TÉCNICO LISBOA

How to disappear online

approaches for digital erasure and evaluation methodologies

João Vasco Almeida Sobral Siborro Reis

Thesis to obtain the Master of Science Degree in

Computer Science and Engineering

Supervisors: Prof. Cláudia Martins Antunes
Prof. David Rogério Povoa de Matos

Examination Committee

Chairperson: Prof. -
Supervisor: Prof. Cláudia Martins Antunes
Member of the Committee: Prof. Mário Jorge Costa Gaspar da Silva

October 2024

Declaration
I declare that this document is an original work of my own authorship and that it fulfills all the requirements of the Code of Conduct and Good Practices of the Universidade de Lisboa.

Acknowledgments

First of all, I want to thank myself. I know I am a highly determined person, and above all, focused. It took me a long time to get here, and I feel like this is a great ending to a story that I have been writing. This journey has had many smiles, a lot of hard work, many tears, and also plenty of parties, but what truly matters at the end of the day is consistency and persistence, and I believe I'm good at that.

Secondly, I want to express my gratitude to my mum and dad for their unwavering support and, of course, for being the sponsors of this journey—haha! A special thanks to my grandparents for their constant encouragement, and to my brother, who's also embarking on a new career path. He's probably the most similar person to me, and though none of them wrote my thesis, they helped in countless other ways and were always there for me.

Without a doubt, I want to thank my supervisors. Starting with Professor Cláudia (my first professor at university and certainly my favorite), she was a teacher who sought to connect with students and provided close guidance. I remember her coming to my seat to ask why I had done an exercise a certain way. Honestly, it felt like I was still in high school, and I thought that university would continue to be like that. However, she was an exception because I never had another professor like her. After that, the distance between professors and students became more noticeable, and I felt it throughout these five years. The fact that she set up weekly meetings forced me to keep working, and honestly, I feel that with other supervisors, I might not have reached the point where I am today. The help from Professor David was also invaluable. In a different field, he helped me adapt quickly to challenges that would have taken me ages on my own—he is super responsible and pragmatic. So, a big thank you to both professors, who have undoubtedly shaped my career at IST the most.

I also want to thank my friends from Colégio Manuel Bernardes, especially the guys who still organize weekly lunches at the Indian restaurant. A thank you to the people I met at the University of Lisbon and to the two girls I met on my first day at IST, who ended up becoming my best friends and a huge support without a doubt. Also a thanks to Trutas for all the late night study times.

Now, the most emotional part for me: a big thank you to Sungkyunkwan University (and Professor Gaya), where I studied during my first year of my master's. Honestly, I had forgotten what it felt like to enjoy learning, to enjoy working, and above all, they fostered what I can call "a healthy learning

environment.” I felt like I was back in high school; all the professors knew me, constantly talked to me about my academic progress and homework. I felt that the professors were truly my friends. I want to write this down because realizing that university can be like this is something I never want to forget. It was the best year of my life—the year I studied the most, had the most fun, never missed a class, matured the most, and learned the most. It didn't require living a life of daily stress to acquire knowledge, and it was also the year where I got the best grades ”ever.”

To my friends from there (korean and Sungkyunkwan University friends), I also want to say a big thank you to my bros: Jejin, Gyuri, YoonChang, Youngmin, Shelby and I could keep this list going indefinitely, so here's a heartfelt thanks to all those who were with me, and a special hug to (PEH GOH REE), where I found the best friends I could ask for. To the friends living there, and those trying to make a life there—you know who you are when you read this—we'll undoubtedly cross paths many more times in Seoul.

To my friends abroad, a huge special thank you to my friends from Spain, the Czech Republic, Mexico, Peru, the United States, France, Malaysia, Turkey, Italy, Germany, Austria and Brazil—you surely know who you are.

A big thank you to the people who are helping me start this new chapter at Unbabel. To be honest, I was really scared, and I've always struggled with a huge sense of impostor syndrome when it comes to computer science. But they've shown me a more human side, which is something not everyone displays in the competitive environment of university. So, thanks especially to the integrations team, from whom I'm learning so much and hope to keep learning even more! They've been such great mentors.

I'm sure there are people I'm forgetting, so a big thank you to everyone who has been part of my journey over the years, whether directly or indirectly. I'd like to end with a special thanks to Cristiano Ronaldo for his determination, which has been a huge source of inspiration for me. Thank you all!

Abstract

In the digital age, personal data privacy has become a critical concern, driven by the proliferation of online platforms and the increasing volume of data storage. This thesis explores the challenges and methodologies surrounding digital erasure, with a particular focus on compliance with privacy regulations such as the General Data Protection Regulation (GDPR), the California Consumer Privacy Act (CCPA), and the Artificial Intelligence Act (AIA). The work investigates machine learning techniques, especially Random Forest and Decision Tree classifiers, to track, match, and prepare personal data for deletion across complex systems. By adapting intrusion recovery methodologies, this research proposes a system for accurately identifying and managing data marked for deletion, ensuring organizations can meet regulatory obligations. The analysis demonstrates how machine learning models are particularly effective for matching data, with this thesis focusing specifically on the identification and matching of logs as a crucial step in the broader process of data deletion. This thesis contributes to the broader discourse on data privacy by addressing the practical challenges of data management in distributed environments, paving the way for an initial approach for what would be one of the steps in order to obtain a more secure and compliant data deletion.

Keywords

Data Privacy; Digital Erasure; Swedish Act; General Data Protection Regulation (GDPR); California Consumer Privacy Act (CCPA); Artificial Intelligence Act (AIA); Intrusion Recovery; Machine Learning;

Resumo

Na era digital, a privacidade dos dados pessoais tornou-se uma preocupação importante, devido ao aumento das plataformas online e ao grande volume de armazenamento de dados. Esta tese explora os desafios e métodos relacionados com a eliminação de dados digitais, com foco no cumprimento de leis de privacidade como o Regulamento Geral de Proteção de Dados (RGPD), a California Consumer Privacy Act (CCPA) e a Lei de Inteligência Artificial (AIA). O trabalho investiga técnicas de machine learning, especialmente os classificadores Random Forest e Decision Tree, para rastrear, corresponder e preparar dados pessoais para eliminação em sistemas complexos. Adaptando métodos de recuperação de intrusões, esta pesquisa propõe um sistema para identificar e gerir com precisão os dados que precisam de ser apagados, garantindo que as organizações cumpram as suas obrigações legais. A análise mostra que os modelos de machine learning são especialmente eficazes na correspondência de dados, sendo que esta tese se concentra na identificação e correspondência de registos, uma parte essencial do processo mais amplo de eliminação de dados. Esta tese contribui para o debate sobre privacidade de dados, abordando os desafios práticos da gestão de dados em ambientes distribuídos, abrindo caminho para uma abordagem inicial de um dos passos necessários para alcançar uma eliminação de dados mais segura e conforme às leis.

Palavras Chave

Privacidade de Dados; Eliminação Digital; Lei Sueca; Regulamento Geral de Proteção de Dados (RGPD); California Consumer Privacy Act (CCPA); Artificial Intelligence Act (AIA); Recuperação de Intrusões; Aprendizagem Automática

Contents

List of Figures

List of Tables

Acronyms

AI Artificial Intelligence

AIA Artificial Intelligence Act

AUC Area Under the Curve

CAGR Compound Annual Growth Rate

CCPA California Consumer Privacy Act

CMS Content Management System

COPPA Children's Online Privacy Protection Act

CSV Comma-Separated Values

DB Database

DNNs Deep Neural Networks

EU European Union

FKGL Flesch Kincaid Grade Level

FP False Positives

FN False Negatives

GDPR General Data Protection Regulation

HTTP Hypertext Transfer Protocol

IBM International Business Machines

IE Traditional Information Extraction

LGPD General Data Protection Law

MAMP Macintosh, Apache, MySQL, PHP

MFTEM Multi Fractal Trust Evaluation Model

PaaS Platform as a Service

PDPA Personal Data Protection Act

PIPA Personal Information Protection Act

PIPEDA Personal Information Protection and Electronic Documents Act

POPIA Protection of Personal Information Act

QoS Quality of Service

RGPD Regulamento Geral de Proteção de Dados

SHAP SHapley Additive exPlanations

SQL Structured Query Language

TN True Negatives

TP True Positives

1

Introduction

Contents

In the contemporary digital era, the significance of information management has escalated more than ever. As elucidated in the work by Gates and Matthews, personal data has emerged as a vital asset in realms such as targeted marketing and advertising, with its value extending even to illicit black markets [1]. This monetization of personal data raises profound concerns regarding data ownership and privacy. The ambiguity surrounding legal ownership of data and the existing legal frameworks pose significant challenges, especially in the context of digital erasure [1].

Organizations are faced with the increasingly difficult challenge of managing and protecting vast amounts of personal data. Regulations grant individuals the right to request the deletion of their personal information, making data erasure a crucial component of compliance. However, ensuring the complete

and secure deletion of data across complex and distributed systems is far from straightforward [2]. Large organizations, with vast databases and intricate data flows, struggle to implement effective solutions for tracking, identifying, and securely erasing data [3]. This thesis addresses this problem by exploring innovative approaches to digital erasure, focusing on the evaluation of techniques that can help organizations prepare for and manage the process of securely removing data in compliance with privacy regulations.

1.1 Overview of Regulatory Frameworks in Data Privacy

To combat data corruption and respect the privacy of individuals, several regulatory approaches have emerged. With the appearance of the General Data Protection Regulation (GDPR) [4], the Artificial Intelligence Act (AIA) [5], and the California Consumer Privacy Act (CCPA) [6], there has been a significant shift in the landscape of data privacy and digital ethics.

The **General Data Protection Regulation (GDPR)**, focuses on protecting and empowering all European Union (EU) citizens' data privacy and reshaping the way organizations across the region approach the topic. It grants individuals greater control over their personal data and harmonizes data privacy laws across Europe, affecting how organizations handle and process personal data.

Meanwhile, the **Artificial Intelligence Act (AIA)**, proposed by the European Commission, is a pioneering framework designed to regulate the use of artificial intelligence within the European Union. The Act aims to ensure the safety, transparency, and accountability of Artificial Intelligence (AI) systems, establishing stringent standards for high-risk AI applications to protect fundamental rights and user safety.

In the United States, the **California Consumer Privacy Act (CCPA)** serves as a landmark law that enhances privacy rights and consumer protection for residents of California. The CCPA empowers California residents with rights such as knowing about which personal data was collected about them, deleting personal data held by businesses, and opting out of the sale of their personal data.

1.2 Challenges of Digital Erasure and Compliance

These frameworks collectively signify a global movement towards enhanced data privacy, ethical deployment of AI, and digital society strengthening consumer rights in the digital domain, setting critical standards for organizations globally. On the other hand, is the absence of practical solutions to erase the data. The implementation of the (GDPR) has posed significant challenges for many technology companies, requiring a comprehensive reevaluation of their data handling practices. Notably, the financial and operational burdens associated with achieving compliance have been substantial, often culminating in heavy sanctions for non-compliance. This situation is exemplified in a study, which delves into the experiences of nine major tech companies represented in Table 1.1, revealing the multifaceted nature

of GDPR compliance issues [7].

Company	Nature of (Alleged) GDPR Violation
Google	In January 2019, the French Data Protection Authority (CNIL) fined Google €50 million for lack of transparency and failure to obtain user consent for processing personal data for ad targeting.
Amazon	In 2021, Amazon was fined €746 million following a complaint by a French privacy rights group for using users' data for ad targeting.
Microsoft	Since 2018, Dutch investigators have been investigating Microsoft for massive data collection in the Netherlands without authorization from data subjects.
IBM	Although IBM has had some records of violation under state laws, there is no evidence of GDPR violation.
Oracle	In 2018, Oracle was alleged to be involved in illegal data gathering activities, including the use of web browser cookies for ad targeting.
Meta (Instagram)	In 2022, the Irish Data Protection Commission fined Instagram €405 million for exposing data of children between ages 13 and 17.
Tesla	In 2022, Germany's Federation of Consumer Association sued Tesla over the sentry mode in its cars for allegedly breaching GDPR.
Apple	Apple was alleged to have its "Personalized Ads" option on by default on iPhones, without user consent.
Twitter	The Irish Data Protection Commission fined Twitter €450,000 for exposing private tweets of its Android users and failing to notify the regulator in time.

Table 1.1: GDPR Violations by American Big Tech Companies - Table From : "Complying with GDPR: The Difficulties American Big Techs Face"

One notable response to the stringent requirements of GDPR has been the decision by some American companies, including prominent news publishers, to restrict access to their platforms for EU users. This measure, while mitigating the risk of penalties, underscores the broader compliance challenges faced by these corporations. Furthermore, the complexities inherent in GDPR compliance are highlighted by a survey cited in the paper, which found that a majority of companies grapple with both financial constraints and the intricacies of deploying automated privacy rights management solutions [7].

1.3 Data Loss and Its Implications

The risk of data loss, whether through hardware failure, human error, or cyber threats, poses significant challenges for both individuals and organizations. As highlighted in [8], data loss prevention is a critical aspect of safeguarding confidential information within an organization's boundaries. The paper emphasizes the importance of understanding and mitigating the risks associated with data loss in the digital age, particularly as businesses navigate the complexities of digital risk management and GDPR compli-

ance. Furthermore, the potential for irreparable consequences due to failures in personal data protection systems underscores the necessity of robust digital erasure methodologies. This thesis, therefore, aims to address these challenges by proposing an effective framework for digital erasure, ensuring the secure and compliant deletion of personal data in line with GDPR requirements. The framework will incorporate an innovative approach for risk mitigation, data protection, and compliance, contributing to the evolving discourse on digital privacy and data management in a world increasingly driven by data.

1.4 Intrusion Recovery and its Relevance in Digital Erasure

Intrusion recovery is a process in information security that focuses on identifying, mitigating, and recovering from unauthorized intrusions or attacks on a system. The goal is to restore the system to a safe, secure state by detecting and eliminating any compromised data or corrupted files. This process typically involves identifying the point of intrusion, analyzing the damage, isolating affected components, and restoring the system without the lingering effects of the attack. In the context of this thesis, intrusion recovery techniques are adapted to address the challenges of digital erasure. The similarities between handling corrupted data in intrusion scenarios and tracking data that must be deleted under privacy regulations, like the GDPR, allow for the application of similar methodologies. Specifically, the focus is on identifying personal data that has been flagged for deletion, tracing it across complex systems, and ensuring that it can be effectively removed in the future. By leveraging the principles of intrusion recovery, the thesis aims to improve the accuracy and efficacy of tracking data for eventual deletion. Intrusion recovery methods, such as identifying digital footprints and isolating compromised components, provide a valuable framework for analyzing how data can be securely marked and prepared for deletion, minimizing the risk of incomplete removal or unauthorized recovery. This approach ensures that data management processes align with the stringent requirements of digital privacy laws.

1.5 Aim and Approach of the Thesis

The primary objective of this thesis is to develop and evaluate a framework for analyzing the effectiveness of digital erasure techniques, particularly in the context of GDPR-compliant data management. Instead of directly implementing data deletion protocols, this work focuses on studying how accurately data can be identified and tracked for eventual deletion, and assessing the performance of various techniques used to facilitate this process.

The thesis is structured around the following key steps:

- **Intrusion Identification and Data Tracking:** The thesis proposes the adoption of methods from intrusion recovery to detect and track personal data marked for deletion. By treating such data as

analogous to compromised or corrupted data in intrusion scenarios, this approach can enable the precise identification of digital footprints that should be erased.

- **Data Matching Using Machine Learning:** A critical part of the work involves using machine learning models, such as Random Forest and decision trees, to evaluate how well they can identify and match data flagged for deletion. This step assesses the capability of these algorithms to accurately trace data points in large, complex datasets, simulating scenarios where deletion is necessary.
- **Analysis of Results:** Instead of carrying out the deletion itself, the thesis evaluates the results of data identification processes. Metrics such as F1 Score, accuracy, and precision are used to determine how effectively data has been matched and isolated, and to assess the feasibility of eventual deletion. The analysis provides insights into the strengths and limitations of the methods in ensuring comprehensive data tracking and deletion preparedness. We additionaly discuss whether the techniques used for data matching and isolation are suitable for real-world digital erasure tasks. This involves analyzing the deletion potential based on the results obtained from the models.

By analyzing these aspects, the thesis contributes to the field of digital privacy and data management by providing insights into the feasibility and effectiveness of machine learning approaches for preparing data for deletion.

1.6 Thesis Outline

The remainder of this thesis is structured as follows:

- **Chapter 2: Background** – Provides the historical context of data privacy regulations, beginning with early milestones such as the Swedish Data Act, and progressing to contemporary laws like the GDPR and the California Consumer Privacy Act (CCPA). It also covers the growth of server infrastructures and the increasing complexity of managing personal data across distributed systems, highlighting the difficulties in ensuring compliance with digital erasure requirements.
- **Chapter 3: Related Work** – Reviews the relevant literature on data deletion techniques and their application in machine learning. It explores challenges such as ensuring compliance with privacy regulations, particularly GDPR, and covers specific topics like machine unlearning, privacy in large language models, and data deletion approaches before GDPR. The chapter also identifies gaps in current methodologies and explains how the thesis aims to address them.
- **Chapter 4: Solution** – The principle for designing a system for digital erasure are presented. The solution leverages intrusion recovery techniques to track personal data flagged for deletion,

scopes the project into its goal which is the matching logs using machine learning algorithms such as random forests and decision trees for data matching and isolation. The architecture of the system, including the key components involved in data tracking and deletion preparation, is also presented.

- **Chapter 5: Scenario 1 – One Form, One Database** – Details the first experimental scenario, which involves a single form interacting with one database. The chapter describes how data logs are tracked, how the system identifies personal data marked for deletion, and the metrics used to evaluate the system's performance, including accuracy, F1 Score, and precision. Results from the experiment are discussed, offering insights into the system's efficacy in this simple scenario.

- **Chapter 6: Scenario 2: Database DB - DB with numerical IDs** – Introducing a more complex scenario in which multiple databases with differing features are used aiming to represent the communication between different databases. The system's ability to match and track data across various databases is tested, and the impact of factors suczh as numerical IDs, encrypted data, and textual variables is examined.

- **Chapter 7: Scenario 3: DB - DB with textual IDs** – The effectiveness of string comparison methods for matching textual IDs across databases is evaluated. Levenshtein distance outperformed Hamming distance, proving to be more effective in handling name variations, insertions, deletions, and substitutions. The Random Forest model achieved perfect classification using Levenshtein distance, making it the preferred approach for data integration scenarios involving inconsistent or encrypted identifiers.

- **Chapter 8: Journey and Self-Reflection** – Presents a personal reflection on the research journey, outlining the challenges faced during the thesis process, the lessons learned, and the overall experience of developing the proposed system. It provides insight into the practical aspects of conducting the research and offers reflections on the outcomes.

- **Chapter 9: Conclusion and Future Work** – Summarizes the key findings of the thesis, focusing on the proposed methodology and its effectiveness as a first step in the deletion of personal data in compliance with GDPR. The chapter also discusses the limitations of the current approach and suggests directions for future research, including potential improvements with the need for experimenting with real data and extensions to the system to handle more complex scenarios.

2

Background

Contents

The consideration of user privacy in the digital domain has become increasingly pertinent. Recent studies highlight key findings in this area: A survey conducted on September 21, 2016, titled “The State of Privacy in Post-Snowden America,” reveals significant user concern regarding privacy. The survey findings indicate that a majority, specifically 65%, of respondents consider it very important to control the data collected about them. Furthermore, a significant 86% of these individuals had actively taken steps to remove or mask their digital footprints, highlighting a growing awareness and proactive behavior towards digital privacy [9].

In an analysis focused on data deletion mechanisms on websites, a concerning gap was identified. This study, which scrutinized 150 websites, found that only 74% of them provided at least one mechanism for users to delete their data. This implies that about 26% of these sites, approximately 39 in total, did not offer any option for users to delete their personal data, suggesting a notable deficiency in user control over personal data [10]. The absence of such mechanisms indicates that users might face significant challenges, or even find it impossible, to delete their personal data through the standard

interfaces or account settings provided by these websites.

Additionally, the complexity of data deletion policies on websites poses further challenges. In the study "An Empirical Analysis of Data Deletion and Opt-Out Choices on 150 Websites" by Habib et al., the analysis revealed that the Flesch Kincaid Grade Level (FKGL) score for text related to data deletion averaged at 14.28. This high FKGL score denotes a complexity level akin to university-level understanding, which is considerably higher than the average comprehension level of the general public, typically ranging between a FKGL score of 7 and 9 [10]. This complexity can make it difficult for average users to fully understand data deletion policies.

Moreover, the same study revealed that a significant 83% of the 150 websites analyzed did not provide clear information regarding when an account would be permanently deleted. This lack of clarity leaves users in a state of uncertainty about the fate of their data, raising concerns about transparency and control in digital data management [10].

These findings collectively underscore the challenges and gaps in current digital privacy practices, emphasizing the need for more user-friendly and transparent policies in managing and deleting personal data online.

2.1 Historical Evolution of Digital Privacy

Digital privacy has been a significant topic in society since at least January 1, 1973 [11]. Over the years, its concepts, approaches, and considerations have evolved. To fully grasp the current implications of digital privacy, it's essential to understand its historical context, recognizing how perspectives and strategies have shifted from earlier times to the present.

2.1.1 Pioneering Legislation: The Swedish Data Act

A landmark in the evolution of data privacy legislation is the Swedish Data Act, regarded as the world's first national data protection law. This pioneering act came into partial force on July 1st, 1973, representing a significant step forward in the protection of personal privacy. It sets a precedent for data protection laws globally by addressing the growing concerns regarding personal data processing and privacy in the digital age. The Swedish Data Act's implementation underscored the urgent need for legal frameworks to safeguard personal data, both in the public and private sectors, thus paving the way for subsequent privacy laws and regulations worldwide [12].

2.1.2 Early Initiatives in Data Privacy: International Business Machines (IBM)'s Example

As early as the 1970s, corporate awareness of personal data privacy was emerging. A notable instance was IBM's revision of its record-keeping practices. The company undertook a comprehensive review and subsequently eliminated several personal data elements from its application forms. This included the date of birth, Social Security number, spouse's employment details, relatives employed by IBM, type of military discharge, and previous address. Additionally, IBM ceased the use of background checks and pre-employment personality tests, reflecting a pioneering approach towards enhancing data privacy and reducing unnecessary personal data collection in corporate practices [13].

2.1.3 Evolving Data Privacy Landscape

In the years following these early initiatives, data privacy concerns continued to grow, leading to various legislative responses worldwide. Notable developments included the European Union's Data Protection Directive in 1995, which was a precursor to the GDPR, and various sector-specific privacy laws in the United States. Key among these are the Health Insurance Portability and Accountability Act (HIPAA), which sets standards for the protection of individuals' medical records and other personal health data [14], and the Children's Online Privacy Protection Act (COPPA), which requires websites and online services directed at children to obtain parental consent before collecting, using, or disclosing personal data from children [15]. These laws represented a shift towards more comprehensive and specific data protection regulations.

2.1.4 Contemporary Milestones in Data Privacy: GDPR, CCPA and AIA

Building on these foundational legislations, the landscape of data privacy law reached new heights with the introduction of the GDPR in the European Union and the CCPA in the United States. GDPR, enforced from 2018, revolutionized privacy laws within the EU, providing comprehensive rights to individuals over their personal data, including the right to access, correct, delete, and restrict processing of their data. It also imposed stringent obligations on data processors and controllers, emphasizing transparency, accountability, and the requirement of explicit consent. Meanwhile, the CCPA, effective from 2020, marked a significant advancement in US privacy law. This act empowers California residents with rights similar to GDPR, such as the right to know about the personal data a business collects and its purposes, the right to delete personal data held by businesses, and the right to opt-out of the sale of personal data. Together, GDPR and CCPA signify a global shift towards stronger individual control over personal data and set a benchmark for future privacy regulations [4] [6]. Mirroring the evolution of data privacy laws, the Artificial Intelligence Act proposed by the EU represents a significant step in AI regulation. Introduced

in 2021, this Act categorizes AI systems based on risk, imposing strict rules on high-risk applications. It mandates compliance with specific requirements for high-risk AI, including transparency, oversight, and user rights protection, while establishing a framework for governance and enforcement. The Act also encourages AI innovation within a regulated environment. Parallel to GDPR's impact on privacy, this Act is poised to shape the future of AI governance in the EU, setting a new standard in AI legislation [5].

The CCPA, established in 2018, represents a landmark legislation in California, home to numerous tech giants. [6] The act emphasizes:

- **Consumer Empowerment**: Californians now have the rights to inquire about, and decide how their personal data is utilized. They can request businesses to reveal the data they have collected, ask for its deletion, and opt out of its sale.
- **Business Adaptation**: To adhere to the CCPA, companies underwent significant operational changes. This involved greater transparency about data practices and offering mechanisms for consumers to exercise their rights.
- **Right to Access and Deletion** : Allows consumers to request access to and deletion of their personal data (Sections 1798.100 to 1798.105). [16]
- **Right to Opt-Out of Data Sale** : Gives consumers the right to opt-out of the sale of their personal data (Section 1798.120). [17]

The GDPR [4], effective from May 2018 in the European Union, is a comprehensive data protection law that replaced the previous Data Protection Directive [18]. It represents a significant shift in data privacy regulation, with a broader scope and stronger enforcement :

- **Consent and Rights of Individuals**: Emphasizes the need for clear and affirmative consent for data processing (Article 7). Grants individuals the right to access their personal data (Article 15). Provides individuals the right to data portability (Article 20). Allows individuals to request the deletion of their personal data under certain conditions ('Right to be Forgotten', Article 17) [19].
- **Data Protection and Privacy**: Mandates organizations to implement appropriate data protection measures (Articles 25 and 32). Sets out stricter conditions for processing sensitive data (Article 9).
- **Breach Notification**: Requires timely data breach notifications to supervisory authorities (Article 33). Requires notification to affected individuals in certain cases (Article 34).
- **Global Impact**: Applies to all companies processing the personal data of individuals residing in the EU (Article 3).

Comparative Analysis with GDPR: While the GDPR introduces a ”Right to be Forgotten” the CCPA focuses on rights to knowledge, access, and opting out of data sale. Both, however, pivot on amplifying consumer data rights and ensuring business transparency.

These laws have set new standards in data privacy, influencing other regions to adopt similar measures. Since the introduction of Sweden’s landmark privacy laws in 1973, numerous countries have followed suit, enacting a variety of legislation aimed at safeguarding citizens’ privacy. This trend includes Brazil’s 2020 initiatives and other earlier efforts like France’s ”Loi Informatique et Libertés”, the ”US Privacy Act”, the UK’s ”Data Protection Act”, Canada’s ”Personal Information Protection and Electronic Documents Act (PIPEDA)”, Singapore’s ”Personal Data Protection Act (PDPA)”, South Korea’s ”Personal Information Protection Act (PIPA)”, South Africa’s ”Protection of Personal Information Act (POPIA)”, China’s ”Cybersecurity Law”, and Brazil’s ”General Data Protection Law (LGPD)”, among many others. These acts, spanning different continents, aim to regulate privacy, as exemplified by the US-EU Safe Harbor Agreement in 2000. The sheer number of these acts, their continual revision and refinement, reflect a global commitment to enhancing personal privacy. To see a chronological summary of 18 significant privacy acts in history, refer to Figure 2.1.

2.2 Growth of the Server Industry and Data Complexity

Even today, data deletion poses a significant challenge. The number of servers in existence is substantial: ”The global server market size was estimated to be USD 89.26 billion in 2022 and it anticipated to grow at a Compound Annual Growth Rate (CAGR) of 9.3% from 2023 to 2030. The growth of the server industry can be attributed to the proliferation of smartphones, an increasing number of data centers, among others.” [20]. This vast extent of servers makes tracking data traffic very difficult. In turn, the complexity of data often complicates its extraction and eventual processing: ”Traditional Information Extraction (IE) systems are inefficient in dealing with this huge deluge of unstructured big data. The volume and variety of big data require the enhancement of computational capabilities of these IE systems. It is necessary to understand the competency and limitations of the existing IE techniques related to data pre-processing, data extraction and transformation, and representations for vast volumes of multidimensional unstructured data.” [21].

Therefore, with the existing number of servers and the rapidly increasing complexity of data, there is undoubtedly a problem that arises about how we can address GDPR issues today while keeping user data secure and allowing for its tracking.

PRIVACY ACTS

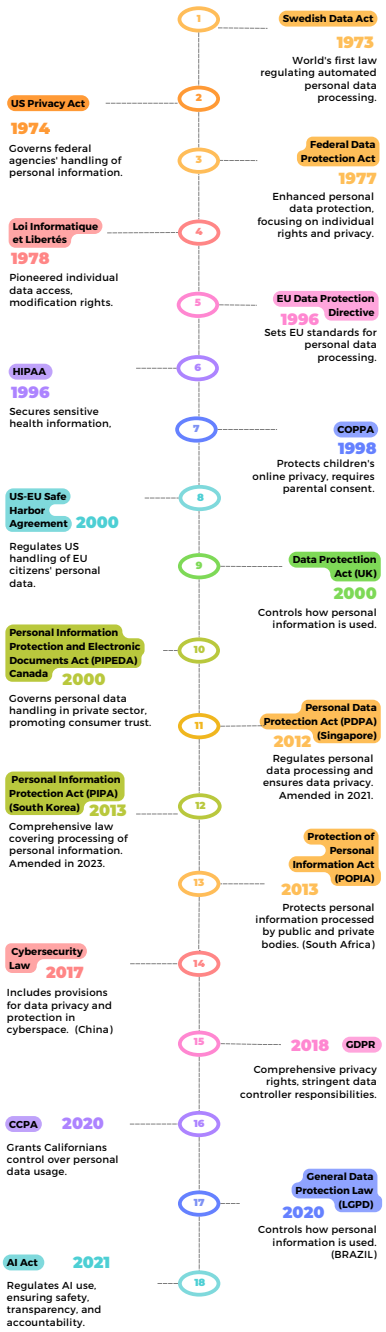


Figure 2.1: Data Privacy Acts

3

Related Work

Contents

Given the growing significance of data in today’s world, one might expect to find extensive research in this field. Surprisingly, this is not the case. Many existing approaches are either in their infancy or simply inadequate. A fundamental step in addressing data-related issues is to first consider whether the data should be deleted, followed by the actual process of deletion. This section delves into the latest methodologies for ”Unlearning,” which involves techniques for eliminating data that has been used in model training. This process is crucial for compliance with regulations like the GDPR. Additionally, we explore some methods in the field of Intrusion Recovery, where efforts are primarily focused on rectifying database intrusions.

3.1 Exploring Data Deletion in Machine Learning

The main problem the paper attempts to solve is how to extract certain data points from a machine learning model that has already been trained, without having to re-train the model from scratch. For reasons like privacy and adhering to laws like the EU's Right To Be Forgotten, this is significant. Traditionally, the only way to ensure a data point is completely removed from a model, is to retrain the model without that data point, which can be highly resource-intensive and impractical. [22]

Ginart et al. introduce the concept of deletion-efficient learning algorithms, specifically designed to update or modify an existing trained model to effectively 'forget' specific data points. They outline four key design principles:

- **Linearity**: Linear models offer easier updates compared to non-linear ones. For example, in linear regression, certain matrix operations can be used for quick model adjustments.
- **Laziness**: In lazy models, like k-nearest neighbors, removing a data point is as simple as excluding it from the dataset used for inference, since there is no explicit model, and each application depends exclusively on the data existing at the application moment.
- **Modularity** and **Quantization**: These principles involve modifying the model in a way that small changes in the data (like deleting a single data point) do not significantly affect the overall model. This can be more complex in pretrained models, especially if they are non-linear or complex, like deep neural networks.

Furthermore, the paper conducts a case study using k-means clustering to propose two new algorithms that enhance data deletion efficacy:

Quantized k-Means (Q-k-means)

This algorithm is a variant of Lloyd's algorithm [23] [24] designed for efficient data deletion.

We investigate algorithmic principles that enable efficient data deletion in ML. [22]

The main strategy involves quantizing the centroids at each iteration, ensuring that the algorithm's centroids remain constant with respect to deletions with high probability. The key aspects of Q-k-means are [22]:

- **Quantization of Centroids:** Each point is mapped to the nearest vertex of a uniform lattice, and a random phase shift is applied to the lattice for de-biasing.
- **Memoization of Optimization State:** The optimization state is stored in the model's metadata at various steps for use during deletion.

- **Balance Correction Step:** This step compensates for imbalanced clusters by averaging current centroids with a momentum term based on previous centroids.
- **Early Termination:** If the loss increases at any iteration, the algorithm terminates early.
- **Deletion Process:** Deletion in Q-k-means involves verifying if the deletion of a specific datapoint would have resulted in a different quantized centroid. If so, retraining from scratch is necessary; otherwise, the metadata is updated to reflect the deletion without recomputing the centroids.

Divide-and-Conquer k-Means (DC-k-means)

This algorithm partitions the dataset into smaller sub-problems, solves each as an independent k-means instance, and recursively merges the results. Its key features include [22]:

- **Tree-Based Partitioning:** The dataset is partitioned using a perfect w-ary tree of height h, with each leaf node representing a sub-problem.
- **Recursive Merging:** After solving k-means for each leaf, the results are merged up the tree, updating centroids at each level.
- **Modularity:** The tree hierarchy modularizes the computation's dependence on the data, allowing for efficient deletion operations.
- **Simplicity and Flexibility:** While the basic version focuses on depth-1 trees with w leaves, the algorithm can be extended to different tree structures and centroid weighting schemes.
- **Deletion Time Complexity:** The algorithm's deletion time complexity can be parameterized based on the tree width w, offering a trade-off between deletion efficacy and statistical performance.

Reflecting on this work, I must express my reservations about the approach taken towards data deletion in machine learning, particularly in the context of clustering. My concern stems from the fact that removing points from a cluster invariably influences the calculation of the centroid's position. This holds true except in rare cases, such as when all points in a cluster form a perfect circle and the point removed is at the center. In such scenarios, the centroid's position might not significantly change, but this should not be the sole criterion for asserting successful data deletion.

Furthermore, in supervised learning models, such as linear regression, every data point used in training, including those on the regression line, contributes to the computation of the model's output. Thus, their removal cannot be trivialized. The only exception I see in machine learning algorithms where data deletion might be straightforward is in the case of Support Vector Machines (SVC). Here, if the data points to be deleted are not among the support vectors, then their removal can indeed be

argued as inconsequential for the model's calculations. This is because support vectors are the critical elements that define the decision boundary in SVCs. Thus, non-support vector points can be considered redundant for the model's performance and their deletion can be seen as effectively non-impactful.

3.2 Ablating Concepts in Text-to-Image Diffusion Models

The research by Kumari et al. [25] delves into the realm of text-to-image diffusion models, focusing on the concept of ablating or removing specific concepts from these models. This challenge arises in the context of copyright concerns and the need for respecting artistic integrity in automated image generation, following the policies of the GDPR. The traditional approach of retraining models to exclude certain content is resource-heavy and often impractical, requiring a more efficient solution.

On the paper it is proposed a concept of ablation that includes a model-based variant, which has shown to be more effective, and a noise-based one :

Model-based Ablation: Suppose the specific prompt is "photo of Grumpy Cat." In model-based ablation, the model is fine-tuned so that when it receives this specific prompt, instead of generating an image specifically of Grumpy Cat, it generates an image more aligned with a general concept like "photo of a cat." This alignment is achieved by adjusting the model internally (minimizing the Kullback-Leibler divergence between the image distributions of the targeted concept and a more general anchor concept) to reduce the difference in how it processes and responds to "photo of Grumpy Cat" compared to "photo of a cat." The focus is on making the model's output distribution for the specific concept similar to that of the general concept.

Noise-based Ablation: The approach here is different. The training data is altered by pairing the specific prompt "photo of Grumpy Cat" with an image that corresponds to a more general concept like "photo of a random cat." The model is then fine-tuned on these new pairs. The goal is to train the model to associate the specific prompt with images that are less specific to Grumpy Cat and more generic or representative of cats in general. This changes the model's learned associations between the text prompts and the images.

This method enables the efficient removal of specific concepts, styles, or memorized images from the model without the need for retraining from scratch, which usually takes a lot of time.

With a particular focus on the challenges and limitations encountered. One notable example discussed is the difficulty in completely removing specific artistic styles, such as those of Van Gogh. While the model successfully ablated the concept of Van Gogh, it could still generate images resembling his famous 'Starry Night' painting. This example underscores a significant challenge: even after ablating a target concept, the model may still generate related content through different text prompts. Which is a relevant topic these days, where we can see it is possible to finess chatGPT restrictions giving different

prompts to obtain the ”restricted goal”.

Difficulties Encountered in Ablation:

- *Incomplete Ablation*: The model's ability to still generate 'Starry Night' after ablating Van Gogh highlights a key limitation. This suggests that ablation is not absolute and may require further refinement, such as explicitly ablating each related concept.
- *Impact on Surrounding Concepts*: The paper notes that ablating a target concept might sometimes lead to a slight degradation in the quality of surrounding concepts, illustrating the intricate balance required in concept ablation.
- *Reintroduction by Users*: The study also acknowledges that users with complete access to the model's weights could potentially reintroduce the ablated concepts, posing a challenge to the permanence of concept ablation.

This critical examination by Kumari et al. of the challenges faced in concept ablation draws parallels to the broader problem of data deletion in machine learning models. Just as removing the influence of Van Gogh from a model does not ensure the complete absence of his style, deleting specific data points from a machine learning model can be similarly complex and fraught with unexpected outcomes. The difficulties highlighted in their work echo the intricate nature of data manipulation in AI models.

Reflecting on this work: While there are instances of positive outcomes, much development is still needed in the field. Specifically, in the realm of ablations, the data remains present, and the current focus is on finding methods to circumvent it. This approach does not yet offer a solution compliant with GDPR standards. Considering the growing prominence of technologies like stable diffusion and DALL-E, further advancements in this area are essential. [26]

3.3 Machine Unlearning

The study by Bourtoule et al. [27] delves into the critical need for machine unlearning in the context of digital privacy. This concept is especially relevant given the rise of privacy regulations like GDPR and CCPA, which mandate the right to be forgotten.

Given dataset (D), one can train various models (like Deep Neural Networks (DNNs)) that effectively learn from this dataset. When a new data point (du) is added, creating a new dataset (D'), there are multiple ways to train a new model on D'. One approach is to use the parameters from a previously trained model (MA) as a starting point for the new model (MB), rather than beginning from scratch. This approach, however, faces challenges in measuring the influence of the new data point (du) on MB's parameters, making it difficult to reverse the process without having saved a copy of MA beforehand. The paper further discusses a strategy called ”slicing” and the concept of plausible deniability in privacy,

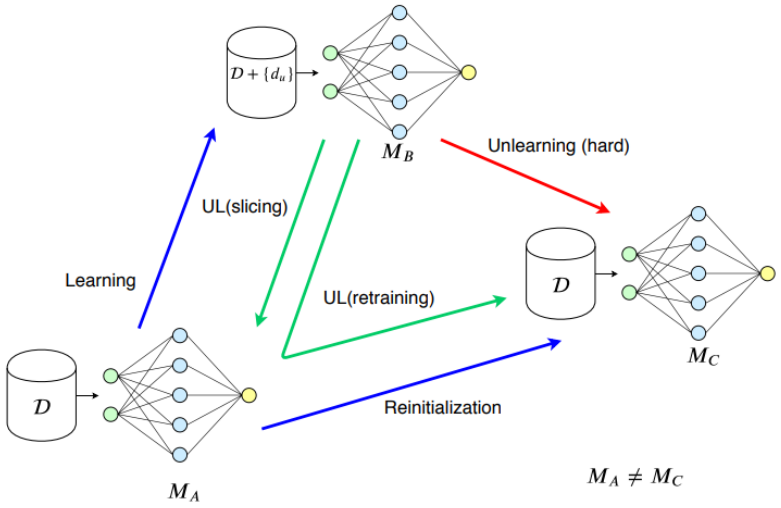


Figure 3.1: "Unlearning (red arrow) is hard because there exists no function that measures the influence of augmenting the dataset D with point du and fine-tuning a model MA already trained on D to train (left blue arrow) a model MB for D+du. This makes it impossible to revert to model MA without saving its parameter state before learning about du. We call this model slicing (short green arrow). In the absence of slicing, one must retrain (curved green arrow) the model without du, resulting in a model MC that is different from the original model MA."
[27]

which involves retraining the model from scratch without the specific data point to ensure it doesn't influence the model's training. For a visual overview of the system proposed on the paper access Figure 3.1.

The 'SISA (Sharded Isolated Sliced and Aggregated) training' framework is presented by the authors as a novel way to accelerate the unlearning process in machine learning models. By dividing the training data into separate shards, this technique restricts the impact of each data point to the appropriate shard. Computational overhead is decreased because only the model linked to the impacted shard needs to be retrained in response to an unlearning request.

The aggregation method mentioned is a label-based majority vote, where each model contributes equally to the final decision. This method is effective but might lose information in scenarios where models assign high scores to multiple classes. To address this, the paper evaluates a refined strategy where the entire prediction vectors (post-softmax vectors indicating a model's confidence in predicting each class) are averaged, and the label with the highest value is selected.

Bourtoule et al. contribute significantly to the field by formulating a definition of unlearning and demonstrating the practicality of SISA training. It was definitely a step forward on a work using Machine learning that was primarily motivated by privacy .However, the approach faces challenges like the creation of weak learners and the need for comprehensive hyperparameter tuning.

3.4 “If I press delete it’s gone” - User Understanding of Online Data Deletion and Expiration

Through interviews with 22 participants and focus groups with 7 data deletion experts, the study uncovers the nuances in user perceptions and the challenges they face in comprehending these concepts. [28]

- **Diverse Views on Data Deletion:** The research identifies two primary perspectives among users regarding data deletion:
  - *UI-Based View:* some users understand deletion purely from a user interface perspective, believing data is gone once it is removed via the interface.
  - *Backend-Aware View:* others have a more nuanced understanding, considering backend processes like data storage on servers or in the cloud.
- **Reasons for Data Deletion:** The reasons for deleting data varied. In email scenarios, common reasons included managing storage limitations and keeping inboxes tidy. In social media contexts, reasons were often related to the content being outdated or potentially embarrassing.
- **Complexity in Data Storage and Deletion:** Participants stated that the intricacy of storing data before erasing it influences whether the data disappears instantly. Cloud storage, servers, and databases were common places to store data. Participants frequently used phrases like ”server” and ”cloud” synonymously, suggesting a general knowledge of data storage but maybe lacking in technical precision.
- **Assumptions about Data Storage:** Users had various assumptions about why data is stored, including for backups, law enforcement, and marketing purposes. In the case of social media, many believed that data is kept for marketing profiling and that deletion from the web is complex or impossible.
- **Data Expiration Concept:** When the idea of data expiration was discussed, the majority of participants connected modifications in the value of data to particular, non-time-bound acts, such as terminating a service provider account. According to this research, the value of data to consumers is very context-dependent and may not increase linearly with time.
- **Expert Recommendations:** Six major topics have been highlighted by experts as helping users better grasp online data deletion: Anonymization, data backups, backend operations, delayed data destruction, and the presence of shared copies are some examples of these. These subjects were thought to be essential for users to make knowledgeable judgments about data privacy.

3.5 Pre-GDPR Secure Data Deletion Approaches

In the context of digital erasure and data privacy, the period before the enforcement of the GDPR is marked by pivotal research, including the paper ”SoK: Secure Data Deletion” by Reardon et al. [29]. This work is instrumental in our understanding of the methodologies and challenges in secure data deletion, a key aspect of disappearing online.

Reardon et al. present a systematic analysis of secure data deletion methods, categorizing them based on their interaction with various storage mediums [29]. They identify key challenges in ensuring data is unrecoverable, such as the complexity of different storage technologies and the varying capabilities of adversaries. The paper highlights that most systems do not securely delete data by default, leaving remnants that could be exploited.

A significant contribution of this work is the taxonomy of adversaries and deletion methods. Reardon et al. classify adversaries based on their access levels, ranging from those with physical access to the storage medium to those with only remote access. This classification is crucial for understanding the effectiveness of different deletion methods in various threat scenarios.

Moreover, the paper discusses the behavioral properties of secure deletion methods, including deletion latency and the physical wear on the storage medium. These aspects are vital for evaluating the practicality and sustainability of secure deletion techniques, especially in environments with high data turnover or limited physical durability.

Prior to this comprehensive analysis, studies like Perlman’s ”The Ephemerizer: Making Data Disappear” [30] and Geambasu et al.’s ”Vanish: Increasing Data Privacy with Self-Destructing Data” [31] laid the groundwork for secure data deletion. However, the breadth and depth of Reardon et al.’s paper provide a more nuanced understanding of the landscape before GDPR, setting a benchmark for subsequent research and development in this field.

3.6 Recovery from malicious transactions

In the context of the current data-driven business environment, managing large volumes of data effectively is a significant challenge. Traditional tracking methods become impractical in large companies, where the sheer volume of data and the complexity of adherence to specific protocols by numerous employees are overwhelming. This scenario underscores the importance of efficient data management strategies, particularly in complex organizational settings. In the field of intrusion recovery, which focuses on restoring corrupted or compromised data, there are several approaches that can be adapted for digital erasure. Sanare, with its ability to accurately trace and eliminate specific data alterations, presents an adaptable solution that extends beyond intrusion recovery to GDPR-compliant data management.

Sanare [32],is an intrusion recovery system specifically designed for web applications that maintain shared state information. The primary functionality of Sanare involves two key processes: first, identifying all changes made by an intrusion and any subsequent changes that depend on the initial intrusion; second, erasing the data altered by the intrusion. This system, leveraging its deep learning scheme Matchare, effectively links Hypertext Transfer Protocol (HTTP) requests to database operations, which is crucial for tracking and reversing unauthorized data changes.

A significant aspect of ensuring data integrity in web applications, particularly in the context of digital erasure, is the recovery from malicious transactions. The work by Ammann, Jajodia, and Liu [33] propose algorithms designed to recover databases from the effects of malicious transactions, aiming to preserve the work of benign transactions.

Ammann et al. detail two recovery models: the Coldstart and Warmstart repair semantics. The Coldstart model involves taking the database offline during the recovery process [33]. This method, while ensuring a thorough rollback of malicious transactions, can be detrimental in environments requiring high database availability.

Introducing dependency graphs to map intertransaction relationships, identifying how changes in one transaction can affect others occur in both models. The concept of dependency graphs is derived from these transaction logs. These logs record the details of all transactions processed by the database.

Conversely, the Warmstart model allows the database to remain operational while recovery is underway [33]. This approach is more suitable for real-time applications but requires sophisticated mechanisms ("The cost of on-the-fly repair is that some new transactions may inadvertently access and subsequently spread damaged data.") to effectively manage on-the-fly repairs and differentiate between malicious and benign transactions. Key mechanisms in the Warmstart model include:

- **Isolation of the Malicious Transaction Recovery Module**: This strategy maintains the malicious transaction recovery module separate from the traditional recovery module. Such isolation is vital in a live, operational setting, as it helps prevent degradation of the traditional module's performance and allows for the seamless integration of the malicious transaction recovery module with the existing database system.
- **Extracting Read Information from Profiles and Input Arguments**: In the Warmstart model, read information is extracted from the profiles and input arguments of transactions, rather than relying on a comprehensive read log. This method is advantageous because it requires storing only the smaller-sized input parameters of each transaction, as opposed to the larger read set.

These mechanisms address the challenge in the Warmstart model that some new transactions may inadvertently access and subsequently spread damaged data during the on-the-fly repair process.

The recovery strategies proposed by Ammann et al. highlight the complexities involved in distinguishing between data that needs to be erased due to malicious intent and legitimate data that should

be preserved.

In ”Recovery From Malicious Transactions,” Ammann et al. [33] present algorithms for recovering databases after malicious transactions. This approach is centered on identifying both the malicious and affected benign transactions using transaction logs. The paper introduces the concept of dependency graphs, which are crucial for understanding the interrelations and impacts of these transactions, thus guiding the recovery process effectively.

Matos, Pardal, and Correia [34] propose Rectify, a service for recovering Platform as a Service (PaaS) web applications from intrusions, in their paper. Rectify operates by logging HTTP requests and database statements, using machine learning to correlate these logs for identifying malicious activities. Unlike the previous approach, which focuses on database transactions, Rectify addresses intrusions at the application level, recovering the system to a state as if the intrusion never occurred, using compensation operations.

While both approaches aim to recover from malicious activities, their methodologies and focus areas differ. Ammann et al. concentrate on the database layer, using transaction logs and dependency graphs, whereas Rectify focus on the application level, leveraging machine learning to associate HTTP requests with database modifications in a PaaS environment.

This section focuses on a contemporary 2023 approach to intrusion detection in cloud environments, as presented in the paper ”Real-Time Multi Fractal Trust Evaluation Model for Efficient Intrusion Detection in Cloud” by S. Priya and R. S. Ponmagal [35].

The Multi Fractal Trust Evaluation Model (MFTEM) represents a significant advancement in cloud security, addressing the increased rate of intrusion attacks in cloud services. This novel approach leverages the concept of trust evaluation across multiple dimensions: service growth, network growth, and Quality of Service (QoS) growth [35].

MFTEM assesses trust through various metrics, such as Trusted Service Support (TSS), Trusted Network Support (TNS), and Trusted QoS Support (TQS), culminating in a Trust User Score (TUS) [35]. Each metric evaluates user behavior and support in different aspects of the cloud service, thus offering a multifaceted approach to detecting and preventing intrusions.

The model’s effectiveness is demonstrated by its ability to improve intrusion detection accuracy by up to 99%, marking it as a substantial improvement over traditional methods. The MFTEM uses service traces and applies comprehensive analyses to evaluate trust, making it a robust solution for modern cloud environments [35].

4

Solution

Contents

This work proposes a method that involves correlating logs from distinct related sources, such as HTTP requests with database logs, for example, allowing us to trace the location and nature of the unauthorized data. Focusing on the matching phase, as an initial step toward the ideal deletion of such data, our approach aims to adapt the Sanare framework to address the ideal goal of detecting and mitigating unauthorized data creation or modification, which we conceptualize as an intrusion that needs to be removed. The process includes multiple steps, with the first being the ability to match the logs and identify where the unauthorized data is located. In the context of GDPR compliance, these steps are crucial, and this thesis concentrates on the critical phase of data matching and preliminary deletion efforts.

4.1 System Proposal

In the context of GDPR, we define *intrusion* as the presence of personal data in a system beyond a subject's request for its deletion. Identifying the last HTTP requests interacting with such data is necessary, as it marks the initiation of the data deletion process under GDPR compliance. The GDPR, as discussed by [36], governs the collection, storage, and processing of personal data, encompassing both direct and indirect identifiers. The complexity of defining anonymity under GDPR is significant, particularly when considering data that must be deleted upon request [36]. Furthermore, the principle of data minimization, as outlined in Article 5 of GDPR [4], is central to this process, requiring the limitation of personal data collection and usage to only what is necessary for the intended purpose [36].

4.1.1 System Definition

For the development of the proof of concept, we consider a system that consists of a web application connected to multiple data repositories, such as a relational database and a file system. The creation of personal data is initiated through HTTP requests, which serve as the mechanism for user interactions with the web application. The personal data itself is stored in a relational database, where it can be tracked and processed according to GDPR requirements. The system logs user activities and stores corresponding HTTP requests, allowing for the identification and potential deletion of data in compliance with privacy regulations. This setup reflects typical web-based systems where data flows between the user interface and the back-end storage repositories, creating a robust framework for evaluating digital erasure methods and data deletion strategies within complex data architectures.

4.1.2 Data Matching Using Machine Learning

The objective at this stage is to establish a robust link between HTTP requests and corresponding database operations, a task critical for accurately mapping data dependencies in the context of GDPR compliance. The HTTP request that is of interest to us is the one which initiated the creation of the user data. To address this, we propose to use machine learning, particularly ensembles known for their effectiveness in pattern recognition and complex data matching tasks [37]. Our proposed approach centers on using and potentially enhancing models like Matchare [32], which have shown promise in similar applications [38]. Given the complexity of data interactions, we consider the use of ensemble methods, and the approach by Random Forests. These methods combine predictions from multiple algorithms, offering improved accuracy over single-model approaches, especially in high-dimensional data scenarios [39]. Effective training and fine-tuning of models are essential to ensure they are tailored to the specific requirements. Random Forest is an ensemble learning method widely used for classification and regression tasks. It was first introduced by Leo Breiman in 2001 [40], and it has since become a

popular choice for various machine learning applications due to its robustness, accuracy, and ability to handle large datasets with numerous input features. The Random Forest algorithm operates by constructing a multitude of decision trees during training. Each tree is built using a random subset of the data and a random subset of the features. The final prediction is made by aggregating the predictions of all individual trees, typically by majority voting for classification tasks or by averaging for regression tasks [40]. The key advantage of Random Forest lies in its ability to reduce overfitting, a common problem with decision trees. By averaging the predictions of multiple trees, the model becomes less sensitive to the variance of individual trees, thereby improving generalization to unseen data [41]. In the context of machine learning, features refer to the input variables used to make predictions. In Random Forest, features are randomly selected at each node in the decision trees to determine the best split. This randomness ensures that the trees are diverse, which contributes to the overall robustness of the model [42]. For each tree in the forest, a subset of features is chosen randomly from the total set of features. This process, known as feature bagging, helps in reducing the correlation between individual trees and enhances the predictive performance of the model [43]. Estimators in the Random Forest algorithm refer to the individual decision trees that are built during the training process. The number of estimators, denoted as $n_{estimators}$, is a crucial hyperparameter in the model. Increasing the number of estimators generally improves the model's accuracy, as it allows the forest to capture more information from the data. However, after a certain point, the marginal improvement diminishes, and computational costs increase [40]. Each estimator is trained on a bootstrap sample of the original dataset, which means that some samples are used multiple times, while others are not used at all. This technique, known as bootstrap aggregating or bagging, enhances the stability and accuracy of the model by reducing the variance [44]. Random Forest has been successfully applied in various domains, including medical diagnostics, finance, and bioinformatics, due to its ability to handle high-dimensional data and complex interactions between variables [45]. Its versatility and effectiveness make it a go-to model for many classification and regression problems. While Random Forest offers robustness against overfitting through the averaging of multiple decision trees, Gradient Boosting is another ensemble method that builds trees sequentially, with each new tree correcting the errors of the previous ones [46]. Gradient Boosting offers high sensitivity to outliers, making it suitable for detecting rare, unauthorized data changes, but it is also prone to overfitting if not carefully managed. Unlike Random Forest, Gradient Boosting is more susceptible to overfitting, particularly when the number of trees or boosting rounds is too high [47]. Since each tree is trained to minimize the residual errors of the previous ones, it may overly adapt to the noise in the training data.

4.1.3 Data Isolation

The objective in this phase ensures that the deletion process is targeted and does not inadvertently affect unrelated data. Its precision is crucial for maintaining the integrity of the remaining data. The methodology involves the strategic usage of the previously constructed dependency graph. This graph, which details all changes and interactions associated with the user's data, serves as a guide to identify and isolate the relevant data records and operations. By tracing these dependencies, we can ensure that only the data directly related with the user's request is targeted for deletion. The successful execution of this phase will result in the precise isolation of the user's data, paving the way for its secure and compliant deletion. [48].

4.1.4 Data Deletion Process

The primary goal in this phase is to comply with the user's GDPR request by thoroughly and securely deleting their data from the system. This process must be comprehensive, ensuring that all traces of the user's data are removed in accordance with GDPR requirements [48]. The deletion process is executed following the isolation of the user's data, as described in the previous phase. This process involves systematically removing all records, dependencies, and traces of the user's data from the system. The process must be exhaustive, ensuring that no residual data remains in the system, whether in databases, backups, or logs. Post-deletion verification is crucial to confirm that all targeted data has been irreversibly removed, adhering to the accountability requirements of GDPR. The deletion process must be secure to prevent unauthorized access or recovery of deleted data. [48].

4.1.5 Architecture

The organizational data landscape encompasses a diverse array of systems, ranging from cloud-based solutions and databases to conventional file systems. To facilitate the data erasure, it is imperative to account for the various data storage modalities employed within an organization. This study proposes a methodology wherein data, specifically logs and HTTP requests, will be logged in a centralized log repository. This initiative aims to leverage ensemble learning techniques to establish a correlation between HTTP requests and log data. By training the model on these synthetically generated datasets, the study seeks to enhance its predictive efficacy in identifying relationships between HTTP requests and log entries, which is where this thesis is focusing its' scope on. Subsequently, the trained model will be applied to the actual organizational data, enabling the matching of logs with corresponding HTTP requests. This approach helps in identifying specific user data within the organization's data structure. Cleanum represents the local component, wherein all data originating from the organization will be consolidated. This includes data sourced from database servers, cloud infrastructures, or file systems. The

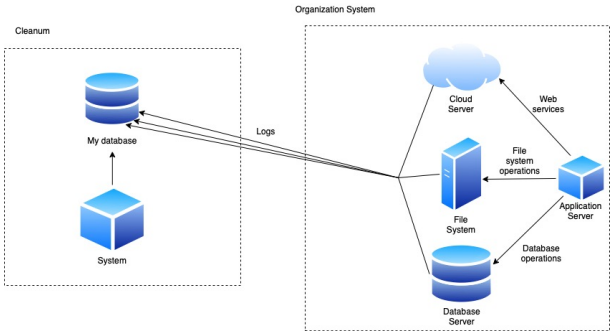


Figure 4.1: In this figure, *Cleanum* represents the local component, wherein all data originating from the organization will be consolidated.

system is tasked with aggregating the generated HTTP requests and logs, which are formulated based on the acquired data. Subsequently, this compiled data will serve as the foundational training set for the system. The primary objective of this configuration is to enable the matching of logs with their corresponding HTTP requests from the organization, thereby establishing an effective and coherent data matching system. For a graphical understanding of Cleanum access Figure 4.1.

4.2 Building the Dataset

This chapter outlines the development of the dataset. To collect the data we deployed a web form designed within WordPress, a platform that accounts for approximately 62.7% of all websites globally, emphasizing the importance of adhering to GDPR guidelines for data collection and storage. The form serves as a critical component of a broader research project aimed at exploring the implications of GDPR compliance in technological environments.

4.2.1 Developing a GDPR-compliant web form

The web form was developed using WordPress, a Content Management System (CMS) that powers millions of websites worldwide. WordPress's flexibility and extensive plugin ecosystem facilitated the creation of a form tailored to the research objectives. The form includes fields for collecting demographic information, such as first name, last name, country, height, weight, gender, ethnicity, hair color, eye color, shirt size, EU foot size, smoking habits, drinking habits, monthly wage, religion, sexual orientation, political orientation, and consent to terms and conditions. Each field was carefully designed to comply with GDPR requirements, ensuring that users' personal data is collected and processed lawfully and transparently. The form was tested in a local environment using Macintosh, Apache, MySQL, PHP (MAMP).

4.2.2 System

For the development of our prototype we deployed the WordPress application in a MAMP server, a software suite designed to emulate a local web server environment on macOS and Windows platforms. MAMP, an acronym for Macintosh, Apache, MySQL, and PHP, encapsulates the core technologies essential for web development and application hosting. By leveraging MAMP, we were able to establish a localized server environment that mirrored the production-level infrastructure, thereby facilitating the development, testing, and deployment phases of my project. This approach allowed for a seamless transition from the development stage to the live environment, ensuring the reliability and efficacy of the final product. Moreover, MAMP's compatibility with a broad spectrum of web technologies, including PHP, MySQL, and Apache, enabled me to integrate various components seamlessly, enhancing the overall functionality and performance of the project.

4.2.2.A Recording the HTTP and database logs

In order to monitor and record HTTP requests reaching the Apache web server component of our MAMP stack, we had to configure the Apache server (the `httpd.conf` file) to ensure that every request gets logged. We also explored the use of dumpio logging [1] for detailed inspection of request and response bodies, setting DumpIOInput and DumpIOOutput to On. To collect the logs with the database operations we configured the database server (`my.cnf` file) to collect every statement that is executed. This would ensure we would get a log in the file system (stored in `/Applications/MAMP/logs/mysql-general-log`).

4.2.3 Form filling

We wrote a script to automate the process of filling out a web form with randomized data, simulating user interactions to collect a large dataset for machine learning purposes. It leverages Python libraries such as Selenium [2] for web automation, Pandas [3] for data manipulation, and regular expressions for parsing log files. This approach allows for efficient data collection without manual intervention, enabling the generation of a comprehensive dataset for analysis.

4.2.3.A Methodology

The script uses a list of names loaded from a Comma-Separated Values (CSV) file, with all the first and last names registered in Portugal in 2017, to generate randomized data for various form fields. This ensures diversity in the collected data. Selenium WebDriver is utilized to automate the interaction with the web application, including navigating to the form, filling out fields, and submitting the form multiple

[1] Dumpio logging: https://httpd.apache.org/docs/2.4/mod/mod_dumpio.html
[2] Selenium: https://www.selenium.dev/
[3] Pandas: https://pandas.pydata.org/

times. Every time it fills the form, it can either fill every field from the form or just a portion of them. The intended way of using this Web Automation was to simulate as much as possible human behavior, as long as the usage of random sleep times while filling each field of the form, in order to also simulate how a human would act. Both Apache and MySQL log files undergo parsing using regular expressions to glean pertinent details concerning database operations associated with form submissions, for the data to be able to be analyzed.

4.2.3.B Implementation details

A script runs the WebDriver so that it can interact with the web application. Then, custom functions parse the log files and extract relevant information about form submissions. Finally, the script appends the collected data to a CSV file, ensuring that each iteration's data is preserved for future analysis. There were some challenges with our approach. Some web elements from the WordPress applicatios are dynamic which makes the process of handling them complex. To solve this, we used explicit waits and dynamic XPath selectors to ensure accurate element identification. Another challenge was related with the log file complexity. Parsing log files involved dealing with complex patterns and potential inconsistencies in log formats, necessitating robust regular expression handling. These processes can be computationally expensive. To manage performance, especially during long-running scripts, optimizations such as minimizing sleep times and efficiently processing log files were implemented.

4.2.4 Variables

WordPress is a extensible platform and it allowed to build personalized user forms. For our use case, the form includes three primary types of fields: text fields, numeric fields, and drop-down menus, each serving a unique purpose in capturing detailed participant profiles.

4.2.4.A Text fields for basic identifiers

Text fields were used to collect basic identifiers such as the participant's first and last names, ensuring a personalized touch to the data collection process. These fields are open-ended, accepting any string value inputted by the user, thereby accommodating a wide range of names across different cultures and languages.

4.2.4.B Numeric fields for quantitative data

Numeric fields were placed to capture quantitative data related to physical attributes. Specifically, the form includes fields for height, weight, and EU foot size, each with predefined numerical ranges to

standardize the data collected. These fields not only facilitate the gathering of essential demographic information but also contribute to the overall accuracy and consistency of the dataset.

4.2.4.C Dropdown menus for structured selection

Dropdown menus, on the other hand, offer a structured selection process for participants to choose from predefined options. This method ensures uniformity in data entry and reduces the likelihood of errors. The form features drop-down menus for selecting the country of origin, gender identity, ethnicity, hair color, eye color, smoking habits, drinking habits, monthly wage, religion, sexual orientation, and political orientation. Each category includes a comprehensive set of options, reflecting the diversity of perspectives and backgrounds among the study's target population. By integrating these three field types, the form effectively balances the need for detailed, qualitative insights with the requirement for standardized, quantitative data. This blend of field types not only enhances the depth of the information gathered but also ensures that the data remains consistent and comparable across all participants. The form's design and functionality underscore the importance of thoughtful form design in research projects, particularly when aiming to capture a broad spectrum of human experiences and characteristics.

4.2.5 Dataset

The innitial approach for the construction of the dataset consisted of extracting and organizing Structured Query Language (SQL) logs to create a comprehensive dataset. This dataset was built aligning entries from our database with corresponding HTTP request logs. Through a logging process, we ensured that each database entry had a direct counterpart in the HTTP request logs, resulting in a dataset where every log entry was perfectly matched. This matching process yielded a dataset consisting of **823** lines where each line represented a pair of a database entry and its corresponding HTTP request. This high level of correspondence between the database and HTTP logs was necessary in the initial stages of dataset construction. This approach facilitated the creation of a labeled dataset and ensured that the dataset was reflective of the actual interactions occurring within our system, providing a basis for further analysis and modeling efforts. To train a model we needed a dataset that was representative of both classes: "matched" and "unmatched" operations. To address this, we identified and incorporated **2177** lines that did not find a direct match in the database entries, labeling them as "unmatched." This strategic addition was motivated by the recognition that not all log entries from HTTP requests would have a corresponding entry in the database, reflecting the inherent complexity and diversity of the data involved. Introducing these "unmatched" lines into our training set was a deliberate decision aimed at enhancing the model's ability to generalize beyond the specific patterns observed in the matched data. By doing so, we sought to create a more representative and training set that could better withstand the

challenges of unseen data in production environments. This approach aims to represent the possibility of encountering HTTP requests that lack a direct match in the database.

4.2.5.A Data cleaning and structuring

In the data treatment phase, the following steps were taken to clean and structure the dataset obtained from SQL logs and HTTP request logs. More specifically we removed irrelevant columns in the SQL logs, such as IDs with no connection to the HTTP request logs. We also eliminated identical values across all logs were also eliminated.

4.2.5.B Encoding

For text fields like first names, last names and the Country variables, since there was not a straightforward way to encode these names to numerical values there was a one to one mapping into a numeric variable based on its' value. For the remaining variables, an encoding was created to translate categorical values into numerical ones. An example of this mapping is shown for the Ethnicity variable:

```
ethnicity-mapping = { 'Black or African American': 10, 'Indian': 7, 'Latino or Hispanic': 5,
'Asian': 2, 'White (Caucasian)': 0, 'Other': -1 }
```

This encoding assigns a unique numerical value to each category, with a sequence that reflects the order or hierarchy of the categories. This approach was adopted for all other variables, ensuring that the numerical representation of the data carried meaningful implications for the algorithms.

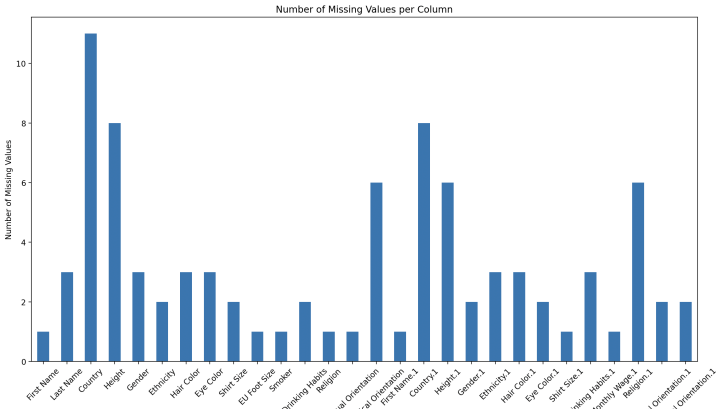


Figure 4.2: Bar chart showing the number of missing values per column. Columns expected to match exhibit higher frequencies of missing values in both corresponding columns, while non-matching columns show more random patterns of missing data.

4.2.5.C Handling missing values and special cases

Missing values were mapped to -2, and the "Other" option was consistently mapped to -1. This consistent mapping helped maintain coherence in the results, ensuring that missing data and special selections were easily identifiable and treated uniformly throughout the analysis. There was also one more variable on HTTP request side and database side that represented when were the actions performed (Timestamp of those actions), the refactor on this case simply on transforming it into date time. In the first approach of this work, where we were labeling matched columns and unmatched ones, it was expected that missing values would exhibit a specific pattern based on whether the columns were meant to match. In cases where a match between columns was intended, missing values would occur in both corresponding columns, indicating that the absence of data was consistent across the match pair. Conversely, in columns designated as non-matching, the missing values would be irrelevant to the matching process and were anticipated to occur randomly across the dataset. On this figure 4.2 we are able to visualize some of the missing values on the dataset.

After this on our dataset we had now on one side the features

4.2.6 Methodology for dataset analysis and initial model selection

At this point, our dataset was fully prepared for training. On one side of the dataset, we had features extracted from the HTTP requests (such as first name from HTTP, last name from HTTP, ethnicity from HTTP), and on the other side, we had corresponding features from the database (first name from the database, last name from the database, ethnicity from the database), along with other variables. Additionally, we included a feature indicating whether each pair of entries (from HTTP and database) matched or not. This structure allowed us to systematically compare and analyze the relationships between the HTTP request data and the corresponding database entries, which will be useful for the classification. Our initial model selection was guided by the nature of the problem at hand – a classification task. Given the complexity and variability inherent in web traffic data, we opted for a Random Forest classifier. The Random Forest algorithm's strength lies in its ability to handle a multitude of input features and its robustness against overfitting, making it particularly well-suited for classification problems where the outcome depends on a combination of factors. With 36 variables in total, including the matched indicator, we believed that the Random Forest model would effectively navigate the intricacies of our dataset, facilitating accurate predictions. This decision marked our first approach to solving the problem, driven by the need for a model that could seamlessly integrate the complexities of our data. As we progressed, we continued to refine our methodology, exploring additional models and techniques to enhance the performance and interpretability of our findings.

5

Scenario 1 : One Form - One Database

Contents

As the simplest approach to data deletion, we aim to represent the most straightforward scenario: attempting to delete a user's data from a single Database (DB), after a form filling. In this case, a form is filled out, generating the corresponding HTTP requests, which populate the database as described in the previous chapter. The data is then stored in one database, which provides the necessary logs. Our goal is to match this data with our dataset, which has already been built for this purpose.

5.1 Original Logs with no correspondence knowledge

This section presents the analysis of the results obtained from the first attempt at training a Random Forest classifier. The objective of the model was to identify matches between the first 18 columns and the subsequent 18 columns of the dataset. An additional column labeled as "Matched" or "Unmatched"

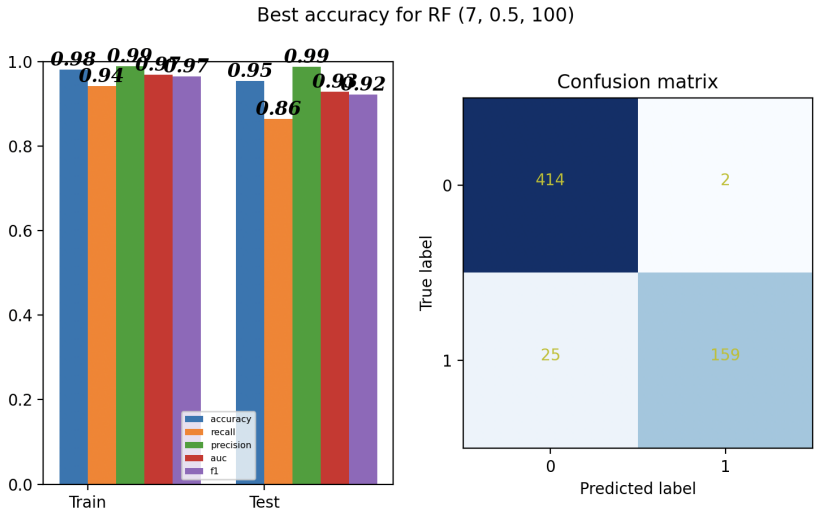


Figure 5.1: Confusion matrix analysis showing 159 True Positives (TP), 414 True Negatives (TN), 2 False Positives (FP), and 25False Negatives (FN).

was used as the target variable for training. While the initial results are promising, it is important to consider the implications of false negatives and false positives in the context of data deletion. In the context of our model, both false positives (FP) and false negatives (FN) present significant challenges. A false positive occurs when the model incorrectly classifies data as a match, leading to the deletion of information that should have been retained. This is undesirable as it risks removing important or valid data. Conversely, a false negative arises when the model fails to identify data that should be deleted, leaving unwanted data in the system, which can lead to privacy or compliance issues. To address these issues, we aim to achieve a model with both high precision and high recall. High precision ensures that when the model predicts data for deletion, it is accurate, minimizing the risk of deleting necessary data. At the same time, high recall ensures that the model identifies all data that should be deleted, preventing unwanted data from remaining in the system. Balancing these two metrics is critical for minimizing both false positives and false negatives, ensuring that we only delete data that should be removed while retaining all relevant information. The confusion matrix and evaluation metrics for both the training and testing datasets are summarized below.

	Predicted Positive	Predicted Negative
Actual Positive	159	25
Actual Negative	2	414

Table 5.1: Confusion Matrix

The evaluation metrics can be accessed on the Figure 5.1 where we can analyze the following results present in the table 5.2:

Metric	Training Set	Testing Set
Accuracy	0.98	0.92
Recall	0.94	0.86
Precision	0.99	0.98
Area Under the Curve (AUC)	0.97	0.93
F1 Score	0.97	0.92

Table 5.2: Performance metrics for the training and testing sets.

5.1.1 Impact of Max Depth, Number of Estimators, and Feature Proportion on Accuracy

The figure below (5.2) illustrates the accuracy of the Random Forest classifier as a function of the number of estimators (”nr estimators”) and the ”max-depth” of the trees. The figure is divided into three subplots, each representing a different value of ”max-depth”: 2, 5, and 7. Each line within the subplots corresponds to a different proportion of features used in the model, denoted by the different colors in the legend (10%, 30%, 50%, 70%, 90%). In the left subplot, where ”max-depth” is set to 2, the accuracy remains relatively flat across different numbers of estimators, with minor variations. The lines are very close to each other, indicating that the model’s performance is not significantly influenced by the proportion of features used when the ”max-depth” is shallow. This suggests that a ”max-depth” of 2 may be too limited to capture complex patterns in the data, resulting in similar performance regardless of the number of estimators or the fraction of features. However, in the best-case scenario, this setup could perform adequately in a simplified situation where there are only two variables—one from the database and one from the form—since the depth would be the same as the number of variables. The middle subplot, with a ”max-depth” of 5, shows slightly more variation in accuracy compared to the ”max-depth = 2” case. There is a noticeable separation between the lines corresponding to different feature proportions, particularly for lower feature fractions (e.g., 10%, 10%). This behavior can be attributed to the combination of shallow trees with a high variation in features (as indicated by the low percentage of features), which causes the trees to fail in representing the matching pairs accurately. As the number of estimators increases, the model’s accuracy generally improves, albeit modestly. This indicates that increasing the ”max-depth” allows the model to capture more complex interactions, leading to slightly better performance, especially when larger feature subsets are used. The right subplot, where ”max-depth” is set to 7, exhibits the highest accuracy among the three configurations. The lines for different feature proportions are more distinct, with a clear trend showing that using a higher fraction of features (e.g., 50%, 70%, 90%) results in better accuracy. This reinforces the earlier observation that a larger number of features helps the model better capture the relationships in the data. Additionally, the accuracy improves gradually as the number of estimators increases. This suggests that a ”max-depth” of 7 is more effective at capturing the complexity of the data, leading to higher accuracy overall. The graph

indicates that using a higher proportion of features and increasing the number of estimators further enhances the model's performance. Overall, the figure demonstrates that as the "max-depth" increases, the Random Forest classifier becomes more capable of capturing the underlying data structure, which results in higher accuracy. Higher proportions of features and more estimators generally contribute to better performance, particularly when the trees are deeper. This visualization highlights the importance of carefully tuning hyperparameters such as "max-depth", number of estimators, and feature proportion to optimize the performance of the Random Forest model.

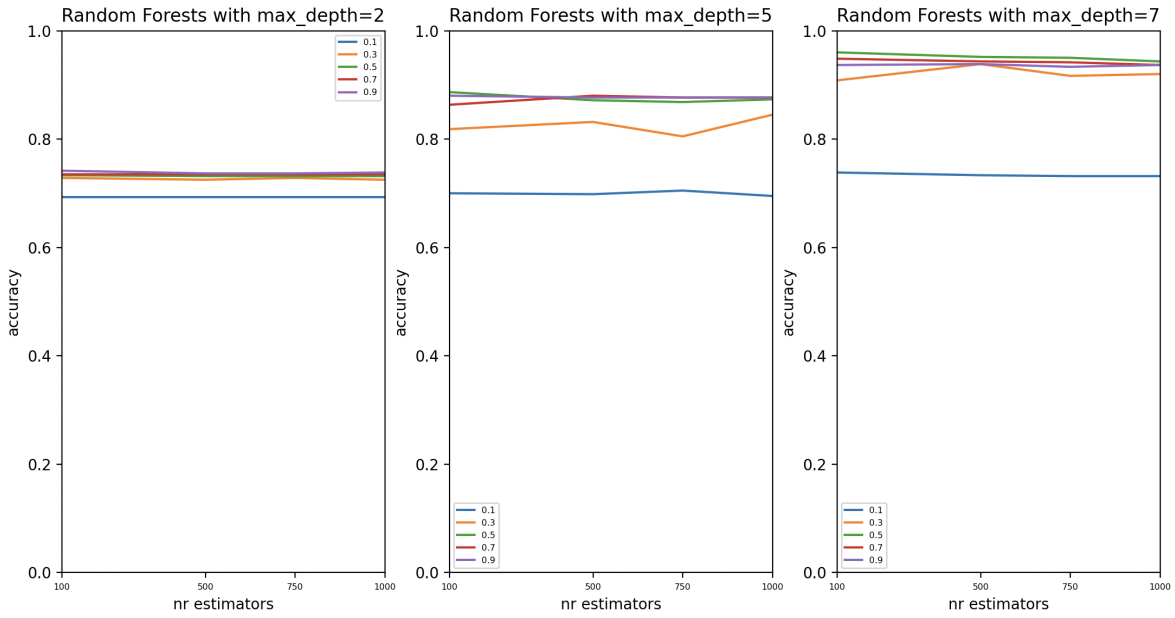


Figure 5.2: Accuracy of the Random Forest classifier with different "max-depth" values (2, 5, 7) as a function of the number of estimators and the proportion of features used. The figure shows that increasing "max-depth", the number of estimators, and the proportion of features generally improves model accuracy.

5.1.2 Interpretation of Results

The Random Forest model demonstrates strong performance on both the training and testing datasets. The high accuracy, recall, precision, AUC, and F1 scores on the training set indicate that the model has learned the underlying patterns effectively. On the testing set, the metrics remain robust, suggesting good generalization capabilities. This first approach was systematically evaluated using different values of 'max-depth', specifically 2, 5, and 7, to understand how the depth of the individual decision trees influences the model's ability to capture the underlying structure of the data. The best results were obtained with a 'max-depth' of 7, which is intuitively reasonable as it allows the model to capture more intricate and non-linear relationships within the dataset, thus enabling the trees to make more nuanced decisions that contribute to higher accuracy. Moreover, the model also tests among different estimators

100, 500 ,750, 1000 the best model configuration included 100 estimators, which refers to the number of individual decision trees in the Random Forest ensemble. A larger number of estimators generally enhances the stability and robustness of the model by reducing the variance of the predictions through the process of averaging the results of multiple trees [40]. Each of these trees is trained on a different bootstrap sample of the data, and by combining their outputs, the ensemble model is less likely to overfit compared to a single decision tree. In addition to the number of estimators, the model used a feature subset size of 0.5, meaning that at each split in every decision tree, only 50% of the available features were considered for selecting the best split. This technique, known as feature bagging or random subspace method [43], plays a critical role in reducing correlation among the trees in the ensemble. By decorrelating the trees, this approach ensures that the trees are more diverse, which improves the overall generalization ability of the Random Forest. The combination of these parameters—100 estimators and 0.5 features per split—optimally balanced bias and variance, yielding the best accuracy observed in the accompanying trees. This configuration highlights the power of ensemble learning methods like Random Forests, where careful tuning of hyperparameters can significantly impact the performance and generalization of the model [49]. From the confusion matrix (Table 5.1), we observe the following: True Positives (TP): 159 instances correctly identified as matches, True Negatives (TN): 414 instances correctly identified as non-matches, False Positives (FP): 2 instances incorrectly identified as matches, and False Negatives (FN): 25 instances incorrectly identified as non-matches. While the overall performance metrics are commendable, the presence of false negatives (FN = 25) and false positives (FP = 2) are a critical consideration. In the context of ensuring data deletion:

- **False Negatives (FN):** These are instances identified incorrectly as matches. In practical terms, this means that some data that should have been flagged for deletion were not, potentially leading to incomplete data removal [50].
- **False Positives (FP):** These are instances where matches were not identified correctly. In practical terms, this means that some data that was flagged for deletion should have not. Which would later on mean the deletion of other entities' data.

5.1.3 Conclusion from the first results

This initial attempt at training a Random Forest classifier has yielded very good results, demonstrating the model's ability to accurately identify matches and non-matches within the dataset. However, the presence of false negatives indicates that the model does not provide 100% safety for data deletion purposes. Specifically, some unmatched data may not be identified correctly, leading to potential retention of data that should have been deleted. It is essential to further refine the model, possibly by tuning hyperparameters [51] or employing additional techniques [52], to minimize false negatives and enhance the

reliability of data deletion processes. Following the initial experiment, where the Random Forest model was trained with various hyperparameters, a second set of experiments was conducted with slight modifications. The primary goal of this analysis was to improve the matching accuracy between two sets of columns, focusing on minimizing false negatives and enhancing the overall model performance. In this these experiments, the `max_depth` parameter of the Random Forest model was systematically increased to values of 7, 15, 20, and 35, compared to the initial experiments where `max_depth` values of 2, 5, and 7 were tested. As hypothesized, increasing the `max_depth` allowed the model to capture more intricate patterns within the data, resulting in nearly perfect accuracy.

5.1.4 Comparative Results

The results from this enhanced model configuration showed significant improvement over the initial setup, we can analyze visualize the results on the Figures (5.3, 5.4, 5.5) and in the table 5.3

Metric	Value
Accuracy	99%
Recall	97%
Precision	100%
AUC	99%
F1 Score	99%
True Positives (TP)	179
True Negatives (TN)	416
False Positives (FP)	5
False Negatives (FN)	0

Table 5.3: Performance metrics for the enhanced model configuration.

- **Accuracy:** The accuracy improved across all experiments, with the best model achieving perfect scores on the training set and near-perfect scores on the testing set.
- **Confusion Matrix:** The confusion matrix reflected an almost flawless classification, with a significant reduction in false positives and false negatives.

Confusion Matrix Analysis:

- **True Positives (TP):** The model correctly identified 179 instances as matches.
- **True Negatives (TN):** It also accurately identified 416 instances as non-matches.
- **False Positives (FP):** The model incorrectly classified only 5 instances as matches.
- **False Negatives (FN):** Importantly, the number of false negatives was reduced to 0, demonstrating a key improvement in the model's ability to identify matches correctly.

5.1.5 Impact of feature proportion

The figure also highlights the impact of using different proportions of features at each split in the decision trees. Specifically, lower proportions of features, such as 0.1, consistently resulted in lower accuracy across all tested `max_depth` values. This outcome is likely because using only 10% of the available features limits the model's ability to capture the full complexity of the data, particularly when the dataset includes a large number of informative features. When a smaller subset of features is used, the trees within the Random Forest may not have access to the most relevant variables for splitting at each node, leading to suboptimal decision boundaries. As the `max_features` value increases (e.g., 0.5, 0.7, 0.9), the model is better able to utilize the available information, which is reflected in the improved accuracy. This observation underscores the importance of balancing feature diversity with the need to maintain a sufficient amount of information at each split.

5.1.6 Conclusion

This second attempt at tuning the Random Forest model successfully addressed the shortcomings of the initial approach. By increasing the `max_depth`, the model was better equipped to handle the complexity of the dataset, achieving almost perfect performance. These results suggest that the Random Forest model, with appropriate hyperparameter tuning, is highly effective for this type of pattern matching task, providing strong assurance in data deletion contexts. Figures 5.3, 5.4, and 5.5.

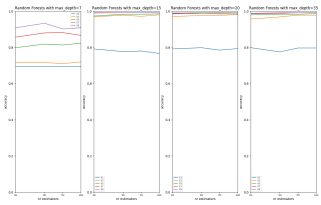


Figure 5.3: Accuracy of Random Forest models with varying `max_depth` values, showcasing consistent performance improvements.

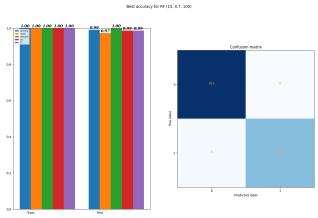


Figure 5.4: Confusion matrix of the Random Forest model with `max_depth=15`, `max_features=0.7`, and 100 estimators, demonstrating near-perfect classification performance.

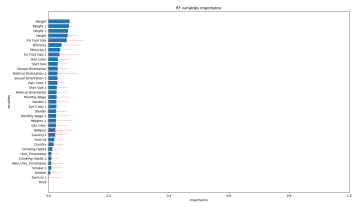


Figure 5.5: Importance of features as determined by the Random Forest model, highlighting the most influential variables in the matching process.

5.2 Exploring the known correspondence between variables

After evaluating the initial and improved approaches, a third experiment was conducted to further enhance the model's ability to accurately identify matches between columns. This final approach involved

a significant alteration to the dataset: the corresponding columns were transformed by taking the difference between them.

5.2.1 Dataset transformation

The core idea behind this transformation was to simplify the matching process for the Random Forest model. Specifically, for each pair of columns that were expected to match, the difference between their values was computed. This resulted in a new dataset where each entry in a transformed column was 0 if the original columns matched, and a non-zero value otherwise. This modification had a considerable impact on the classification task. In this new setup:

- **Matched Rows:** rows that were labeled as "matched" would now contain only zeros across all columns. This represented a perfect match across all paired columns.
- **Unmatched Rows:** any row containing a non-zero value indicated a mismatch in at least one column pair, leading to its classification as "unmatched."

5.2.2 Results

This transformation made the classification task trivial for the Random Forest model. The model achieved perfect accuracy across all tested configurations, regardless of the number of features or estimators used. The confusion matrix confirmed these results, showing no false positives or false negatives, and the feature importance analysis indicated that all transformed features contributed equally to the classification task.

Key Observations:

- **Perfect classification:** the Random Forest model was able to perfectly distinguish between matched and unmatched rows, with 100% accuracy, recall, precision, AUC and f1 on both the training and testing sets.
- **Feature Importance:** the feature importance analysis showed that, due to the uniform nature of the transformed data, each feature had an equal role in the decision-making process, leading to uniformly high importance scores.

5.2.3 Conclusion

This final approach demonstrated that by transforming the dataset to directly reflect the matching condition through simple arithmetic operations, the Random Forest model could achieve perfect classification performance. This method effectively reduced the complexity of the problem, allowing the model to

make clear and accurate distinctions between matched and unmatched data, irrespective of the model's configuration. This result underscores the potential of data preparation, in general, and feature engineering, in particular in enhancing machine learning model performance, particularly in scenarios requiring precise and reliable matching. Figures 5.6, 5.7, and 5.8

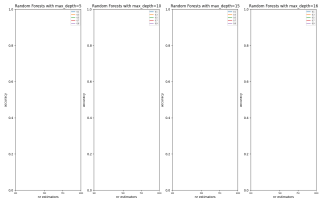


Figure 5.6: Accuracy of the Random Forest model across various configurations, demonstrating perfect classification with the transformed dataset.

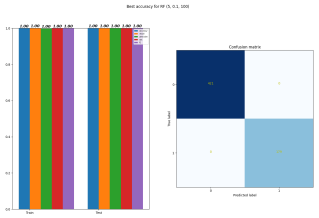


Figure 5.7: Confusion matrix showing perfect classification performance, with no false positives or false negatives.

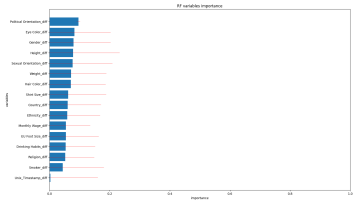


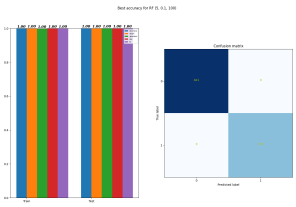
Figure 5.8: Feature importance analysis indicating uniform importance across all transformed features, reflecting the equal role of each feature in the decision-making process.

In this section, we present a grid of confusion matrices (Figure 5.9) generated from different configurations of the Random Forest model. The configurations vary in terms of the number of variables and the maximum depth of the trees. The results demonstrate that, even with smaller max depths and a reduced number of variables, the model was still able to achieve perfect accuracy. This indicates the robustness of the model and its ability to generalize well even under constrained conditions.

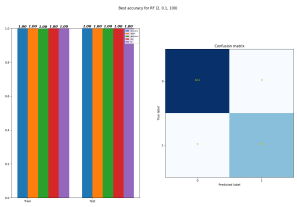
5.3 Study on the impact of the number and type of variables

In this grid (Figure 5.10), we explore how the Random Forest model's performance changes as we reduce the number of variables and adjust the maximum depth. The variables removed first were those that exhibited higher importance in initial models, often those with a wider range of possible values, such as weight and height. From these graphs, it is evident that variables with a larger range of potential outcomes tend to have greater influence on the model's predictions. This guided the strategy of progressively removing these influential variables to pinpoint when the model's performance would begin to degrade. Notably, even as the number of variables and the model's depth decreased, the model maintained strong predictive accuracy, suggesting a high degree of robustness.

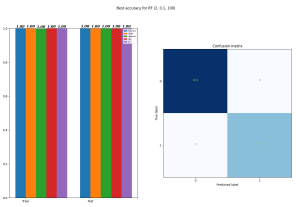
To further investigate the impact of different types of variables on the model's performance, we conducted experiments using only numeric variables and compared them with experiments where categorical variables were included. Specifically, we focused on the accuracy achieved with a smaller set of variables and varying importance levels. The analysis revealed that even when using a minimal set



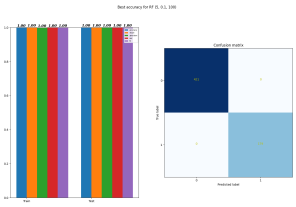
(a) 16 variables, max depth 5-16



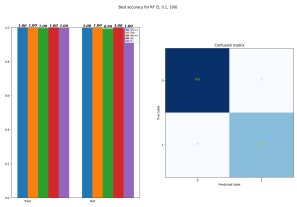
(b) 16 variables, max depth 1-4



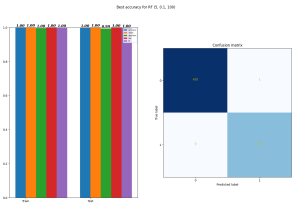
(c) 12 variables, max depth 1-4



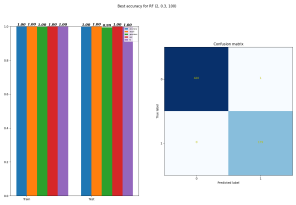
(d) 12 variables, max depth 5-16



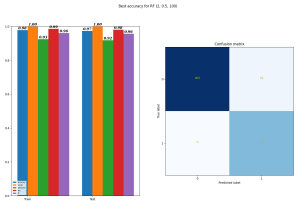
(e) 8 variables, max depth 5-16



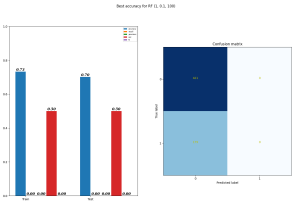
(f) 8 variables, max depth 5-8



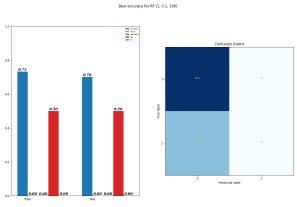
(g) 8 variables, max depth 1-4



(h) 4 variables, max depth 1-4

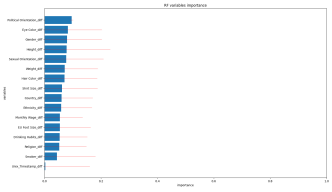


(i) 2 variables, max depth 1-2

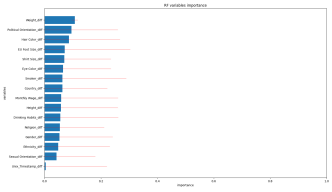


(j) 1 variable, max depth 1

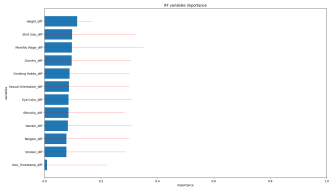
Figure 5.9: Grid of Confusion Matrices with varying number of variables and max depth values.



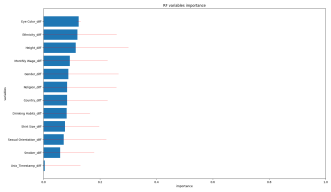
(a) 16 variables, max depth 5-16



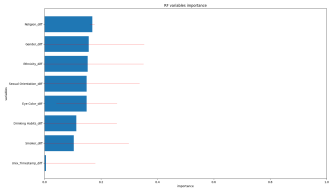
(b) 16 variables, max depth 1-4



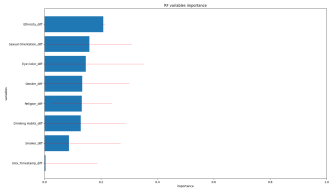
(c) 12 variables, max depth 1-4



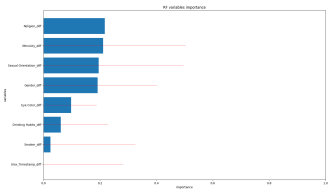
(d) 12 variables, max depth 5-16



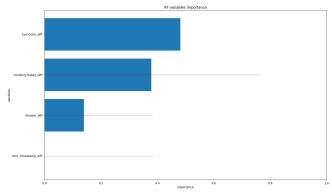
(e) 8 variables, max depth 5-16



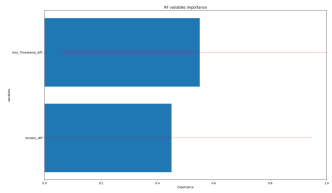
(f) 8 variables, max depth 5-8



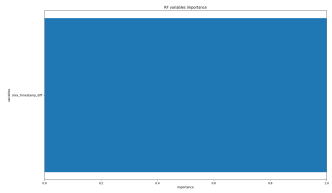
(g) 8 variables, max depth 1-4



(h) 4 variables, max depth 1-4



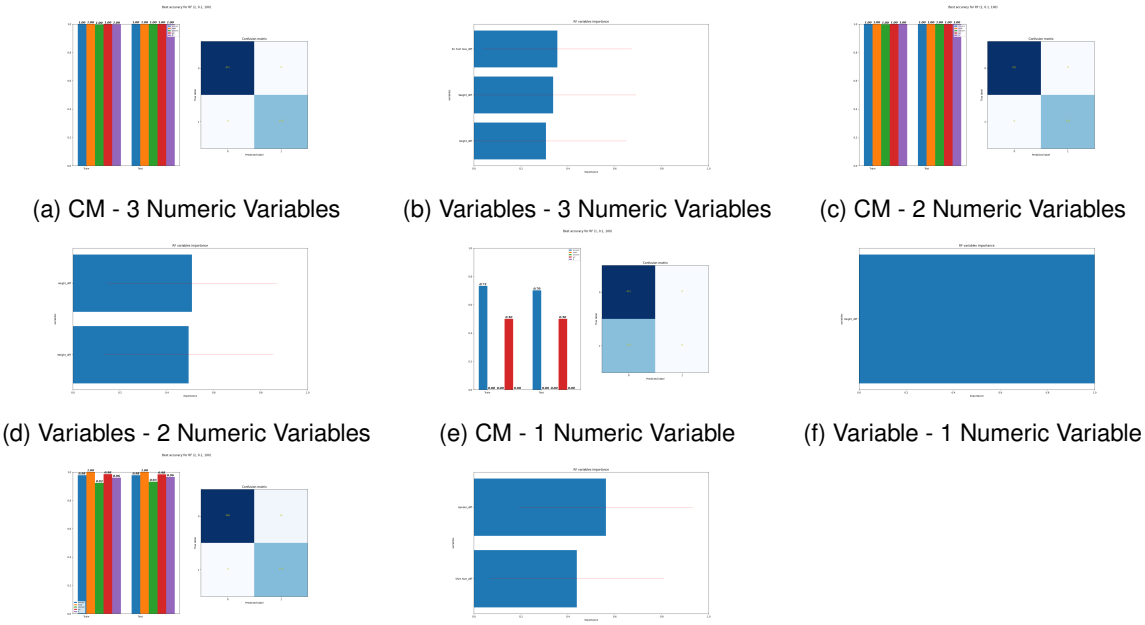
(i) 2 variables, max depth 1-2



(j) 1 variable, max depth 1

Figure 5.10: Grid of variable importance's with varying number of variables and max depth values.

of numeric variables as we can see in a,b,c,d of (Figure 5.11), the model achieved perfect accuracy. In contrast, when only categorical variables (such as drop-down options) were considered (g), (h) on the (Figure 5.11) , the performance dropped significantly. This observation is consistent across various model configurations, as seen in the confusion matrices and the corresponding variable importance plots. This experiment emphasizes the strong influence of numeric variables, especially those with a higher number of possible values (e.g., height, weight), on the model's accuracy. As for example on (Figure 5.11) (g) with the same amount of variables we achieve worse results (lowerd accuracy, auc, precision, f1) then in (c).



(a) CM - 3 Numeric Variables (b) Variables - 3 Numeric Variables (c) CM - 2 Numeric Variables

(d) Variables - 2 Numeric Variables (e) CM - 1 Numeric Variable (f) Variable - 1 Numeric Variable

(g) CM - 2 Dropdown Variables (h) CM - 2 Dropdown Variables

Figure 5.11: Comparison of Model Performance with Numeric and Categorical Variables.

5.4 Conclusion

The knowledge of variable pairs enables the model to achieve near-perfect performance in identifying matches and non-matches. This is evident from the significantly high accuracy, recall, precision, and other evaluation metrics achieved during the experiments. However, it is important to note that decision tree nodes based on numeric variables tend to create more general partitions, which requires deeper trees to isolate the distinct records. This need for greater tree depth can negatively impact model performance, as deeper trees become necessary, and Random Forests with lower max depths may struggle to correctly classify these records. Therefore, careful consideration of feature types and appropriate tree depth are essential to maintaining model efficacy and accuracy.

6

Scenario 2: DB - DB with numerical IDs

Contents

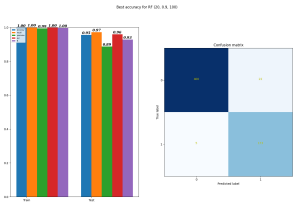
Given the excellent results achieved in the initial phase of this research, where the correlation between HTTP requests and database logs was successfully identified, it seemed natural to extend this investigation to encompass connections between logs from two different databases. This scenario mirrors real-world conditions where information is typically dispersed across multiple databases.

6.1 Scenario description

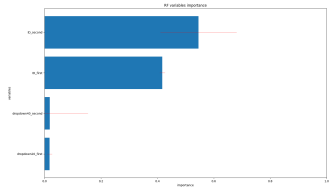
To simulate this environment, a second form was developed with a significant modification: user authentication. In this setup, users are required to authenticate themselves before submitting data to a database. Subsequently, the same users fill out another form, using the same credentials, but this time the data is stored in a different database. The forms were completed randomly to ensure that the databases contained varied results. In the initial phase of this study, all variables were selected from dropdown menus with up to four options, as illustrated in Figure **??**. These two datasets represent the user's actions across different services. Since the authentication key is known and consistent across both services, we display all the variables, including the authentication key, in our dataframe. The goal is to understand how the Random Forest model handles the scenario where a consistent key exists across multiple databases, which effectively links the datasets. In our database, we begin by storing a total of 42 variables for each user. These variables include 20 values from each instance of form submission, resulting in 40 variables that represent the user's actions and data input across two different services. Additionally, we include 2 extra variables that represent the authentication keys used during each login process. In terms of the dataset, we have 2177 rows corresponding to non-matched correspondences and 823 rows corresponding to correct matches, representing a balance in the dataset where 27.43% of the rows are matched correspondences. As explained earlier, with the 42 variables, we now have 21 variables corresponding to the HTTP request logs and 21 variables corresponding to the database logs, along with an additional variable to label whether the row is matched or unmatched, leaving a total of 43 features (being 1 the label). These keys are crucial as they serve to link the data from the two separate instances, providing a consistent identifier across the datasets. The objective of this experiment is to determine how the Random Forest model performs under these conditions, where the existence of a key is present but unknown and how it affects the dataset. This approach allows us to simulate and analyze the Random Forest model's behavior in scenarios where data integration across multiple databases is required.

6.2 With Different IDs

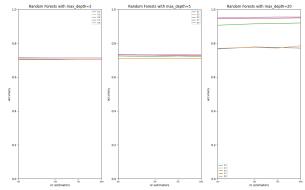
In this section, we analyze the impact of varying the number of input variables on the performance of a Random Forest model. The experiments were conducted using three different configurations: one with 42 variables (including 2 IDs), another with 12 variables (including 2 IDs), and the last one with just 4 variables (including 2 IDs). The goal was to understand how the number of variables influences the model's ability to correctly identify patterns, particularly focusing on the authentication keys that are consistently present across both datasets. We can visualize the results of the performance on (Figures 6.1) Starting with just 4 variables, the model demonstrated the best performance. The reduced complexity



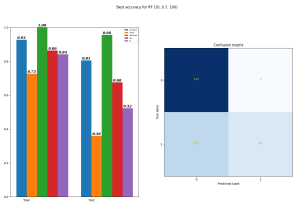
(a) Confusion Matrix (4 variables)



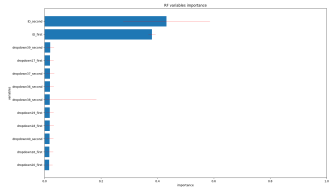
(b) Variable Importance (4 variables)



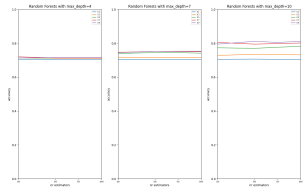
(c) Accuracy vs Estimators (4 variables)



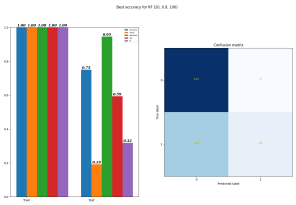
(d) Confusion Matrix (12 variables)



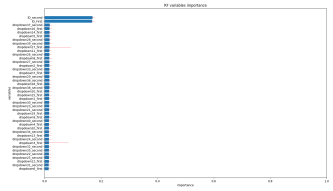
(e) Variable Importance (12 variables)



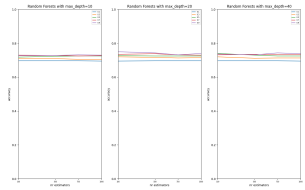
(f) Accuracy vs Estimators (12 variables)



(g) Confusion Matrix (42 variables)



(h) Variable Importance (42 variables)



(i) Accuracy vs Estimators (42 variables)

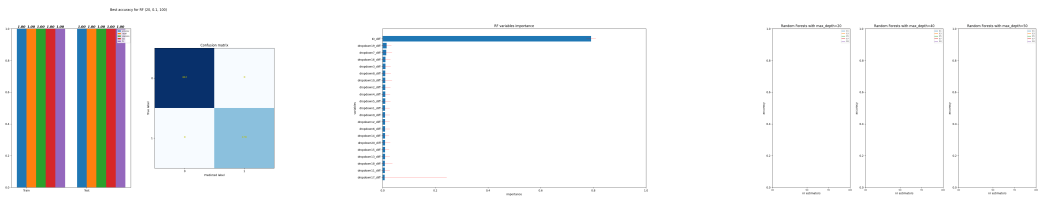
Figure 6.1: Comparison of Random Forest performance with different numbers of variables: 4, 22, and 42 variables (including 2 IDs in each case).

and minimal noise allowed the Random Forest to clearly focus on the most relevant features, particularly the ID keys, which led to significant improvements in accuracy. The confusion matrix highlights this improvement, as the simplicity of the dataset made the underlying patterns more distinguishable, resulting in better generalization. When the number of variables increased to 12, the performance remained strong, but there was a slight decline in efficacy. Although the reduction in noise still helped the model focus on key variables, the introduction of additional features added some complexity. This made it more difficult for the model to consistently test the ID variables, as it had more features to consider for matching, which affected its overall performance slightly, as seen in the metrics. Finally, with all 42 variables, the model struggled to generalize well. The large number of irrelevant variables made it less likely for the ID variable to be selected, which hindered the model's ability to recognize patterns in the test set. The confusion matrix shows that the model's performance suffered because the ID, the most important variable in the test set, was not consistently tested. As the model encountered IDs it hadn't seen before, it treated them as random selections, leading to discrepancies between training and test data. The imbalance caused by relying on a single ID without enough other matching variables contributed to this decline in performance. With fewer variables, the model tested more IDs, as there were fewer other variables for matching, leading to better outcomes.

These experiments highlight the importance of careful variable selection in machine learning. When too many irrelevant variables are included, the model can become overwhelmed and fail to generalize well, even when critical variables like the ID keys are present. By reducing the number of variables, the model's focus on the most important features increases, leading to better performance.

6.3 With the same ID

After conducting the initial experiments where the Random Forest model struggled with varying numbers of variables and to deal adequately with the id, due its numerical nature, a new approach was tested: computing the difference between corresponding columns when the authentication key is known. The results of this approach were impressive, with the model achieving perfect accuracy, precision, recall, and F1 scores across the board. This outcome is intuitive. When we know the location of the key in both datasets, subtracting the corresponding columns (i.e., calculating the difference) effectively isolates the effect of the key. This simplifies the problem significantly for the model, as all other variables become irrelevant. The key's presence and consistency across both datasets become the sole determinant of the model's output. The attached image (Figures 6.2) illustrates this point. The confusion matrix shows perfect classification, with all instances correctly identified. The variable importance chart highlights the overwhelming dominance of the key, as expected, and the accuracy chart further confirms that, regardless of the number of estimators or the depth of the trees, the model consistently performs at its best. This experiment underscores an interested insight: when the key is known and used to link datasets, the model's performance can reach perfection. It also highlights the importance of understanding the nature of the data and the relationships between variables when building predictive models. In cases where a key exists, leveraging it effectively can simplify the model's task and lead to significantly improved performance. Which links to the importance of the importance of knowing the IDs for matching.



(a) Confusion Matrix with Diff (b) Variable Importance with Diff (c) Accuracy vs Estimators with Diff

Figure 6.2: Random Forest performance after calculating the difference between corresponding columns when the key is known. The model achieves perfect accuracy due to the significant impact of the key.

6.4 Experiment with text variables

In this section, we explore a variation of the previously discussed experiment. The difference lies in the nature of the input variables. While the initial setup involved dropdown selections with limited options, this new approach focuses on text inputs, in this form the user can only add text inputs. The authentication mechanism remains consistent with the prior setup, ensuring that users must log in to submit their data. However, the change to text-based inputs introduces a more complex and varied dataset. Each text box was encoded into a unique, random value, as the specific content within the text boxes was not relevant to the analysis. The objective here was to assess how the Random Forest model performs with more complex input types and whether it can effectively generalize from this data. As anticipated, the results were notably worse compared to those obtained with dropdown inputs. The attached graphs illustrate the outcomes of this experiment. It is evident that the model's performance has deteriorated, which is likely due to the increased complexity and variability introduced by the text inputs. The confusion matrices, accuracy vs. estimator plots, and variable importance graphs show poorer performance metrics compared to the previous tests. This decline in performance could be attributed to several factors:

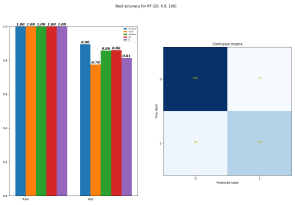
1. **Increased input variability:** text inputs introduce a much higher level of variability compared to dropdown selections, making it harder for the model to identify consistent patterns.

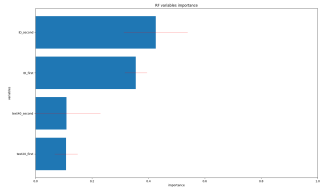
2. **Overfitting:** Some of the results suggest potential overfitting, where the model performs well on the training data but fails to generalize to unseen data. This could be due to the model attempting to learn from noise rather than from relevant variables.

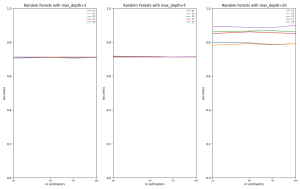
3. **Feature importance:** The variable importance indicate that the model is relying on the key features, but the overall impact of these features is less pronounced in the presence of more complex and varied inputs, which leads to worse results since they are contemplating irrelevant (and wrong) features for the matching.

These findings highlight the challenges associated with handling more diverse data types in machine learning models. They emphasize the importance of feature engineering and the potential need for more sophisticated preprocessing techniques when working with text data. The results can be identified in : (Figures 6.3,6.4, and 6.5)

These results clearly demonstrate the impact of variable complexity on model performance, reaffirming the need for careful consideration of input types in predictive modeling.

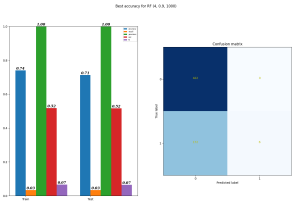


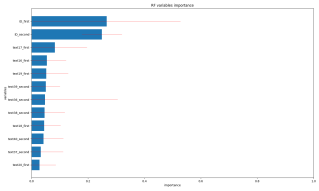


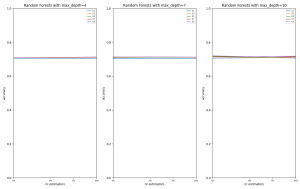


Confusion Matrix (4 Text Variables) Variable Importance (4 Text Variables) Accuracy vs Estimators (4 Text Variables)

Figure 6.3: Random Forest performance with 4 text variables (including 2 IDs).

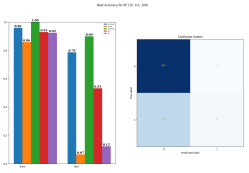


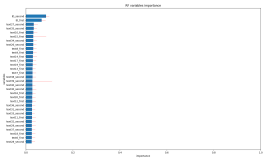


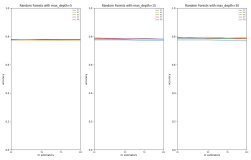


Confusion Matrix (12 Text Variables) Variable Importance (12 Text Variables) Accuracy vs Estimators (12 Text Variables)

Figure 6.4: Random Forest performance with 12 text variables (including 2 IDs).







Confusion Matrix (32 Text Variables) Variable Importance (32 Text Variables) Accuracy vs Estimators (32 Text Variables)

Figure 6.5: Random Forest performance with 32 text variables (including 2 IDs).

6.5 Analysis of SHAP Values in Three Experimental Scenarios

SHAP values provide a unified measure of feature importance by assigning each feature an attribution value that reflects its contribution to the model's output. SHAP values leverage concepts from cooperative game theory, particularly the Shapley value, to fairly distribute the "payout" (i.e., model prediction) among the features [53]. In this section, we analyze SHAP values for three distinct experimental scenarios to interpret the impact of different sets of features on model predictions.

6.5.1 SHAP Analysis of Feature Differences

The first experiment involved computing SHAP values for a dataset with 10 variables representing differences between attributes captured in the first form. The resulting SHAP summary plot (Figure 6.6) reveals the relative impact of each feature on the model's predictions.

- **Height_diff** is the most influential feature, with high SHAP values that significantly impact model predictions. The plot shows that large differences in height (red) tend to push predictions in a positive direction, indicating a strong impact on the model outcome.
- **Weight_diff** and **EU Foot Size_diff** also demonstrate considerable influence, with their SHAP values distributed symmetrically around zero. This suggests that while these features can impact predictions, the effect is more balanced between increasing and decreasing the model's output.
- Other features like **Ethnicity_diff** and **Gender_diff** have a more moderate impact, often centered around zero. The varying shades of blue and red illustrate how specific feature values contribute differently to model predictions.

These findings suggest that physical characteristic differences are significant contributors to the model's decision-making process in this context.

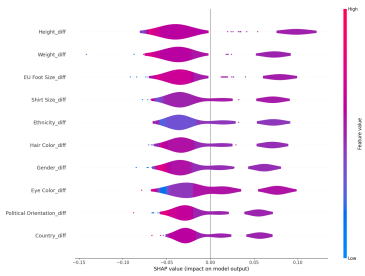


Figure 6.6: SHAP summary plot for the first experiment analyzing feature differences from the first form.

6.5.2 SHAP Analysis of Dropdown Selections

The second experiment evaluated SHAP values for a dataset containing dropdown selections from a form. Figure 6.7 illustrates the SHAP summary plot for these features.

- **ID_second** and **ID_first** are the most impactful features, dominating the model's prediction landscape. The distribution of SHAP values shows a wide range of impacts, from strongly positive to strongly negative, highlighting the high variability and influence of these identifiers.

Dropdown features like **dropdown18_first** and **dropdown19_first** show moderate contributions to the model's predictions, with a tendency for higher values (red) to increase the likelihood of a positive outcome. Other features, such as **dropdown39_second** and **dropdown17_first**, exhibit more symmetrical SHAP distributions around zero, indicating a balanced but still notable influence on predictions. Some dropdowns display more importance than others; however, this is largely due to randomness, as the only truly impactful features for the model are the **ID_first** and **ID_second** identifiers.

6.5.3 SHAP Analysis of Text Features

The third experiment involved SHAP analysis for text features extracted from form inputs. Figure 6.8 presents the SHAP summary plot for this set of features.

- As seen in the previous experiment, **ID_second** and **ID_first** continue to dominate the feature importance landscape. Their wide SHAP value distribution indicates substantial influence across a range of predictions.

Textual features like **text19_first** and **text16_first** show moderate contributions to the model's predictions, where higher or lower values influence the outcome in a noticeable way. Other features, such as **text39_second** and **text36_second**, exhibit more balanced SHAP distributions around zero, indicating a weaker but still relevant effect on predictions. Similar to the dropdown features, the importance of these textual features is driven by randomness rather than any particular rationale, as the truly impactful features for the model remain the **ID_second** and **ID_first** identifiers.

This experiment underscores the importance of textual data, as specific text-based features can heavily impact model decisions, especially when identifiers are also involved.

The SHAP value analyses across these three experiments reveal that certain features consistently exert significant influence on model predictions. Particularly, identifiers and physical characteristic differences emerge as key drivers. The importance of a variable is directionally proportional to its variance, meaning that features with higher variability have a greater impact on the model's predictions. SHAP values provide an interpretable and fair way to distribute the impact of features, offering valuable insights

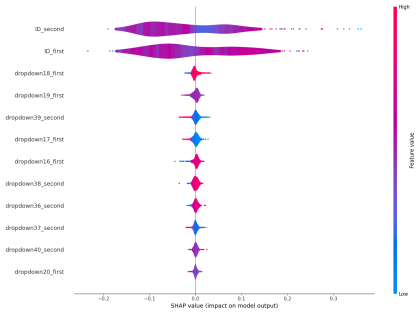


Figure 6.7: SHAP summary plot for the second experiment focusing on dropdown selections.

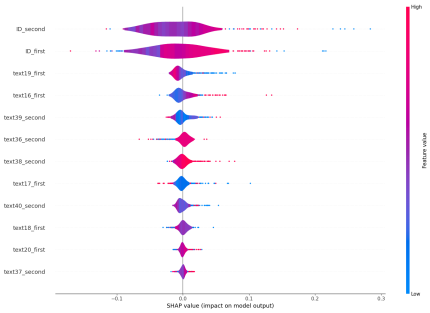


Figure 6.8: SHAP summary plot for the third experiment analyzing textual form inputs.

into the model's decision-making process [53]. This approach helps in validating model behavior and aligns the predictive power with domain knowledge.

6.6 Comparison of Decision Tree and Random Forest Classifiers

In this experiment, a classification problem was addressed using two distinct approaches, now with a Decision Tree classifier. The dataset comprised four features, where two columns matched each other, and the other two were random (a total of 4 variables to compare with previous results, the dataset chosen was the one from the dropdown form). Initial experiments were conducted using a Decision Tree model, followed by testing with a Random Forest model. The motivation for incorporating a Decision Tree model in this analysis was to provide additional means of comparison within classifiers, to observe the structure of a single tree, and to understand to what extent a single tree could replicate the performance of a Random Forest. In Random Forests, we used a relatively small number of estimators, and the number of trees did not vary significantly. Therefore, we aimed to assess the impact of using just one tree in comparison to multiple estimators in Random Forests. The results indicate that the Random Forest consistently outperformed the Decision Tree in terms of classification accuracy, as evidenced by the performance metrics and confusion matrices presented in Figures 6.10 and 6.1.

6.6.1 Decision Tree Performance Analysis

Figure 6.9 presents the accuracy of Decision Tree models using two different criteria, Gini and entropy, as a function of tree depth. The accuracy curves show that performance generally improves with increasing tree depth, with the Gini criterion slightly outperforming entropy at higher depths. However, the incremental gains diminish as the tree becomes deeper, reflecting the tendency of Decision Trees to overfit the training data as complexity increases. **Figure** 6.10 provides a detailed analysis of the Deci-

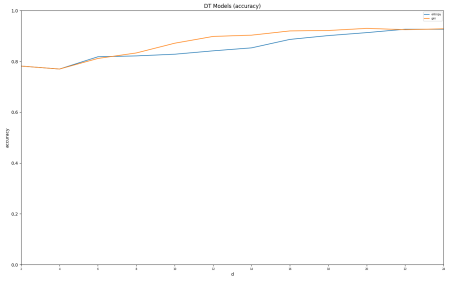


Figure 6.9: Decision Tree results: Accuracy as a function of tree depth for different criteria (entropy and Gini).

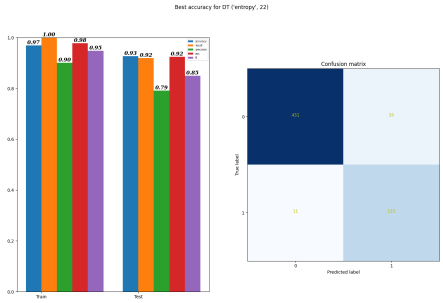


Figure 6.10: Decision Tree performance: Accuracy, Recall, Precision, AUC, and F1-score metrics along with the confusion matrix.

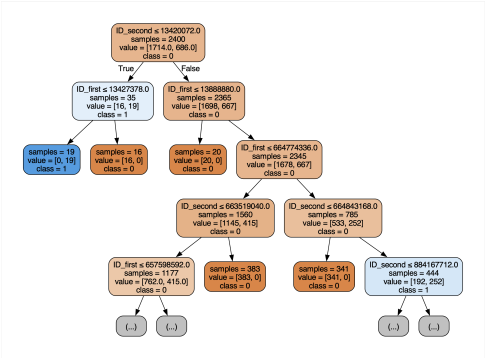


Figure 6.11: Decision tree variable chosing

sion Tree's performance on both the training and test datasets. The bar chart shows five key metrics: accuracy, recall, precision, AUC, and F1-score. It is evident that while the Decision Tree performs well on the training data (high accuracy and recall), the performance drops on the test set, particularly in terms of recall and F1-score. This drop, where the model captures specific patterns from the training data that are not useful and do not generalize well to new, unseen data. The accompanying confusion matrix in Figure 6.10 shows that the Decision Tree makes a substantial number of false positives (33 instances) and a few false negatives (11 instances). This imbalance suggests that the Decision Tree struggles to accurately classify certain instances, which negatively impacts the model's overall precision and recall. The Decision Tree visualization in Figure 6.11 highlights that 'ID-first' and 'ID-second' are the most important features driving the model's decisions. Although there are a total of four columns in the dataset, the other two features are not prominently used in the initial splits, indicating that 'ID-first' and 'ID-second' have the strongest influence on the classification outcomes. This suggests that the model heavily relies on these two features for decision-making, which is correct, underscoring their predictive power compared to the other columns, on the other hand worst results were observed on the test.

6.6.2 Random Forest performance comparison

Figure 6.1 (with the same amount of variables, 4) displays the performance of the Random Forest classifier, showcasing the same set of metrics as in the Decision Tree analysis. The Random Forest demonstrates superior performance across all evaluated metrics on both the training and test datasets, with particularly notable improvements in recall, precision, and F1-score. The bar chart in Figure 6.1 shows that the Random Forest achieves near-perfect accuracy on the training set and maintains a high level of performance on the test set, unlike the Decision Tree, which exhibits a performance drop. This stability indicates that the Random Forest, thanks to its ensemble nature. The confusion matrix for the Random Forest in Figure 6.1 reveals a much lower number of misclassifications compared to the Decision Tree. The false positive rate is significantly reduced, with only 23 instances, and false negatives are almost negligible at 4 instances. This balanced performance across all classes highlights the Random Forest's robustness and its ability to handle variability in the data more effectively than a single Decision Tree.

6.6.3 Superior performance of Random Forest

The Decision Tree model exhibited high training accuracy but a notable drop in test performance, indicating potential overfitting. This is typical behavior for decision trees due to their nature of fitting data very closely without accounting for variance and noise. On the other hand, the Random Forest classifier demonstrated significantly improved performance on the test set, achieving higher accuracy, recall, and F1-score compared to the Decision Tree. Several factors contribute to this superior performance:

- **Ensemble learning:** In this work, Decision Trees often relied too much on specific ID values seen during training. This became a problem during testing when the model encountered new, unseen IDs, leading to misclassifications. Random Forest, however, mitigated this issue by constructing multiple trees, each trained on different samples. This reduced the model's dependence on specific ID values, as the ensemble approach allowed Random Forest to generalize better by averaging predictions across trees, leading to more robust and accurate results.
- **Feature randomization:** In this experiments, Decision Trees focused heavily on ID variables, which led to overfitting and poor generalization. Random Forest addressed this by introducing randomness in feature selection at each split, allowing different trees to consider various combinations of features. This diversification prevented the model from over-relying on ID variables, resulting in better overall performance and more balanced predictions across different features.
- **Handling random and correlated features:** In datasets where some features are random and others are highly correlated, a Decision Tree might overly rely on the correlated features, leading

to biased splits and suboptimal generalization. This reliance can cause overfitting, as the model memorizes specific patterns in the training data but struggles when presented with unseen or slightly different data during testing. For example, in Figure 6.11, if the value of *IDsecond* is 13420072.0, the tree splits to the left. However, even if *IDfirst* is 13427378.0 (a different ID), the model incorrectly labels this as a match, because it relies on thresholds learned during training. Since the model did not see these specific combinations of ID values during training, it fails to generalize properly. Random Forest mitigates this issue by distributing reliance across multiple features, reducing the risk of overfitting to specific correlated values and improving generalization on unseen data.

- **Reduction of variance:** Decision Trees are high-variance models; a small change in data can drastically alter the tree structure. Random Forest reduces variance by averaging the outputs of multiple trees, resulting in a smoother decision boundary that captures the underlying data structure more effectively, as seen in the comparison of confusion matrices.

The Decision Tree (Figure 6.10) achieved a relatively high accuracy on the training set but struggled to maintain that performance on unseen data. Conversely, the Random Forest (Figure 6.1) maintained slightly higher performance on both training and test sets, highlighting its ability to generalize better to new data.

These findings align with established literature, which underscores the advantage of Random Forest models in scenarios where data complexity, noise, and feature interaction are significant [40]. By averaging multiple decision trees, Random Forest can handle variability more effectively than a single Decision Tree, making it a preferred choice for robust and reliable predictions in many classification tasks.

6.7 Perfect classification with known IDs using Decision Tree

The Decision Tree results, as shown in Figures 6.12 and 6.13, demonstrate that when the “IDdiff” is included as a feature, the model achieves perfect classification. The tree splits directly on the “IDdiff”, perfectly separating the classes, resulting in 100 accuracy. Specifically, when “IDdiff” smaller or equal to 58690.0 (which symbolizes the id match since in the database it would be a 0 for this), the model accurately classifies all samples as class 1, and when “IDdiff” higher then 58690.0, it classifies all samples as class 0. This is because the subtraction of two equal numbers results in 0, while all other differences were greater than 58690 (as could be another number) when the subtraction was performed. This indicates that the inclusion of “IDdiff” allows the model to perfectly distinguish between the two classes, highlighting the feature’s critical importance in the classification task.

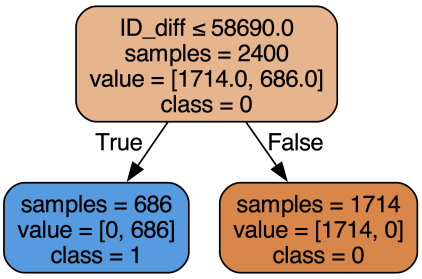


Figure 6.12: Decision Tree with IDdiff as the main split, perfectly classifying all samples.

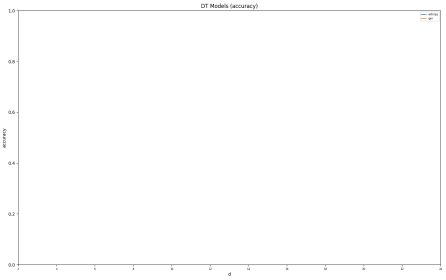


Figure 6.13: Accuracy of Decision Tree models showing perfect classification with known IDdiff.

6.8 Conclusion

The experiments conducted in this study highlighted key insights into the behavior of Decision Trees and Random Forests when matching features from HTTP logs and database logs.

First, when the features between the two datasets were identical(One Form - One Database), the problem was trivial, and a Decision Tree was sufficient to achieve perfect classification. The variables dominated the model's decision-making process, and the model performed with high accuracy since it only needed to match identical keys.

However, when the features were not identical ((DB) x (DB)), but instead represented different values from separate databases, the model's ability to generalize declined. Although the ID's remained the most important features, the model struggled with generalization, as shown in the Decision Tree experiment. This is because the model overly relied on specific values observed during training and failed to adapt when encountering unseen combinations of IDs in the test set.

The third key finding is the significance of the number and type of variables used for matching. In the experiments, reducing the number of irrelevant variables improved the model's performance by allowing it to focus on the critical features, particularly the ID keys. In contrast, increasing the number of irrelevant features added complexity, making it harder for the model to generalize effectively, as seen in the results with 42 variables. The Random Forest model, with its ensemble learning and feature randomization, was better able to mitigate these issues to some extent by distributing the reliance on multiple features and reducing the risk of overfitting.

Overall, the results emphasize the importance of carefully selecting relevant variables and understanding the nature of the dataset when building predictive models, particularly when working with different databases and diverse input features.

7

Scenario 3: DB - DB with textual IDs

Contents

In many data integration scenarios, especially involving personal data, a key challenge is ensuring the integrity and consistency of identifiers, such as person names, across multiple databases. This issue is particularly relevant in contexts where a person's name may appear with slight variations, abbreviations, or even missing components in one database compared to another [54]. In our study, we explored this scenario by simulating a situation where names from forms had slight differences, including variations in abbreviations, missing letters, additional characters, or differing numbers of names.

To evaluate the effectiveness of different string comparison methods, we considered longer names with an average of five components (first name, middle names, and surnames). This reflects realistic complexities encountered in multi-source data environments. To measure how similar or different two names were, we tested both Hamming and Levenshtein distances. The Hamming distance, which

counts the number of differing characters at the same positions in strings of equal length, quickly proved inadequate due to its limitations: it only works well when comparing strings of the same length and fails to account for insertions or deletions [55].

7.1 Levenshtein distance: a superior metric

The Levenshtein distance [56] (Edit Distance), on the other hand, measures the minimum number of single-character edits (insertions, deletions, or substitutions) required to change one string into another [57]. This approach is particularly suited for our scenario because it accurately captures real-world variations in names. For instance, it can handle cases where one name includes additional middle names or where abbreviations are used. Figure 7.2 and Figure 7.1 illustrate how the Levenshtein distance provided significantly more reliable classification results compared to Hamming distance, as can be observed from the results on the confusion matrix in Figure 7.4. The Random Forest model using Levenshtein distance achieved perfect classification with zero misclassifications in the confusion matrix, as seen in Figure 7.2, indicating a robust and accurate identification of name matches.

7.2 Performance analysis

Figures 7.3 shows the feature importance derived from the model using Levenshtein distance. Levenshtein distance consistently emerged as the most influential feature, underscoring its superior role in distinguishing matching and non-matching names. This further highlights why Levenshtein distance is preferable in data linkage tasks involving person names.

Our results demonstrate that Levenshtein distance offers a more practical and effective solution for comparing names across databases, particularly in scenarios involving varied name formats and inconsistencies. This approach allows for more accurate data integration, reducing the risk of mismatches and ensuring greater integrity in person identification [58].

7.3 Testing Name Matching with Shorter Names

In this experiment, we aimed to investigate the performance of name matching algorithms when dealing with shorter names, as some individuals, particularly in certain cultural contexts, may have only two names (e.g., ”John Doe”). The goal was to understand how the effectiveness of the matching process would change when the names used as identifiers are shorter and more likely to overlap across different individuals [57].

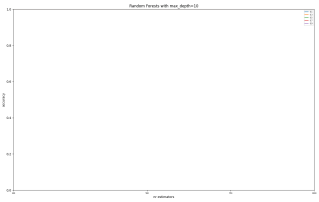


Figure 7.1: Accuracy results using Levenshtein distance. The model achieved perfect accuracy, highlighting its effectiveness in identifying matches.

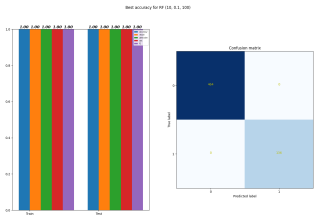


Figure 7.2: Confusion matrix showing zero misclassifications when using Levenshtein distance, demonstrating its superior accuracy in classifying name matches.

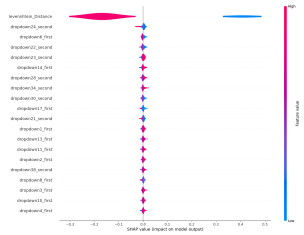


Figure 7.3: SHAP values for the Levenshtein distance model, showing its high impact on predictions.

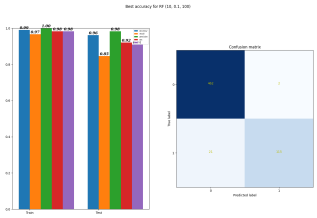


Figure 7.4: Confusion matrix showing the results when using Hamming distance, highlighting its limitations compared to Levenshtein distance.

7.3.1 Scenario and Methodology

We simulated a scenario with shorter names to test the effectiveness of both Hamming and Levenshtein distances in distinguishing between different individuals. In contrast to the previous experiment with longer names, this setup presented a more challenging situation due to the higher probability of identical name sequences appearing for distinct individuals, such as ”John Doe” appearing for two different people [55].

7.3.2 Performance Analysis

As expected, the results were considerably worse compared to scenarios involving longer names. This is primarily because shorter names offer less uniqueness as identifiers. In some cases, names of non-matching individuals were identical, leading to false positives in the classification task. For instance, if two distinct individuals were recorded as ”John Doe,” the algorithm would struggle to differentiate them based on name alone, significantly undermining the integrity of the matching process [59].

Figures 7.5, 7.6, and 7.7 show the performance of the Random Forest model using Hamming distance for shorter names. Notably, the confusion matrix in Figure 7.6 demonstrates an increase in misclassifications, confirming the difficulty in distinguishing between distinct records with similar names. On

the other hand, Figures 7.8, 7.9, and 7.10 illustrate that while Levenshtein distance still performed better than Hamming, the overall accuracy decreased compared to the scenario with longer names.

These findings highlight the challenges of using shorter names as keys in data integration tasks and reinforce the importance of using more robust identifiers. While Levenshtein distance remains a preferable option over Hamming distance, its effectiveness is diminished when dealing with less distinctive name sets. This underscores the critical need for comprehensive approaches in managing person identifiers, especially in contexts where data integrity is essential [55, 57, 59].

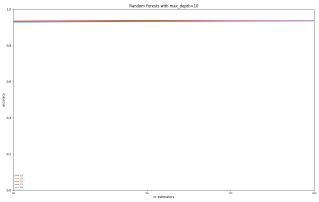


Figure 7.5: Accuracy results using Hamming distance for shorter names. The model's performance decreased significantly due to identical shorter names.

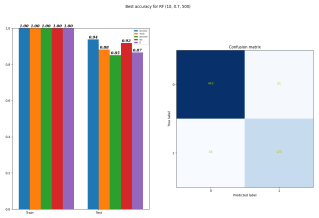


Figure 7.6: Confusion matrix using Hamming distance with shorter names, showing increased misclassifications.

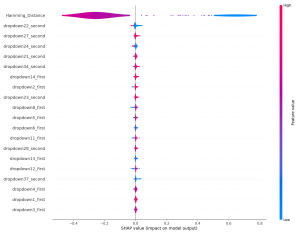


Figure 7.7: SHAP values for the Hamming distance model, highlighting its challenges in classifying short, similar names.

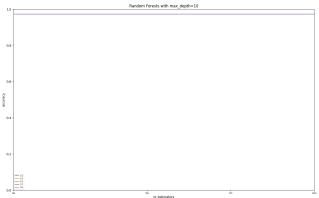


Figure 7.8: Accuracy results using Levenshtein distance for shorter names, showing better performance than Hamming but still facing challenges.

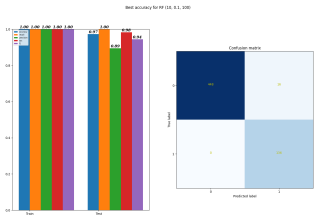


Figure 7.9: Confusion matrix using Levenshtein distance with shorter names, showing improved accuracy compared to Hamming distance.

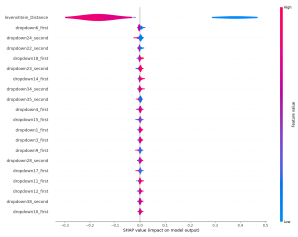


Figure 7.10: SHAP values for the Levenshtein distance model, showing its importance in distinguishing short names.

7.4 Encrypted identifiers and text data

In this experiment, we aimed to evaluate the model's performance using six variables, including two unknown identifiers (IDs) and four text data columns. This setup simulates a real-world scenario where each participant is required to write their name as an identifier and provide two texts describing themselves. To train a Random Forest model under these conditions, we encoded the data by calculating the

Hamming distances between all possible combinations of columns. This approach was chosen because the specific columns representing the IDs were unknown, making it necessary to consider all column pairs.

Despite the inherent challenges, the model demonstrated the ability to find matches with reasonable accuracy, as seen in the results. This scenario reflects real-life situations where certain information in databases, such as personal identifiers, is encrypted along with associated data. However, users may not know which column specifically corresponds to the encrypted identifier. In our simulation, the individual's name was considered as the key, representing a common case in database matching tasks. The model was still able to identify most of the corrected matched cases even with the ID being unknown. Figures 7.11, 7.12, 7.13, and 7.14 illustrate the model's performance under these conditions. Figure 7.11 shows the accuracy achieved by the model, while Figure 7.12 presents the confusion matrix, highlighting the model's ability to correctly identify matches even with encrypted identifiers. Figure 7.13 displays the SHAP values indicating feature importance, and Figure 7.14 demonstrates the importance of each variable in the Random Forest model.

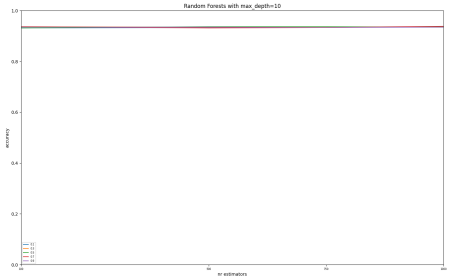


Figure 7.11: Accuracy results of the Random Forest model using Hamming distances between all column combinations.

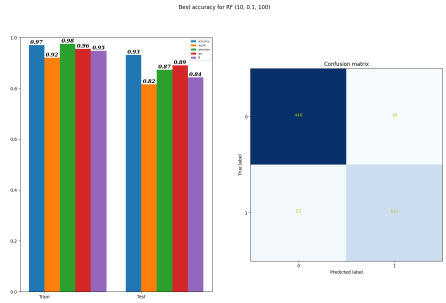


Figure 7.12: Confusion matrix showing the model's classification performance with encrypted IDs and text data.

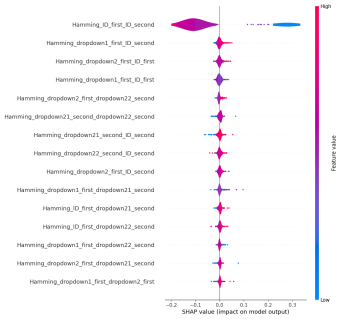


Figure 7.13: SHAP values indicating the impact of each feature on the model's predictions.

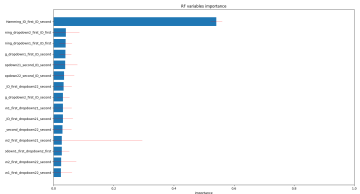


Figure 7.14: Variable importance plot highlighting the significance of each column in the Random Forest model.

7.5 Results with Levenshtein Distance and Conclusion

When the same experiment was conducted using Levenshtein distance, the results significantly improved (again). Levenshtein distance, which accounts for insertions, deletions, and substitutions, provided a more comprehensive measure of similarity between strings, enhancing the model's ability to identify correct matches even when identifiers were encrypted or varied in structure.

Figures 7.15, 7.16, and 7.17 illustrate the superior performance of Levenshtein distance in this context. The Random Forest model achieved higher accuracy, fewer misclassifications, and demonstrated the critical role of Levenshtein distance features in distinguishing between matching and non-matching records.

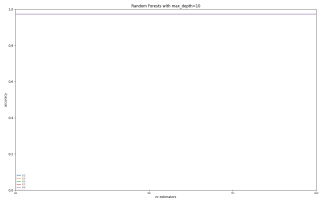


Figure 7.15: Accuracy results using Levenshtein distance, showing a marked improvement in model performance.

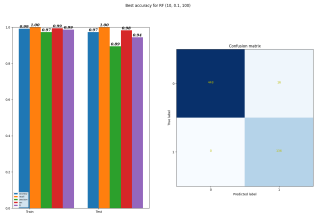


Figure 7.16: Confusion matrix for the Levenshtein distance, with significantly fewer misclassifications.

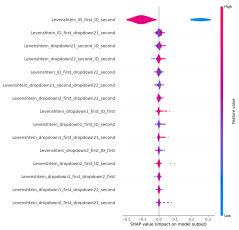


Figure 7.17: SHAP values showing the high impact of Levenshtein distance on model predictions.

The comparison shows that Levenshtein distance is a more effective metric for matching encrypted or anonymized data across databases, especially when the structure of identifiers is not predetermined. The enhanced ability to handle varied and complex edits makes Levenshtein distance a preferred choice over Hamming distance, providing a more reliable solution for data integration tasks involving person identifiers.

8

Journey and Self-Reflection

In a world overwhelmed by data, where digital footprints are constantly expanding and exposure seems inevitable, the laws and ethical frameworks governing privacy often struggle to keep pace with technology's rapid evolution. Amidst this complex digital landscape, I chose to focus my research on a pressing issue that remains largely unexplored: *"How to Disappear Online: Approaches for Digital Erasure and Evaluation Methodologies."*. This topic delves into the challenges of maintaining privacy in an age where data is the new currency, and for many, the ability to remain unseen is becoming increasingly difficult.

My fascination with this subject stems from the surge of data as the new currency of our world. Data has become a lucrative asset, but what about those who simply wish to remain under the radar? Not everyone aspires to live in the public eye, and there ought to be methodologies to help track and maintain digital anonymity.

The journey of this dissertation began with a phone call from Seoul, where I was studying, to Lisbon. I conveyed my interest in this field to my supervisor, acknowledging the challenges ahead—there were no guidelines, no established paths, just a passionate conviction that this was the right course of action. With a year to dedicate, I resolved to pursue something meaningful, something that would not only contribute to the academic world but also hold profound relevance for my career.

The investigation began in earnest. I scoured every available resource, reached out to experts in the field, and scheduled meetings with those who could potentially guide my work. While the enthusiasm for

my project was evident, I quickly realized that no one had yet contributed anything substantial to the topic of my thesis. The so-called “applications to disappear online” were merely tools to unsubscribe users from platforms, driven by commercial interests rather than ethical considerations or genuine efforts to explore methods of digital erasure.

My research journey led me to meticulously study every existing regulation from the inception of the Swedish Data Act to more contemporary laws like GDPR, CCPA, and the AIA. I sought to understand the foundational rules and principles governing digital data, documenting their evolution and differences with unprecedented detail. This historical perspective was crucial, as comprehending the nuances of these regulations was essential to advancing a study in this uncharted territory.

Despite diving into existing literature, I found that the work currently available was only tangentially related to my goals. The focus was consistently misaligned, none directly addressing the core purpose of my investigation. It was at this juncture that my co-supervisor, a professor with experience in intrusion recovery, joined the journey. At that time, I was engrossed in reading papers discussing whether data recovery from partial remnants meant the data was never truly deleted—a critical intersection that revealed a deeper connection between deletion and recovery.

This revelation crystallized the direction of my work. Drawing parallels between malicious intrusions and data deletion, I saw a commonality: the first step in erasure is identifying the data’s location and understanding how it can be found. Just as one would attempt to remove evidence of an intrusion, the challenge of disappearing online involves pinpointing where one’s digital footprint exists. This became the foundation of my approach to digital erasure—mapping data traces to devise effective strategies for ethical and complete deletion.

The real challenge emerged early on: a glaring absence of existing data relevant to my research. The lack of publicly available datasets on digital erasure was not just a hurdle; it was a stark indicator of the deficiency in this field. With no real-world data to anchor my experiments, I was compelled to generate synthetic data, which became a significant limitation of my project. While I meticulously designed various scenarios and approaches to mirror real-life applications as closely as possible, the nature of generated data inevitably introduces biases and deviations from authentic digital environments. This disconnect underscores a critical flaw—not just in my work but in the broader research landscape, where the scarcity of accessible data stymies meaningful progress.

In my dissertation, I documented the generation of synthetic data that mimicked personal information and focused initially on the simplest case: tracking data locally. This seemingly straightforward task quickly revealed unexpected complexities. I constructed a form to populate a database, capturing HTTP request logs and extracting data from the network. Locally, the data from these logs and the MySQL database were strikingly similar, allowing me to achieve near-perfect results with 100 per cent accuracy, recall, and other performance metrics using appropriate string-matching models.

However, this initial success felt hollow. It became apparent that the first approach, as outlined in my dissertation proposal, was too trivial; it failed to capture the complexity of real-world data environments. To push the boundaries, I began testing the limits of these perfect results. My findings demonstrated that as long as the data had more than eight variables—comprising text, numeric, and dropdown inputs—the results maintained the necessary precision (in my form). Precision in this context is crucial: in digital erasure, inaccurate deletions are unacceptable, and any remaining undeleted data nullifies the entire purpose. Yet, as the variables expanded under eight (which would be indeed less variables to be able to match, and consequentelly we would different people with exactly the same caracteristics and names which would lower our results), the integrity of the results began to falter, revealing weaknesses.

I further explored the impact of variable types on data matching accuracy, investigating tables dominated by dropdown variables, which have limited value ranges, against those with numeric variables, where values vary widely (e.g., height or age). These experiments highlighted that variables with higher ranges and diversity performed better, revealing that variable type significantly influences data matching and identification processes, because it would be harder to match different people if the data can be more diverse.

To better simulate real-world conditions where data traverses multiple systems globally, I introduced the challenge of identifying relationships across different databases. This scenario replicated an environment where encrypted data is distributed among various databases without clear indicators of the linking features. The task was to train models to detect correlations between tables containing encrypted, shared data, where the identifying key was unknown. This investigation exposed significant difficulties: matching results were notably poorer compared to scenarios with known IDs, emphasizing the need for mantaining a known id across databases.

Next, I examined scenarios where names served as the primary identifier—a common yet flawed approach in data systems. I tested this method across databases, varying between longer, distinguishable names and shorter, more common ones. Longer names, due to their uniqueness, allowed for near one-to-one correspondence between records. However, with shorter, more frequent names like ”John Doe,” ambiguity arose, leading to multiple matches and uncertainties about which records to delete. This highlighted the limitations of name-based matching, stressing the importance of advanced string-matching algorithms like Levenshtein distance, which performed best in these conditions.

I then pushed further by eliminating the assumption of knowing the ID features in each database. In this more complex scenario, I deployed string matching to infer which columns corresponded across datasets (to try in find which ones would have the id encrypted), testing the model's adaptability when the key identifier was unknown. As expected, the results declined significantly compared to controlled conditions, revealing a critical gap in current methodologies when dealing with real-world complexities of data linkage and erasure.

This research faced numerous challenges and was a pioneering step into largely uncharted territory. My work shines a light on the urgent need for deeper exploration and development in this field. The limitations, particularly the reliance on synthetic data, underscore the necessity of access to real-world datasets. Such data would make findings far more robust and reflective of actual scenarios. The discrepancies between generated and real data illustrate that while this study lays an important foundation, it also highlights the broader inadequacies in the current state of digital erasure research. Without genuine datasets, there will always be a degree of separation from reality—a gap that future research must strive to bridge.

9

Conclusion and Future Work

Contents

The motivation behind this thesis comes from the increasing need for organizations to comply with data privacy regulations, particularly the General Data Protection Regulation (GDPR) [4]. As organizations collect data across various distributed systems, it becomes critical to trace where unauthorized data is located, especially in cases of breaches or unauthorized modifications. This work addresses the challenge of improving data tracking and deletion processes in compliance with GDPR. One key step toward achieving this goal is the matching of logs from different systems (such as HTTP requests and database logs). Through the proposed machine learning techniques, we aimed to develop methods that enhance this matching process, allowing for more accurate identification of unauthorized data and thus enabling timely and effective responses to data breaches or privacy violations [54, 57].

9.1 General Description of the Approach

This thesis proposes a method that focuses on correlating logs from HTTP requests with database logs, aiming to detect unauthorized data actions and facilitate its potential removal. The proposed solution, which adapts the SANARE framework, emphasizes the data matching phase as an essential first step toward identifying where unauthorized data exists, before considering further actions such as deletion. This approach is highly relevant in the context of GDPR, where organizations must quickly locate and mitigate data breaches [60]. By linking logs from different data sources, we can more effectively trace data origins and identify security breaches. This thesis specifically focuses on the initial task of matching logs from different systems as a precursor to later data deletion or containment measures.

9.2 Description of the Simulations

To test the proposed models, synthetic datasets were generated, representing realistic user actions and submissions across multiple systems [54]. These datasets included HTTP request logs and database entries, simulating real-world interactions where users provide information that is stored in different databases. The experiments involved varying the number of input features (from 4 to 42 variables) to evaluate the model's ability to recognize patterns under different conditions. The focus was on understanding the performance of Decision Trees [61] and Random Forests [40] when handling both well-aligned and noisy datasets, providing insights into how these models behave in scenarios where data is incomplete or contains irrelevant information.

9.3 Summary of Results

The experiments yielded several important findings regarding the performance of the models under different conditions:

1. **If IDs are the same:** When the values between the HTTP requests and database logs were perfectly aligned, the models—both Decision Tree and Random Forest—performed with exceptional accuracy. This confirms that when the data identifiers are consistent across different sources, the models are highly effective in matching the logs.

2. **If IDs are different:** When dealing with misaligned data from different databases, the models' performance suffered. The ID key (key shared between both databases) remained the most important feature in both models. In this scenario, Random Forest demonstrated a clear advantage due to its ability to handle variability better than a single Decision Tree, which can fall more easily to overfitting. The ensemble learning approach of Random Forest, combined with its feature

randomization, allowed it to perform better in cases where the IDs did not perfectly match, as it could rely on multiple trees to generate a consensus decision, mitigating over-reliance on specific ID values. [40]

3. **DB x DB with textual identifiers:** In scenarios where textual identifiers were used across databases, it became crucial to apply effective string-matching techniques. The Levenshtein distance proved to be particularly advantageous over the Hamming distance for handling variations in textual data, such as person names. Levenshtein distance, which accounts for insertions, deletions, and substitutions, allowed for a more flexible and accurate comparison of strings with minor differences, like abbreviations, missing letters, or additional characters. [56] This flexibility was critical when matching similar but not identical identifiers across databases, ensuring greater accuracy in identifying correct matches. The Hamming distance, limited to fixed-length string comparisons, failed to accommodate such variations, making Levenshtein the superior choice for scenarios with inconsistent or partially aligned textual identifiers.

9.4 Future Work

Building on the findings of this thesis, there are several important directions for future research:

- **Real-world data:** Future experiments should incorporate real-world data to provide more comprehensive and practical insights. While synthetic data allowed for controlled experiments, testing the models on real-world datasets would reveal their behavior under true operational conditions, where unpredictability, noise, and incomplete data are more prevalent.

- **Exploration of other classifiers:** Additional machine learning models, such as Gradient Boosting [46] and like Multilayer Perceptrons (MLP) [62], should be explored. Gradient Boosting offers high sensitivity to outliers, making it suitable for detecting rare, unauthorized data changes, but it is also prone to overfitting if not carefully managed. On the other hand, MLP models, which handle continuous data well, might struggle with discrete data like dropdown options, as used in this study. However, both Gradient Boosting and MLP remain viable alternatives for future research, potentially offering improvements in accuracy and generalization over the models tested in this thesis.

This thesis laid the groundwork for understanding how machine learning models can be applied to the task of log matching across different systems. While the proposed methods performed well in controlled simulations, further work is needed to refine these models for real-world applications. Future efforts should focus on expanding the dataset to include real-world information and exploring additional

classifiers that may offer improvements. This will enhance the models' effectiveness in real-world scenarios, ensuring better log matching and bringing us one step closer to full compliance with data privacy regulations.

Bibliography

[1] C. Gates and P. Matthews, "Data is the new currency," in *Proceedings of the 2014 workshop on New Security Paradigms Workshop, Victoria, BC, Canada, September 15-18, 2014*, K. Beznosov, A. Somayaji, T. Longstaff, and P. C. van Oorschot, Eds. ACM, 2014, pp. 105–116. [Online]. Available: https://doi.org/10.1145/2683467.2683477

[2] M. Alharbi and I. Yahya, "Challenges and issues of gdpr compliance for developing countries," *Journal of Information Security*, vol. 12, no. 1, pp. 1–12, 2021.

[3] S. Rosenbaum, C. Müller-Bloch, and H.-T. Wagner, "Gdpr compliance implementation issues and solutions: The perspective of german smes," *Journal of Business Research*, vol. 122, pp. 903–914, 2020.

[4] European Commission, "General data protection regulation," https://commission.europa.eu/law/law-topic/data-protection/data-protection-eu_en, 2018, accessed: 2024-01-10.

[5] E. Commission, "Artificial intelligence act," European Commission proposal for a regulatory framework, 2021, proposal for a regulation of the European Parliament and of the Council laying down harmonised rules on artificial intelligence (Artificial Intelligence Act).

[6] California Office of the Attorney General, "California consumer privacy act (ccpa)," State of California - Department of Justice - Office of the Attorney General, 2020, accessed: 2024-01-10. [Online]. Available: https://oag.ca.gov/privacy/ccpa

[7] P. Oyakhare, "Complying with gdpr: The difficulties american big techs face," https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4275535, 2022.

[8] M. Domb, S. Joshi, and P. Roshan, "Risk mitigation model for data loss: A case study approach," *Journal of Advanced Research in Dynamical and Control Systems*, vol. 11, pp. 440–447, 09 2019.

[9] Pew Research Center, "The state of privacy in post-snowden america," https://www.pewresearch.org/short-reads/2016/09/21/the-state-of-privacy-in-america/, 2016, accessed: 2024-01-10.

[10] H. Habib, Y. Zou, A. Jannu, N. Sridhar, C. Swoopes, A. Acquisti, L. F. Cranor, N. M. Sadeh, and F. Schaub, "An empirical analysis of data deletion and opt-out choices on 150 websites," in *Fifteenth Symposium on Usable Privacy and Security, SOUPS 2019, Santa Clara, CA, USA, August 11-13, 2019*, H. R. Lipford, Ed. USENIX Association, 2019. [Online]. Available: https://www.usenix.org/conference/soups2019/presentation/habib

[11] B. A. Gordon, "Global data privacy laws: Forty years of acceleration," *SSRN Electronic Journal*, 2011. [Online]. Available: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2205064

[12] J. Freese, "The swedish data act," https://www.ojp.gov/pdffiles1/Digitization/49670NCJRS.pdf, November 1977, accessed: 2024-01-10.

[13] Privacy Protection Study Commission, *Personal Privacy in an Information Society*, July 1977.

[14] "Health insurance portability and accountability act of 1996," U.S. Department of Health & Human Services, 1996, accessed: 2024-01-10. [Online]. Available: https://www.hhs.gov/hipaa/index.html

[15] "Children's online privacy protection act of 1998," Federal Trade Commission, 1998, accessed: 2024-01-10. [Online]. Available: https://www.ftc.gov/enforcement/rules/rulemaking-regulatory-reform-proceedings/childrens-online-privacy-protection-rule

[16] "Sections 1798.100 to 1798.105 of the california consumer privacy act (ccpa)," https://oag.ca.gov/privacy/ccpa, 2018.

[17] "Section 1798.120 of the california consumer privacy act (ccpa)," https://oag.ca.gov/privacy/ccpa, 2018.

[18] "Directive 95/46/ec of the european parliament and of the council of 24 october 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data," https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:31995L0046, 1995.

[19] E. Commission. (2024) Article 17 gdpr – right to erasure ('right to be forgotten') - general data protection regulation (gdpr). Accessed: 2024-01-10. [Online]. Available: https://gdpr-info.eu/art-17-gdpr/

[20] G. V. Research, "Server market size, share & growth analysis report, 2030," Retrieved from https://www.grandviewresearch.com/industry-analysis/server-market, 2023, accessed: 2024-01-10.

[21] K. Adnan and R. Akbar, "An analytical study of information extraction from unstructured and multidimensional big data," *Journal of Big Data*, vol. 6, no. 91, 2019.

[22] A. Ginart, M. Y. Guan, G. Valiant, and J. Zou, "Making AI forget you: Data deletion in machine learning," in *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, and R. Garnett, Eds., 2019, pp. 3513–3526. [Online]. Available: https://proceedings.neurips.cc/paper/2019/hash/cb79f8fa58b91d3af6c9c991f63962d3-Abstract.html

[23] S. P. Lloyd, "Least squares quantization in PCM," *IEEE Trans. Inf. Theory*, vol. 28, no. 2, pp. 129–136, 1982. [Online]. Available: https://doi.org/10.1109/TIT.1982.1056489

[24] D. Arthur and S. Vassilvitskii, "k-means++: the advantages of careful seeding," in *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2007, New Orleans, Louisiana, USA, January 7-9, 2007*, N. Bansal, K. Pruhs, and C. Stein, Eds. SIAM, 2007, pp. 1027–1035. [Online]. Available: http://dl.acm.org/citation.cfm?id=1283383.1283494

[25] N. Kumari, B. Zhang, S. Wang, E. Shechtman, R. Zhang, and J. Zhu, "Ablating concepts in text-to-image diffusion models," *CoRR*, vol. abs/2303.13516, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2303.13516

[26] K. Busch, "Generafive arfificial intelligence and data privacy: A primer (no. r47569; p. 8). congressional research service," 2023, accessed: 2024-01-11. [Online]. Available: https://crsreports.congress.gov/product/pdf/R/R47569

[27] L. Bourtoule, V. Chandrasekaran, C. A. Choquette-Choo, H. Jia, A. Travers, B. Zhang, D. Lie, and N. Papernot, "Machine unlearning," in *42nd IEEE Symposium on Security and Privacy, SP 2021, San Francisco, CA, USA, 24-27 May 2021*. IEEE, 2021, pp. 141–159. [Online]. Available: https://doi.org/10.1109/SP40001.2021.00019

[28] A. Murillo, A. Kramm, S. Schnorf, and A. De Luca, "If i press delete it's gone: User understanding of online data deletion and expiration," in *Proceedings of the Fourteenth Symposium on Usable Privacy and Security*. Baltimore, MD, USA: USENIX Association, August 2018. [Online]. Available: https://www.usenix.org/conference/soups2018/presentation/murillo

[29] J. Reardon, D. A. Basin, and S. Capkun, "Sok: Secure data deletion," in *2013 IEEE Symposium on Security and Privacy, SP 2013, Berkeley, CA, USA, May 19-22, 2013*. IEEE Computer Society, 2013, pp. 301–315. [Online]. Available: https://doi.org/10.1109/SP.2013.28

[30] R. Perlman, "The ephemerizer: Making data disappear," *ACM Transactions on Information and System Security - TISSEC*, vol. 1, 03 2005.

[31] R. Geambasu, T. Kohno, A. A. Levy, and H. M. Levy, "Vanish: Increasing data privacy with self-destructing data," in *18th USENIX Security Symposium, Montreal, Canada, August 10-14, 2009, Proceedings*, F. Monrose, Ed. USENIX Association, 2009, pp. 299–316. [Online]. Available: http://www.usenix.org/events/sec09/tech/full_papers/geambasu.pdf

[32] D. R. Matos, M. L. Pardal, and M. Correia, "Sanare: Pluggable intrusion recovery for web applications," *IEEE Trans. Dependable Secur. Comput.*, vol. 20, no. 1, pp. 590–605, 2023. [Online]. Available: https://doi.org/10.1109/TDSC.2021.3139472

[33] P. Ammann, S. Jajodia, and P. Liu, "Recovery from malicious transactions," *IEEE Trans. Knowl. Data Eng.*, vol. 14, no. 5, pp. 1167–1185, 2002. [Online]. Available: https://doi.org/10.1109/TKDE.2002.1033782

[34] D. R. Matos, M. L. Pardal, and M. Correia, "Rectify: black-box intrusion recovery in paas clouds," in *Proceedings of the 18th ACM/IFIP/USENIX Middleware Conference, Las Vegas, NV, USA, December 11 - 15, 2017*, K. R. Jayaram, A. Gandhi, B. Kemme, and P. R. Pietzuch, Eds. ACM, 2017, pp. 209–221. [Online]. Available: https://doi.org/10.1145/3135974.3135978

[35] R. S. P. S. Priya, "Real-time multi fractal trust evaluation model for efficient intrusion detection in cloud," *Intelligent Automation & Soft Computing*, vol. 37, no. 2, pp. 1895–1907, 2023. [Online]. Available: http://www.techscience.com/iasc/v37n2/53254

[36] N. Gruschka, V. Mavroeidis, K. Vishi, and M. Jensen, "Privacy issues and data protection in big data: A case study analysis under GDPR," in *IEEE International Conference on Big Data (IEEE BigData 2018), Seattle, WA, USA, December 10-13, 2018*, N. Abe, H. Liu, C. Pu, X. Hu, N. K. Ahmed, M. Qiao, Y. Song, D. Kossmann, B. Liu, K. Lee, J. Tang, J. He, and J. S. Saltz, Eds. IEEE, 2018, pp. 5027–5033. [Online]. Available: https://doi.org/10.1109/BigData.2018.8622621

[37] S. González, S. García, J. Del Ser, L. Rokach, and F. Herrera, "A practical tutorial on bagging and boosting based ensembles for machine learning: algorithms, software tools, performance study, practical perspectives and opportunities," *Information Fusion*, vol. 64, pp. 205–237, 2020.

[38] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[39] T. G. Dietterich, "Ensemble methods in machine learning," in *Multiple Classifier Systems, First International Workshop, MCS 2000, Cagliari, Italy, June 21-23, 2000, Proceedings*, ser. Lecture Notes in Computer Science, J. Kittler and F. Roli, Eds., vol. 1857. Springer, 2000, pp. 1–15. [Online]. Available: https://doi.org/10.1007/3-540-45014-9_1

[40] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[41] T. K. Ho, “Random decision forests,” in *Proceedings of 3rd international conference on document analysis and recognition*, vol. 1. IEEE, 1995, pp. 278–282.

[42] Y. Amit and D. Geman, “Shape quantization and recognition with randomized trees,” in *Advances in neural information processing systems*, 1997, pp. 130–136.

[43] T. K. Ho, “The random subspace method for constructing decision forests,” in *IEEE transactions on pattern analysis and machine intelligence*, vol. 20, no. 8. IEEE, 1998, pp. 832–844.

[44] L. Breiman, “Bagging predictors,” *Machine learning*, vol. 24, no. 2, pp. 123–140, 1996.

[45] R. D. Cutler, T. C. Edwards Jr, K. H. Beard, A. Cutler, J. Hess, D. J. Gibson, and J. J. Lawler, “Random forests for classification in ecology,” *Ecology*, vol. 88, no. 11, pp. 2783–2792, 2007.

[46] J. H. Friedman, “Greedy function approximation: a gradient boosting machine,” *Annals of statistics*, pp. 1189–1232, 2001.

[47] ——, “Stochastic gradient boosting,” *Computational statistics & data analysis*, vol. 38, no. 4, pp. 367–378, 2002.

[48] K. Hjerppe, J. Ruohonen, and V. Leppänen, “The general data protection regulation: Requirements, architectures, and constraints,” in *2019 IEEE 27th International Requirements Engineering Conference (RE)*, 2019, pp. 265–275.

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

[50] M. Sokolova and G. Lapalme, “A systematic analysis of performance measures for classification tasks,” *Information processing & management*, vol. 45, no. 4, pp. 427–437, 2009.

[51] J. Bergstra and Y. Bengio, “Random search for hyper-parameter optimization,” *Journal of Machine Learning Research*, vol. 13, no. Feb, pp. 281–305, 2012.

[52] M. Hossin and M. Sulaiman, “A review on evaluation metrics for data classification problems,” *International Journal of Data Mining & Knowledge Management Process*, vol. 5, no. 2, pp. 1–11, 2015.

[53] S. M. Lundberg and S.-I. Lee, “A unified approach to interpreting model predictions,” *Advances in Neural Information Processing Systems*, vol. 30, pp. 4765–4774, 2017.

[54] E. Rahm and H.-H. Do, “Data cleaning: Problems and current approaches,” *IEEE Data Engineering Bulletin*, vol. 23, no. 4, pp. 3–13, 2000.

[55] L. Yujian and L. Bo, “A normalized levenshtein distance metric,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 6, pp. 1091–1095, 2007.

[56] V. I. Levenshtein, “Binary codes capable of correcting deletions, insertions, and reversals,” *Soviet Physics Doklady*, vol. 10, no. 8, pp. 707–710, 1966.

[57] G. Navarro, “A guided tour to approximate string matching,” *ACM Computing Surveys*, vol. 33, no. 1, pp. 31–88, 2001.

[58] A. Cuzzocrea, L. Bellatreche, and I.-Y. Song, “Data warehousing and olap over big data: current challenges and future research directions,” in *Proceedings of the 16th International Workshop on Data Warehousing and OLAP*. ACM, 2013, pp. 67–70.

[59] W. W. Cohen, P. Ravikumar, and S. E. Fienberg, “A comparison of string distance metrics for name-matching tasks,” in *Proceedings of the IJCAI-03 Workshop on Information Integration on the Web (IIWeb-03)*, 2003, pp. 73–78.

[60] Q. Liu, Z. Xiao, and C. Wang, “Privacy-preserving data sharing in cloud computing,” *Security and Communication Networks*, vol. 2017, pp. 1–9, 2017.

[61] L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen, *Classification and regression trees*. CRC press, 1984.

[62] F. Rosenblatt, *Principles of neurodynamics. perceptrons and the theory of brain mechanisms*. Spartan Books, 1961.