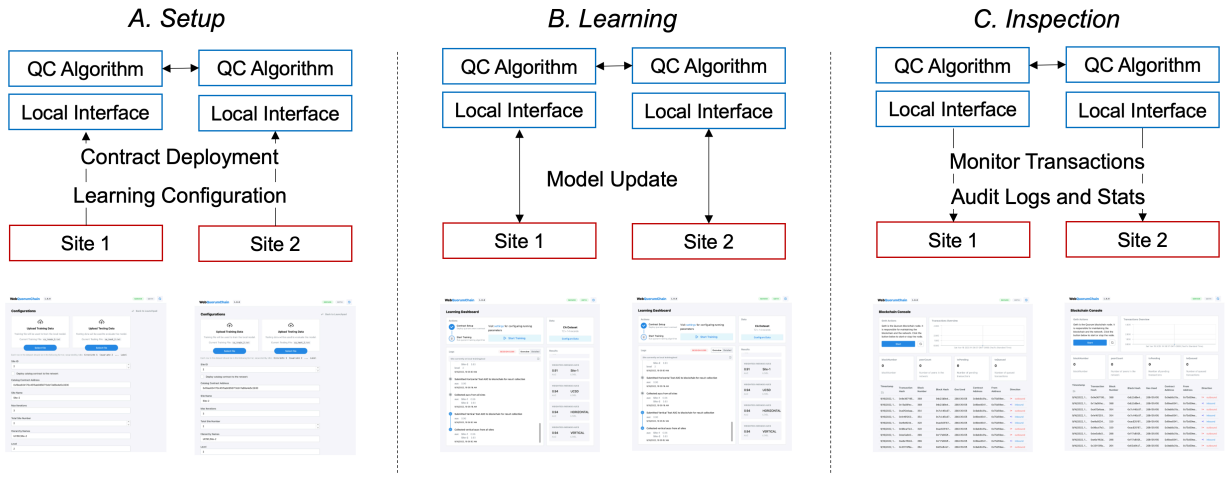


Figure 7.1: High-level overview of WebQuorumChain:

A) The setup component helps initiate the learning system by integrating smart contract deployment and hyperparameter configuration through a local interface.

B) The learning component helps wrap the core learning algorithm and handles model updates.

C) The inspection component helps monitor model transactions and allows the auditing of learning and blockchain logs with viewable statistics.

7.3.2 User Workflow

From a user's perspective, the WebQuorumChain service offers a straightforward and user-friendly approach to administer learning functionalities, all of which are fully equipped with intuitive clickables. Its implementation commences with the user selecting the appropriate command buttons to configure and initiate the blockchain network through the provided GUI; following this initial setup, the user can prepare for learning by linking the training and testing datasets and deploying the essential contracts for the DFL protocol (**Figure 7.2.A**). After the contracts are deployed, the learning process occurs (**Figure 7.2.B**), and the user may actively interact with the graphical inspection tools to track its progress (**Figure 7.2.C**). In particular, logs are generated throughout the learning span to encapsulate critical information pertaining to model-specific details such as gradient contents, and other algorithm state transitions. These logs can be scrutinized in real-time with visual indicators for easy comprehension. Once learning completes, the evaluation metrics for all ensemble levels, as well as the final model, are displayed for further analysis and validation.

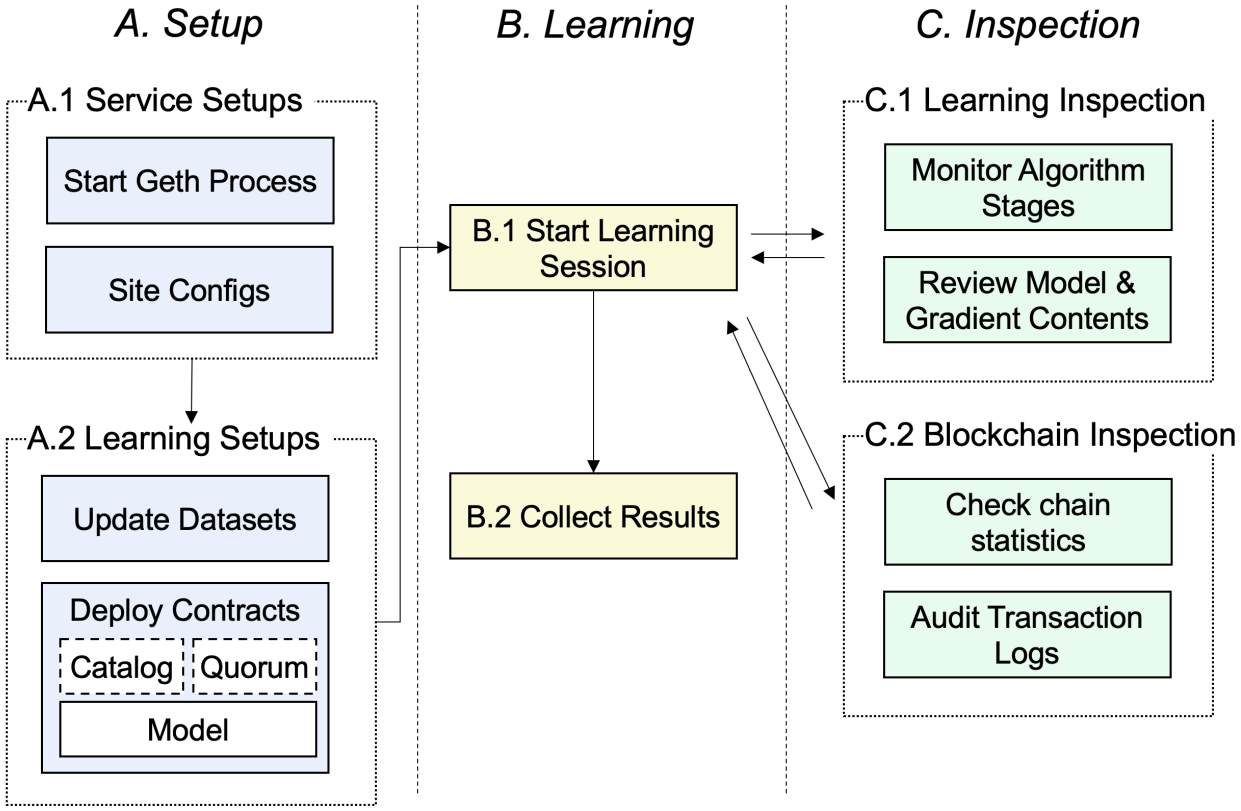


Figure 7.2: User workflow and GUI functionalities:

A) To start the service, the user first sets up network configs by starting the Geth blockchain node and editing site configs via the configuration web interfaces. The user then deploys smart contracts Quorum, Model, and optionally Catalog as defined by the QuorumChain algorithm, as well as linking to their datasets.

B) The learning process starts until results may be collected.

C) Inspection tools are readily available during the life cycle of learning. Event logs including algorithm states and model information such as gradient contents are emitted for review. Upon learning conclusion, evaluation metrics for all ensemble levels as well as the final model are displayed.

7.3.3 Underlying System Layers

A robust encapsulation of the QuorumChain implementation, WebQuorumChain employs an unobtrusive approach that avoids change to the original architecture. Between the local executable and the decentralized blockchain network, the on-chain activities can be effectively relayed, scheduled, and monitored through a front-facing web client and its graphical elements, as enumerated in **7.3.2**. In return, this client interacts with the intermediate local "server" (local host) within a virtual node that oversees the underlying blockchain transactions (which are coordinated by the blockchain node, i.e., Go-Ethereum (Geth) Process [132]), the message queue in which events and transaction logs are piped, as well as the algorithmic functions of QuorumChain (the QuorumChain Process) that regulate the fundamental operations of the federated learning. This strategy ensures minimal chance of computational manipulation or state modification, while still offering a convenient view of the execution status. Thus, the enhanced features and functionality of WebQuorumChain do not violate the integrity and reliability of the underlying QuorumChain algorithms. The architecture is depicted in **Figure 7.3**, and more details about each layer are provided in **7.3.4**.

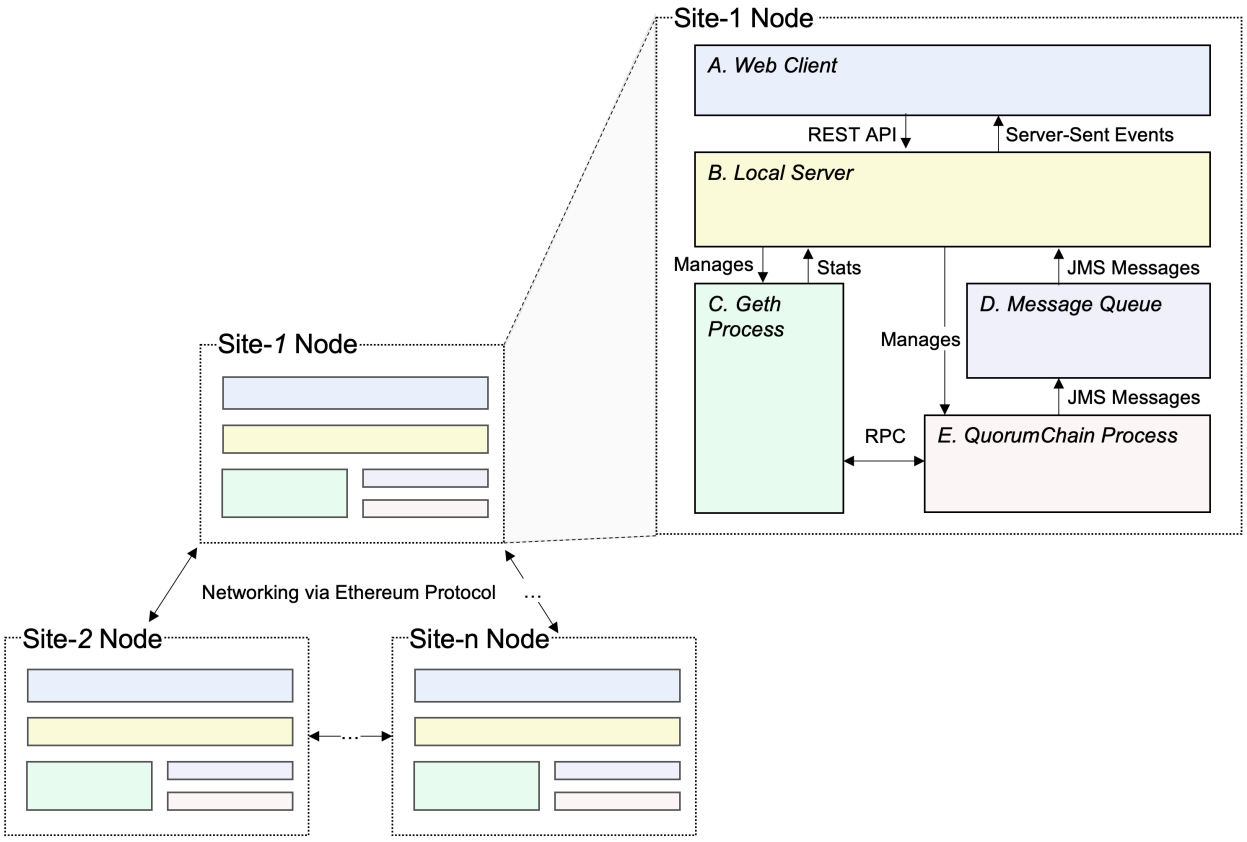


Figure 7.3: Underlying architecture breakdown: *n* sites participate in the FL session and form a peer-to-peer network communicating via blockchain protocols. Per-node architecture is shown in Site-1 Node diagram:

A) web client exposes the functionalities of the server by providing a user-friendly interface;

B) local server mediates between user interactions and the execution of learning units;

C) Geth (Go-Ethereum) Process is the blockchain client that executes smart contract code and the networking agent for communicating among sites;

D) message queue of logs and events emitted during the process; and

E) the QuorumChain Process carries the fundamental training algorithms.

7.3.4 Back-end Modules and Front-end Interface

While the Geth Process converts smart contract definitions into executables on the Ethereum network and allows subsequent interactions with the blockchain, the QuorumChain Process communicates with this on-chain component to synchronize the order of model exchange, aggregation, quorum formation, and learning continuation across quorum members. It also dispatches event logs to the Java Message Service (JMS), which are then funneled into a local Message Queue (MQ) for buffering. An event log can contain information such as when a participant enters or leaves the training session, and updates to the model/gradient contents. At the invocation of the inspection tool, the event messages will flow to the web client for visual display. Without this mechanism, such critical information would not be accessible outside of the algorithm. Finally, the web client brings the server's functionalities to the user by providing a user-friendly and intuitive interface. It listens to the inspection logs and renders them for visualization. It also defines a set of operations to enable easy configuration and control of the training states. Some of these interface designs are showcased in **Figure 7.4**.

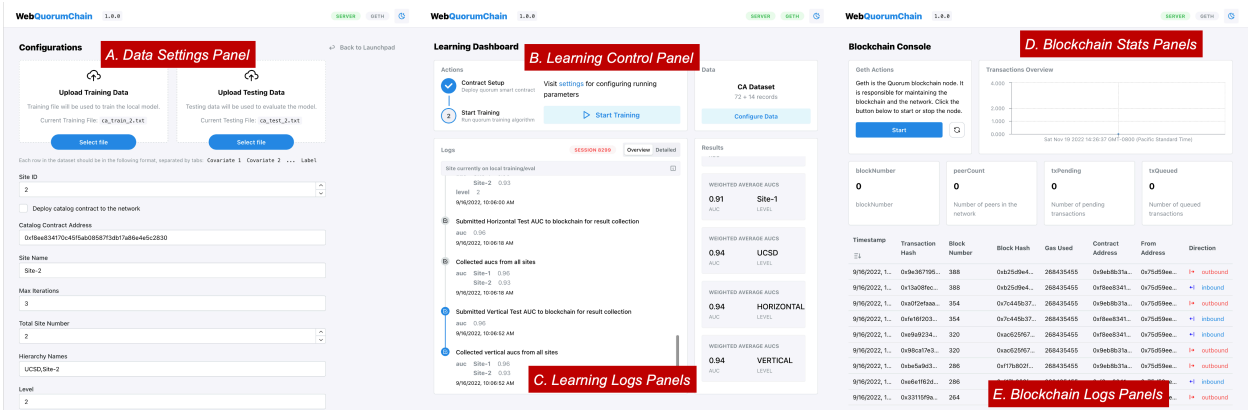


Figure 7.4: Details of GUI design:

A) a configuration interface through which users may upload training/testing data via drag-and-drop;

B) the learning control panel at a ready state where training session can be started;

C) blockchain stats panel including the transaction workload time series plot, current block number, and peer count;

D) learning logs panel in overview mode such that detailed algorithm logs are filtered out; and exhibited logs presented with previewable model/gradient updates; and

E) blockchain logs that provides insights into the underlying contract transactions; fields like timestamp, block hash, and in/outbound directions are shown.

7.3.5 Data

We evaluated the system on two public datasets, Cancer Biomarker (CA)[256] and Myocardia Infarction (Edin);[257] both with a binary prediction target of disease presence. The Myocardial Infarction (Edin) dataset has 9 covariates and 1,253 samples,[257] whereas the Cancer Biomarker (CA) dataset has 2 covariates and 141 samples.[256] The details of each dataset are further summarized in **Table 7.1**.

Table 7.1: Description of Cancer Biomarker (CA) and Myocardial Infarction (Edin) datasets including covariates, sample size, class distribution, and outcome.

Dataset	Number of Covariates	Number of Samples	Class Distribution
Cancer Biomarker (CA)	2	141	0.638 / 0.362
Myocardial Infarction (Edin)	9	1,253	0.219 / 0.781

7.3.6 Evaluation Setting

To test the WebQuorumChain system, we set up a distributed federated learning framework on a private Ethereum network consisting of 2 sites, who collaborated to train single-layer logistic regression models for binary classification tasks over 3 epochs. For the CA dataset, the QuorumChain hyperparameters were set with a waiting time period of 4 seconds, and a quorum percentage threshold of 51%. For the Edin dataset, the waiting time period was 16 seconds, and the quorum threshold was 60%. Both datasets used a pooling time period of 1 second, and a maximum of 100 iterations per level. All training configurations were initiated via the web client of WebQuorumChain. Next, we ran 30 trials of the collaborative learning per dataset and measured the final AUCs (Area Under the Curve) of the global models, and runtimes either in the training or evaluation phases, or both (i.e., total execution time, which also includes system overheads). These times were measured end-to-end on the client-side, considering the overhead of the application layer. Our implementation employed Geth (Go-Ethereum) v1.10.20 [85] for the blockchain node, ActiveMQ 5.17.1 [258] as the Java Message Service (JMS) provider, Spring Boot 2.7.2 [259] as the server framework, and React 18.2.0 [260] for the web client framework. These experiments were conducted on Alpine Linux [261] utilizing Java SE 17 [262] within docker containers.

7.4 Results

7.4.1 Prediction Accuracy and Runtime Efficiency

The experimental results are presented in **Table 7.2**. Both datasets completed their respective 30 trials. The AUC results were consistently high across all trials given the deterministic nature of the logistic regression algorithm. For the CA dataset, the final AUC of the collaborative model was 0.94 whether at the horizontal (i.e., the prediction score of a site is generated using the scores from all models at the same level, as weighted averaged by its respective training data sizes) or vertical (i.e., the weighted-average prediction score of a site is generated using the scores from each level related to that site) ensemble.[250] For the Edin dataset, both vertical and horizontal AUC was 0.96. Execution times were measured from the start to the end of the corresponding learning periods on the web client, taking system latencies into account, and thus would exceed the sum of training and evaluation times. In particular, it took less than 2 minutes to train the CA dataset, ~2.5 minutes for evaluation, with a total execution time around 5.3 minutes inclusive of system overhead. The larger Edin dataset took ~9 minutes for training, ~7.8 minutes for evaluation, and ~20 minutes for total execution time.

Table 7.2: Evaluation results and performance metrics. The metrics include the Area Under the Curve (AUC) for vertical and horizontal ensembles, training time, evaluation time, and total execution time for the entire trial which included network latency (the times are all measured in seconds).

Dataset	Vertical AUC	Horizontal AUC	Training Time	Evaluation Time	Total Time
CA	0.94 ± 0.00	0.94 ± 0.00	107.01 ± 4.37	149.89 ± 3.21	317.03 ± 4.50
Edin	0.96 ± 0.00	0.96 ± 0.00	538.98 ± 0.13	470.00 ± 0.03	1236.00 ± 0.15

7.4.2 Inspection Dashboard

With regards to the inspection function, example strip plots for learning events and blockchain events emitted by WebQuorumChain from both sites and datasets were presented in **Figure 7.5**. These logs were collected in real-time and timestamped at the web client, showing that the algorithm executes consistently throughout the session. The web client was able to effectively capture information from the ongoing execution process. The learning event categories included INFO, MODEL, and RESULT. INFO events provide general information about various stages along the progression of the algorithm, MODEL events represent emitted gradient and model contents, and RESULT events mark the completion of evaluation stages. Another event category was blockchain (BLOCKCHAIN), which depicted blockchain transaction events.

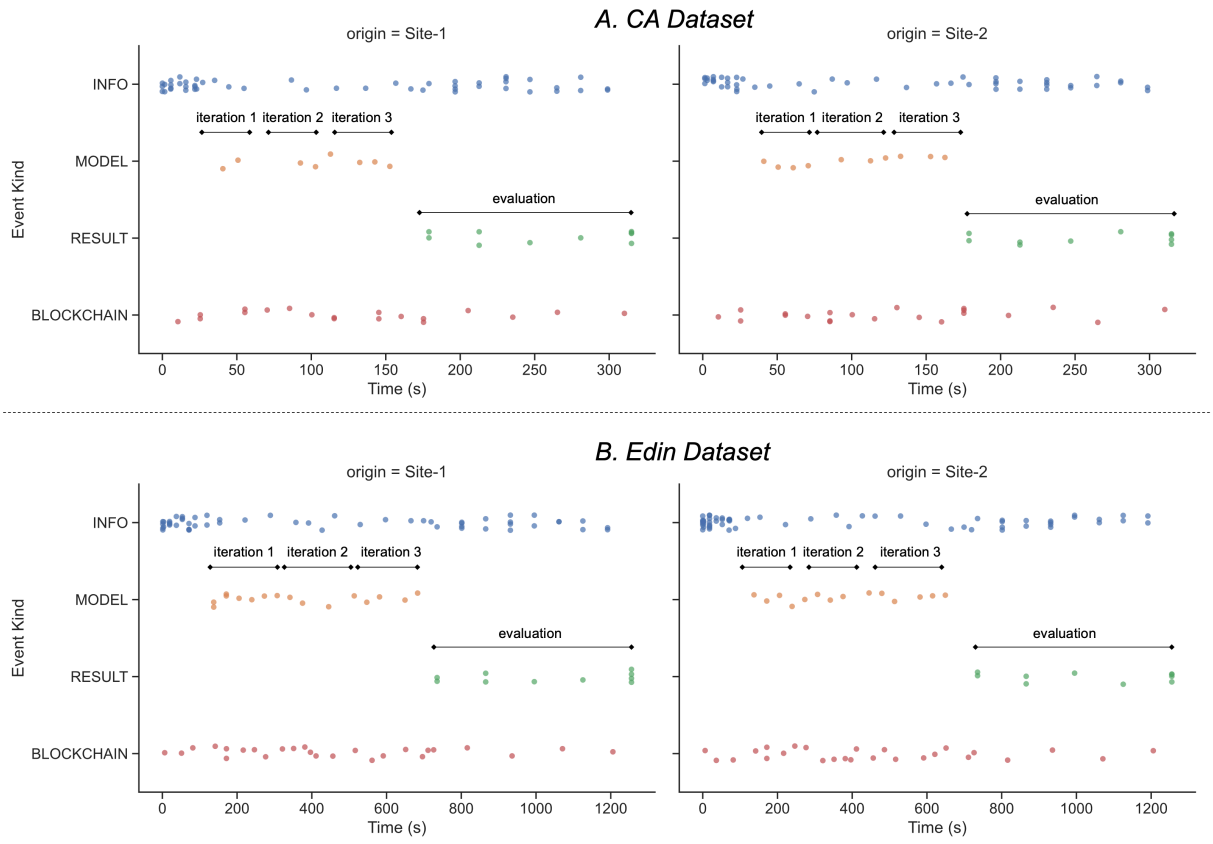


Figure 7.5: Examples of learning events (categorized into INFO, MODEL, and RESULT) and blockchain (BLOCKCHAIN) events for

A) CA

B) Edin datasets.

All timestamps are measured on the client side. Data points are split into their respective two sites.

7.5 Discussion

7.5.1 Findings

Our experimental results demonstrate that WebQuorumChain is an efficient extension to the original QuorumChain algorithm for decentralized federated learning. Despite the addition of application layers, the system completed each learning trial within a reasonable timeframe, without errors, and yielded consistently high AUC. The variations in training, evaluation and/or total execution time may be attributed to nondeterministic network and/or scheduling latencies. Nonetheless, such variations are negligible, as evidenced by their relatively small standard deviation. This suggests that the overhead introduced by our architectural integration is minimal, confirming the technical practicality of WebQuorumChain. Another benefit WebQuorumChain offers is the interactive interface with intuitive visual tools for inspection. It is noted from **Figure 7.5** that the inspected events align with the stages described in the QuorumChain algorithm. For example, the MODEL logs containing model statistics were emitted through each iteration of the training phase, whereas RESULT timely reflected the evaluation stages. On the other hand, both INFO and BLOCKCHAIN events occurred steadily over the course of the learning session, showing the workload was evenly distributed between the two sites.

7.5.2 Enhanced User Experience without Security Compromise

From the perspective of users, WebQuorumChain is an effective upgrade over the command line interfaces provided in previous works. It offers “clickable” setup, learning, and inspection tools with fine-grained user control, while seamlessly managing the low-level jobs and algorithms defined by the QuorumChain algorithm. At the same time, our integrated system remains loyal to the privacy-preserving and security-robust nature of its original counterpart QuorumChain. The newly-introduced local host only manages client-side logic, while peer-to-peer communication still relies solely on the blockchain network and thus maintains compliance to its decentralized, transparent, and immutable principle.

7.5.3 Limitations

Our study is limited by the following constraints:

1. Lack of real-life user testing. Thorough user testing on a wider range may establish the efficacy of our implemented usability features such as on-interface step-by-step guidance. Further studies may need to focus on the human-computer interaction aspects of the system.
2. Limit on datasets and number of sites. The system was validated on a 2-site network using two public health datasets. A larger scale study to test the scalability of our system as the number of learning agents grow, and/or when the size of the dataset changes.
3. Lack of full authentication/authorization to the web interface. For simplification, our proposed system has yet to include full-fledged authentication mechanisms such as 2-factor authentication[263] or role-based access control[264]. Further investigation with regards to user identity management is needed to enhance the security and privacy protection of WebQuorumChain.

7.6 Chapter Conclusion

Based on our results, the integrated system WebQuorumChain with a web-based interface that simplifies the deployment and administration of the collaborative QuorumChain algorithm can improve the utilization of decentralized federated learning schemas. For users, WebQuorumChain provides intuitive and interactive tools to help set up, run, and inspect the progression of their models. From a technical perspective, the system helps schedule and monitor low-level jobs and processes without violating the fundamental security promises of QuorumChain. It also maintains both predictive accuracy and runtime efficiency in the presence of additional system layers. WebQuorumChain can be useful to promote meaningful collaboration among healthcare institutions, who can retain full control of their patient data privacy while contributing to the rise of data-driven discoveries.

7.7 Aim 3 Statement of Impact

Our works in Chapter 6 and Chapter 7 bolster cross-institutional research by innovating the current process of handling data-access requests, and by improving the usability of the distributed FL workflow. With regards to accelerating data access, our proposed blockchain-based mechanism can efficiently record and query requests, lower human errors, and further improve HIPAA compliance by providing an immutable audit trail as per who has asked for what data and when. In the same spirit, our effort to enhance FL usability by providing users with visually interactive and technologically-friendly learning tools may promote collaborative learning, while still adhering to privacy constraints in cross-institutional healthcare data. Our ongoing work may thereby improve the overall transparency, security, and efficiency of cross-institutional biomedical research.

7.8 Acknowledgements

Chapter 7, in part, has been submitted for publication of the material as it may appear in Informatics in Medicine Unlocked, 2024. Shao, Xiyan;* Pham, Anh;* Kuo, Tsung-Ting, Elsevier, 2024. The dissertation author was one of the primary researchers and co-first authors of this paper. (* These authors contributed equally to this work)

REFERENCES

1 Blumenthal, D. & Tavenner, M. The "meaningful use" regulation for electronic health records. *New England Journal of Medicine* **363**, 501-504 (2010).

2 Foronda, M. The AI revolution in cancer. (2020). <https://www.nature.com/articles/d42859-020-00082-9>.

3 Tresp, V., Overhage, J. M., Bundschus, M., Rabizadeh, S., Fasching, P. A. & Yu, S. Going digital: a survey on digitalization and large-scale data analytics in healthcare. *Proceedings of the IEEE* **104**, 2180-2206 (2016).

4 Sutton, R. T., Pincock, D., Baumgart, D. C., Sadowski, D. C., Fedorak, R. N. & Kroeker, K. I. An overview of clinical decision support systems: benefits, risks, and strategies for success. *NPJ digital medicine* **3**, 17 (2020).

5 Peiffer-Smadja, N., Rawson, T. M., Ahmad, R., Buchard, A., Georgiou, P., Lescure, F.-X., Birgand, G. & Holmes, A. H. Machine learning for clinical decision support in infectious diseases: a narrative review of current applications. *Clinical Microbiology and Infection* **26**, 584-595 (2020).

6 Horng, S., Sontag, D. A., Halpern, Y., Jernite, Y., Shapiro, N. I. & Nathanson, L. A. Creating an automated trigger for sepsis clinical decision support at emergency department triage using machine learning. *PloS one* **12**, e0174708 (2017).

7 Antoniadi, A. M., Du, Y., Guendouz, Y., Wei, L., Mazo, C., Becker, B. A. & Mooney, C. Current challenges and future opportunities for XAI in machine learning-based clinical decision support systems: a systematic review. *Applied Sciences* **11**, 5088 (2021).

8 Fitriyani, N. L., Syafrudin, M., Alfian, G. & Rhee, J. HDPM: an effective heart disease prediction model for a clinical decision support system. *IEEE Access* **8**, 133034-133050 (2020).

9 Mani, S., Ozdas, A., Aliferis, C., Varol, H. A., Chen, Q., Carnevale, R., Chen, Y., Romano-Keeler, J., Nian, H. & Weitkamp, J.-H. Medical decision support using machine learning for early detection of late-onset neonatal sepsis. *Journal of the American Medical Informatics Association* **21**, 326-336 (2014).

10 Karthikeyan, A., Garg, A., Vinod, P. & Priyakumar, U. D. Machine learning based clinical decision support system for early COVID-19 mortality prediction. *Frontiers in public health* **9**, 626697 (2021).

11 Collier, M., Fu, R., Yin, L. & Christiansen, P. Artificial intelligence: healthcare's new nervous system. *Accenture plc* (2017).

12 Horgan, D., Romao, M., Morré, S. A. & Kalra, D. Artificial intelligence: power for civilisation–and for better healthcare. *Public health genomics* **22**, 145-161 (2020).

13 Kalkman, S., van Delden, J., Banerjee, A., Tyl, B., Mostert, M. & van Thiel, G. Patients' and public views and attitudes towards the sharing of health data for research: a narrative review of the empirical evidence. *Journal of medical ethics* **48**, 3-13 (2022).

14 He, J., Baxter, S. L., Xu, J., Xu, J., Zhou, X. & Zhang, K. The practical implementation of artificial intelligence technologies in medicine. *Nature medicine* **25**, 30-36 (2019).

15 *2018 Annual Report*, <https://www.hhs.gov/sites/default/files/2018-annual-report.pdf> (2018).

16 *Future Health Index 2019 Transforming Healthcare Experience*.

17 Cakir, Z. & Savas, H. (2020).

18 Li, M. M., Pham, A. & Kuo, T.-T. Predicting COVID-19 county-level case number trend by combining demographic characteristics and social distancing policies. *JAMIA open* **5**, ooac056 (2022).

19 Keshavamurthy, R., Dixon, S., Pazdernik, K. T. & Charles, L. E. Predicting infectious disease for biopreparedness and response: A systematic review of machine learning and deep learning approaches. *One Health*, 100439 (2022).

20 Wahab, O. A., Mourad, A., Otrok, H. & Taleb, T. Federated machine learning: Survey, multi-level classification, desirable criteria and future directions in communication and networking systems. *IEEE Communications Surveys & Tutorials* **23**, 1342-1397 (2021).

21 Appari, A. & Johnson, M. E. Information security and privacy in healthcare: current state of research. *International journal of Internet and enterprise management* **6**, 279-314 (2010).

22 Schumaker, E. *Florida agents raid COVID-19 data scientist's home after controversial firing*, <https://abcnews.go.com/US/florida-agents-raid-covid-19-data-scientists-home/story?id=74603007> (2022).

23 Grind, K. M. Meta Employees, Security Guards Fired for Hijacking User Accounts. (2023).

24 Zhao, Q., Huang, C. & Dai, L. VULDEFF: Vulnerability detection method based on function fingerprints and code differences. *Knowledge-Based Systems* **260**, 110139 (2023).

25 Marczak, B., Scott-Railton, J., McKune, S., Abdul Razzak, B. & Deibert, R. Hide and seek: Tracking NSO group's Pegasus spyware to operations in 45 countries. (2018).

26 Bertin, P., Bonjour, S. & Bonnin, J.-M. in *GLOBECOM 2009-2009 IEEE Global Telecommunications Conference.* 1-6 (IEEE).

27 Chanthadavong, A. *Sabre Systems IT outage cripples airline operations globally*, <https://www.zdnet.com/article/sabre-systems-it-outage-cripples-airline-operations-globally/> (2023).

28 Tsidulko, J. *United Airlines, NYSE Outages Reveal Poor Redundancy Architecture, Insufficient Testing | CRN*, <https://www.crn.com/news/security/300077385/united-airlines-nyse-outages-reveal-poor-redundancy-architecture-insufficient-testing.htm> (July).

29 Nakamoto, S. Bitcoin: A peer-to-peer electronic cash system. *Decentralized business review*, 21260 (2008).

30 Hasselgren, A., Kralevska, K., Gligoroski, D., Pedersen, S. A. & Faxvaag, A. Blockchain in healthcare and health sciences—A scoping review. *International Journal of Medical Informatics* **134**, 104040 (2020).

31 Kuo, T.-T., Kim, H.-E. & Ohno-Machado, L. Blockchain distributed ledger technologies for biomedical and health care applications. *Journal of the American Medical Informatics Association* **24**, 1211-1220 (2017).

32 Randall, D., Goel, P. & Abujamra, R. Blockchain applications and use cases in health information technology. *Journal of Health & Medical Informatics* **8**, 8-11 (2017).

33 Bell, L., Buchanan, W. J., Cameron, J. & Lo, O. Applications of Blockchain Within Healthcare. *Blockchain in healthcare today* (2018).

34 Liu, P. T. S. in *Information and Communications Security: 18th International Conference, ICICS 2016, Singapore, Singapore, November 29–December 2, 2016, Proceedings 18.* 254-261 (Springer).

35 Kamel Boulos, M. N., Wilson, J. T. & Clauson, K. A. Vol. 17 1-10 (BioMed Central, 2018).

36 Hirtan, L., Krawiec, P., Dobre, C. & Batalla, J. M. in *2019 IEEE 24th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD).* 1-7 (IEEE).

37 Kuo, T.-T., Pham, A., Edelson, M. E., Kim, J., Chan, J., Gupta, Y., Ohno-Machado, L. & Consortium, R. D. Blockchain-Enabled Immutable, Distributed, and Highly Available Clinical Research Activity Logging System for Federated COVID-19 Data Analysis from Multiple Institutions. *Journal of the American Medical Informatics Association*, ocad049 (2023).

38 Di Pierro, M. What is the blockchain? *Computing in Science & Engineering* **19**, 92-95 (2017).

39 Mackey, T. K., Kuo, T.-T., Gummadi, B., Clauson, K. A., Church, G., Grishin, D., Obbad, K., Barkovich, R. & Palombini, M. 'Fit-for-purpose?'–challenges and

opportunities for applications of blockchain technology in the future of healthcare. *BMC medicine* **17**, 1-17 (2019).

40 Azaria, A., Ekblaw, A., Vieira, T. & Lippman, A. in *2016 2nd international conference on open and big data (OBD)*. 25-30 (IEEE).

41 Team, S. *Solidity Programming Language*, <https://soliditylang.org//> (2023).

42 Safran, C., Bloomrosen, M., Hammond, W. E., Labkoff, S., Markel-Fox, S., Tang, P. C. & Detmer, D. E. Toward a national framework for the secondary use of health data: an American Medical Informatics Association White Paper. *Journal of the American Medical Informatics Association* **14**, 1-9 (2007).

43 Botsis, T., Hartvigsen, G., Chen, F. & Weng, C. Secondary use of EHR: data quality issues and informatics opportunities. *Summit on Translational Bioinformatics* **2010**, 1 (2010).

44 Schlegel, D. R. & Ficheur, G. Secondary use of patient data: review of the literature published in 2016. *Yearbook of medical informatics* **26**, 68-71 (2017).

45 *NOT-OD-14-124: NIH Genomic Data Sharing Policy*, <https://www.ncbi.nlm.nih.gov/pubmed/> (2022).

46 Kim, H., Bell, E., Kim, J., Sitapati, A., Ramsdell, J., Farcas, C., Friedman, D., Feupe, S. F. & Ohno-Machado, L. iCONCUR: informed consent for clinical data and bio-sample use for research. *Journal of the American Medical Informatics Association* **24**, 380-387 (2017).

47 Stone, M. A., Redsell, S. A., Ling, J. T. & Hay, A. D. Sharing patient data: competing demands of privacy, trust and research in primary care. *British journal of general practice* **55**, 783-789 (2005).

48 Kim, J., Kim, H., Bell, E., Bath, T., Paul, P., Pham, A., Jiang, X., Zheng, K. & Ohno-Machado, L. Patient perspectives about decisions to share medical data and biospecimens for research. *JAMA network open* **2**, e199550-e199550 (2019).

49 Kaye, J., Curren, L., Anderson, N., Edwards, K., Fullerton, S. M., Kanellopoulou, N., Lund, D., MacArthur, D. G., Mascalzoni, D. & Shepherd, J. From patients to partners: participant-centric initiatives in biomedical research. *Nature Reviews Genetics* **13**, 371-376 (2012).

50 Prictor, M., Lewis, M. A., Newson, A. J., Haas, M., Baba, S., Kim, H., Kokado, M., Minari, J., Molnar-Gabor, F. & Yamamoto, B. Dynamic consent: an evaluation and reporting framework. *Journal of Empirical Research on Human Research Ethics* **15**, 175-186 (2020).

51 Kuo, T.-T., Gabriel, R. A., Cidambi, K. R. & Ohno-Machado, L. EX pectation P ropagation LO gistic RE g R ession on permissioned block CHAIN (ExplorerChain):

decentralized online healthcare/genomics predictive model learning. *Journal of the American Medical Informatics Association* **27**, 747-756 (2020).

52 Kumar, N. M. & Mallick, P. K. Blockchain technology for security issues and challenges in IoT. *Procedia Computer Science* **132**, 1815-1823 (2018).

53 Kuo, T.-T. & Ohno-Machado, L. Modelchain: Decentralized privacy-preserving healthcare predictive modeling framework on private blockchain networks. *arXiv preprint arXiv:1802.01746* (2018).

54 Liu, V., Musen, M. A. & Chou, T. Data breaches of protected health information in the United States. *Jama* **313**, 1471-1473 (2015).

55 Reddy, J. *Data breaches in healthcare security systems*, University of Cincinnati, (2021).

56 Grady, H. More than decade long snooping of patient records finally brought to light - Data Privacy - Nixon Peabody blog | Nixon Peabody LLP. (2021). <https://www.nixonpeabody.com/insights/articles/2021/07/20/more-than-decade-long-snooping-of-patient-records-finally-brought-to-light>.

57 Seh, A. H., Zarour, M., Alenezi, M., Sarkar, A. K., Agrawal, A., Kumar, R. & Ahmad Khan, R. Healthcare data breaches: insights and implications. *Healthcare* **8**, 133 (2020).

58 *Patient perspectives around data privacy*, <https://www.ama-assn.org/system/files/ama-patient-data-privacy-survey-results.pdf> (

59 Harry, D. The Human Genome Diversity Project. *Abya Yala News* **8**, 13-15 (1993).

60 Foulks, E. F. Misalliances in the Barrow alcohol study. *American Indian and Alaska Native Mental Health Research* **2**, 7-17 (1989).

61 Drabiak-Syed, K. Lessons from Havasupai tribe v. Arizona state university board of regents: recognizing group, cultural, and dignity harms as legitimate risks warranting integration into research practice. *J. Health & Biomedical L.* **6**, 175 (2010).

62 Kuo, T.-T., Jiang, X., Tang, H., Wang, X., Bath, T., Bu, D., Wang, L., Harmanci, A., Zhang, S. & Zhi, D. Vol. 13 1-12 (BioMed Central, 2020).

63 Dannen, C. *Introducing Ethereum and solidity*. Vol. 1 (Springer, 2017).

64 Macdonald, M., Liu-Thorrold, L. & Julien, R. The blockchain: a comparison of platforms and their uses beyond bitcoin. COMS4507-Adv. *Computer and Network Security* (2017).

65 Yu, H., Sun, H., Wu, D. & Kuo, T.-T. in *AMIA Annual Symposium Proceedings.* 1266 (American Medical Informatics Association).

66 Chowdhury, M. J. M., Ferdous, M. S., Biswas, K., Chowdhury, N., Kayes, A., Alazab, M. & Watters, P. A comparative analysis of distributed ledger technology platforms. *IEEE Access* **7**, 167930-167943 (2019).

67 Tellew, J. & Kuo, T.-T. CertificateChain: decentralized healthcare training certificate management system using blockchain and smart contracts. *JAMIA open* **5**, ooac019 (2022).

68 Zhang, L., Cheng, L., Alsokhiry, F. & Mohamed, M. A. A novel stochastic blockchain-based energy management in smart cities using V2S and V2G. *IEEE Transactions on Intelligent Transportation Systems* **24**, 915-922 (2022).

69 Mohamed, M. A., Mirjalili, S., Dampage, U., Salmen, S. H., Obaid, S. A. & Annuk, A. A cost-efficient-based cooperative allocation of mining devices and renewable resources enhancing blockchain architecture. *Sustainability* **13**, 10382 (2021).

70 Yin, F., Hajjiah, A., Jermsittiparsert, K., Al-Sumaiti, A. S., Elsayed, S. K., Ghoneim, S. S. & Mohamed, M. A. A secured social-economic framework based on PEM-blockchain for optimal scheduling of reconfigurable interconnected microgrids. *IEEE Access* **9**, 40797-40810 (2021).

71 Mann, S. P., Savulescu, J., Ravaud, P. & Benchoufi, M. Blockchain, consent and prosent for medical research. *Journal of medical ethics* **47**, 244-250 (2021).

72 Velmovitsky, P. E., Miranda, P. A. D. S. E. S., Vaillancourt, H., Donovska, T., Teague, J. & Morita, P. P. A blockchain-based consent platform for active assisted living: modeling study and conceptual framework. *Journal of medical Internet research* **22**, e20832 (2020).

73 Román-Martínez, I., Calvillo-Arbizu, J., Mayor-Gallego, V. J., Madinabeitia-Luque, G., Estepa-Alonso, A. J. & Estepa-Alonso, R. M. Blockchain-based service-oriented architecture for consent management, access control, and auditing. *IEEE Access* **11**, 12727-12741 (2023).

74 Rahman, M., Hasan, M., Rahman, M. & Momotaj, M. A Framework for Patient-Centric Consent Management Using Blockchain Smart Contracts in Pre-dictive Analysis for Healthcare In-dustry. *International Journal of Health Systems and Medical Sci-ences* **3**, 45-59 (2024).

75 De Sousa, H. R. & Pinto, A. in *Blockchain and Applications: International Congress.* 54-61 (Springer).

76 Hu, C., Li, C., Zhang, G., Lei, Z., Shah, M., Zhang, Y., Xing, C., Jiang, J. & Bao, R. CrowdMed-II: a blockchain-based framework for efficient consent management in health data sharing. *World Wide Web* **25**, 1489-1515 (2022).

77 Anderson, C., Carvalho, A., Kaul, M. & Merhout, J. W. Blockchain innovation for consent self-management in health information exchanges. *Decision Support Systems* **174**, 114021 (2023).

78 Jaiman, V. & Urovi, V. A consent model for blockchain-based health data sharing platforms. *IEEE access* **8**, 143734-143745 (2020).

79 *Interact with your contracts - Truffle Suite*, <https://trufflesuite.com/docs/truffle/how-to/contracts/interact-with-your-contracts/> (2023).

80 Kuo, T.-T., Jiang, X., Tang, H., Wang, X., Harmanci, A., Kim, M., Post, K., Bu, D., Bath, T. & Kim, J. The evolving privacy and security concerns for genomic data analysis and sharing as observed from the iDASH competition. *Journal of the American Medical Informatics Association* **29**, 2182-2190 (2022).

81 *iDASH Privacy & Security Workshop 2021 - Secure Genome Analysis Competition*, <http://www.humangenomeprivacy.org/2021/index.html> (2021).

82 Labs, W. *Web3j*, <https://docs.web3j.io/4.8.7/> (2023).

83 De Angelis, S., Aniello, L., Baldoni, R., Lombardi, F., Margheri, A. & Sassone, V. in *CEUR workshop proceedings.* (CEUR-WS).

84 Robinson, P., Ramesh, R. & Johnson, S. Atomic crosschain transactions for ethereum private sidechains. *Blockchain: Research and Applications* **3**, 100030 (2022).

85 Ethereum, G. Official Go implementation of the Ethereum protocol. *URl: https://geth. ethereum. org (visited on 01/25/2019)* (2017).

86 Kostamis, P., Sendros, A. & Efraimidis, P. in *2021 3rd Conference on Blockchain Research & Applications for Innovative Networks and Services (BRAINS).* 53-60 (IEEE).

87 Merlec, M. M., Lee, Y. K., Hong, S.-P. & In, H. P. A smart contract-based dynamic consent management system for personal data usage under GDPR. *Sensors* **21**, 7994 (2021).

88 Agarwal, R. R., Kumar, D., Golab, L. & Keshav, S. in *2020 ieee international conference on blockchain and cryptocurrency (icbc).* 1-9 (IEEE).

89 Temesgen, Z. M., DeSimone, D. C., Mahmood, M., Libertin, C. R., Palraj, B. R. V. & Berbari, E. F. in *Mayo Clinic Proceedings.* S66-S68 (Elsevier).

90 Else, H. COVID IN PAPERS. *Nature* **588**, 553 (2020).

91 Wu, X., Zheng, H., Dou, Z., Chen, F., Deng, J., Chen, X., Xu, S., Gao, G., Li, M. & Wang, Z. A novel privacy-preserving federated genome-wide association study framework and its application in identifying potential risk variants in ankylosing spondylitis. *Briefings in bioinformatics* **22**, bbaa090 (2021).

92 Pezoulas, V. C., Kalatzis, F., Exarchos, T. P., Goules, A., Gandolfo, S., Zampeli, E., Skopouli, F., De Vita, S., Tzioufas, A. G. & Fotiadis, D. I. in *8th European Medical and Biological Engineering Conference: Proceedings of the EMBEC 2020, November 29–December 3, 2020 Portorož, Slovenia.* 306-313 (Springer).

93 Petkova, E., Antman, E. M. & Troxel, A. B. Pooling data from individual clinical trials in the COVID-19 era. *Jama* **324**, 543-545 (2020).

94 McBride, O., Murphy, J., Shevlin, M., Gibson-Miller, J., Hartman, T. K., Hyland, P., Levita, L., Mason, L., Martinez, A. P. & McKay, R. Monitoring the psychological, social, and economic impact of the COVID-19 pandemic in the population: Context, design and conduct of the longitudinal COVID-19 psychological research consortium (C19PRC) study. *International journal of methods in psychiatric research* **30**, e1861 (2021).

95 Kochunov, P., Jahanshad, N., Sprooten, E., Nichols, T. E., Mandl, R. C., Almasy, L., Booth, T., Brouwer, R. M., Curran, J. E. & de Zubicaray, G. I. Multi-site study of additive genetic effects on fractional anisotropy of cerebral white matter: comparing meta and megaanalytical approaches for data pooling. *Neuroimage* **95**, 136-150 (2014).

96 Cao, H., McEwen, S. C., Forsyth, J. K., Gee, D. G., Bearden, C. E., Addington, J., Goodyear, B., Cadenhead, K. S., Mirzakhanian, H. & Cornblatt, B. A. Toward leveraging human connectomic data in large consortia: generalizability of fMRI-based brain graphs across sites, sessions, and paradigms. *Cerebral Cortex* **29**, 1263-1279 (2019).

97 Drew, D. A., Nguyen, L. H., Steves, C. J., Menni, C., Freydin, M., Varsavsky, T., Sudre, C. H., Cardoso, M. J., Ourselin, S. & Wolf, J. Rapid implementation of mobile technology for real-time epidemiology of COVID-19. *Science* **368**, 1362-1367 (2020).

98 Edelson, M. & Kuo, T.-T. Generalizable prediction of COVID-19 mortality on worldwide patient data. *JAMIA open* **5**, ooac036 (2022).

99 Kim, J., Neumann, L., Paul, P., Day, M. E., Aratow, M., Bell, D. S., Doctor, J. N., Hinske, L. C., Jiang, X. & Kim, K. K. Privacy-protecting, reliable response data discovery using COVID-19 patient observations. *Journal of the American Medical Informatics Association* **28**, 1765-1776 (2021).

100 Kim, J., Neumann, L., Paul, P., Day, M. E., Aratow, M., Bell, D. S., Doctor, J. N., Hinske, L. C., Jiang, X., Kim, K. K., Matheny, M. E., Meeker, D., Pletcher, M. J., Schilling, L. M., SooHoo, S., Xu, H., Zheng, K., Ohno-Machado, L. & Consortium, f. t. R. D. Privacy-protecting, reliable response data discovery using COVID-19 patient observations. *Journal of the American Medical Informatics Association* (2021). https://doi.org:10.1093/jamia/ocab054

101 Lardinois, F. *Microsoft says GitHub now has a $1B ARR, 90M active users*, <https://techcrunch.com/2022/10/25/microsoft-says-github-now-has-a-1b-arr-90m-active-users/> (2022).

102 Docs, G. *Access permissions on GitHub*, <https://docs.github.com/en/get-started/learning-about-github/access-permissions-on-github> (

103 McLaughlin, K. The Florida COVID-19 data 'whistleblower' crashed the state's dashboard and locked out her manager before she was fired, the National Review reports. (2021). <https://www.businessinsider.com/rebekah-jones-florida-covid-19-whistleblower-dashboard-national-review-2021-5>.

104 Grind, K. M., Robert. in *The Wall Street Journal* (The Wall Street Journal, 2023).

105 Pallini, T. *American Airlines and others carriers were left helpless after a system outage crippled operations, causing delays*, <https://www.yahoo.com/now/american-airlines-others-carriers-were-183048782.html> (2021).

106 Dutton, J. & Joyner, A. *United Airlines says system outage resolved after U.S. flights stopped*, <https://www.newsweek.com/united-airlines-ground-stop-computer-systems-down-1630138> (2021).

107 *Colonial Pipeline Cyber Incident*, <https://www.energy.gov/ceser/colonial-pipeline-cyber-incident> (2022).

108 Strickland, E. *5 Major Hospital Hacks: Horror Stories from the Cybersecurity Frontlines*, <https://spectrum.ieee.org/5-major-hospital-hacks-horror-stories-from-the-cyber-security-frontlines> (2016).

109 Harwell, D. *New York Stock Exchange outage adds to fears on financial markets*, <https://www.washingtonpost.com/business/economy/nyse-outage-raises-questions-about-whether-regulators-can-keep-up/2015/07/08/e8f7b02a-258d-11e5-b77f-eb13a215f593_story.html> (2023).

110 Ballinger, K. *April service disruptions analysis*, <https://github.blog/2020-05-22-april-service-disruptions-analysis/> (2020).

111 Ballinger, K. *An update on recent service disruptions*, <https://github.blog/2022-03-23-an-update-on-recent-service-disruptions/> (2022).

112 Nakamoto, S. Bitcoin: A peer-to-peer electronic cash system. *Decentralized Bus Rev*, 21260 (2008).

113 Kuo, T.-T., Kim, H.-E. & Ohno-Machado, L. Blockchain distributed ledger technologies for biomedical and health care applications. *Journal of the American Medical Informatics Association (JAMIA)* **24**, 1211-1220 (2017). https://doi.org:10.1093/jamia/ocx068

114 Atlassian. *git amend | Atlassian Git Tutorial*, <https://www.atlassian.com/git/tutorials/rewriting-history> (2023).

115 Göbel, T. & Baier, H. Anti-forensics in ext4: On secrecy and usability of timestamp-based data hiding. *Digital Investigation* **24**, S111-S120 (2018).

116 Ekblaw, A., Azaria, A., Halamka, J. D. & Lippman, A. in *ONC/NIST Use of Blockchain for Healthcare and Research Workshop* (Gaithersburg, Maryland, United States, 2016).

117 Azaria, A., Ekblaw, A., Vieira, T. & Lippman, A. in *International Conference on Open and Big Data (OBD)*. 25-30 (IEEE).

118 Kuo, T.-T. & Pham, A. Quorum-based Model Learning on a Blockchain Hierarchical Clinical Research Network using Smart Contracts. *International Journal of Medical Informatics (IJMI)* **169**, 104924 (2022). https://doi.org:10.1016/j.ijmedinf.2022.104924

119 Kuo, T.-T. & Pham, A. Detecting Model Misconducts in Decentralized Healthcare Federated Learning. *International Journal of Medical Informatics (IJMI)* **158**, 104658 (2022). https://doi.org:10.1016/j.ijmedinf.2021.104658

120 Kuo, T.-T., Kim, J. & Gabriel, R. A. Privacy-Preserving Model Learning on Blockchain Network-of-networks. *Journal of the American Medical Informatics Association (JAMIA)* **27**, 343–354 (2020). https://doi.org:10.1093/jamia/ocz214

121 Kuo, T.-T., Gabriel, R. A., Cidambi, K. R. & Ohno-Machado, L. EXpectation Propagation LOgistic REgRession on permissioned blockCHAIN (ExplorerChain): decentralized online healthcare/genomics predictive model learning. *Journal of the American Medical Informatics Association (JAMIA)* **27**, 747-756 (2020). https://doi.org:10.1093/jamia/ocaa023

122 Kuo, T.-T. The Anatomy of a Distributed Predictive Modeling Framework: Online Learning, Blockchain Network, and Consensus Algorithm. *Journal of the American Medical Informatics Association Open (JAMIA Open)* **3**, 201-208 (2020). https://doi.org:10.1093/jamiaopen/ooaa017

123 Kuo, T.-T., Gabriel, R. A. & Ohno-Machado, L. Fair compute loads enabled by blockchain: sharing models by alternating client and server roles. *Journal of the American Medical Informatics Association (JAMIA)* **26**, 392-403 (2019). https://doi.org:10.1093/jamia/ocy180

124 Kuo, T.-T., Jiang, X., Tang, H., Wang, X., Bath, T., Bu, D., Wang, L., Harmanci, A., Zhang, S., Zhi, D., Sofia, H. J. & Ohno-Machado, L. iDASH Secure Genome Analysis Competition 2018: Blockchain Genomic Data Access Logging, Homomorphic Encryption on GWAS, and DNA Segment Searching. *BMC Medical Genomics* **13**, 98 (2020). https://doi.org:10.1186/s12920-020-0715-0

125 Kuo, T.-T., Bath, T., Ma, S., Pattengale, N., Yang, M., Cao, Y., Hudson, C. M., Kim, J., Post, K., Xiong, L. & Ohno-Machado, L. Benchmarking Blockchain-Based Gene-Drug Interaction Data Sharing Methods: A Case Study from the iDASH 2019 Secure Genome Analysis Competition Blockchain Track. *International Journal of Medical Informatics (IJMI)* **154**, 104559 (2021). https://doi.org:10.1016/j.ijmedinf.2021.104559

126 Li, M. M. & Kuo, T.-T. Previewable Contract-Based On-Chain X-Ray Image Sharing Framework for Clinical Research. *International Journal of Medical Informatics (IJMI)* **156**, 104599 (2021). https://doi.org:10.1016/j.ijmedinf.2021.104599

127 Tellew, J. & Kuo, T.-T. CertificateChain: decentralized healthcare training certificate management system using blockchain and smart contracts. *Journal of the American Medical Informatics Association Open (JAMIA Open)* **5** (2022). https://doi.org:10.1093/jamiaopen/ooac019

128 Kuo, T.-T., Jiang, X., Tang, H., Wang, X., Harmanci, A., Kim, M., Post, K., Bu, D., Bath, T., Kim, J., Liu, W., Chen, H. & Ohno-Machado, L. The evolving privacy and security concerns for genomic data analysis and sharing as observed from the iDASH competition. *Journal of the American Medical Informatics Association (JAMIA)* (2022). https://doi.org:10.1093/jamia/ocac165

129 Kuo, T.-T., Zavaleta Rojas, H. & Ohno-Machado, L. Comparison of blockchain platforms: a systematic review and healthcare examples. *Journal of the American Medical Informatics Association (JAMIA)* **26**, 462-478 (2019). https://doi.org:10.1093/jamia/ocy185

130 Yu, H., Sun, H., Wu, D. & Kuo, T.-T. in *AMIA Annual Symposium.* (American Medical Informatics Association, Bethesda, MD, 2019).

131 Buterin, V. A next-generation smart contract and decentralized application platform. *white paper* **3**, 2-1 (2014).

132 TheEthereumCommunity. Go Ethereum (Geth). <https://github.com/ethereum/go-ethereum>.

133 Wood, G. Proof-of-Authority (PoA) Private Chains. (2015). <https://github.com/ethereum/guide/blob/master/poa.md>.

134 TheEthereumCommunity. Clique PoA protocol & Rinkeby PoA testnet. (2017). <https://github.com/ethereum/EIPs/issues/225>.

135 TheEthereumCommunity. The Solidity Contract-Oriented Programming Language. (2022). <https://github.com/ethereum/solidity>.

136 Web3Labs. web3j: Web3 Java Ethereum Dapp API. . <https://github.com/web3j/web3j>.

137 Liu, Z., Miao, Z., Zhan, X., Wang, J., Gong, B. & Yu, S. X. in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition.* 2537-2546.

138 Raza, S., Schwartz, B. & Rosella, L. C. CoQUAD: a COVID-19 question answering dataset system, facilitating research, benchmarking, and practice. *BMC bioinformatics* **23**, 1-28 (2022).

139 Badia, A. Question answering and database querying: Bridging the gap with generalized quantification. *Journal of Applied Logic* **5**, 3-19 (2007).

140 Hirschman, L. & Gaizauskas, R. Natural language question answering: the view from here. *natural language engineering* **7**, 275-300 (2001).

141 Fordtran, J. S. Colitis Due to Clostridium Difficile Toxins: Underdiagnosed, Highly Virulent, and Nosocomial. *https://doi.org/10.1080/08998280.2006.11928114* (2017). https://doi.org:10.1080/08998280.2006.11928114

142 M, R., MH, W. & DN, G. Clostridium difficile infection: new developments in epidemiology and pathogenesis. *Nature reviews. Microbiology* **7** (2009). https://doi.org:10.1038/nrmicro2164

143 Jump, R. L. & Donskey, C. J. Clostridium difficile in the long-term care facility: prevention and management. *Current geriatrics reports* **4**, 60-69 (2015).

144 *Clostridioides difficile Infection*, <https://www.cdc.gov/hai/organisms/cdiff/cdiff_infect.html> (2020).

145 Prevention, C. f. D. C. a. *Antibiotic Resistance Threats in the United States, 2019*, 2019).

146 *Current HAI Progress Report*, <https://www.cdc.gov/hai/data/portal/progress-report.html> (2019).

147 Alrawashdeh, M., Rhee, C., Hsu, H., Wang, R., Horan, K. & Lee, G. M. Assessment of Federal Value-Based Incentive Programs and In-Hospital Clostridioides difficile Infection Rates. *JAMA Network Open* **4**, e2132114-e2132114 (2021).

148 *Healthcare-Associated Infections Report UC Sand Diego Health*, 2017).

149 C, D., LP, H., R, B. & LT, J. Biocide Resistance and Transmission of Clostridium difficile Spores Spiked onto Clinical Surfaces from an American Health Care Facility. *Applied and environmental microbiology* **85** (2019). https://doi.org:10.1128/AEM.01090-19

150 Edwards, A. N., Karim, S. T., Pascual, R. A., Jowhar, L. M., Anderson, S. E. & McBride, S. M. Chemical and stress resistances of Clostridium difficile spores and vegetative cells. *Front Microbiol* **7**, 1698 (2016).

151 Rineh, A., Kelso, M. J., Vatansever, F., Tegos, G. P. & Hamblin, M. R. Clostridium difficile infection: molecular pathogenesis and novel therapeutics. *Expert Rev Anti Infect Ther* **12**, 131-150 (2014).

152 Longtin, Y., Paquet-Bolduc, B., Gilca, R., Garenc, C., Fortin, E., Longtin, J., Trottier, S., Gervais, P., Roussy, J.-F., Lévesque, S., Ben-David, D., Cloutier, I. & Loo, V. G. Effect of Detecting and Isolating Clostridium difficile Carriers at Hospital Admission on the Incidence of C difficile Infections: A Quasi-Experimental Controlled Study. *JAMA*

Internal Medicine **176**, 796-804 (2016). https://doi.org:10.1001/jamainternmed.2016.0177

153 Lee, H. S., Plechot, K., Gohil, S. & Le, J. Clostridium difficile: Diagnosis and the Consequence of Over Diagnosis. *Infectious diseases and therapy*, 1-11 (2021).

154 (ed Washington State Department of Health).

155 Zacharioudakis, I. M., Zervou, F. N., Pliakos, E. E., Ziakas, P. D. & Mylonakis, E. Colonization with toxinogenic C. difficile upon hospital admission, and risk of infection: a systematic review and meta-analysis. *Official journal of the American College of Gastroenterology| ACG* **110**, 381-390 (2015).

156 Curry, S. R., Muto, C. A., Schlackman, J. L., Pasculle, A. W., Shutt, K. A., Marsh, J. W. & Harrison, L. H. Use of multilocus variable number of tandem repeats analysis genotyping to determine the role of asymptomatic carriers in Clostridium difficile transmission. *Clinical Infectious Diseases* **57**, 1094-1102 (2013).

157 Biswas, J. S., Patel, A., Otter, J. A., van Kleef, E. & Goldenberg, S. D. Contamination of the hospital environment from potential Clostridium difficile excretors without active infection. *infection control & hospital epidemiology* **36**, 975-977 (2015).

158 Wiens, J., Horvitz, E. & Guttag, J. Patient Risk Stratification for Hospital-Associated C. diff as a Time-Series Classification Task. *Advances in Neural Information Processing Systems* **25** (2021).

159 Dubberke, E. R., Yan, Y., Reske, K. A., Butler, A. M., Doherty, J., Pham, V. & Fraser, V. J. Development and validation of a Clostridium difficile infection risk prediction model. *Infection Control & Hospital Epidemiology* **32**, 360-366 (2011).

160 Oh, J., Makar, M., Fusco, C., McCaffrey, R., Rao, K., Ryan, E. E., Washer, L., West, L. R., Young, V. B. & Guttag, J. A generalizable, data-driven approach to predict daily risk of Clostridium difficile infection at two large academic health centers. *infection control & hospital epidemiology* **39**, 425-433 (2018).

161 Katz, D. A., Lynch, M. E. & Littenberg, B. Clinical prediction rules to optimize cytotoxin testing for Clostridium difficile in hospitalized patients with diarrhea. *The American journal of medicine* **100**, 487-495 (1996).

162 Pham, A., El-Kareh, R., Ohno-Machado, L. & Kuo, T. in *AMIA Annual Symposium.*

163 *Pharmacologic Class*, <https://www.fda.gov/industry/structured-product-labeling-resources/pharmacologic-class> (

164 Kuo, T.-T., Rao, P., Maehara, C., Doan, S., Chaparro, J. D., Day, M. E., Farcas, C., Ohno-Machado, L. & Hsu, C.-N. in *AMIA Annual Symposium Proceedings.* 1880 (American Medical Informatics Association).

165 Kuo, T.-T., Kim, J. & Gabriel, R. A. Privacy-preserving model learning on a blockchain network-of-networks. *Journal of the American Medical Informatics Association* **27**, 343-354 (2020).

166 *Scikit-learn: Machine Learning in Python*, <https://scikit-learn.org/stable/> (2022).

167 *Matplotlib: Visualization with Python*, <https://matplotlib.org/> (2022).

168 SciPy. (2022).

169 Lasko, T. A., Bhagwat, J. G., Zou, K. H. & Ohno-Machado, L. The use of receiver operating characteristic curves in biomedical informatics. *Journal of biomedical informatics* **38**, 404-415 (2005).

170 Klünemann, M., Andrejev, S., Blasche, S., Mateus, A., Phapale, P., Devendran, S., Vappiani, J., Simon, B., Scott, T. A., Kafkia, E., Konstantinidis, D., Zirngibl, K., Mastrorilli, E., Banzhaf, M., Mackmull, M.-T., Hövelmann, F., Nesme, L., Brochado, A. R., Maier, L., Bock, T., Periwal, V., Kumar, M., Kim, Y., Tramontano, M., Schultz, C., Beck, M., Hennig, J., Zimmermann, M., Sévin, D. C., Cabreiro, F., Savitski, M. M., Bork, P., Typas, A. & Patil, K. R. Bioaccumulation of therapeutic drugs by human gut bacteria. *Nature* **597**, 533-538 (2021). https://doi.org:10.1038/s41586-021-03891-8

171 Imhann, F., Vich Vila, A., Bonder, M. J., Lopez Manosalva, A. G., Koonen, D. P., Fu, J., Wijmenga, C., Zhernakova, A. & Weersma, R. K. The influence of proton pump inhibitors and other commonly used medication on the gut microbiota. *Gut microbes* **8**, 351-358 (2017).

172 Aldape, M. J., Packham, A. E., Nute, D. W., Bryant, A. E. & Stevens, D. L. Effects of ciprofloxacin on the expression and production of exotoxins by Clostridium difficile. *Journal of medical microbiology* **62**, 741 (2013).

173 Louppe, G., Wehenkel, L., Sutera, A. & Geurts, P. Understanding variable importances in forests of randomized trees. *Advances in neural information processing systems* **26**, 431-439 (2013).

174 Baxter, S. L., Marks, C., Kuo, T.-T., Ohno-Machado, L. & Weinreb, R. N. Machine learning-based predictive modeling of surgical intervention in glaucoma using systemic data from electronic health records. *American journal of ophthalmology* **208**, 30-40 (2019).

175 Pépin, J., Valiquette, L. & Cossette, B. Mortality attributable to nosocomial Clostridium difficile–associated disease during an epidemic caused by a hypervirulent strain in Quebec. *Cmaj* **173**, 1037-1042 (2005).

176 Deshpande, A., Pasupuleti, V., Thota, P., Pant, C., Rolston, D. D., Sferra, T. J., Hernandez, A. V. & Donskey, C. J. Community-associated Clostridium difficile infection and antibiotics: a meta-analysis. *Journal of Antimicrobial Chemotherapy* **68**, 1951-1961 (2013).

177 Pépin, J., Saheb, N., Coulombe, M.-A., Alary, M.-E., Corriveau, M.-P., Authier, S., Leblanc, M., Rivard, G., Bettez, M., Primeau, V., Nguyen, M., Jacob, C.-É. & Lanthier, L. Emergence of Fluoroquinolones as the Predominant Risk Factor for Clostridium difficile–Associated Diarrhea: A Cohort Study during an Epidemic in Quebec. *Clinical Infectious Diseases* **41**, 1254-1260 (2005). https://doi.org:10.1086/496986

178 Vardakas, K. Z., Trigkidis, K. K., Boukouvala, E. & Falagas, M. E. Clostridium difficile infection following systemic antibiotic administration in randomised controlled trials: a systematic review and meta-analysis. *International journal of antimicrobial agents* **48**, 1-10 (2016).

179 Leffler, D. A. & Lamont, J. T. Clostridium difficile infection. *New England Journal of Medicine* **372**, 1539-1548 (2015).

180 Warn, P., Thommes, P., Sattar, A., Corbett, D., Flattery, A., Zhang, Z., Black, T., Hernandez, L. D. & Therien, A. G. Disease progression and resolution in rodent models of Clostridium difficile infection and impact of antitoxin antibodies and vancomycin. *Antimicrobial agents and chemotherapy* **60**, 6471-6482 (2016).

181 Starr, J., Campbell, A., Renshaw, E., Poxton, I. & Gibson, G. Spatio-temporal stochastic modelling of Clostridium difficile. *Journal of Hospital Infection* **71**, 49-56 (2009).

182 Zhang, S., Palazuelos-Munoz, S., Balsells, E. M., Nair, H., Chit, A. & Kyaw, M. H. Cost of hospital management of Clostridium difficile infection in United States—a meta-analysis and modelling study. *BMC infectious diseases* **16**, 1-18 (2016).

183 Rodrigues, R., Barber, G. E. & Ananthakrishnan, A. N. A comprehensive study of costs associated with recurrent Clostridium difficile infection. *infection control & hospital epidemiology* **38**, 196-202 (2017).

184 *How Much Does an Stool, C-Diff Test Cost Near Me? - MDsave*, <https://www.mdsave.com/procedures/stool-c-diff-test/d787ffc4> (2021).

185 Murray, S. G., Yim, J. W. L., Croci, R., Rajkomar, A., Schmajuk, G., Khanna, R. & Cucina, R. J. Using Spatial and Temporal Mapping to Identify Nosocomial Disease Transmission of Clostridium difficile. *JAMA Internal Medicine* **177**, 1863-1865 (2017). https://doi.org:10.1001/jamainternmed.2017.5506

186 Margolis, R., Derr, L., Dunn, M., Huerta, M., Larkin, J., Sheehan, J., Guyer, M. & Green, E. D. The National Institutes of Health's Big Data to Knowledge (BD2K) initiative: capitalizing on biomedical big data. *J Am Med Inform Assoc* **21**, 957-958 (2014). https://doi.org:10.1136/amiajnl-2014-002974

187 Tedersoo, L., Kungas, R., Oras, E., Koster, K., Eenmaa, H., Leijen, A., Pedaste, M., Raju, M., Astapova, A., Lukner, H., Kogermann, K. & Sepp, T. Data sharing practices and data availability upon request differ across scientific disciplines. *Sci Data* **8**, 192 (2021). https://doi.org:10.1038/s41597-021-00981-0

188 Rights, O. f. C. *A decision tool: Limited Data Set (LDS)*, <https://www.hhs.gov/hipaa/for-professionals/special-topics/emergency-preparedness/limited-data-set/index.html > (2022).

189 Division, D. C. *What is phi?* , <https://www.hhs.gov/answers/hipaa/what-is-phi/index.html > (2021).

190 *Guidance on the protection of personal identifiable information*, < https://www.dol.gov/general/ppii> (

191 *Limited Data Set: The Complete Guide*, <https://www.clarity-ventures.com/glossary/limited-dataset#:~:text=Why%20Is%20a%20Limited%20Data,treatments%2C%20and%20advance%20medical%20research.> (

192 *HIPAA privacy rule and its impacts on research*, <https://privacyruleandresearch.nih.gov/pr_08.asp> (

193 Rights, O. f. C. *Research*, <https://www.hhs.gov/hipaa/for-professionals/special-topics/research/index.html > (2021).

194 *Research, ethics, and compliance training*, <https://about.citiprogram.org/> (

195 Rights, O. f. C. *A decision tool: Data Use Agreement*, <https://www.hhs.gov/hipaa/for-professionals/special-topics/emergency-preparedness/data-use-agreement/index.html.> (2022).

196 *Data ownership*, <https://ori.hhs.gov/content/Chapter-6-Data-Management-Practices-Data-ownership> (

197 Champieux, R., Solomonides, A., Conte, M., Rojevsky, S., Phuong, J., Dorr, D. A., Zampino, E., Wilcox, A., Carson, M. B. & Holmes, K. Ten simple rules for organizations to support research data sharing. *PLoS Comput Biol* **19**, e1011136 (2023). https://doi.org:10.1371/journal.pcbi.1011136

198 C., T. *Manual Process vs Automated Process: A Comparison Guide*, <https://www.comidor.com/blog/business-process-management/a-comparison-of-manual-and-automated-workflow-processes/ > (2019).

199 Gumrukcu, E., Arsalan, A., Muriithi, G., Joglekar, C., Aboulebdeh, A., Alparslan Zehir, M., Papari, B. & Monti, A. in *2022 4th Global Power, Energy and Communication Conference (GPECOM)* 506-511 (2022).

200 Kuo, T. T., Kim, H. E. & Ohno-Machado, L. Blockchain distributed ledger technologies for biomedical and health care applications. *J Am Med Inform Assoc* **24**, 1211-1220 (2017). https://doi.org:10.1093/jamia/ocx068

201 R, L. *Amazon's server outage broke fast food apps like McDonald's and Taco Bell*, <https://www.theverge.com/2023/6/13/23759857/amazon-aws-down-outage-taco-bell-mcdonalds-burger-king> (2023).

202 D., T. *Pa. Firm alleges ex-partners 'secretly planned' to join Armstrong Teasdale*, <https://today.westlaw.com/Document/I1c80a1d070ac11ebb555947e94fe83f6/View/FullText.html?transitionType=SearchItem&contextData=%28sc.Default%29&firstPage=true> (2021).

203 U.S. Attorney's Office, W. D. o. W. *Former Seattle Tech worker convicted of wire fraud and computer intrusions. Former Seattle tech worker convicted of wire fraud and computer intrusions*, <https://www.justice.gov/usao-wdwa/pr/former-seattle-tech-worker-convicted-wire-fraud-and-computer-intrusions> (2022).

204 Jahansoozi, J. & McManus, T. Organization-stakeholder relationships: exploring trust and transparency. *Journal of Management Development* **25**, 942-955 (2006). https://doi.org:10.1108/02621710610708577

205 Lacson, R., Yu, Y., Kuo, T. T. & Ohno-Machado, L. Biomedical blockchain with practical implementations and quantitative evaluations: a systematic review. *J Am Med Inform Assoc* **31**, 1423-1435 (2024). https://doi.org:10.1093/jamia/ocae084

206 Dimitrov, D. V. Blockchain Applications for Healthcare Data Management. *Healthcare Informatics Research* **25** (2019). https://doi.org:10.4258/hir.2019.25.1.51

207 Yaqoob, I., Salah, K., Jayaraman, R. & Al-Hammadi, Y. Blockchain for healthcare data management: opportunities, challenges, and future recommendations. *Neural Computing and Applications* **34**, 11475-11490 (2021). https://doi.org:10.1007/s00521-020-05519-w

208 Kuo, T. T., Jiang, X., Tang, H., Wang, X., Harmanci, A., Kim, M., Post, K., Bu, D., Bath, T., Kim, J., Liu, W., Chen, H. & Ohno-Machado, L. The evolving privacy and security concerns for genomic data analysis and sharing as observed from the iDASH competition. *J Am Med Inform Assoc* **29**, 2182-2190 (2022). https://doi.org:10.1093/jamia/ocac165

209 Kuo, T. T., Pham, A., Edelson, M. E., Kim, J., Chan, J., Gupta, Y., Ohno-Machado, L. & Consortium, R. D. Blockchain-enabled immutable, distributed, and highly available clinical research activity logging system for federated COVID-19 data analysis from multiple institutions. *J Am Med Inform Assoc* **30**, 1167-1178 (2023). https://doi.org:10.1093/jamia/ocad049

210 *Medicalchain*, <https://medicalchain.com/en/> (

211 Kuo, T. T., Bath, T., Ma, S., Pattengale, N., Yang, M., Cao, Y., Hudson, C. M., Kim, J., Post, K., Xiong, L. & Ohno-Machado, L. Benchmarking blockchain-based gene-drug interaction data sharing methods: A case study from the iDASH 2019 secure genome analysis competition blockchain track. *Int J Med Inform* **154**, 104559 (2021). https://doi.org:10.1016/j.ijmedinf.2021.104559

212 Kuo, T. T., Jiang, X., Tang, H., Wang, X., Bath, T., Bu, D., Wang, L., Harmanci, A., Zhang, S., Zhi, D., Sofia, H. J. & Ohno-Machado, L. iDASH secure genome analysis competition 2018: blockchain genomic data access logging, homomorphic encryption on GWAS, and DNA segment searching. *BMC Med Genomics* **13**, 98 (2020). https://doi.org:10.1186/s12920-020-0715-0

213 *BurstIQ*, <https://burstiq.com/> (

214 Li, M. M. & Kuo, T. T. Previewable Contract-Based On-Chain X-Ray Image Sharing Framework for Clinical Research. *Int J Med Inform* **156**, 104599 (2021). https://doi.org:10.1016/j.ijmedinf.2021.104599

215 *ProCredEx*, <https://procredex.com/ > (

216 Christidis, K. & Devetsikiotis, M. Blockchains and smart contracts for the internet of things. *IEEE access* **4**, 2292-2303 (2016).

217 Curbera, F., Dias, D. M., Simonyan, V., Yoon, W. A. & Casella, A. Blockchain: An enabler for healthcare and life sciences transformation. *IBM Journal of Research and Development* **63**, 8:1-8:9 (2019). https://doi.org:10.1147/jrd.2019.2913622

218 Pincheira, M., Donini, E., Vecchio, M. & Kanhere, S. A Decentralized Architecture for Trusted Dataset Sharing Using Smart Contracts and Distributed Storage. *Sensors (Basel)* **22** (2022). https://doi.org:10.3390/s22239118

219 Purohit, S., Calyam, P., Alarcon, M. L., Bhamidipati, N. R., Mosa, A. & Salah, K. HonestChain: Consortium blockchain for protected data sharing in health information systems. *Peer Peer Netw Appl* **14**, 3012-3028 (2021). https://doi.org:10.1007/s12083-021-01153-y

220 Tellew, J. & Kuo, T. T. CertificateChain: decentralized healthcare training certificate management system using blockchain and smart contracts. *JAMIA Open* **5**, ooac019 (2022). https://doi.org:10.1093/jamiaopen/ooac019

221 Nyaletey, E., Parizi, R. M., Zhang, Q. & Choo, K.-K. R. in *2019 IEEE International Conference on Blockchain (Blockchain)* 18-25 (2019).

222 H, Y., H, S., D, W. & T-T, K. in *AMIA Annual Symposium: American Medical Informatics Association.*

223 Kuo, T. T., Zavaleta Rojas, H. & Ohno-Machado, L. Comparison of blockchain platforms: a systematic review and healthcare examples. *J Am Med Inform Assoc* **26**, 462-478 (2019). https://doi.org:10.1093/jamia/ocy185

224 *Poa private chains*, <https://github.com/ethereum/guide/blob/master/poa.md> (

225 *Web3j: Lightweight java and Android Library for integration with Ethereum clients*, <https://github.com/web3j/web3j > (

226 Agarap, A. F. M. in *Proceedings of the 2nd international conference on machine learning and soft computing.* 5-9.

227 Battineni, G., Sagaro, G. G., Chinatalapudi, N. & Amenta, F. Applications of machine learning predictive models in the chronic disease diagnosis. *Journal of personalized medicine* **10**, 21 (2020).

228 Kononenko, I. Machine learning for medical diagnosis: history, state of the art and perspective. *Artificial Intelligence in medicine* **23**, 89-109 (2001).

229 Olsen, C. R., Mentz, R. J., Anstrom, K. J., Page, D. & Patel, P. A. Clinical applications of machine learning in the diagnosis, classification, and prediction of heart failure. *American Heart Journal* **229**, 1-17 (2020).

230 Richens, J. G., Lee, C. M. & Johri, S. Improving the accuracy of medical diagnosis with causal machine learning. *Nature communications* **11**, 3923 (2020).

231 Kassania, S. H., Kassanib, P. H., Wesolowskic, M. J., Schneidera, K. A. & Detersa, R. Automatic detection of coronavirus disease (COVID-19) in X-ray and CT images: a machine learning based approach. *Biocybernetics and Biomedical Engineering* **41**, 867-879 (2021).

232 Wang, S. & Summers, R. M. Machine learning and radiology. *Medical image analysis* **16**, 933-951 (2012).

233 Thrall, J. H., Li, X., Li, Q., Cruz, C., Do, S., Dreyer, K. & Brink, J. Artificial intelligence and machine learning in radiology: opportunities, challenges, pitfalls, and criteria for success. *Journal of the American College of Radiology* **15**, 504-508 (2018).

234 Choy, G., Khalilzadeh, O., Michalski, M., Do, S., Samir, A. E., Pianykh, O. S., Geis, J. R., Pandharipande, P. V., Brink, J. A. & Dreyer, K. J. Current applications and future impact of machine learning in radiology. *Radiology* **288**, 318-328 (2018).

235 Lavecchia, A. Machine-learning approaches in drug discovery: methods and applications. *Drug discovery today* **20**, 318-331 (2015).

236 Patel, L., Shukla, T., Huang, X., Ussery, D. W. & Wang, S. Machine learning methods in drug discovery. *Molecules* **25**, 5277 (2020).

237 Ekins, S., Puhl, A. C., Zorn, K. M., Lane, T. R., Russo, D. P., Klein, J. J., Hickey, A. J. & Clark, A. M. Exploiting machine learning for end-to-end drug discovery and development. *Nature materials* **18**, 435-441 (2019).

238 Artificial Intelligence: Healthcare's New Nervous System. (2020).

239 Callahan, A. & Shah, N. H. in *Key advances in clinical informatics* 279-291 (Elsevier, 2017).

240 Javaid, M., Haleem, A., Singh, R. P., Suman, R. & Rab, S. Significance of machine learning in healthcare: Features, pillars and applications. *International Journal of Intelligent Networks* **3**, 58-73 (2022).

241 Abouelmehdi, K., Beni-Hessane, A. & Khaloufi, H. Big healthcare data: preserving security and privacy. *Journal of big data* **5**, 1-18 (2018).

242 Sheller, M. J., Edwards, B., Reina, G. A., Martin, J., Pati, S., Kotrotsou, A., Milchenko, M., Xu, W., Marcus, D. & Colen, R. R. Federated learning in medicine: facilitating multi-institutional collaborations without sharing patient data. *Scientific reports* **10**, 12598 (2020).

243 Kuo, T.-T. & Pham, A. Detecting model misconducts in decentralized healthcare federated learning. *International journal of medical informatics* **158**, 104658 (2022).

244 Kuo, T.-T., Gabriel, R. A. & Ohno-Machado, L. Fair compute loads enabled by blockchain: sharing models by alternating client and server roles. *Journal of the American Medical Informatics Association* **26**, 392-403 (2019).

245 Zerka, F., Barakat, S., Walsh, S., Bogowicz, M., Leijenaar, R. T., Jochems, A., Miraglio, B., Townend, D. & Lambin, P. Systematic review of privacy-preserving distributed machine learning from federated databases in health care. *JCO clinical cancer informatics* **4**, 184-200 (2020).

246 Kim, Y. J. & Hong, C. S. in *2019 20th Asia-Pacific Network Operations and Management Symposium (APNOMS)*. 1-4 (IEEE).

247 Zhang, H., Bosch, J. & Olsson, H. H. in *2020 27th Asia-Pacific Software Engineering Conference (APSEC)*. 385-394 (IEEE).

248 Hu, C., Jiang, J. & Wang, Z. Decentralized federated learning: A segmented gossip approach. *arXiv preprint arXiv:1908.07782* (2019).

249 Kuo, T.-T. The anatomy of a distributed predictive modeling framework: online learning, blockchain network, and consensus algorithm. *JAMIA open* **3**, 201-208 (2020).

250 Kuo, T.-T. & Pham, A. Quorum-based model learning on a blockchain hierarchical clinical research network using smart contracts. *International journal of medical informatics* **169**, 104924 (2023).

251 Ludwig, H., Baracaldo, N., Thomas, G., Zhou, Y., Anwar, A., Rajamoni, S., Ong, Y., Radhakrishnan, J., Verma, A. & Sinn, M. Ibm federated learning: an enterprise framework white paper v0. 1. *arXiv preprint arXiv:2007.10987* (2020).

252 Jiang, W., Li, P., Wang, S., Wu, Y., Xue, M., Ohno-Machado, L. & Jiang, X. WebGLORE: a web service for Grid LOgistic REgression. *Bioinformatics* **29**, 3238-3240 (2013).

253 Burlachenko, K., Horváth, S. & Richtárik, P. in *Proceedings of the 2nd ACM International Workshop on Distributed Machine Learning*. 1-7.

254 Matschinske, J., Späth, J., Nasirigerdeh, R., Torkzadehmahani, R., Hartebrodt, A., Orbán, B., Fejér, S., Zolotareva, O., Bakhtiari, M. & Bihari, B. The featurecloud ai store for federated learning in biomedicine and beyond. *arXiv preprint arXiv:2105.05734* (2021).

255 Liu, W., Zhang, Y. H., Li, Y. F. & Zheng, D. A fine-grained medical data sharing scheme based on federated learning. *Concurrency and Computation: Practice and Experience* **35**, e6847 (2023).

256 Zou, K. H., Liu, A., Bandos, A. I., Ohno-Machado, L. & Rockette, H. E. *Statistical evaluation of diagnostic performance: topics in ROC analysis*. (CRC press, 2011).

257 Kennedy, R., Fraser, H., McStay, L. & Harrison, R. Early diagnosis of acute myocardial infarction using clinical and electrocardiographic data at presentation: derivation and evaluation of logistic regression models. *European heart journal* **17**, 1181-1191 (1996).

258 Christudas, B. & Christudas, B. ActiveMQ. *Practical Microservices Architectural Patterns: Event-Based Java Microservices with Spring Boot and Spring Cloud*, 861-867 (2019).

259 Walls, C. *Spring Boot in action*. (Simon and Schuster, 2015).

260 Gackenheimer, C. *Introduction to React*. (Apress, 2015).

261 Ely, D., Savage, S. & Wetherall, D. in *3rd USENIX Symposium on Internet Technologies and Systems (USITS 01)*.

262 *Java | Oracle*, <https://www.java.com/en/> (2023).

263 Ometov, A., Bezzateev, S., Mäkitalo, N., Andreev, S., Mikkonen, T. & Koucheryavy, Y. Multi-factor authentication: A survey. *Cryptography* **2**, 1 (2018).

264 Sandhu, R. S. in *Advances in computers* Vol. 46 237-286 (Elsevier, 1998).