Predicting the Impact of IoT Data Gathering on User’s Privacy Preferences

Ghassen Kilani

Follow this and additional works at: https://repository.fit.edu/etd

Part of the Computer Sciences Commons
Predicting the Impact of IoT Data Gathering on User’s Privacy Preferences

by

Ghassen Kilani

Master of Science
Information Assurance And Cybersecurity
Florida Institute Of Technology
2016

Bachelor of Science
Electrical Engineering
Florida Institute Of Technology
2014

Bachelor of Science
Computer Science
Florida Institute Of Technology
2013

A dissertation
submitted to the College of Engineering and Science
at Florida Institute of Technology
in partial fulfillment of the requirements
for the degree of

Doctor of Philosophy
in
Computer Science

Melbourne, Florida
July, 2022
We the undersigned committee hereby approve the attached dissertation

Predicting the Impact of IoT Data Gathering on User’s Privacy Preferences by
Ghassen Kilani

William H. Allen, Ph.D.
Associate Professor
Computer Engineering and Sciences
Major Advisor

Siddhartha Bhattacharyya, Ph.D.
Associate Professor
Computer Engineering and Sciences

Deborah S. Carstens, Ph.D.
Professor
College of Aeronautics

William Shoaff, Ph.D.
Associate Professor
Computer Engineering and Sciences

Khaled Slhoub, Ph.D.
Assistant Professor
Computer Engineering and Sciences

Philip J. Bernhard, Ph.D.
Associate Professor and Department Head
Computer Engineering and Sciences
Abstract

Title:
Predicting the Impact of IoT Data Gathering on User’s Privacy Preferences

Author:
Ghassen Kilani

Major Advisor:
William H. Allen, Ph.D.

The proliferation of Internet of Things (IoT) devices has increased data sharing, profiling, and manipulation on various networks. The rapid growth of information disclosure has caused system users to lose motivation to enhance their data privacy. The repeated breaches on different networks worldwide have made people feel discouraged, as they perceive privacy schemes as futile. IoT systems introduce another dimension of privacy leakage due to their expendability nature and information collection features. The situation worsens when users have to manage multiple IoT devices, each following different security protocols, leading to poor decision-making and privacy leakage. This tremendous flow of unsecured data gathered about users without their knowledge or consent, combined with complicated privacy policies, leads to information overload and privacy fatigue. Users’ continuously changing privacy behavior further complicates the problem as they opt to disregard privacy for better service quality or monetary gain. Many users have become tired of implementing security controls due to privacy intrusiveness and a lack of knowledge. Additionally, the concept of
privacy differs between individuals, leading to a variety of expectations that are not yet implemented in IoT environments. To protect users from privacy invasions in a climate where sensors are omnipresent, a shift toward context-centric privacy and adaptive preference approach is required. In this dissertation, we stress for a tailored privacy preference experience unique to each individual that aims at offering an automated contextual privacy preferences recommendation based on user experience. We first conduct a replication study to understand the sensitive information gathered through IoT sensors. This study allowed us to discover what factor combinations impact users’ privacy decisions in different IoT environments. The replication study also helped us classify sensors and locations depending on how sensitive the data collected is. Next, we used the collected data and feedback to experiment with how Machine Learning (ML) algorithms behave using different techniques and features from the original study. This step allowed us to study how well different features can predict an individual’s decision to allow or deny entry to a specific IoT location. The experiment allowed us to measure how feasible it is to build a Machine Learning approach capable of predicting users’ preferences while mimicking real-world scenarios and eliminating factors that users do not have control over. We then introduced PPM (Privacy Preference Manager), a recommender system that uses a simple yet powerful approach to predict user privacy preferences. PPM is a ML approach built on all the feedback collected from our previous experiments, uses a minimalistic feature collection, and shifts from a binary classification to a privacy risk recommendation approach. We finally designed IoTPP (Internet of Things Privacy Preference), a web application that enforces user privacy preferences and gives them control over their environment. IoTPP was explicitly created to help users manage their privacy and guide them when interacting with IoT spaces. IoTPP aims to provide users with a tailored privacy experience based on their privacy expectations. We also examined IoTPP’s ability to mitigate the privacy fatigue phenomenon and analyzed its usability as a tool in enforcing privacy management.
# Table of Contents

Abstract ................................. iii

List of Figures ............................ ix

Acknowledgments .......................... xiii

1 Introduction ............................ 1
   1.1 IoT Privacy ........................... 2
      1.1.1 Privacy Paradox and User Awareness 4
      1.1.2 Smart Devices and Sensing Technologies in IoT 5
      1.1.3 Privacy Preferences in IoT 7
      1.1.4 Security and Privacy Fatigue 8
   1.2 Research Problem .................... 10
      1.2.1 Motivation ...................... 10
      1.2.2 Research Question ............... 11
      1.2.3 Proposed Approach ............... 12
   1.3 Research Description ................. 15

2 Internet of Things ..................... 18
   2.1 The concept of Internet of Things (IoT) 18
      2.1.1 Brief History of IoT ............ 18
      2.1.2 What is IoT? ................... 19
2.2 IoT Elements ......................................................... 21
  2.2.1 Identification, Sensing, and Communication Technologies .......... 22
  2.2.2 Computation ................................................. 24
  2.2.3 Services ..................................................... 24
  2.2.4 IoT Semantics .............................................. 25
2.3 IoT Architecture .................................................. 26
  2.3.1 Possible Architectures ..................................... 26
  2.3.2 The Service Oriented Architecture .......................... 27
2.4 IoT Applications .................................................. 28
  2.4.1 Security and Surveillance .................................. 29
  2.4.2 Social IoT .................................................... 30
  2.4.3 Wearable Devices ......................................... 31
2.5 IoT Security ........................................................ 33
  2.5.1 Confidentiality in IoT ....................................... 34
  2.5.2 Integrity in IoT ............................................. 35
  2.5.3 Availability in IoT ......................................... 35
  2.5.4 Identification and Authentication in IoT ....................... 35
  2.5.5 Data Privacy in IoT ....................................... 36
  2.5.6 Trust in IoT ................................................ 38
2.6 IoT Security Issues ............................................... 38
  2.6.1 Conventional vs. IoT Security ............................... 38
  2.6.2 IoT Networking Challenges ................................ 39
    2.6.2.1 Network Security and Protocol Issues .................. 40
    2.6.2.2 Identity Management Issues ............................ 40
2.7 Attacks on IoT .................................................... 41
  2.7.1 Perception Layer Attacks .................................. 41
6  A New Survey of User’s Privacy Preferences .......................... 102
   6.1  PPM Survey ..................................................... 102
       6.1.1  Creating The Survey ..................................... 103
       6.1.2  Recruiting Participants ................................. 105
   6.2  Data Cleaning, Coding and Engineering ......................... 106
   6.3  Machine Learning Approach .................................. 111
       6.3.1  Design ..................................................... 111
       6.3.2  Results ................................................... 112
       6.3.3  Discussion ................................................ 117

7  IoTPP Web Application ................................................ 119
   7.1  Web App: Overview ............................................. 119
   7.2  Design and Methodology ........................................ 120
   7.3  Evaluation ...................................................... 125
       7.3.1  NASA TLX ................................................ 125
       7.3.2  System Usability Scale ................................. 125
   7.4  Observations, Limitations, and Future Work .................... 128

8  Conclusion ............................................................ 131

References .............................................................. 135
# List of Figures

1.1 Privacy by Design Principles ........................................ 3
1.2 Privacy Paradox .................................................... 5
1.3 Privacy Fatigue ..................................................... 10
1.4 Privacy Preference Overview ........................................ 14

2.1 Evolution of Internet of Things - Projected transition beyond the year 2020[155] ................................................................. 19
2.2 Internet of Things paradigm - initiated with RFID technology and used in different industries [41]. .............................................. 21
2.3 The IoT building blocks and technologies adapted from [30] ................. 22
2.4 The IoT architecture. (a) Three-layer. (b) Middle-ware based. (c) SOA based. (d) Five-layer.[30] ............................................................. 26
2.5 SOA-based architecture for the IoT middleware. ............................. 28
2.6 IoT’s overall pure showing the vertical markets and the way they integrate horizontally [30] .............................................................. 29
2.7 Wearable devices Timeline .............................................. 32
2.8 Privacy by Design Principles ............................................. 37
2.9 Internet of Things main Security challenges adapted from [221]. .............. 44

3.1 Supervised Machine Learning Work Flow ................................ 56
3.2 Unsupervised Machine Learning Work Flow ................................ 57
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.3</td>
<td>Reinforcement Learning Work Flow</td>
<td>58</td>
</tr>
<tr>
<td>3.4</td>
<td>Support Vector Machine example</td>
<td>64</td>
</tr>
<tr>
<td>3.5</td>
<td>Logistic Regression example</td>
<td>65</td>
</tr>
<tr>
<td>3.6</td>
<td>Decision Tree example</td>
<td>66</td>
</tr>
<tr>
<td>3.7</td>
<td>Random Forest example</td>
<td>67</td>
</tr>
<tr>
<td>4.1</td>
<td>Factors used in our replication study, their levels, and detailed explanation</td>
<td>75</td>
</tr>
<tr>
<td>4.2</td>
<td>Example of Data categorization using Data Type as a factor</td>
<td>77</td>
</tr>
<tr>
<td>4.3</td>
<td>Relation between Data collection and users comfort level</td>
<td>80</td>
</tr>
<tr>
<td>4.4</td>
<td>GLMM Regression model for Comfort Level</td>
<td>83</td>
</tr>
<tr>
<td>4.5</td>
<td>GLMM BIC Summary for Comfort Level model</td>
<td>84</td>
</tr>
<tr>
<td>4.6</td>
<td>GLMM Regression model for Allow/Deny</td>
<td>87</td>
</tr>
<tr>
<td>4.7</td>
<td>GLMM Regression Summary of BIC values for the Allow/Deny model</td>
<td>88</td>
</tr>
<tr>
<td>5.1</td>
<td>Privacy Level Distribution</td>
<td>91</td>
</tr>
<tr>
<td>5.2</td>
<td>Feature levels</td>
<td>92</td>
</tr>
<tr>
<td>5.3</td>
<td>ML Models used.</td>
<td>93</td>
</tr>
<tr>
<td>5.4</td>
<td>Sample illustrating different factor level combinations in the Dataset</td>
<td>94</td>
</tr>
<tr>
<td>5.5</td>
<td>Data type distribution</td>
<td>95</td>
</tr>
<tr>
<td>5.6</td>
<td>K-Fold evaluation</td>
<td>96</td>
</tr>
<tr>
<td>5.7</td>
<td>K-Fold iterations</td>
<td>96</td>
</tr>
<tr>
<td>5.8</td>
<td>DT classification report</td>
<td>97</td>
</tr>
<tr>
<td>5.9</td>
<td>Training data results</td>
<td>98</td>
</tr>
<tr>
<td>5.10</td>
<td>Model Comparison</td>
<td>99</td>
</tr>
<tr>
<td>6.1</td>
<td>Factor Interactions</td>
<td>103</td>
</tr>
<tr>
<td>6.2</td>
<td>Privacy Levels distribution</td>
<td>108</td>
</tr>
<tr>
<td>6.3</td>
<td>Distribution Techniques</td>
<td>108</td>
</tr>
</tbody>
</table>
Acknowledgements

For my dad who made my dream a reality.
For my mom who always believed in me.
For my wife who stood by me through thick and thin.
For my kids who made my journey memorable.
For my advisor and mentor Dr. Allen who never stopped pushing me toward the right track.
Chapter 1

Introduction

The history of the Internet of Things (IoT) traced back to 1999 and was considered a revolution in humankind’s development [41]. This emerging technology facilitated the access to electronic services such as health care, assisted living and driving, security, and social interaction while creating new security issues and breaches never dealt with before. The fragile nature of the IoT security and its scalability makes it the perfect target for malicious activity and privacy leaks. IoT security has evolved over the recent years to enhance data privacy and integrity owing to its gradual evolution towards becoming a considerable driver in controlling and monitoring applications [41]. The billions of things in the network are powered using the user’s private information to provide personalized services and tailored experiences. However, users are generally unaware of the sensitive information being gathered about them nor understand what this private data will be used for outside of the network [76]. The heterogeneous nature of the IoT technology combined with unaware users, who are considered the weakest link in security, presents a huge technical challenge in the IoT world. Actually, when developing a new secure IoT approach, researchers have to consider the human factor along with different architectures, protocols, performance levels, and technical requirements in different dynamic scenarios [170]. All these requirements make accomplishing this task
preposterous as privacy and usability perspectives differ between individuals.

In this work, we aim to address IoT Privacy preferences through guiding users better manage and control their private information in different IoT environments. To this end, we designed IoTPP, a risk assessment tool that utilizes Machine Learning to enforce user privacy expectations in smart spaces. We start by understanding the sensitive information gathered through IoT sensors and analyze how users perceive contextual privacy. We applied privacy levels and factors that users control to Machine Learning to examine its impact on privacy decisions. Our analysis exhibits the need to shift from a binary classification to a privacy risk assessment approach. Thus we designed a recommender system that predicts users’ privacy needs while requiring minimal user interaction. We examine IoTPP workload in addressing the privacy fatigue phenomenon. We also focus on studying its usability capabilities in guiding users through their privacy management process.

1.1 IoT Privacy

Data privacy is a fundamental human right that ensures users full access and control over their data and safeguard against unauthorized data usage or manipulation [158]. Privacy also includes the right to stay anonymous as well as the right to control distraction and attention [140]. Ann Cavoukian, a former privacy commissioner, stressed that the core structure of IoT should implement privacy preferences and promote a user-centric approach through the whole design process [66]. This could be done through the Privacy by Design Framework (PbD), which is a significant pillar of the General Data Protection Regulation (GDPR) in Europe. In fact, privacy should be applied at the manufacturing phase to make IoT privacy transparent and protect the sensitive data collected through smart devices. The Federal State Commission also reported that PbD is the most promising approach to protect user’s privacy and make the IoT technology trustworthy [208].
It can be seen from Figure 1.1 that the PbD Framework is composed of seven principles that mainly promote privacy in a holistic way rather than in a technical way [66]. These principles mainly focus on embedding privacy as a default in the design phase and preserve data privacy through the entire communication process. This approach should be tightly followed in our research and design as it offers optimal privacy and security for the user who should be considered a valuable asset, not a liability.

The primary threat facing the evolution of IoT in various industries is its fragile design. Billions of IoT devices have already "invaded" our environment and embedded themselves in our daily lives without our consent. This new technology, not yet fully understood, is still in the development phase, making it insecure and unpredictable. Extensive research has been made to mitigate its security flaws, though its heterogenous nature and complex interactions make it difficult to overcome its imperfections [210]. In fact, no ideal solutions have been proposed to overcome IoT’s individual problems, which pushed the researchers to study closely its gaps [128].
1.1.1 Privacy Paradox and User Awareness

The exponential increase in information security management and complexities in its deployment has caused system users to lose motivation in enhancing data privacy. Besides, the repeated breaches on different networks worldwide have made people feel discouraged, as they perceive the security schemes as futile [75]. Some individuals’ behavior tends to change when using online services and opt to disregard privacy for better service quality or monetary gain [138]. In fact, IoT systems introduce another dimension of privacy leakage due to its expendability nature and information collection features. Many users have become tired of implementing security controls due to privacy intrusiveness and a lack of robust protection provided by security companies [75]. Therefore, user behavior and miseducation, perceived as the weak chain in privacy breaches, makes it easier for attackers to compromise their system, leading to data loss. This leads us to the privacy paradox concept illustrated in Figure 1.2, explaining the contrast between users’ privacy attitude and their actual behavior. Extensive research have been conducted to understand this phenomenon [60, 228, 26, 186] and all pointed out that users have developed a privacy-compromising tendency when using online services, resulting in a disparity between privacy-protection attitudes and their actual actions and behaviors. Users claim that their privacy should be safeguarded, and their private information kept confidential, yet their actual behavior does not support this belief. Studies have shown that users will usually consider the actual benefit without thinking about future risks. User attitude is closely tied to the perceived risk (negative outcome exceeds instant rewards) thus when the actual benefits outweigh future risks, their behavior is dramatically altered. IoT technology will magnify the privacy paradox phenomenon as sensors will silently collect critical user data to deliver quality services and provide optimal results. In an embedded environment, smart devices will autonomously learn user preferences and use ingenuity to offer the best incentives to the administrator. Users unfamiliarity with IoT scalability and massive communication capabilities will even deepen privacy risks. Further-
more, users’ data will be trapped indefinitely without the ability to be controlled or deleted, unlike conventional technologies [247]. Hence, there is a profound need to display all necessary information, feature, and data flow associated with each IoT device in our research to protect careless users and inform them about potential risks.

1.1.2 Smart Devices and Sensing Technologies in IoT

The introduction of smartphones and smartwatches has increased the profiling of collected data on users sharing their daily activities on various networks. In fact, using default security settings made penetrating user’s personal data by intruders a common practice [199]. The proliferation of smart objects and sensors will deepen even more information exposure, raising the threat to data privacy. Therefore, IoT sensing presents one of the most significant gaps where researchers have not been able to formulate reliable solutions [19]. Identification
in IoT technology links private users’ information (name, email, addresses, etc.) with the object ID and stores it in insecure commercialized devices such as cameras or RFID tags. Those methods raise privacy issues because smart devices often create unpredictable communications with unknown devices [258]. Identity management solution and k-anonymity techniques may be helpful in establishing a secure IoT network with the required privacy levels through anonymization and minimal encryption techniques. However, the proposed solutions are far from ideal as they can be easily broken using auxiliary data or homogeneity attacks [165, 199]. Another critical issue in IoT privacy is tracking and profiling, made easy through autonomous IoT technologies, in an era where face recognition is a common governmental practice. Smart devices will improve location-based services (LBS) accuracy, even in an indoor environment, using less invasive techniques. For instance, data will be collected in a passive and pervasive way where hidden sensors are continuously tracking users without their knowledge or consent [258]. This breach of the Declaration of Human Rights (Article 12) [81] has pushed the IT community to devise diverse privacy preserving solutions. Maganis et al. [166] proposed ”Tricoder”, a system capable of detecting Third-party sensors using smartphones. This framework enables users to inquire data being collected from sensors and displays privacy polices attached to each application. This approach is clearly a step toward data transparency and should be implemented and refined for future research. Another research conducted by Enck et al. [92] showed that it is possible to monitor the flow of Personally Identifiable Information (PII) in a smart device communicating with third party application. The authors achieved this through TaintDroid, an Android application capable of monitoring the flow of users sensitive data in real-time. While those research are promising, they do not provide context for knowing if this flow of information is easily understandable by the general population or overwhelming for the user. Unfortunately, no smart detection of sensitive data content or automatic user categorization (privacy level) in a smart environment has yet been deployed [258, 240]. Additionally, after conducting exten-
sive research no solution was found on locating embedded smart devices and restricting them from collecting physical data, which makes it even a more serious problem to user privacy. Thus, the proposed approach will focus on allowing users to control the captured data to ensure users full data ownership.

### 1.1.3 Privacy Preferences in IoT

To protect users from privacy invasions in a climate where sensors are omnipresent, Konings and Schaub [140] developed "PriPref", a privacy context awareness application. Through their application, users can broadcast their privacy preferences in the surrounding environment in different scenarios for better coexistence. To better assess user's privacy concerns, the authors first conducted a survey to enlist common user’s disturbances and strategies used to handle them. Based on the survey results, Konings et al. developed an application that displays the prevalent privacy preferences in neighboring environments. In fact, PriPref processes all users’ preferences and displays the current environment status to all participants for better privacy effectiveness. Users can also enable dynamic privacy adaptation, where the app will automatically modify users’ phone setting to accommodate the current environment preferences. This approach sheds light on an essential aspect of user privacy preferences, where the location plays an important part in users privacy decisions. This shift toward context-centric privacy and adaptive security is one of the goals of our proposed research as users may have different privacy needs in diverse environments. The concept of privacy differs between individuals, which could lead to a variety of expectations that should be examined and managed in an IoT environment. The Differential Privacy (DP) framework is a widely used technique that provides privacy preference anonymity for individuals in a group. This model’s main limitation is that users may belong to a group that is over/under protecting them, which may not reflect their actual privacy preferences. To mitigate this gap, Jorgensen et al. [130] proposed Personalized Differential Privacy (PDP) framework
that offers individual data privacy flexibility. Their result indicates that PDP can be achieved efficiently while preserving all the virtues and security expected by DP. Lin et al.[156] also experimented the feasibility of categorizing user’s privacy preferences through profiles in order to manage mobile app permissions. Their result indicates that it was possible to cluster users into four different privacy groups that served as a basis for more complex privacy needs. This demonstrates that privacy profiles are a viable technique on mobile apps but may not necessarily reflect users’ privacy preferences in an IoT environment where various heterogeneous sensors are deployed. Tsai et al.[235] also developed a privacy mobile manager based on contextual privacy preferences called "TurtleGuard". This novel privacy permission manager used a Machine Learning algorithm to lessen the decision burden on its users. By selecting few contextual circumstances preferences, the applications provided its users with the necessary feedback from various applications with the ability to modify or audit the automated decisions. The aforementioned researches stress the need for a tailored privacy preference experience unique to each individual and reflect his/her needs in different scenarios. The majority of privacy preference research focuses mainly on mobile apps permissions in particular and the Internet in general. Still, their work [130, 156, 235] can not be applied to IoT ecosystems due to its scalability, heterogeneity, and continuously changing user behavior.

1.1.4 Security and Privacy Fatigue

The term "security fatigue" was conceived by Steven Furnell and Kerry Thomson in 2009 through their finding that security is perceived as tiresome and an obstacle when acquiring services [98]. Fatigue can be defined as a feeling of tiredness caused by extreme stress when people are faced with overwhelming demands and decisions to make. This situation may cause psychological collapse and frustration, leading to decreased productivity and efficacy [111, 80]. SGI Global, a risk management company, surveyed 1280 participants and found
out that more than a third admitted that security is considered a barrier when using online services [17]. In fact, multiple agencies provide different privacy protocols to their subscribers that might seem complicated and nonsensical to the average users. Consequently, people feel burdened with excessive security options leading to the inability to make rational decisions and reaching a saturation state [229]. The IoT era has made it easier for users to share their private information to enjoy a wide range of online services; however, they are bombarded with an excessive number of security protocols to be followed. The situation worsens when users have to manage multiple IoT devices, each following different security protocols, leading to poor decision making and privacy leakage. According to Acquisti et al., "bounded rationality" is an inhibitor of the ability to implement information privacy controls. When a certain threshold is reached, factors such as cognitive limitation, lack of knowledge, and time may hinder people’s willingness to invest in privacy [27]. People always tend to find alternative ways to avoid additional security requirements when they reach the cost-benefit limit, resulting in restrained intellectual capabilities. Current research in IoT focuses on providing users with more information than they can handle [229, 27], and do not consider the increased burden associated with smart devices’ heterogeneity and scalability. This burden will lead to an unprecedented level of information disclosure and erode user trust in IoT privacy and security [75, 229]. Figure 1.3 depicts all factors that might lead to security and privacy fatigue, as explained in this section. Thus, the proposed research should focus on providing ways to reduce the amount of information through efficient visualization techniques and data filtering approaches in order to only display important information and valuable data. This will enable the user to control his/her data and make rational decisions when entering an IoT ecosystem.
1.2 Research Problem

1.2.1 Motivation

The Internet of things considered as the fourth industrial revolution[178], has revolutionized how we perceive our world and modernized the communication between the real world and the virtual one. This emerging research topic has enabled the deployment of embedded devices everywhere and added new dimensions to the internet realm, creating new privacy and usability challenges for the scientific community. This technology, utilizing users’ private information, made autonomous personalized services part of our routine while invading our personal space. This tremendous flow of unsecured data being gathered about users without their knowledge or consent, combined with complicated privacy policies, lead to information overload and loss of control. In 2017, Jason Hong [118], a pioneer in data privacy, challenged the IoT community to find ways to identify sensors and detect data flows in less than 30 seconds when entering an IoT environment. The IoT community has attempted to individually
solve independent IoT gaps through providing privacy management solutions [156, 161, 235] to make privacy categorization easier, Machine Learning techniques to predict users’ privacy preferences [140, 179, 152] and data visualization approaches that are generally perceived as complex by an average user [231, 92]. However, researchers have not attempted to combine those solutions to offer automated contextual privacy preferences based on continuous user needs and experience. Therefore, those existing challenges have motivated us to present a solution that will incorporate optimal data/sensor visualization approaches and knowledge sharing between individuals. In fact, as privacy is perceived differently between individuals, anonymous data sharing and privacy level categorization should be beneficial for the IoT ecosystem. IoT environments are increasingly more complex with the inclusion of a wider range of sensors utilized in the same location, thus analysis of user’s privacy perceptions must consider multi-sensor combinations. Implementing these approaches can revolutionize how users perceive privacy and lead to better privacy choices and data control. The Proposed approach should also account for IoT device scalability and data overflow that could lead to privacy fatigue and information disclosure.

1.2.2 Research Question

This thesis presents a method for protecting a user’s privacy in modern IoT-rich environments that gather a wide range of personal data automatically by anticipating the data gathering that could occur in a specific location and providing recommendations to the user regarding the potential impact on the user’s privacy preferences before they enter the environment.

Our research will address that challenge by providing individualized guidance to assist users in determining whether the IoT environment they are about to enter presents privacy risks that do not match the user’s own privacy preferences. This is accomplished by achieving the following goals:
• Goal 1: Collect data on user’s privacy preferences and perceptions regarding their exposure to a range of IoT devices.
  - This goal is addressed in Chapter 4

• Goal 2: Categorize the impact that data gathering by different types of IoT sensors in different IoT environments has on user’s privacy perceptions.
  - This goal is addressed in Chapter 4 and 5.

• Goal 3: Compare Machine Learning approaches to determine the optimal method for predicting the impact of IoT data gathering on a user’s privacy preferences.
  - This goal is addressed in Chapter 5.

• Goal 4: Design a Privacy Preference Manager to predict the impact that IoT data gathering will have on a user’s privacy before the user is exposed to the IoT sensor environment.
  - This goal is addressed in Chapter 6.

• Goal 5: Determine how accurately the prediction results match the user’s privacy preferences.
  - This goal is addressed in Chapter 7.

1.2.3 Proposed Approach

Our proposed approach will address the challenges stated in Section 1.2.2 through developing IoTPP (Internet of Things Privacy Preference) Web App capable of assigning privacy levels depending on the users’ privacy perceptions and allowing them full control over different sensors in various IoT environments. Our research will focus on a combination of user privacy knowledge and preference matching to make effective decisions and allow the user
to control the disclosure of information in specific locations. To do so, we will first try to categorize sensors and locations in various categories depending on how sensitive the gathered data is. Additionally, we need to effectively understand the relationship between sensor combinations and contextual privacy to phantom user needs and perceptions. To classify different IoT features and grasp users’ privacy needs, we will conduct a replication study based on Naeini et al. approach [179]. This study will be the basis of our research as it will clarify user privacy needs and help us develop a survey that will be used to collect more informative data. The application will be designed to reduce the amount of unnecessary information given to the user to lower the feeling of privacy fatigue. Our survey will only include critical information needed to understand user preferences toward different sensor categories and locations. This application’s overreaching goal is to develop a flexible solution that will provide users with enough information and guidance when interacting with an IoT space. This will be achieved through a combination of user experiences and Machine Learning algorithms. We will first compare different Machine Learning techniques and factor combinations to Naeini et al. [179] research and eventually create a new Privacy Preference Manager (PPM) based on all the knowledge and feedback collected. IoTPP Web App will incorporate our final ML approach (PPM) to automatically provide users with a tailored privacy-enhanced recommendation before entering an IoT space. Users will be able to interact with different sensor combinations and share their experience in specific IoT locations by providing feedback on the recommender’s prediction and the type of data captured that may put their privacy at risk. This unique user-centric privacy risk visualization will enable users to better manage their privacy preferences in specific IoT spaces. Figure 1.4 illustrates a general overview of the interaction between the user and the application. This research will leverage the field of privacy management, users shared experience, and sensor visualization to bring optimal decision making and provide transparency and usability.
Figure 1.4: Privacy Preference Overview
1.3 Research Description

This section will provide details about the different phases we will be undertaking to implement our proposed solution.

- **Phase 1**: Analyze the variety of sensitive information gathered by different types of IoT sensors and determine how a user’s privacy perceptions with respect to this information will impact their contextual privacy preferences.

  Phase one is a critical part of our work as it will help us understand and classify how sensitive the data collected through sensors are. Comprehending this data will help us understand how users perceive privacy preferences and help them make optimal decisions with less effort. We will be undertaking a replication study of the Naeini et al. paper [179]. This vignette study will help us determine users’ preferences regarding collecting their data in an IoT-based environment. Furthermore, surveying users using different scenarios will help us determine what factors might impact their privacy decisions. Phase one steps will be explained in detail in Chapter 4, and its results and analysis will be used as input for our next phase.

- **Phase 2**: Evaluate different Machine Learning approaches to determine which is most effective for categorizing the potential effect of data gathering on a user’s privacy preferences.

  In this phase, we experimented on the collected data from the replication study to see how well different features can contribute in predicting an individual decision to allow or deny entry to a specific IoT location. The main goal of this approach is to build a system capable of mimicking real-world scenarios where users have control over their shared data sustainably. To that end, we decided to eliminate factors that users do not have control over. Factors such as user perceived benefits and retention time are...
hypothetical and may depend on the authority managing them. Phase one enabled us to engineer new feature categorizations that would help us create new factors to provide a better ML pattern building. To develop our models, we decided to test three different ML models to assess the best approach for our binary classification and compare it to Naeini et al. research [179]. Phase two allowed us to understand users’ privacy needs better and helped us design a better approach to mitigate user privacy fatigue and enhance users’ privacy experience. Phase two steps will be explained in detail in Chapter 5 and based on the observation collected, we will build PPM explained in phase three.

• **Phase 3**: Design a recommender system that we call the Privacy Preference Manager (PPM) with the goal of predicting the impact data gathering in a specific IoT environment will have on a user’s privacy preferences.

This phase’s primary goal is to apply all the enhancements collected from the previous phases and find an optimal ML method that will report to users the privacy risk associated when entering a specific IoT location. We start by Building a simpler survey that collects only factors that users have control over while creating new features around them to avoid user information overload. Our survey was built to contain an even question distribution between different factor interactions. After collecting the necessary data, we evaluated different coding techniques to enhance our results and find the best approach that satisfies our criteria. PPM also provides a Shift from a binary classification (Allow/Deny) to a recommender system with different privacy risk levels (High Risk, Medium Risk, Low Risk). In terms of ML, we applied multiple boosting techniques to maximize our final model performance. In this phase, we compared numerous ML algorithms performances to determine the best approach that will be integrated into our IoTPP web APP. Phase three steps will be explained in detail in
Chapter 6

- **Phase 4**: Implement and evaluate a web-based version of the Privacy Preference Manager to determine how effectively it guides users in making decisions regarding their privacy in IoT environments.

After grasping all the necessary data from phase three, we now understand how different factors impact users’ privacy decisions in an IoT environment. This knowledge allowed us to develop a web app to guide users when visiting different locations. IoTPP relays on the trained model implemented in phase three and provides users with a tailored privacy recommendation based on their privacy level. IoTPP will provide its users with a sensor/location category information that is most critical to them and use minimal input to predict if entering an IoT space will affect their privacy requirements. Users will also be able to choose from different locations and sensor types in a drop-down menu fashion for an easy visualization process. Our Web App was designed to help users apprehend the ML prediction and allow them to anonymously share their experiences to optimize our system. Users shared experiences and preferences will help us collect more data and improve our ML model efficiency. To cope with the problem of privacy fatigue, we used the NASA-TLX(Task Load Index) scale [20] to determine the users’ perceived workload and the SUS scale [59] to study the usability of IoTPP for future App design enhancement. Details about the web application development will be described in Chapter 7.
Chapter 2

Internet of Things

2.1 The concept of Internet of Things (IoT)

2.1.1 Brief History of IoT

Kevin Ashton, executive director of Auto-Id labs, was the first person to propose the IoT concept in 1999 where he described it as a network of unique objects connected with a radio frequency identification (RFID) technology [155]. Figure 2.1 describes the growth of the IoT concept beginning with the emergence of wireless networks (Wi-Fi, Bluetooth, Electromagnetic waves) that laid the foundation for implementation of sensor fusion and mobile computing. In 2005, technology evolved to provide Internet users with wireless sensor networks (WSN) that could sense the physical world. The era of WSNs came with cloud computing, low energy communication, and Web 2.0 [155]. Since 2010, IoT has evolved exponentially to enhance interoperability, reliability, and efficiency in global communication through mobile computing, and smart object cooperation. Recent advances in the innovation include the introduction of predictive analysis, advanced sensor fusion, and faster connectivity in wireless networks [3]. The evolution of IoT led to a complex network of multiple technologies
that range from wireless sensors to micro-electromechanical systems (MEMS) [155]. Figure 2.1 shows the history and evolution of IoT since 2000 [155]:

2.1.2 What is IoT?

There are many proposed definitions for IoT. In fact, formulating a standard description of the global concept of IoT is still in progress [155]. In 1926 Nikola Tesla stated that “When wireless is perfectly applied the whole earth will be converted into a huge brain... and the instrument through which we shall be able to do this will be amazingly simple compared with our present telephone.” [219]. This quote implies that Nicola Tesla envisioned the birth of IoT, where wireless instruments and sensors will have the ability to independently co-operate together for a better global communication. IoT was derived from the concept of RFID tags which are perceived as simple objects [41]. Auto-ID Labs, a leading company in networked RFID technology, was the first to invent the term Internet of Things 15 years ago [3]. Auto-ID Labs perceives IoT as “The basic counterparts that interlinks the real world
and the digital world” [3, 219]. According to Li et al., IoT is a global network infrastructure with the ability to configure itself dynamically based on various interoperable standards and protocols [155]. The ITU (International Telecommunication Union) supports this definition by describing IoT as a global infrastructure that supports advanced communication of virtual and physical objects based on dynamic information technologies [219]. Therefore, it can be seen that IoT is a complex system of interconnected devices and objects that may be uniquely identified to provide a service. Atzori et al. defines IoT as a novel paradigm through which different things interact using distinguished addressing schemes and communication protocols to achieve common objectives [41]. The concept of “Internet of Things” includes the “Internet” oriented vision and the “Things” oriented vision where different technologies can be wirelessly connected through the Internet using standard protocols. When these two visions are merged, a disturbing level of innovation arises due to the huge amount of valuable information exchanged between the Things. Consequently, a new “Semantic” oriented vision is needed to facilitate data processing and execution. Figure 2.2 encapsulates the convergence of these three visions presented by Atzori et al. in [41]:

In [4] Miorandi et al. assert that IoT is a “global network interconnecting smart objects by means of extended Internet technologies” and techniques required for the visualization of a virtual environment. The concept is therefore perceived as a global system of smart devices with which humans can interact through an interface connected to the Internet. Other authors, such as Al-Fuqaha et al., describe the IoT concept as a system whose objective is to produce a new class of application through the collaboration of a smart sensor embedded in objects and the Internet [30]. Furthermore, IoT is a system that equips everyday objects with capabilities to sense, identify, and process information after connecting with other devices and services via the Internet [243]. Therefore, technology is made part of everyday activities by its diverse applications in various fields.

Diversity within the ICT (Information and Communications Technology) world, such as
background, visions and stakeholder needs has resulted in a multiplicity of definitions for IoT. After exploring all those IoT definitions from different perspectives, we define IoT, in this work, as collection of smart sensors that are uniquely distinguishable and operates under low computational capabilities. Those sensors communicate through specific protocols over the network in order to securely deliver the intended services to the user or organization.

### 2.2 IoT Elements

IoT is a collection of multiple elements that work together to achieve the intended objectives. The concept can be explained through a systematic analysis of the six building blocks of the IoT. The six building blocks presented by Al-Fuqaha et al. (Figure 2.3 top part) include identification, communication, sensing, computation, services, and semantics [30].
Each building block in Figure 2.3 has a specific function that supports the communication between objects and users. However, the functionalities depend on each other through specialized technologies to enhance interoperability and reliability of the system.

### 2.2.1 Identification, Sensing, and Communication Technologies

Identification is the IoT unit that uniquely identifies each object before accessing and collecting its information. The process of identifying things is important because it helps in establishing the object class (Camera, heat sensor, fridge etc...), its capabilities and access right in the IoT space [175]. In the next section we discuss in detail the technologies and protocols used in each IoT block as illustrated in Fig 3. IoT has many available identification techniques that are considered as unique numbers attached to each object in an RFID tag [30]. These include ubiquitous codes (uCode), QR codes, and the electronic product codes (EPC). These codes can help in reading the tags and returning an identifier that allow users to query a database to retrieve information about an object or service [175]. IoT sensing is the process of collecting data from various smart objects within the system and sending it
to a central database or cloud for processing. Some of the units that collect the data include actuators, smart sensors, and wearables with sensing capabilities [30]. For instance, companies like Belkin provide their users with smart hubs and sensors through WeMo home control that enables them to control their appliances with a simple click [211]. Wireless Sensor Network (WSN) is a technology that is fundamental in the sensing process. WSN is a collection of multiple sensors that interact to monitor environmental characteristics such as temperature, humidity, pressure and surveillance. WSN passes the collected information through the sensors to a database while the actuators perform the required action [243]. Some of the actions that actuators may perform in response to the sensed data include but are not limited to sound, light or vibrations.

An example of a WSN that implements a sensor-actuator technology is a system that detects fire in a house and uses an actuator to start a fire sprinkler. The communication subsystem in the IoT network is responsible for connecting multiple objects to share information and deliver the required services [30]. In the presence of noisy or lossy transmission links, IoT nodes use low power communication to send and retrieve information. Some of the technologies used in the communication process include Bluetooth, Wi-Fi, RFID, Z-waves and Near Field Communication (NFC) [30]. Communication in IoT is often a hard task because of the limited resources and low computational power the devices can offer. All the devices in IoT are supposed to be organized in a manner such that they can be accessible and available for communication with each other at any time [155]. Wireless Sensor networks, NFC, and the RFID systems play a crucial role in IoT Identification, sensing and communication. RFID uses radio frequency to receive and send information from an object to a receiver [175]. Active RFID tags continuously emit a signal from the transceiver and harvest the necessary energy from an on-board battery or the signal received giving it more transmission range (Up to a 1000 feet) but higher production cost [184]. Passive tags do not emit any signal but rather wait to be contacted and modulate the data received in order
to add the information [184]. The tags can use either high or low frequency, making them more appropriate for IoT, as having a wide communication spectrum enables them to operate under diverse channels depending on the task to be accomplished [41]. NFC, on the other hand, has a faster data rate of, up to 10 Mbps, compared to RFID hence making it preferable for modern communication. NFC is practicable for short ranges, not exceeding 10 cm, and uses high radio frequencies [30].

2.2.2 Computation

IoT computation involves hardware and software applications that perform processing functions within the network. The main core of IoT are microprocessors and microcontrollers. These systems also require hardware platforms such as UDOO [21], Arduino, Raspberry PI, and Intel Galileo to run the required applications [30]. IoT computation also requires special software that helps in running the processing functions of the Operating Systems with real-time capabilities such as Riot OS [44], TinyOS [153], and LiteOS [63]. These Operating systems suit the specification of IoT as they are flexible and can operate under restricted memory and processing speed.

2.2.3 Services

According to Gigli et al. [101], IoT services fall into four categories that include Identity-based, ubiquitous, collaborative-aware, and information aggregation services. Identity-related services can be passive or active. Active services depend on power sources for identification and passive services do not require any computational power and need an external mechanism in order to identify the physical environment to the virtual world. Examples of these services include shipping, where IoT can help improve the tracking of products when they are in transit though unique identification of packages at all times [101]. Information ag-
Aggregation services gather, analyze, and summarize the data from sensors before processing and reporting it to the appropriate IoT application [30]. Examples of these services include the supervision of green agriculture for an optimal production [101]. Collaborative-aware services use the sensor data processed and reported from aggregation services to react and make decisions [101]. Examples of collaborative-aware services are smart buildings, smart homes, and industrial automation using 6LOWPAN as an IP infrastructure [30, 101].

2.2.4 IoT Semantics

IoT semantics is the ability of the system to extract information and knowledge from machines to perform the intended services to meet the needs of users. The extraction of knowledge is a process that involves discovering relevant data, using appropriate resources, and modeling the right information using a set of rules and features. Through knowledge extraction the system is able to interpret and examine the data to formulate a decision during service delivery [30, 48]. In [30], the authors compare the IoT semantics to a brain that sends demands to the appropriate resource for a particular response. An example of semantic web technologies that support IoT interpretability is the Web Ontology Language (OWL). OWL was designed to provide a detailed description of things and their relationship (ontology) using a universal web language that can be understood by different operating systems. In fact, OWL will help IoT integrate and process information between things explicitly making communication between devices and the web easier. In addition, the World Wide Web Consortium (W3C) introduced Efficient XML Interchange (EXI) in 2011 to help the IoT network optimize XML-based applications when operating in environments with resource constraints [216]. The innovation by W3C is helpful in that it reduced the bandwidth required by IoT without affecting the system performance and memory. EXI achieves its efficiency by converting the XML texts to binary thus decreasing data size and enabling efficient communication between sensors [216].
2.3 IoT Architecture

2.3.1 Possible Architectures

The increasing need to interconnect numerous smart objects via the Internet requires flexibility, scalability and interpretability [30]. Currently, there is no standard architecture followed by IoT that provides a reference model accepted in all domains [41, 155, 175]. When designing an architecture, the focus is on a certain domain or an industry sector, which makes it hard to standardize to meet all possible requirements [144]. Projects such as IoT-A [8], a European initiative, have tried to design a common architecture that can address a wide range of applications, but the project was discontinued due to unsatisfying results [175]. The diagram below shows four frequently used IoT architectures:

The three-layer model in Figure 2.4 (a) is the most basic architecture and was used in the initial stages of IoT development at a small scale. This model lacks abstraction, management and does not take business layers into consideration [249]. The middleware-based model in Figure 2.4 (b) adds more abstraction by breaking the network layer into middle-
ware, coordination, and backbone layers. The middleware layer is responsible for managing the information and services between devices of the same category and processing the data to make the right decisions [134]. The five-layer model in Figure 2.4 (d) is beneficial in understanding the way business processes may integrate with the IoT functionalities through the business layer. This approach is also able to manage, monitor, and analyze Big Data through proposing business model and building flowcharts [30, 134]. As can be seen, IoT architecture are diverse and have different approaches, though the Service Oriented Architecture (SOA) approach in figure 2.4 (c) is the most common approach because it helps in breaking down complex systems into simpler applications composed of distinct units [30, 41, 155, 175, 243, 192].

2.3.2 The Service Oriented Architecture

According to Miorandi et al., a Service-Oriented Architecture (SOA) is a group of services that use a set of standardized protocols and languages to communicate and interact with each other through a published platform [175, 192]. SOA is an architectural style of the Service Oriented Computing (SOC) models that implement a low computational resource distributed architecture. The SOC approach treats entities of a system as blocks and uses standard interfaces to access them [175]. SOA uses HTTP and other web-based protocols (SOAP, XML, etc.) to enhance operability and make the web services behave like virtual systems to meet specific user needs. According to Papazoglou, SOA improves flexibility and heterogeneity in the IoT system and offers integrated services [191]. The Service-oriented Approach also allows hardware and software to be reused individually since it does not enforce any service implementation technology [41, 193]. Re-using software also reduces the testing and the maintenance costs. The SOA approach also enables the use of standards protocols and interfaces implemented in an enterprise system [41]. SOA model is also beneficial for business processes through transparent service coordination which helps in adapting to market
evolution [41]. Figure 2.5 shows the SOA architecture with all its layers:

The SOA model layers shown above and proposed by Atzori et al. encompass the functionalities required in an IoT network by taking into consideration all issues encountered by the middleware approach [41]. From top to bottom, SOA architecture is composed of Application, Service Composition, Service Management, Object Abstraction, and Objects. Trust, Privacy and security are also taken into consideration and may be incorporated at any layer, making it a powerful approach. Each layer has its function but depends on other layers for complete and reliable service delivery.

### 2.4 IoT Applications

IoT made the virtual realm a part of our daily life through diverse applications. The applications of IoT are infinite and can be applied in different domains and environments. Connecting billions of smart devices and giving them the ability to cooperate and analyze the environment, modernized our life, and made it more convenient.
In 2015 there were approximately 15.4 billion interconnected devices deployed and is expected to reach 75 billion by 2025 [77]. For instance, in 2016, $737 billion was spent on IoT across markets. This figure is expected to grow up to $1.29 trillion by 2020, which is approximately 15.6% of the annual growth rate [77]. It can be seen from figure 2.6 that IoT has been deployed in diverse sectors such as agriculture, transportation, industry, smart infrastructure, education, etc. However, this research will focus on topics such as security and surveillance, Social IoT, and wearable devices.

2.4.1 Security and Surveillance

China and the UK are considered as surveillance states with respectively 176 million and 5.9 million cameras deployed [23, 13]. IoT technologies can bring radical change to security
and surveillance by providing cheaper and less invasive techniques to monitor and track people’s activities [175]. Security and surveillance became a necessity in the modern world in order to monitor public places and borders and help protect assets. IoT systems can solve these problems by replacing the widespread deployment of cameras with tiny computers and smart sensors while preserving privacy. The technology is beneficial as it will be able to provide real-time positioning and behavior analysis [175]. Besides, smart sensors have the ability to monitor and analyze their surroundings to prevent catastrophes such as gas leaks and earthquakes. Another advantage of using IoT technology is the ability to implement flexible access control policies that may change over time due to logistic change and business evolution [175].

2.4.2 Social IoT

The concept of Social Internet of Things originated in 2001 and was proposed by Holmquist et al. In their paper the authors were trying to wirelessly connect smart artefacts (sensors) through a set of relationships [42, 117]. In fact, Holmquist et al. [117] used contextual and spatial proximity to temporarily connect artifacts with the same parameters. Users with the same preferences can than detect each other’s in a crowded place thus enabling social relationship [117]. Social Internet of Things (SIoT) is an application of IoT that can enhance service discovery and perform various social activities. SIoT can enhance privacy and confidentiality of information by deploying the current security measures used in human social networks [155]. Social networks can collaborate with IoT to improve the experience and trust of interacting with other people though automatic sensing of users’ location and sharing it with friends. IoT technology can also sense when objects are in proximity of each other for a certain period of time and automatically trigger friendship between individuals and share their information [42].
2.4.3 Wearable Devices

A wearable device or computer is a technology created through the fusion of embedded devices and ubiquitous computing. Wearable devices have the ability to communicate through networks and are considered to be mobile to deliver specific services [239]. According to Barfield et al, a wearable device is a “fully functional, self-powered, self-contained computer that is worn on the body... provides access to information, and interaction with information, anywhere and at any time” [47]. Steve Mann, considered as the father and founder of smart wearable devices, added that wearable devices must be controlled by their owner. According to Jian et al, wearables are light weight portable devices that have the same abilities of a smart phone and are sometimes more convenient to use to accomplish tasks faster [126].

In 1997 Bass defined wearables as a set of devices that have five characteristics: can be used while in motion, can be used without user interaction, can be body attached or part of clothes, must be controlled by its wearer and must be available at any time [126, 49]. The first wearable technology can be traced all the way back to the 13th century during the invention of the first eyeglasses. In the 16th century, Peter Henlein created the first wearable clock in Europe called Nuremberg eggs hence initiating the journey of applying IoT technology through wearables [234]. The intention of these portable watches was for them to be worn around the neck, making them a popular symbol during the 16th and 17th century. Abacus, a ring created in china in the 17th century, was considered as the first smart counting ring. In 1838, a scientist, Charles Wheatstone, demonstrated that the brain perceives differently two-dimension images from each eye and project it onto a single three-dimensional object [7]. He developed the first stereoscope that brought a sense of immersion to users for virtual tourism hence becoming the first experience of Virtual Reality (VR).

In 1966, Professor Edward Thorp reveled in his book “Beat the Dealer” how he cheated at roulette by embedding a tiny computer in his shoe [137]. He implemented a timing device to help him predict the position that the ball would land next, giving him a leading edge
of 44 percent in the game. In 1975, the Hamilton company released the world’s first wrist calculator [126, 137]. In 1977, CC Collins designed the first head mounted device for the blind to help them convert images into tactile grids [126]. In 1987, the founder of Visual Programming Lab (VPL), Jaron Lanier, formally introduced the concept of “virtual reality” by developing multiple virtual reality devices such as virtual goggles and Virtual Gloves [7]. Since then, technology has evolved exponentially, and wearable devices markets have emerged. Companies such as Nike developed the first sport kit in 2006 to gather movement information, followed by FITBIT in 2007 [137]. 2013 became the climax of wearables when Samsung lunched their first smart watch with full computing capabilities. Nowadays, the market for wearable devices is fully deployed and devices such as smart glasses have emerged to bring a new era of visualization. A timeline for wearable devices is illustrated in Figure 2.7.

There are two standards used to classify wearable technologies according to Jiang et al. These standards include the form-based and the function-based criteria [126]. Smart glasses are considered as form-based wearable technology that brings a computer monitor in front of a user’s eye. Smart glasses use augmented reality technology to capture the physical world and translate it to a virtual environment [203]. According to Philipp et al. smart glasses are a combination of wearable devices and augmented reality and uses a mixture of devices such as cameras, GPS and sensors to capture information from the physical world [202].
Smart glasses are an efficient way to access data dynamically because they will learn human interaction and behavior quickly and enable a better understanding of the surroundings. It can also be used in industries to enhance performance and help users accomplish tasks more effectively. Google Glass [6], Microsoft HoloLens [11], and EverySight [5] are examples of commercialized smart glasses.

2.5 IoT Security

The security of IoT has evolved over the recent years to enhance data privacy and integrity owing to its gradual evolution towards becoming a significant driver in controlling and monitoring applications [41]. The primary reason for security improvement is the scalability and heterogeneity of wireless smart objects. The billions of things in the network lack a unified structure that may allow universal security measures, making it difficult to improve their data security and privacy. IoT devices have diverse standards and protocols that require multiple solutions for their individual security issues [255]. Wireless communication systems, such as the RFID, have promoted the advancement of IoT to embrace various standards and services anywhere and at any time [210]. For instance, an IoT system is capable of collecting data about the environment and its surroundings and share it with other IoT components for better decision making. However, this high level of diversity and the expendable nature of IoT makes the network vulnerable to security and privacy threats [221]. The IoT network contains a tremendous number of smart Things that store, transmit, and process confidential data about people, businesses, and governments over the Internet. The presence of highly valuable data and personal information in the system may tempt hackers to utilize malicious techniques to compromise and control the system to acquire critical data for future bargain such as ransom or profiling. Therefore, network administrators, users should implement security measures to protect the confidentiality, integrity, and availability of information, driv-
ing the need for better and robust IoT security [210]. Additionally, hardware manufacturer, considered as the main core of IoT technology, should address security primitives (encryption, authentication, etc.) in a creative way at the design level in order to enable developer to implement basic protection [250].

For an IoT network to be secure, one should implement specific security measures that allow its users to access, share, and store information in a trustworthy manner without experiencing any privacy or security concerns. Due to the heterogeneity of smart devices, IoT has to account for six security pillars. The security features are composed of the CIA triad: Confidentiality, Integrity, Availability and three other security information standards: Authentication, Privacy and Trust [158].

### 2.5.1 Confidentiality in IoT

According to [158], confidentiality is an aspect of the IoT network that ensures information is accessible to authorized users only. An information system within the IoT network should ensure that data is accessible and available to users or devices with the required permission and authority.

For instance, an IoT system should be able to protect its users from unintended or unauthorized access, such as eavesdropping, when neighboring nodes are communicating private information. Moreover, smart sensors, such as RFID tags, should protect the stored data from leaking to other components without the owner’s authorization. IoT may achieve a higher level of confidentiality by implementing access control, secure key management, and encryption in wireless systems [64].
2.5.2 Integrity in IoT

Integrity in IoT involves protecting data against unauthorized or hazardous manipulation during data transfer among devices [158]. The security of the IoT should enhance user data protection from intrusions and accidental changes for accurate output. Processing forged information may result in the execution of erroneous operations that may impede the decision-making process causing monetary and physical damage.

2.5.3 Availability in IoT

Availability in IoT requires that the system and its components are accessible and functional when requested by authorized users anywhere and at any time. Most IoT subsystems involve real-time services that users with relevant permissions should access instantaneously [158]. Furthermore, Stored information should always be available and protected from malicious attacks and natural disasters. Denial-of-service (DoS) is a common attack that leads to the loss of data availability [158]. System administrators and users may safeguard against security threats affecting availability by deploying real time backup techniques and using secure routing protocols such as RAED (Robust formally analyzed routing protocol for WSN deployment) that help against flood attacks in a resource constrained environment [169].

2.5.4 Identification and Authentication in IoT

Identification is the process of establishing the identity of a network user or device in order to grant permissions or assign responsibilities. The IoT should ensure that unauthorized devices are locked out of specific subsystems to protect its users from malicious activities such as data tampering which can lead to wrong data analytics. Authentication determines whether the users and devices accessing certain network resources are legitimate and have authorized access. Authentication is a primary way of assigning permission to data owners to
help them reduce intrusions from illegitimate sources [158]. Internet is a complex environment containing an abundant number of nodes that need to be identified and authenticated by the system before delivering the intendent services. The use of Biometrics may be an efficient approach to help ensure the legitimacy of users during the authentication process in IoT. Biometric traits are unique and unchangeable making them more resilient compared to complexes approaches such as passwords [119].

2.5.5 Data Privacy in IoT

Smith [224] defined privacy as “the desire by each of us for physical space where we can be free of interruption, intrusion, embarrassment, or accountability and the attempt to control the time and manner of disclosures of personal information about ourselves”. According to Lin et al. data privacy in IoT ensures that legitimate users have full access and control over their private data and no other entity or organization can manipulate their data [158]. Privacy also includes the right to stay anonymous as well as the right to control distraction and attention [140]. Privacy also requires that network users should control their data through the whole IoT cycle (as shown in Figure 2.3) in order to prevent unnecessary harm. According to Ann Cavoukian, the core structure of IoT should implement privacy preferences and promote a user-centric approach through the whole design process [66]. This could be done through the Privacy by Design Framework (PbD) created in 2010 and translated into 37 languages [68]. The PbD principle (part of the General Data Protection Regulation) should be applied at the manufacturing phase in order to make IoT privacy transparent and protect the sensitive data collected between devices. The Federal State Commission also reported that PbD is the most promising approach to protect user’s privacy and make IoT technology trustworthy [208]. It can be seen from Figure 2.8 that the PbD Framework is composed of seven principles that mainly promote privacy in a holistic way rather than in a technical way [68]. These principle are as follows:
Figure 2.8: Privacy by Design Principles

1. Proactive not reactive: Hinder harm before it happens.

2. Privacy as default: User data should be protected by default at all times.

3. Privacy embedded into Design: Privacy should be accounted for from early design stages.

4. Full Functionality: Enhance user experience while reinforcing security and privacy.

5. End-to-End Lifecycle Protection: Protect user data through the entire communication process.

6. Visibility and transparency: Making sure that the system and stakeholders are not exploiting the technology.

7. Respect for user privacy: It is all about the user.
2.5.6 Trust in IoT

According to Andrea e al. trust is a complex concept that enforces security and privacy in IoT [37]. Trust in IoT aims at developing a proper interactive behavior between IoT devices, applications, system layers and users [37, 158]. Trust is built on different concepts related to security and privacy such reliability, which refers to the ability of an entity to decide whether it will share or disclose information with other devices or users [251]. System administrators may achieve security and privacy by implementing trust management systems to monitor the interaction of heterogeneous objects within the IoT layers [37]. Therefore, trust enforces all the above IoT pillars by determining the nature of the relationship between smart objects and users when cooperating with each other [31].

2.6 IoT Security Issues

The CIA triad is a basis for developing the security and privacy mechanisms of a network. The communication between different users within the IoT network should meet the trust-worthiness goals described in the six pillars of the system. The major differences in deploying the security controls in networks within the IoT and other wireless systems is the nature of the involved technology and its implementation [91]. For instance, modern security technologies may be difficult to deploy in IoT systems leading to the need to develop lightweight but robust security measures.

2.6.1 Conventional vs. IoT Security

The IoT handles security and privacy differently from conventional approaches due to the limited processing power on LLN (Low Power and Lossy Networks) [31]. The primary key for developing a secure mechanism in IoT systems is to consider the limited computing power and memory requirements. Therefore, the implementation of traditional security
components, such as firewalls, Intrusion Detection Systems, and encryption algorithms, may be difficult. Besides, traditional systems have dynamic topologies that use applications with more computational requirements than IoT systems [31]. In [133], Kasinathan et al. proposed an IDS framework (DEMO) for 6LoWPAN. DEMO is capable of monitoring and detecting DoS attacks in real time. This IDS is built on SIEM (Security Incident and Event Manager), a system that can combine captured data from different nodes to analyses potential breaches. For better resilience, the authors used FAM (Frequency Agility Manager), that is capable of detecting the best available channel when an attack occurs and redirect all communications to it [133]. Conventional and IoT security both carry the same issues and their goals are similar, but they are both designed differently. Hence, lightweight technologies, such as encryption, authentication, etc., are needed to ensure secure communication between smart things.

2.6.2 IoT Networking Challenges

The major challenge in achieving the security features of an IoT network is the complex architecture of the system that requires innovators to develop solutions to protect the immense scalability among things. Besides, the connected objects interact at different layers, with diverse applications, and allow people with different permissions to access them in real-time [210]. The global connectivity among devices widens the exposure of the network to vulnerabilities that may lead to physical damage [210]. Moreover, the heterogeneity of multiple objects operating at different layers of the IoT systems while exchanging data with other systems complicates the deployment of security techniques, resulting in significant security issues.
2.6.2.1 Network Security and Protocol Issues

The heterogenous nature of the IoT system increases the risk of network and protocol breaches by exposing the communication channels between constrained devices in unknown environments. For example, a compromised node may communicate malicious data to other trusted entities and compromise the whole system. Other issues may arise when talking about IoT due to the lack of uniformity and computational capabilities. Entities in a distributed environment may communicate with other unknown constrained machines and exchange information without any proper secure configuration. The authors of [210] believe that security experts should implement adequate security protocols and strong end-to-end encryption between devices handled by efficient key management systems or universal digital certificates such as the X.509 certificate.

2.6.2.2 Identity Management Issues

Identity management (IdM) aims at ensuring that the information sent from a specific network entity contains the correct data generated by legitimate users or devices. The existence of various devices running on different operating systems and communicating over unreliable channels emphasizes the need to create a universal authentication system to manage identities. The interaction between smart things can be quite dynamic and one cannot know in advance the type of connection or services that can be established in a connected world. Another important concept is managing access control in order to limit an intruder’s violation toward generating fake data and limiting the access to authorized users only [210]. Traditional IdMs approaches like EPC and ucode have been discussed in section 2.2.1, thus there is a need for a robust and reliable authentication system that manages users and devices access across the web.
2.7 Attacks on IoT

IoT uses wireless and cloud technologies as a median to enhance connectivity across the network. IoT technology can be deployed in several domains such as healthcare, smart cities, and governments where infinite flow of valuable information is generated. This enormous amount of data is the primary driver for attracting attackers [31]. Besides, the scalability and heterogeneity of connected devices may create multiple paths that attackers may use to execute malicious attacks. In the next section we will discuss the possible attacks on IoT devices and efficient techniques to mitigate them.

2.7.1 Perception Layer Attacks

The perception layer in a basic IoT architecture is responsible for identifying objects and collecting information flowing between nodes [134]. Attacks at the perception layer aim at performing unauthorized data manipulation collected over the network [158]. A node capture attack is a perception layer intrusion where an attacker can tamper with a device within the IoT network or alters its functionality to take over the system [255]. The adversary aims at faking the legitimacy of a node to connect to other trusted nodes and extract sensitive information. One can prevent a node capture attack by implementing an effective node monitoring mechanism, such as Cooperative Distributed Detection (CDD) [51]. CDD works through acquiring nodes information’s and sharing it between neighboring devices. If a node A fails to send its information within a certain frame time to an adjacent node, CDD raises a flag and stops all incoming communication from the alleged captured node until further processing [51]. Malicious code injection is another perception layer attack that allows attackers to gain unauthorized access to an IoT system by injecting malevolent instruction in one of the node’s program to change its behavior [158]. The compromised node can be used to
execute specific malevolent tasks as well as to provide a bridge to compromise the whole IoT system. One method to prevent this type of attack is to use attestation techniques to verify the legitimacy of the memory content [218]. Attestation techniques are procedures that determine the legitimacy of the memory content of embedded devices. In the case of IoT, attestation techniques have to be software based in order to avoid higher production rate. This approach works through sending a random (to avoid precomputation) message authentication code key that needs to be computed and sent back to the verifier [218]. Eavesdropping attacks occur in IoT when an intruder intercepts the data transmitted between nodes or inject noise to hinder with the information [31]. To counter eavesdropping attacks secure end-to-end encryption scheme and noise filtering techniques should be implemented in all wireless communication channels. An ideal approach of noise filtering techniques for scalable and dynamic embedded systems is the use of Kalman Filter (KF). KF is considered as an optimal estimating algorithm that is capable of statistically analyzing valuable data in presence of noise. The Kalman Filter work by recursively studying past measurements to precisely predict the outcome of the current state [104].

### 2.7.2 Network Layer Attacks

The primary function of the network layer is to send information collected from sensors to other layers of the IoT infrastructure [134]. Most IoT breaches in this layer are related to the wireless nature of the environment [158]. DoS attack occurs by bombarding components of the IoT network with massive number of requests to exhaust the system and make it unreachable or unavailable [31, 158]. An attacker can inject an infinite loop generating a large volume of traffic into the IoT network to degrade the limited battery power, memory and bandwidth of smart things, preventing legitimate devices and authorized users from accessing the system. In 2016 Mirai (“Future” in Japanese) botnet was able to generate 1.1 Tbps of traffic and took down several websites such as Netflix, Twitter and Reddit. This attack
was led by compromising four hundred thousand IoT devices that had weak authentication process [139]. A defense mechanism against DoS may involve but is not limited to Secure Bootstrapping, using light IDS, implementing patches and using Telnet to find open ports [50]. A Man-in-the-Middle (MitM) attack requires an adversary to insert a malicious node between two legitimate devices to gain access to other entities in the network [158]. The malicious device imitates the communicating device by acquiring identity information from them and acting as a medium for storing, transmitting, and manipulating data. A MitM attack allows a malicious device to send spoofed information to other legitimate entities to alter their behavior and capture sensitive information [31]. MitM attacks can be prevented by deploying strong identity authentication and encryption protocols such as SSL (Secure Socket Layer).

2.7.3 Application Layer Attacks

The application layer is the layer responsible for providing users with the necessary services and support. According to Lin et al, the primary security threats in this layer involve the exploitation of software vulnerabilities, such as viruses and malware [158]. Phishing attacks allow an attacker to steal user credentials through specially crafted websites designed to appear like legitimate sites. Moreover, an attacker may use corrupted links embedded in emails to fool unaware users and obtain unauthorized access to their system [37]. Malicious scripts are attacks that utilize malevolent programs that can run on top of a legitimate software to alter its functionalities. Malicious scripts can be embedded into trusted platforms which make it easier for an intruder to bait its users into running them [37, 158]. Consequently, the attacker can gain access to confidential information and modify the system parameters for illegal gain. Network administrators may guard against malicious scripts by implementing script detectors such as honeypots [159]. Honeypots are an effective way to deflect attacks and learn about the system flaws for future mitigations. Honeypot are programs able to
mimic the application behavior and appear to contain valuable information to lure attackers [38]. In 2015, Symantec deployed the first IoT honeypot in order to study the extent of smart devices attacks. Their report indicates an average of 9 attacks per hour and an average of two minutes to compromise the system [214].

2.8 IoT Challenges and Solutions

The primary security challenges in IoT arise due to the large number of unpredictable interconnection between nodes sharing unprotected data through the Internet. Billions of things deployed in the network lack a strong security implementation, making them an easy target to acquire [221]. Figure 2.9 illustrates the main IoT security challenges that will be discussed
in detail along with their proposed solutions in the next section.

2.8.1 Authentication and Confidentiality Solutions

An IoT systems encompass countless devices that communicate without any proper authentication mechanism. Thus, there is an urgency to ascertain their unique identity and ensure that private data is kept confidential. According to Zhao et al.[256], authentication could be improved through the use of a custom encapsulation technique and a cross platform communication scheme. This protocol, known as ISSAP (Intelligent Service Security Application Protocol), will deliver low packet overhead combined with a light encryption and authentication design. ISSAP is currently in the development phase and will potentially enhance services and speed up communications among businesses through providing a flexible US-PIOT (User Service Platform) and a secure standard packet approach. In [142], the authors introduce the first functional two-way authentication approach for IoT that utilizes current security technologies. In this approach, Thomas et al. use existing security protocols such as Datagram Transport Layer Security (DTLS) and RSA authentication. Moreover, their framework is ideal for the IoT network because it uses IPv6 over 6loWPAN, which is widely used in IoT environments [142]. Using DTLS protocol makes the communication between devices more secure as communication between layers do not need any security support or enhancement. Their architecture was able to securely identify communicating nodes and achieved a high level of confidentiality while using low energy and memory consumption [142]. In [252], the authors used Elliptic Curve Cryptography (ECC) as an approach for their authentication and session key establishment in constrained environments. In their architectural design, Ye et al.[252] used Attribute Authority to manage attributes of users and devices, and achieve mutual authentication between heterogeneous entities. This approach is mainly built on Attribute- based Access Control policy (ABAC) to establish secure communication between legitimate devices and defend against attacks such as eavesdropping or
MitM attacks. As can be seen from previous work, an effective authentication mechanism for IoT should consider resource constraints properties, reliable connectivity through trusted authorities, and the establishment of standard protocols among heterogeneous devices [221].

### 2.8.2 Access Control Challenges and Solutions

When it comes to IoT technology, access control solutions must consider the large amount of information circulating in the network and how to grant rightful privileges to appropriate IoT users to access them [221]. The authors of [164] propose a hierarchical scheme for data acquisition layers that classifies nodes with their respective security level. This access control approach allows nodes to sense diverse types of data and grant access to users with the appropriate permission type. The approach also takes into account memory and computational restraints as it requires only one key per user and per node. Additional keys will then be derived using a deterministic algorithm to obtain the requested information, hence expending security. Simulation results show low chances of information leakage thanks to the key splitting approach (Only part of the code is stored in the node) and local key update. Their performance analysis also revealed good system reliability and less overhead, compared to traditional approaches, while maintaining high network performance [164].

[121] describes an identity approach for accessing user location during emergency situations. The algorithm proposed in this approach uses policy subsystems to determine the level of emergency through the use of the RMA (Registration and Management Authority) before sharing the location details of a person in distress. The RMA is a trusted entity that stores VIDs attached with personal information. Once the permission is granted, the location of the user will be visible to the applicants and help can be provided. [191] addresses the issue of access control on outsourced data streams through the use of continuous authentication. Papadopoulos et al [191] developed CADS, a continuous query authentication technique that allows a service provider to send and update query results combined with verification.
information to authorized clients. This approach determines the authenticity and temporal completeness of the received data streams. CADS performance exhibited low object verification time (10 millisecond for one hundred thousand queries), enabling service providers to send more queries and updates. More importantly, CADS offered minimal communication overhead without jeopardizing QoS which makes it suitable for wireless restrained devices [191]. Rimma et al. [183] propose a solution that exploits metadata to improve security in a stream-centric scheme. In their novel approach, security policies are linked directly with the data to reduce the overhead. To guarantee full access control, users can specify what information can be shared or accessed by the DSMS (Data Stream Management System). To address interpretability and scalability in IoT environment, Cherkaoui et al. built an authentication and access control framework for constrained machines. In their research, they use Physical Unclonable Functions (PUF) associated with embedded SIM (eSIM). PUF is a new approach for device identification that uses device manufacturing entropy to extract unique physical parameters from sensors. To this end, PUF is able to generate a unique fingerprint for each device making this approach reliable and tamper proof. On the other hand, eSIM is an expansion of regular SIM card that provides scalable over-the-air connectivity for dynamic machine to machine communication. In fact, eSIM can remotely enable the change of operators or service providers through encrypted packets. eSIM can also provide secure routing when new subscribers are added into the network and secure key storage for safe M2M authentication [73]. Combining these technologies can provide flexibility, lightness and strong security visage for IoT.

2.8.3 Privacy Challenges and Solutions

Privacy in IoT can be volatile as embedded devices can silently gather information from the environment and track users without their consent or knowledge. In an IoT ecosystem, the information will be processed and shared with other smart devices for further analysis and
better decision-making. This insecure critical flow of information creates an urgent need for
vigorous data protection and privacy preservation solutions [221]. According to Evans et al. [94], security experts should guarantee privacy by tagging data using Information Flow Control (IFC). Due to the nature of IoT interaction and to provide customizable services, sensors may collect and communicate information about users’ needs and preferences in order to provide the best services. To overcome this privacy invasion, the authors proposed a static data tagging that associate security labels with the data sent. The intention of the tags is to help the system manage data integrity and information disclosure [94]. However, the solution is ineffective in IoT systems due to the excessive overhead that may generate high costs (more memory and processing power needed) when creating privacy tags for large and sensitive data [221]. In [124, 62], the authors propose approaches based on privacy policies based on anonymity to protect data using access control protocols. According to Huang et al. [124], users are able to share information using security levels and control what data is being collected by intended parties at all times. The information shared will be displayed to other users using a k-anonymity technology for a complete data generalization [124]. Unlike the previous static approach, Cao et al. [62] enhanced their results by introducing Continuously Anonymizing STreaming data via adaptive cLustEring (CASTLE). Unlike k-anonymity schemes designed for static data streams, CASTLE uses tuple clustering to ensure freshness (satisfies delay constraints) and supports a better attribute diversity. This approach uses QI (quasi-identifiers) to build space metrics and categorizes tuples based on their numerical and categorical attributes [62]. Su et al. [232] argue about the necessity of implementing a signature-based approach that acknowledges attribute policies to ensure user privacy. The authors of [232] developed ePASS, an innovative Attribute Based Scheme that uses Diffie-Hellman algorithm combined with an attribute tree to guarantee unforgeability and privacy for signers. This scheme uses AND/OR threshold gates and the tree public key to ensure that only users carrying policies with satisfying attribute can decrypt the messages. Further-
more, signers may hide their identity and remain undetectable among policy holders with the same satisfying attributes. In [196], the researchers focus on the protection of RFID and WSN system privacy by describing a mutual authentication protocol based on key-changed schemes. RFID tags are widely used in various industries due to their low manufacturing cost, so there is a need for a reliable security approach based on existing protocols and standards. Li et al.[155] proposed a scheme that not only protects user privacy but also mitigates against known RFID breaches such as replay attacks. The KMAP (Key-changed Authentication Protocol) proposed generates random numbers for both communicating readers and tags in real time when a query is requested. The generated query is sent from the reader to the server that uses a one-way hash and XOR functions to make sure that the information provided exists in the database and updates the system accordingly. To protect users from privacy invasions in a climate where sensors are omnipresent, Konings and Schaub developed “PriPref”, a privacy context awareness application. Through their application users can broadcast their privacy preferences in the surrounding environment in different scenarios for better coexistence [140]. To better assess user’s privacy concerns, the authors first conducted a survey to enlist common user’s disturbances and strategies used to handle them. Results show that loudness (music/noise), disturbance (typing, phone ringing etc...), nearby conversations, contacting and media (pures and video) where users main disturbance factors. Based on the survey results, Konings et al.[140] developed an application that displays the prevalent privacy preferences in neighboring environments. In fact, PriPref processes all users’ preferences and displays the current environment status to all participants for a better privacy effectiveness. Users can also enable dynamic privacy adaptation, where the app will automatically modify users’ phone setting to accommodate the current environment preferences. Usability results show that users were satisfied with the application performance and will most likely use it in a study or travel scenarios to avoid unwanted disturbance and awkward situations [140].
Blockchain (BC) is also a rising technology where intensive research work is conducted to fit IoT technologies due to its decentralized and security features. BC technology is mainly based on the concept of DLT (Distributed Ledger Technology), which is a combination of infrastructures and protocols that is able to simultaneously validate and record transactions through a network of trusted entities. Blockchain inherits its security and privacy aspects from the use of multiple authority vectors and POW (proof of work) where miners (trusted network users) solve a problem to confirm and add the block to the trusted chain. Blockchain is able to guarantee real time transaction updates, immutability and absolute anonymity over the network [173]. IoT researches such as Dorri et al.[90] implemented the blockchain approach in a smart home setting, but their approach could be replicated to work in industries as well. Their architecture is composed of a three-tier foundation: network overlay for efficient distribution, cloud storage for accessing/storing data and a home miner for managing secure communications. To meet the IoT requirements and to deliver faster services, the authors disposed of the POW services and used Cluster Heads (CH), elected through a group of nodes, for minimal network delays. To provide better privacy, transactions (Communication between nodes) can be periodically monitored by owners and safeguarded using a shared key. Lightweight hashing is also deployed to make sure that transactions have not been altered or manipulated by malicious devices. The local BC is responsible for adding/removing new devices (blocks) to the chain through the use of the hash of the previous block header for immutability and policy header for enforcing user control over devices. In fact, blocks are managed by smart miners that are responsible for authorization, authentication and shared key updates. The simulation done by Dorri et al. showed low time overhead and minimal power consumptions, which is acceptable to ensure optimal security and privacy [90].
2.8.4 Middleware Challenges and Solutions

A middleware refers to the software or layer within the IoT network that serves as an interface between smart objects while enabling the connectivity between diverse types of components. Researchers have proposed different types of middleware that focus on device interoperability with the common ones being SOCRADES [18] and LINKSMART [97]. SOCRADES and LINKSMART middleware both use an SOA architecture and are able to handle standard protocols and security at the application layer [94]. Thus, the heterogeneous nature of IoT requires the middleware layers to ensure security and privacy of exchanged information over the communication channel while allowing efficient transfer of information based on specific security policies [221]. In [78], the authors show that security and network challenges have led to the creation of VIRTUS Middleware, an event-based approach built on existing IoT standards. Conzon et al.[78] designed the VIRTUS Middleware with the intention of providing encrypted communications that utilize XMPP (eXtensible Messaging and Presence Protocol), which is an open XML technology widely used in smart devices. XMPP can easily accommodate the exchange of information among diverse smart things. VIRTUS is a Publisher-Subscriber approach that is able to discover other devices running on the same network and send notification when services are requested. This Middleware also offers secure authentication by creating an XMPP account for each module to uniquely identify each entity in the network. The VIRTUS middleware was deployed in multiple scenarios (energy monitoring, manufacturing and homecare) to test its scalability and security performance. Test results indicate that VIRTUS architecture was able to guarantee user privacy through storing private data locally and sharing public information, configured by the user, to authenticated devices only [78]. The authors of [102] propose Otsopack, a Triple Space Computing (TSC) middleware that provides Ambient Intelligence (AmI) and operates on multiple operating systems. This middleware uses HTTP to provide flexible exchange of information among devices. TSC is a coordination scheme that uses indirect communication through seman-
tics to share services. The content within an IoT application can be simply described in the shared space and stored in a graph. The data provided can then be accessed and used by other nodes when requested. Security is also a built-in feature of Otsopack as it uses OpenID [24] to authenticate users and providers and is able to restrict access to certain graphs. This AmI middleware is also able to autonomously work without any human interaction. As an example, sensors embedded in a company can monitor gas levels and automatically decide who to contact and what steps should be taken when the threshold is reached. Aitor et al. [102] case study demonstrates that Otsopack simplicity and use of HTTP enabled it to handle multiple platform (XBee sensors, FoxG20, etc...) and dynamically acclimates different features and scenarios. Additionally, Otsopack exhibits fast response time in embedded devices (600 milliseconds for 35 requests) which is considered sufficient for IoT system [102]. Various IoT communities and leading organizations are also working on developing a global middleware for machine to machine communication. OneM2M [97] in an ongoing project that is geared towards the design of a universal platform to unify the diversity that exists among IoT devices. OneM2M will act like a universal operating system where communication will be made possible across different technologies. This horizontal middleware will encapsulate existing protocols and incorporate security standards in its design. OneM2M will also support the communication with non oneM2M systems to enable the communication between different industries for optimal results [233].
Chapter 3

Introduction to Machine Learning

3.1 What is Machine Learning

The term Artificial Intelligence (AI) was coined around 1956 at the Dartmouth Summer Research Project [85]. Marvin Minsky and John McCarthy, considered the AI founding fathers, described advanced computer intelligence as "Every aspect of learning or any other feature of intelligence can in principle be so precisely described that a machine can be made to simulate it" [205]. In other words, AI can be defined as an intelligent system capable of autonomously interpreting data and flexibly learning from such data to perform complex associations that may be difficult for the human brain to discover [85]. Within AI, Machine Learning (ML) emerged as a subset of AI and was considered as the method of choice across several industries where data-intensive issues are a concern [129]. This rapidly growing technical field was formally defined in 1997 by Tom Mitchell [176] as "A Computer program is said to learn from an experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E.” The core concept of ML is that machines can learn independently without being explicitly programmed, which attracted the scientific community and made it one of the most promising tools of
AI [205]. Over the past decades, the study of ML has shifted from the effort of computer
scientists trying to explore how computers can autonomously play games and statisticians
ignoring computational considerations to an extensive discipline that generated fundamental
and accurate theories [177]. This technology revolutionized how algorithms are designed
and routinely used in commercial systems and altered how data mining might be interpreted
by discovering hidden regularities [177]. Machine Learning, considered as the outgrowth of
the intersection of Computer Science and Statistics, addresses the question of how to build
autonomous programs that improve gradually through experience [129]. In other words, ML
can be described as a self-monitoring system capable of continuously analyzing, retrieving,
and merging data patterns for optimum task completion [177, 129, 205]. Driven by different
optimization problems and industrial needs, several ML algorithms candidates have emerged
to find the best method to optimize tasks performance metrics [177]. Conceptually, ML al-
gorithms can perform classification tasks, pattern recognition, clustering, etc. [254]. ML
programs are trained using statistical models to analyze data and correlate extracted features
and labels. The data obtained during the training phase is then used to identify patterns and
make the corresponding decision/prediction [129].

### 3.2 Machine Learning Types

Several Machine Learning algorithms have been deployed to cope with different problems
and solve complex dilemmas [220]. Those algorithms have been broadly classified into three
categories based on their properties, learning characteristics, and how they handle data and
process it [34]: Supervised Learning, Unsupervised Learning, and Reinforcement Learning.
This type of classification is essential as it will help us identify the type of data pattern to be
used, the appropriate model for our application, and our output structure [43]. In the next
section, we will explore each ML type and its associated algorithms to determine the best
• **Supervised Learning:** Supervised Learning is a Machine Learning task that infers a function mapping labeled input data to an output based on example pairs. It uses a labeled training data set to classify and predict future events using output data patterns known to the system [43]. Supervised Learning aims to generate a model that relies on input-output patterns to predict an outcome based on a class type [220]. This type of ML needs external assistance to divide the input labeled data into training and testing data. In the training phase, we use known data sets with their corresponding labels (features) to train our model and produce a function that will be used on the test data for prediction or classification [220]. The learning algorithm can also compare its outputs with the correct intended data classes (Truth Table) to find errors and correct itself accordingly via backpropagation [168]. After sufficient training, the trained model will make predictions on unseen observations and provide targets for new inputs. The workflow of a Supervised Learning approach is explained in figure 3.1. Supervised learning techniques are best used for Regression (Predicting real continuous values such as stock prices) and classification (Predicting the category of an output variable) problems [168]. Known algorithms in Supervised Learning are Support Vector Machines (SVM), Linear/Logistic Regression, K-Nearest Neighbors, Decision Trees, etc. [189].

• **Unsupervised Learning:** Unsupervised Learning is a Machine Learning task that infers a function capable of discovering a hidden structure from unlabeled input data [168]. Unlike Supervised Learning, there is no need for labeled and classified datasets. This self-organized learning technique explores the underlying patterns from the input data and provides a prediction. In Unsupervised Learning, there is no need for human intervention, and there are no specific recommendations [83]. Instead, the system ex-
explores raw data, draws inferences from the input dataset to depict common structures and features, and clusters them coherently. When new data is introduced, the model uses the previously clustered features to recognize the class of the data and make a prediction based on a collection of perceptions [100]. Figure 3.2 illustrates the workflow of an Unsupervised Learning approach. Unsupervised Learning techniques are best used for clustering (Unveiling inherent grouping in data), feature reduction (reducing the number of unnecessary features), and associative rule mining (discovering association rules) problems. Known algorithms in Unsupervised Learning are K-mean clustering, Neural Networks, Apriori, Anomaly detection, etc.[168].

- **Reinforcement Learning**: Reinforcement Learning (RL) is a Machine Learning technique that uses agents to learn interactively from its environment using the trial-and-error approach [82]. Unlike the previous approaches, Reinforcement Learning uses rewards and punishments as signals for positive and negative agent behavior [131]. In fact, agents can dynamically interact with their environment by moving from one state to another and getting rewarded on successful tasks or penalized on failures. In this way, an agent learns through delayed feedback until it finds a suitable/ideal path.
to maximize its cumulative rewards within a specific context and enhance the model performance [82]. Reinforcement Learning algorithms are mainly used in the field of robotics, industrial automation, aircraft control, gaming, etc. [168]. These learning techniques deal with exploration/exploitation, value learning (Q-Learning), model-based Learning (using models to solve problems), and Markov’s decision processes (MDP) [131, 241]. Almost all RL problems can be formalized using MDP. Figure 3.3 illustrates the action-reward feedback loop of an MDP where the Agent and Environment interact in a discrete, timely manner (t=0,1,2, 3...). This method works as follows: an Agent observes a state ”S” at a time ”t,” produces an Action ”A,” collects a resulting reward ”R” at t+1 and moves to next state ”S (t+1)” [82].
3.3 Introducing IoT to Machine Learning

The recent evolution and spread of ML have been driven by the development of new learning techniques and algorithms but primarily by the rapid increase of available online data and low-cost computational mechanisms [177]. Over the last decade, industrial companies have been competing to collect more data and pushing the scientific community to propose data-intensive ML methods, leading to better decision-making and enhanced predictions through data translation [177]. Currently, applications based on mobile devices, embedded systems, and wireless sensors have become smarter, enabling improved communication and enhanced task executions among computerized devices [254]. With the number of IoT devices increasing ceaselessly, surpassing the world population [223], the amount of data generated and collected by those devices escalated exponentially. In fact, traditional data collection and processing techniques may be outdated and unable to scale with the heterogeneity of the data generated [125]. Hence, to harness the potential of the IoT technology, new mechanisms are needed to make meaningful data correlation and advanced decision making. Thus, there is an urgent need to develop intelligent IoT devices able to create automated applications with appropriate resource allocation and communication to infer valuable knowledge.
In this context, the convergence of IoT systems and ML paves the way for a sturdy improvement in the IoT infrastructure through enhanced efficiency, accuracy, and productivity [220]. Providing IoT with embedded intelligence (ML) can help smart devices deduce proper knowledge from a stream of data and vary their behavior based on the surrounding environment for improved controllability [28]. Combining IoT and ML has an enormous potential in improving human life quality and helping industrial applications grow significantly. Advanced machine intelligence techniques have made data mining from IoT devices possible and delivered better insight into a wide range of complex real-world problems as well as the ability to make crucial decisions [28]. Multiple industrial fields have already deployed computer vision, malware detection, bioinformatics, privacy preservation, and behavior prediction applications to solve complex computational problems [129]. The following section explores how IoT leverages ML to identify privacy preferences through behavior prediction.

3.3.1 Machine Learning and Privacy Preferences

In section 1.1.3, we examined prior solutions on managing IoT privacy preferences [140, 130, 235, 163]. Still, we did not investigate the capabilities of integrating ML algorithms to offer better resolutions for the IoT systems. This section examines relevant prior work focusing on integrating ML approaches for managing privacy preferences, which may help us choose the right approach for our Privacy Preference Manager (PPM).

Both Android and iOS platforms rely on App permission mechanisms to control users’ privacy settings and preferences [162]. The end result is an overwhelming number of decisions with an average of 100 permissions [2] the user is expected to make. To better help users assess their Privacy needs and manage their App permissions, Liu et al. [162] proposed a Personalized Privacy Assistant (PPA) for Android mobile users. The researchers designed a personalized Privacy profile approach that accesses users’ installed Apps to engage with them in a specific manner collecting their perceptions on specific privacy permissions. Using
the collected information, Liu et al. were able to cluster users in different privacy profiles and capture their permission preferences based on different App categories. On the server-side, the researchers extracted various features from the privacy profile and built an SVM (Support Vector Machine) classifier to generate privacy recommendations for new users. The SVM model was trained using a five-fold cross-validation approach and utilized the assigned profile, the application category, and the permission settings as features to predict the user privacy permission behavior [162]. New users were then prompted to answer a few tailored questions about their privacy preferences to assign the correct Privacy profile. Using these specific profiles, the PPA recommended some privacy setting changes and dynamically adapted if the recommendation was denied. Despite the small number of users (72 participants), PPA achieved 78 % recommendation accuracy that users are likely to adopt. PPA also led to more restrictive privacy settings changes to help users better control their permission settings without jeopardizing their comfort level [162].

As shown in section 4.3, privacy preferences differ between individuals and account for different context and scenario parameters. To cope with this problem, Wijesekera et al. [245] studied the feasibility of dynamically granting privacy permissions based on the contextuality behind user past privacy decisions and behavior. Previous mobile privacy permission approaches such as Ask-On-First-Use (AOFU) and Ask-On-Install (AOI) showed to be ineffective as they do not account for circumstances under which users made specific privacy choices [244]. Those approaches could violate privacy preferences and defy users’ privacy expectations [245]. The researchers’ approach consisted of using the Experience Sampling Method (EMS) to continuously ask participants about recent privacy permission events and whenever they would have approved or denied that data collection if given a choice. Using those EMS probes allowed them to capture participants’ behavioral traits, past privacy preferences, and contextual circumstances surrounding their privacy choices. Using the collected data from 131 Android users over 32 days, Wijesekera et al. [245] designed a ML model
over the last decades, the introduction of diverse IoT devices and infrastructures with restricted user interfaces has deepened users’ burden in making the appropriate privacy decisions. To that end, Lee and Kobsa [152] surveyed 172 participants to study the feasibility of modeling and predicting users’ privacy preferences in a Campus-wide IoT environment. To collect participants’ privacy preferences in different IoT environments, the researchers used the EMS approach through a Google Glass application to simulate different scenarios around campus buildings. The authors collected 33090 data points that were analyzed and used to cluster participants, using the K-mode clustering algorithm, in four groups based on the similarities of their privacy choices. Through their clustering approach, Lee and Kobsa [152] were able to evaluate the impact of different contextual factors, such as ”where,” ”what,” ”who”, and ”purpose”, on different users’ categories based their desire for privacy control and potential privacy risks. Consequently, the researchers built a Machine Learning model using inference trees from the collected data of their initial survey. They used the extracted contextual feature and the clustered privacy profiles to train their model using a 10-fold cross-validation approach. The model was able to predict, with an accuracy of 77%, users’ privacy decisions on either allowing or denying data monitoring in given IoT scenarios [152]. The contextual factors were limited and focused on one location (Campus). Another drawback is that participant recruitment was demographically tied to campus students only, resulting in
sampling bias that may not necessarily reflect the general population’s privacy opinion.

Naeini et al. [179] study was similar to [152] but accounted for more features, included different location categories, a more diverse participant population, and different factors interactions [33]. The researchers first conducted a large-scale vignette study with more than 1000 participants to study the contribution of eight different environmental IoT factors on user privacy expectations [179]. They also studied the impact of those factors’ interactions on users’ comfortableness and willingness regarding data collection in 14 different realistic and futuristic scenarios. After analyzing the most pertinent factor interactions (GLMM) that may impact users’ privacy preferences, Naeini et al. [179] built a ML model able to predict users’ privacy preferences. In regard to users’ willingness to accept or deny data collection, where the classification is binary, they used Logistic/Linear Regression, K-Nearest Neighbor, SVM, and AdaBoost to train their model with 75% training data. The AdaBoost classifier using Scikit Python library [195], with a logistic regression base, yielded the best results with an average accuracy of up to 86% for comfort level [179]. The main limitation in this study is that some of the factors used, such as sharing details, retention time, and benefits, may not be necessarily controllable by the users and will not reflect real-world IoT scenarios. As seen from [162, 245, 235], users desperately need a tailored privacy preferences experience, built on ML approach, unique to their privacy expectations and surrounding environment and context. However, their research cannot be directly tied to IoT technology due to their heterogeneity, scalability, and dynamically changing ecosystem. Naeini et al. [179] and Lee et al. [152] showed that even though it is difficult to automatically detect precise IoT context and user privacy behavior, IoT privacy can be enhanced using environmental factors and Machine Learning. To conclude, the above-mentioned research stresses the need to use ML technique to potentially improve users’ privacy preferences and control their personal data by interacting with their constantly changing surroundings.

In the next section, we will explore the potential ML algorithms that might help us build
our PPM system from the collected data in our replication study [33]. This will help us assess the most suitable features and procedures to be used in our actual proposed solution.

### 3.3.2 PPM and Machine Learning

After exploring different Machine Learning techniques applicable to users’ privacy preferences from the literature [162, 245, 152, 179] and examining different Machine Learning types in section 3.2, we decided to focus our efforts on Supervised Learning. This choice stems from the fact that our collected data is already labeled and maps to an output pattern already known to the system. Supervised Learning techniques are also best suited for classification problems in general and Binary categorization in particular. In our preliminary testing phase, we are going to map labeled data containing various extracted features to a binary output function (Allow, Deny) capable of predicting users’ privacy preferences. This procedure will help us grasp the necessary information needed (advantages and deficiencies) to collect a newly improved data set and design a ML model that will provide users with a suitable recommendation instead of a binary classification. Next, we are going to examine different Supervised Machine Learning Algorithms to test our initial approach.

- **Support Vector Machine:** The first SVM algorithm (Linear classification) was introduced by Vladimir N. Vapnik and Alexey Chervonenkis in 1963 and was considered an essential part of the computational learning theory [220]. SVM is regarded as one of the most commonly used supervised learning techniques for binary/multi-class classification and regression analysis [220, 168]. The SVM algorithm works by mapping data points into an "n" dimensional space and classifies them into different groups using an "n-1" dimensional hyperplane approach. To separate classes, the SVM algorithm draws a plane with a maximum margin between datapoints for an optimum gap width, hence minimizing the classification error [220, 167]. Figure 3.4 illustrates an
example of a 2-dimensional SVM approach where line A depicts the maximum margin between the two groups, and the data points close to the separation line are called support vectors. Support vectors are significant components of the SVM structure, as they influence the position and orientation of the hyperplane [204]. In 1992, Vapnik et al. [54] introduced non-linear SVM classification, also known as the "Kernel Trick." This proposed technique transforms non-linear input data to a higher dimensional feature space (Hilbert Space), where the transformed data becomes linearly separable [238]. SVM algorithms are prevalent in classification and categorization problems and perform well with overfitting issues [204].

- **Logistic Regression**: Logistic Regression is a statistical model introduced by Joseph Berkson in 1994 [79]. Unlike linear Regression that generates continuous variables, Logistic Regression is a classification algorithm that assigns observations based on a discrete set of classes [189]. Logistic Regression can perform binary classification as well as multi-class categorization using a Sigmoid function that maps real values to a
probability estimate between 0 and 1. The cost function is then used to classify data into different categories [220]. The mathematical function for Logistic Regression is given by: \( h(x) = S(a_0 + a_1x_1 + a_2x_2 + \ldots + a_nx_n) \), where \( S() \) is the sigmoid function given by \( S(z) = \frac{1}{1+e^{-z}} \) and illustrated in Figure 3.5 [79]. The Sigmoid function described in Figure 3.5 and represented by a "Sigma-like" line uses the dot product of vector parameters to map the resulting probability value to a prediction [79]. Logistic Regression algorithms are computationally efficient, provide high accuracy for binary classification, and are not affected by data noise [204].

**Decision Tree:** The concept of decision trees originated in 1963 at the University of Wisconsin and was formally introduced in 1984 by Breiman et al. [58] with the CART software [103]. Decision trees, a flow chart structure, work by classifying instances based on their feature values [143]. Each node in a Decision Tree describes a feature in an instance to be classified, and each branch represents a value assigned to that particular attribute. Decision trees classify instances at the root node and sort them using their
feature weight [189]. This process is repeated on each node subset recursively until each node is assigned the most common class of the training instance. Pruning is a good technique for evaluating the performance of a Decision Tree as it removes nodes that do not provide value to the prediction target, thus reducing its size [174]. Figure 3.6 represents a Decision Tree model example where each root node denotes a test on a particular attribute, and each tree branch represents an outcome value. The same process is repeated at each leaf node level until the data is divided into subsets of the same class [143]. Decision trees are efficient Supervised Learning techniques widely used for solving classification and regression problems, do not require any parameter settings, and are capable of dealing with missing values [204].

Figure 3.6: Decision Tree example

• **Random Forest**: Random Forest (RF) is a supervised learning procedure coined by Leo Breiman in 2001 [57]. The development of the RF algorithm was greatly in-
fluenced by the work of Amit and German [36] (Random geometric features), Ho (Random Subspace) [116] and Dietterich (Random split theory) [87]. RF is a simple yet powerful technique based on the principle of divide and conquer approach [53]. RF classifiers are a combination of several individual Decision Trees that are trained simultaneously with bootstrapping and aggregation, also known as bagging [160]. Bootstrapping ensures that each DT is uniquely trained in parallel using a different subset of the training data. Consequently, each subtree casts a vote for the most relevant class and aggregates those results to get the final model [160]. The final result is deduced through a majority voting system based on the following function: $H(x) = \arg\max_Y \sum_{i=1}^{k} I(h_i(x) = Y)$. $H(x)$ represents the combination of multiple classifier instances and $h_i$ describes a single tree model [57]. RF bagging mechanism is considered one of the most computationally intensive methods that improve unbalanced estimates and yield high classification accuracy without overfitting issues [160, 53]. RF algorithms are commonly used as they can be applied to a wide range
of prediction problems. They are efficient in regression and classification due to their flexibility and tuning parameters simplicity [160]. As shown in figure 3.7, RF is considered as an Ensemble technique where multiple Trees are trained in parallel and cast a vote for the most popular class. RF then constructs a predictor from each Tree category and determines the most suitable prediction.

- **XGBoost**: Gradient Tree boosting, also referred to as Gradient Boosting Machine (GBM), is an optimization technique that minimizes the loss function by using multiple iterations of weak learners [213]. The idea is built around gradient descent, the first iteration of the optimized algorithm applied on the weak classifiers. As weak learners train on the previous residual errors of the prior learners, misclassified observations are given more importance and weight for the subsequent iterations [181]. Intuitively, in the next cycle, new learners are trained in areas where existing learners performed poorly until the overall error function is minimized. Decision trees are added one step at a time until the loss function no longer improves the external validation process [181]. XGBoost (eXtreme Gradient Boosting) is an extension of GBM and was designed to improve the performance and scalability of tree boosting [71]. XGBoost was created by Chen and Guestrin in 2016 [71] and gained a lot of momentum after dominating the Kaggle competitions (17 winners used XGBoost) [213]. The success of XGBoost stems from the fact that several systems and software refinement were applied, making it ten times faster than traditional tree approaches [71]. Chen et al. optimized their algorithm through an efficient split finding approach capable of handling spare data using node default direction and managing instances weight using merge and prune techniques [71]. Another important refinement applied to XGBoost is the addition of regularization and cross-validation, which aims to create a more sustainable and efficient loss function [71, 213].
3.4 Machine Learning Performance

Improving the performance of a Machine Learning model might be challenging when we cannot compare our results to similar models. There is no specific strategy that can be constantly applied to outperform current results as ML is still in its early stages. In this case, research, data analysis/visualization, and testing are critical components in re-structuring and enhancing the accuracy of our ML approach. When building a supervised ML model, it is essential to understand how different algorithms operate and how they make use of the data and features provided. Analyzing the result and comprehending the distribution and correlation between data points is critical in formulating the correct hypothesis to improve the chosen model. In this section, we will explore different techniques that may help us improve our future model and deliver optimum predictions:

- **Exploratory Data Analysis:** Data is a fundamental artifact when building a ML model. Researchers should be able to ensure that the data is consistent, free from redundancies, and find ways to deal with missing values [56]. This preprocessing step is essential in establishing a solid base of high-quality raw data as ML algorithms are susceptible to noisy data [145]. The next step should be to apply different ML algorithms to determine the most suitable approach for our particular data set. Another important step is to visualize our results and use statistical analysis, variable correlation, and variance analysis to find potential anomalies.

- **Feature Engineering/Creation:** Feature Engineering is the task of constructing a suitable feature from one or multiple given features [180]. Creating new features helps extract more information from existing data and improve the predictive model performance. Feature Engineering is a convenient way to generate and scale data feature space to create new dependencies and patterns. This process can be accomplished through categorical transformation, mathematical formulation, and numerical classifi-
cation [180]. Feature engineering relies on an iterative trial and error approach as ML models can build hidden patterns when training the data [56].

- **Feature Selection:** Feature selection is a well-known technique that became popular in the last decades for dealing with high-dimensional datasets [61]. It became an essential part of data processing and indispensable to Machine Learning classification improvement [106]. Feature selection is simply the process of selecting relevant subsets of the original feature set based on different criteria [123]. Feature selection techniques can preprocess ML algorithms and automatically test different feature combinations to improve predictive accuracy and comprehensibility [69]. In fact, Feature Selection selects the most suitable feature combination by eliminating noise and irrelevant/redundant data that degrades the learning process [147]. Features are chosen through data correlations and dependencies as well as Euclidian distance measurements [61]. Multiple Feature selection techniques have been developed for finding the ideal subset collection. Among those methods, Recursive Feature Elimination (RFE) has shown promises when dealing with small sample classification problems [107]. RFE reduces the model complexity through removing weaker classifiers based on feature importance until it reaches peak performance [72].

- **Data Augmentation:** The main issue with small data set is the existence of significant gaps between data samples that may not provide sufficient information to the ML algorithm to build strong patterns [187]. Small data sets may also exhibit discrepancies in data distribution between classes and make the dataset unbalanced [115]. Therefore, when dealing with small and unbalanced datasets, there is a need for a mechanism that provides high accuracy for minority classes without jeopardizing the accuracy of the overall model. In this case, data analysis through assessment matrices and precision-recall is essential in building a robust dataset. Data augmentation is the process of
generating artificial data samples from the original dataset to cope with weak classifiers and improve the performance of the ML model [187]. Previous research has shown that balanced datasets built upon data augmentation have improved the overall classifier distribution and performance [242, 150, 93]. Nonetheless, data augmentation can be challenging as it may distort and transform data labels [115]. The key solution to dealing with those specific issues is finding and targeting weak classifiers with random oversampling until the specific distribution is adjusted [242].

• **Data Normalization:** Data normalization is a commonly used technique that proved its importance in improving data quality and subsequently in boosting Machine Learning performance [222]. Normalization is the process of scaling or transforming the data in a commonly known range to reduce the domination of high numerical feature values. The main objective behind this approach is to reduce the bias between different features (decrease pattern class discrimination), making features contribute equally to the learning process [99]. Multiple research has proven the utility and impact of data normalization in increasing the classification performances of numerous Machine Learning algorithms [29, 226, 120]. Over the last decades, many normalization techniques have been proposed based on different statistical measures [222]. A well-known technique that works along categorical classification is the Min-Max approach [194]. This method rescales unnormalized data using lower and upper bounds linearly [109]. The following equation \( X'_{i,n} = \frac{X_{i,n} - \min(x_i)}{\max(x_i) - \min(x_i)}(nMax - nMin) + nMin \) denotes the principle behind Min-Max scaling, where \( \min/max \) represent the minimum and maximum \( i^{th} \) feature and \( N\min \) and \( N\max \) indicate the rescaled data boundary [120].
Chapter 4

Replication Study

4.1 Description

In order to thoroughly achieve the potential of IoT Technology, individuals should be fully aware of sensor interactions and data captured during communications to make knowledgeable decisions. To do so, smart devices should always inform their users about captured private information and respect their preferences and decisions. Solving these problems requires a complex understanding of users’ social norms, culture, context, and education as privacy needs is perceived differently between individuals [185]. This dilemma even worsen when multiple devices communicate together and make unpredictable interactions. The first step in resolving this issue is trying to understand and identify the contribution of different factors that might impact users’ choices in an IoT environment. Thus, we conducted a replication study of the Emami-Naeini paper entitled “Privacy Expectation and Preferences in an IoT World” [179]. We choose this Vignette study as it was built upon prior work conducted by Lin et al.[157] and Liu et al. [163], showing that it was possible to accurately predict users’ privacy preferences in a mobile app domain.
This replication study will allowed us to:

- Understand user’s feeling (Comfort Level and Allow/Deny) concerning data collection.
- Analyze and evaluate the contribution of different factors that may impact users privacy choices.
- Compare our findings with the original study conducted in 2017 by Naeini et al. [179] to see if users privacy decisions have evolved over the years.
- Design a more transparent and informative IoT privacy preference framework that will be used in our future work.
- Build a privacy preference model that will be considered as the backbone of Phase 2.

4.2 Methodology

In this section, we will describe in detail the steps we overtook to reproduce the scenarios and methodology used by Naeini et al. [179]. After conducting extensive research, it is worth mentioning that we could only find two scenarios attached to this research.

- Step1: Design of The Replication Study

  We created eight realistic scenarios and four different versions to diversify the possible outcome of our vignette study. Each of those 32 scenarios were crafted to mimic real-world examples in a specific IoT environment. Within those vignettes, we varied six different factors that could impact the user’s privacy preferences. These factors are as follows: Location, Data Type, Device Type, Retention Time, Sharing details, purpose. These factors were considered to be important in previous research when combined together [132, 172, 151, 152]. Integrating the six factors will allow us to
simultaneously study their impact and importance in participants’ decisions regarding privacy. To make our scenarios reliable, we tried to introduce these factors in the same order as not to confuse the participants. The following example is a scenario from version 1 of our replication study: “You are at a department store. This store has product sensors that track customer purchases as they shop around the store. The data is shared with the customer to facilitate the checkout process. The collected data will not be deleted”. All the factors mentioned above have different levels, as explained in Figure 4.1, and randomly combining them to create scenarios may not be feasible as it may not reflect legitimate examples. To overcome this problem, we made sure to select each factor level individually and carefully include all possible combinations. We also tried to have different factor levels in each of our four versions to minimize redundancy and cover all possible interactions. Each subset of our study comprised four parts as follow:

Sample survey Scenarios: Participants were asked how comfortable they were with collecting their personal data given a particular scenario, whenever or not the use of their data is perceived as beneficial for them, and how often they want to be notified about the collection of their data. All those questions were coded using a five-point Likert scale from “Strongly Agree” to “Strongly Disagree”. Participants were also asked about their willingness to allow or deny data collection given a set of factors.

Summary Questions: Bearing in mind all eight scenarios, participants were asked how often they would like to see a summary of their collected data and what factors most impact their comfort level when sharing their data.

IUIPC Questions: Internet Users’ Information Privacy Scale is used to gauge participant privacy concerns and focuses on control, awareness, and collection [171].

Demographic Questions: Participants were asked about their age, gender, educa-
tion, and income, which will be compared with the US average.

<table>
<thead>
<tr>
<th>Factors</th>
<th>Levels</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Location</td>
<td>Department Store, Friends House, Home, Library, Public Restroom, Workplace</td>
<td>The location of the data being collected</td>
</tr>
<tr>
<td>Data Type</td>
<td>Biometrics, Tracking, Temperature, Video</td>
<td>The category of the data being collected</td>
</tr>
<tr>
<td>Device Type</td>
<td>Camera, Fingerprint Scanner, Facial Recognition, Iris Scanner, Presence, Smart Phone, Smart Watch</td>
<td>The type of device collecting the data</td>
</tr>
<tr>
<td>User Benefit</td>
<td>Collector, User, Both</td>
<td>Who benefits from the data being collected (Determined by the researchers)</td>
</tr>
<tr>
<td>User Perceived Benefit</td>
<td>Beneficial, Not Beneficial</td>
<td>Participant perception of whether the scenario would be beneficial to them</td>
</tr>
<tr>
<td>Retention Time</td>
<td>Forever, Purpose Satisfied, Unspecified, Week, Year</td>
<td>How long the data will be kept</td>
</tr>
<tr>
<td>Sharing</td>
<td>Mentioned, Not Mentioned</td>
<td>Are data sharing details mentioned in the scenario</td>
</tr>
<tr>
<td>Purpose</td>
<td>The purpose of collecting the data</td>
<td>The purpose may be mentioned or not and will be used to determine User Benefit when creating models</td>
</tr>
</tbody>
</table>

Figure 4.1: Factors used in our replication study, their levels, and detailed explanation

- **Step 2: Recruiting participants**

Unlike the original research that used 1014 Mechanical Turk workers, we tried to advertise our vignette through convenience sampling methods such as Florida Tech mailing lists, personal invitations, word spreading, and snowball sampling methods. To attract more participants, we added an incentive stating that participants will have a chance to enter a draw to win one of four Amazon gift card upon completion. People of all ages, genders, and backgrounds were welcome to join our study as long as they are over 18 years of age and reside in the United States. We were able to gather 77 participants across a whole semester, each exposed to eight different vignettes presenting numerous IoT collection scenarios for a total of 616 data points.

- **Step 3: Data gathering and Cleaning**

All four versions were created using Google forms service as it presents a convenient way to craft the scenarios and store the answers on an excel sheet. The totality of
the responses were kept on Google servers until the survey was completed and will be moved and held on Florida Tech facilities when the research is concluded. Additionally, all Gmail accounts associated with the survey were deleted for security and privacy purposes. After gathering all the data, we used “Notion.so” to clean and categorize it by factor of interest as shown in Figure 4.2. We did not use any scripts for data segmentation as the results were manageable, and we wanted to make sure that all data input was reliable and legitimate. We also used different data scale coding than the original study to make results more significant and understandable. Data scaling specifics will be explained in detail in the next section, along with model creation and analysis.

- Step 4: Predicting preferences

We used Generalized Mixed Effect Model (GLMM) with a random intercept per participant to construct our models [95]. GLMM is a useful statistical approach for repeated measurement on the same person and very flexible when it comes to studying the interactions of different factors and their dependencies [95]. Using this approach enables us to find the best interactions between various factors, thus finding the best model in our research. GLMM also uses the Bayesian Information Criterion (BIC) for model selection. BIC, derived from Gideon Schwarz [182], is a widely used statistical approach that balances the number of parameters and data points against the maximum likelihood function and measures the efficiency of parameterized models when predicting the output. BIC is calculated as follows: “K*Ln(n) – 2*Ln(L)” where “K” denotes the number of parameters, “n” the number of data points, and “L” the maximum likelihood of a function [182]. A lower BIC always indicates a better model where δBIC has to be above six between models to be considered a significant improvement [179, 206]. We used R programming with the Lme4 package to construct our models, giving us access to most linear models. Regression tables will be pre-
sent in the next section using a significance threshold of 0.05 to determine the most significant factors and their interactions.

4.3 Result and Analysis

This section will first introduce the general distribution of the comfort levels across different factors and then present results regarding various model selections. This will be done by analyzing numerous models and trying to classify factor interactions regarding user decision to allow/deny data collection and their comfort level in different scenarios. Factor categorization and interactions will be used as a reference when we develop our final model in future research.

- **The impact of data collection on Comfort Level**

  The first step in understanding user perception of privacy and what individual factors most impact their decisions in an IoT space was to comprehend their comfort level
regarding the collection of their data in different scenarios. Participants were asked to answer a five-point Likert scale ranging from “Very Comfortable” to “Very Uncomfortable”. To further understand participant behavior, we asked them to express their comfortableness regarding the same scenarios if no retention or sharing details were given. Figure 4.3 illustrates our results that emphasize on four different categories, each encompassing different levels as follows: Data type, Device type, location, and Retention time. Figure 4.3 shows that Biometrics and Video (41% and 42%, respectively) were the most influential factors in Data Type that users feel very uncomfortable sharing. This result could be linked to the fact that Biometrics and Video are considered identifiable and sensitive information tied to the user persona. When it comes to device type, smartphones outperformed all other devices scoring 79% uncomfortability level. Around 3.9 billion smartphone users worldwide use their devices for more than 4 hours a day, capturing a combination of identifiable information (Fingerprint, Facial, Iris, etc.) and location [188]. Our survey clearly indicates that even though phones are a necessity, users are aware and concerned by smartphones capturing their sensitive data continuously. Regarding location, users were uncomfortable with data being collected in sensitive places such as restrooms (44% of the users were very uncomfortable). When it comes to retention time, we noticed that most users were very uncomfortable when data was retained for no reason. Rather, they prefer data kept until the purpose was satisfied and then deleted. We also noticed that users tend to be more conservative about sharing their sensitive data if no sharing detail or retention specification is stated. The second table of figure 4.3 portrays a slight increase in uncomfortableness toward all factors as users are unaware to whom their data will be shared with or how long it will be used for. In particular, we noticed a measurable increase towards biometrics as data type and department store as location. We can hypothesize that this outcome could be tied to the fact that users consider biometrics
as sensitive data that could be used for identity theft and mistrust commercial entries that own such stores.

This preliminary study conducted five years apart from the original research revealed somewhat different results, especially toward smartphones (54% vs. 25%), tracking (33% vs. 11%), and workplace (37% vs. 17%). This might be related to the fact that our study was focused on participants with Engineering background and privacy knowledge. Additionally, privacy and security awareness has evolved over the years as users are more aware of technology’s risks. The smartphone industry has grown exponentially over the past few years and incorporated various sensing technology such as fingerprint scanners, facial/iris recognition, and tracking, which made users’ sensitive information exposed to third-party partners and various leak incidents. This feeling of vulnerability combined with users not having the choice of filtering their sensitive data due to the lack of transparency made them more aware of the risk associated with smartphones. Those smart devices can be tightly related to an IoT ecosystem encompassing various sensors capturing information continuously and communicating with other unprotected entities. The obtained results are a good indicator that there is an urgent need to involve user’s privacy preferences and make them the sole decision makers when disclosing or sharing their private data.

• **Factors affecting Comfort Level**

Using the GLMM regression model, we were able to order various factors and their dependencies using the change in BIC to study their impact and contribution on user’s comfort level. Figure 4.5 illustrates the top 12 factors ordered by their BIC size (From the most influential factor combination to the lowest). It can also be seen from figure 4.4 that the grayed-out columns represent the statistically significant dependencies (p <= 0.05), where a positive estimate depicts a tendency toward comfort and a nega-
Figure 4.3: Relation between Data collection and users comfort level
tive estimate an inclination toward discomfort. The model built indicated that User Perceived Benefits was the factor that most affected users toward being comfortable sharing their personal information with a BIC value of 604.9. This comes in line with some comments gathered from the survey indicating that participants were comfortable sharing any type of data as long as they see a valuable benefit from the process (incentive). Location and data type factor combination also came in our top 3 regression models with a BIC of 680.3 and five significant dependencies. Scenarios in which Biometric/video was involved with sensitive locations (according to the participant’s perception) such as public restrooms or friend’s house had a considerable negative impact on participants comfort level ($P = 0.00$, coefficient $= -8.7$). This outcome also confirms our preliminary quantitative analysis that will be used to categorize different factors in our future work. Based on our model analysis illustrated in Figure 4.5, we found out that users are less comfortable when their data is kept or stored for no particular reason. Instead, users prefer sharing their data, knowing that it will be deleted when the purpose is satisfied, especially in scenarios where their information is shared with law enforcement or the government. We can hypothesize that participants are aware of the risks associated with sharing their sensitive information that may lead to information leakage or unauthorized usage of their data, which clearly explains the mistrust between the parties concerned (provider and consumer). Our GLMM model also confirmed the privacy paradox phenomenon where users behave irrationally since they do not have the choice in an IoT environment. It should also be pointed out that in many countries IoT standard regulations are non-existent, which makes implementing privacy a choice, not an obligation. Conclusively our quantitative results revealed that User Perceived Benefit, Location, and Datatype are the top 3 factors that most affected users’ perception concerning comfort level and will be used to develop our future model where transparency is the primary key.
<table>
<thead>
<tr>
<th>Factor</th>
<th>Estimate</th>
<th>Std. Error</th>
<th>Z-value</th>
<th>P-value</th>
<th>BIC</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>User Perceived Benefit</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>604.9</td>
</tr>
<tr>
<td>Reference : Beneficial</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Not Beneficial</td>
<td>-2.80</td>
<td>0.30</td>
<td>-9.26</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td><strong>Location + User Perceived Benefit</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>617.3</td>
</tr>
<tr>
<td>Reference : Workplace+Beneficial</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Library:Not Beneficial</td>
<td>-0.63</td>
<td>0.94</td>
<td>-0.67</td>
<td>0.50</td>
<td></td>
</tr>
<tr>
<td>Department Store:Not Beneficial</td>
<td>-0.19</td>
<td>1.21</td>
<td>-0.15</td>
<td>0.87</td>
<td></td>
</tr>
<tr>
<td>Public Restroom: Not Beneficial</td>
<td>-3.71</td>
<td>1.23</td>
<td>-3.02</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>Friend house: Not Beneficial</td>
<td>-1.72</td>
<td>1.00</td>
<td>-1.71</td>
<td>0.08</td>
<td></td>
</tr>
<tr>
<td>Home: Not Beneficial</td>
<td>0.31</td>
<td>1.04</td>
<td>0.3</td>
<td>0.76</td>
<td></td>
</tr>
<tr>
<td><strong>Location + Data Type</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>680.3</td>
</tr>
<tr>
<td>Reference : Temperature + Library</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Biometric: Workplace</td>
<td>4.42</td>
<td>1.25</td>
<td>3.52</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>Presence: Workplace</td>
<td>0.91</td>
<td>1.29</td>
<td>0.7</td>
<td>0.47</td>
<td></td>
</tr>
<tr>
<td>Biometric: Department Store</td>
<td>-17.09</td>
<td>2202.30</td>
<td>-0.008</td>
<td>0.99</td>
<td></td>
</tr>
<tr>
<td>Presence: Department Store</td>
<td>-0.28</td>
<td>1.23</td>
<td>-0.23</td>
<td>0.81</td>
<td></td>
</tr>
<tr>
<td>Biometric: Public Restroom</td>
<td>-6.57</td>
<td>2.17</td>
<td>-3.01</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>Video: Public Restroom</td>
<td>-8.7</td>
<td>2.07</td>
<td>-4.19</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>Biometric: Friend house</td>
<td>-2.84</td>
<td>1.96</td>
<td>-1.44</td>
<td>0.14</td>
<td></td>
</tr>
<tr>
<td>Presence: Friend house</td>
<td>-5.12</td>
<td>2.08</td>
<td>-2.45</td>
<td>0.01</td>
<td></td>
</tr>
<tr>
<td>Video: Friend house</td>
<td>-5.79</td>
<td>2.05</td>
<td>-2.81</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td><strong>Location + User Benefit</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>687.5</td>
</tr>
<tr>
<td>Reference : Workplace + User</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Collector:Library</td>
<td>2.24</td>
<td>1.07</td>
<td>2.07</td>
<td>0.03</td>
<td></td>
</tr>
<tr>
<td>Both:Library</td>
<td>2.68</td>
<td>1.09</td>
<td>2.46</td>
<td>0.01</td>
<td></td>
</tr>
<tr>
<td>Collector: Department Store</td>
<td>0.68</td>
<td>1.16</td>
<td>-0.59</td>
<td>0.55</td>
<td></td>
</tr>
<tr>
<td>Both: Department Store</td>
<td>0.08</td>
<td>1.34</td>
<td>0.06</td>
<td>0.94</td>
<td></td>
</tr>
<tr>
<td>Collector: Public Restroom</td>
<td>-6.73</td>
<td>1.89</td>
<td>-3.54</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>Collector: Friend house</td>
<td>3.44</td>
<td>1.77</td>
<td>-1.94</td>
<td>0.05</td>
<td></td>
</tr>
<tr>
<td>Both: Friend house</td>
<td>-4.65</td>
<td>1.80</td>
<td>-2.58</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>Collector: Home</td>
<td>2.48</td>
<td>1.04</td>
<td>2.39</td>
<td>0.01</td>
<td></td>
</tr>
<tr>
<td><strong>Data type</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>691.7</td>
</tr>
<tr>
<td>Reference : Temperature</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Biometric</td>
<td>-2.31</td>
<td>0.37</td>
<td>-6.16</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>Presence</td>
<td>-1.77</td>
<td>0.38</td>
<td>-4.57</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>Video</td>
<td>-2.05</td>
<td>0.39</td>
<td>-5.27</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td><strong>Data type + Happening Today</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>695.6</td>
</tr>
<tr>
<td>Reference : Biometric + No</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Presence: Yes</td>
<td>-1.30</td>
<td>0.73</td>
<td>-1.76</td>
<td>0.07</td>
<td></td>
</tr>
<tr>
<td>Video: Yes</td>
<td>17.27</td>
<td>200.82</td>
<td>0.08</td>
<td>0.93</td>
<td></td>
</tr>
<tr>
<td>Temperature: Yes</td>
<td>-0.78</td>
<td>1.38</td>
<td>-0.56</td>
<td>0.57</td>
<td></td>
</tr>
<tr>
<td>-----------------</td>
<td>-------</td>
<td>------</td>
<td>-------</td>
<td>------</td>
<td></td>
</tr>
<tr>
<td><strong>Retention</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Reference : Unspecified</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Forever</td>
<td>0.35</td>
<td>0.33</td>
<td>1.03</td>
<td>0.29</td>
<td></td>
</tr>
<tr>
<td>Until Satisfied</td>
<td>1.34</td>
<td>0.39</td>
<td>3.40</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>Week</td>
<td>-1.08</td>
<td>0.44</td>
<td>-2.45</td>
<td>0.01</td>
<td></td>
</tr>
<tr>
<td>Year</td>
<td>0.81</td>
<td>0.33</td>
<td>2.43</td>
<td>0.01</td>
<td></td>
</tr>
<tr>
<td><strong>User Benefit</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Reference : User</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Both</td>
<td>0.42</td>
<td>0.27</td>
<td>1.51</td>
<td>0.12</td>
<td></td>
</tr>
<tr>
<td>Collector</td>
<td>-0.82</td>
<td>0.28</td>
<td>-2.93</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td><strong>Shared</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Reference : Not shared</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Shared</td>
<td>1.36</td>
<td>0.38</td>
<td>3.57</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td><strong>Happening Today</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Reference : No</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Yes</td>
<td>1.20</td>
<td>0.33</td>
<td>3.57</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td><strong>Location</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Reference : Public restroom</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Department store</td>
<td>-0.96</td>
<td>0.44</td>
<td>-2.16</td>
<td>0.03</td>
<td></td>
</tr>
<tr>
<td>Friend house</td>
<td>0.48</td>
<td>0.38</td>
<td>1.26</td>
<td>0.20</td>
<td></td>
</tr>
<tr>
<td>Home</td>
<td>0.38</td>
<td>0.37</td>
<td>1.00</td>
<td>0.31</td>
<td></td>
</tr>
<tr>
<td>Library</td>
<td>0.77</td>
<td>0.36</td>
<td>2.10</td>
<td>0.03</td>
<td></td>
</tr>
<tr>
<td>Workplace</td>
<td>-0.65</td>
<td>0.41</td>
<td>-1.58</td>
<td>0.11</td>
<td></td>
</tr>
<tr>
<td><strong>Data type + Retention</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Reference : Biometric + Until Satisfied</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Presence: Unspecified</td>
<td>2.29</td>
<td>1.21</td>
<td>1.88</td>
<td>0.05</td>
<td></td>
</tr>
<tr>
<td>Video: Unspecified</td>
<td>-1.91</td>
<td>0.88</td>
<td>-2.17</td>
<td>0.02</td>
<td></td>
</tr>
<tr>
<td>Temperature: Forever</td>
<td>0.40</td>
<td>1.16</td>
<td>0.34</td>
<td>0.72</td>
<td></td>
</tr>
<tr>
<td>Presence: Forever</td>
<td>3.27</td>
<td>1.26</td>
<td>2.59</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>Video: Forever</td>
<td>-1.95</td>
<td>0.86</td>
<td>-2.268</td>
<td>0.02</td>
<td></td>
</tr>
<tr>
<td>Presence: Week</td>
<td>5.73</td>
<td>1.76</td>
<td>3.25</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>Video: Week</td>
<td>2.31</td>
<td>1.28</td>
<td>1.80</td>
<td>0.07</td>
<td></td>
</tr>
<tr>
<td>Temperature: Year</td>
<td>4.11</td>
<td>1.94</td>
<td>2.11</td>
<td>0.03</td>
<td></td>
</tr>
<tr>
<td>Presence: Year</td>
<td>2.78</td>
<td>1.20</td>
<td>2.31</td>
<td>0.02</td>
<td></td>
</tr>
</tbody>
</table>

Figure 4.4: GLMM Regression model for Comfort Level
Factors affecting Allow/Deny Decisions

We also built a regression model to understand if participants would allow or deny data collection if they were given a choice. This model will help us identify user’s real behavior toward the collection of their data and how they value their privacy in an IoT environment where sensors are omnipresent. Figure 4.6 depicts the results obtained from our GLMM model, where a positive coefficient indicates the likeliness of allowing the collection of data and a negative coefficient the likelihood of denying the collection of information. We again ordered the top 12 factors based on their BIC size to compare their results with our previous comfort level model as shown in Figure 4.7. We found out that User Perceived Benefits had the most effect on user likeliness to Allow/Deny the collection of their data while the combination of Location and Retention Time had the smallest impact. User Perceived Benefit factor aligned with our previous ranking, showing that participants’ comfortableness and behavior might be similar in real-world scenarios. This model also revealed that the combination of Data Type and
User Perceived Benefits highly affect user’s willingness to Allow/Deny the collection of their data. In fact, our statistical model showed that users are most likely to deny the collection of their sensitive data if they do not gain any benefit from it.

Analyzing various estimates among different factors reveals that participants will deny the collection of their data in scenarios where they believe data should be kept confidential and no compensation is awarded. For instance, participants will deny the collection of video/biometrics in a department store or at friend’s house as they do not gain any profit from it and to avoid profiling. Nonetheless, users are willing to allow the collection of their presence (Tracking) for emergency purposes. Among the statistically significant combinations, not all the factors that made participants comfortable sharing their information influenced their allow/deny decisions. For example, the location and retention factors did not have a high effect size as our preliminary model. We can hypothesize that being comfortable with sharing sensitive information does not systematically mean that users are willing to allow data collection when given a choice, which indicates that participant privacy perception shifts when they control their environment.
<table>
<thead>
<tr>
<th>Factor</th>
<th>Estimate</th>
<th>Std. Error</th>
<th>Z-value</th>
<th>P-value</th>
<th>BIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>User Perceived Benefit</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reference : Beneficial</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Not Beneficial</td>
<td>-3.28</td>
<td>0.28</td>
<td>-11.56</td>
<td>0.00</td>
<td>612.9</td>
</tr>
<tr>
<td>Data Type + User Perceived Benefit</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reference : Temperature + Beneficial</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Biometric: Not Beneficial</td>
<td>-1.57</td>
<td>0.90</td>
<td>-1.74</td>
<td>0.08</td>
<td>626.4</td>
</tr>
<tr>
<td>Presence: Not Beneficial</td>
<td>-0.03</td>
<td>0.88</td>
<td>-0.04</td>
<td>0.96</td>
<td></td>
</tr>
<tr>
<td>Video: Not Beneficial</td>
<td>-2.21</td>
<td>0.93</td>
<td>-2.37</td>
<td>0.01</td>
<td></td>
</tr>
<tr>
<td>User Perceived Benefit: Retention</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reference : Not Beneficial + Unspecified</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>654</td>
</tr>
<tr>
<td>Forever: Beneficial</td>
<td>-1.50</td>
<td>0.72</td>
<td>-2.06</td>
<td>0.03</td>
<td></td>
</tr>
<tr>
<td>Until Satisfied: Beneficial</td>
<td>14.37</td>
<td>418.04</td>
<td>0.03</td>
<td>0.97</td>
<td></td>
</tr>
<tr>
<td>Week: Beneficial</td>
<td>-0.45</td>
<td>0.92</td>
<td>-0.49</td>
<td>0.62</td>
<td></td>
</tr>
<tr>
<td>Year: Beneficial</td>
<td>-0.36</td>
<td>0.79</td>
<td>-0.45</td>
<td>0.64</td>
<td></td>
</tr>
<tr>
<td>User Benefit</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reference : User</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>787.5</td>
</tr>
<tr>
<td>Collector</td>
<td>-1.32</td>
<td>0.25</td>
<td>-5.18</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>Both</td>
<td>0.04</td>
<td>0.25</td>
<td>0.17</td>
<td>0.86</td>
<td></td>
</tr>
<tr>
<td>Data Type + User Benefit</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reference : Biometrics + Collector</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>789.9</td>
</tr>
<tr>
<td>Temperature : User</td>
<td>0.27</td>
<td>0.79</td>
<td>0.34</td>
<td>0.72</td>
<td></td>
</tr>
<tr>
<td>Presence : User</td>
<td>-1.56</td>
<td>0.59</td>
<td>-2.62</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>Temperature : Both</td>
<td>2.46</td>
<td>1.09</td>
<td>2.25</td>
<td>0.02</td>
<td></td>
</tr>
<tr>
<td>Presence : Both</td>
<td>-0.53</td>
<td>0.82</td>
<td>-0.64</td>
<td>0.51</td>
<td></td>
</tr>
<tr>
<td>Video: Both</td>
<td>1.88</td>
<td>0.66</td>
<td>2.84</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>Data Type + Location</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reference : Workplace + Temperature</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>804.5</td>
</tr>
<tr>
<td>Biometric: Department Store</td>
<td>-1.59</td>
<td>1.24</td>
<td>-1.27</td>
<td>0.20</td>
<td></td>
</tr>
<tr>
<td>Presence: Department Store</td>
<td>0.03</td>
<td>1.16</td>
<td>0.03</td>
<td>0.97</td>
<td></td>
</tr>
<tr>
<td>Biometric: Library</td>
<td>-3.24</td>
<td>1.05</td>
<td>-3.07</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>Presence: Library</td>
<td>-2.45</td>
<td>1.07</td>
<td>-2.27</td>
<td>0.02</td>
<td></td>
</tr>
<tr>
<td>Biometric: Public Restroom</td>
<td>-4.93</td>
<td>1.32</td>
<td>-3.71</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>Video: Public Restroom</td>
<td>-3.87</td>
<td>1.41</td>
<td>-2.72</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>Biometric: Friend House</td>
<td>-2.63</td>
<td>1.33</td>
<td>-1.97</td>
<td>0.04</td>
<td></td>
</tr>
<tr>
<td>Presence: Friend House</td>
<td>-1.39</td>
<td>1.48</td>
<td>-0.94</td>
<td>0.34</td>
<td></td>
</tr>
<tr>
<td>Video: Friend House</td>
<td>-1.37</td>
<td>1.66</td>
<td>-0.82</td>
<td>0.41</td>
<td></td>
</tr>
<tr>
<td>Happening Today</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reference : No</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>806.5</td>
</tr>
<tr>
<td>Yes</td>
<td>1.19</td>
<td>0.29</td>
<td>4.09</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>Data Type</td>
<td>Reference : Biometrics</td>
<td>808</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>------------------------</td>
<td>--------------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Presence</td>
<td>0.19</td>
<td>0.24</td>
<td>0.82</td>
<td>0.40</td>
<td></td>
</tr>
<tr>
<td>Temperature</td>
<td>1.5</td>
<td>0.32</td>
<td>4.66</td>
<td>0.00</td>
<td></td>
</tr>
<tr>
<td>Video</td>
<td>-0.17</td>
<td>0.23</td>
<td>-0.74</td>
<td>0.45</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Location</th>
<th>Reference : Library</th>
<th>808.8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Friends house</td>
<td>-0.61</td>
<td>0.29</td>
</tr>
<tr>
<td>Home</td>
<td>-0.2</td>
<td>0.29</td>
</tr>
<tr>
<td>Department Store</td>
<td>-1.31</td>
<td>0.33</td>
</tr>
<tr>
<td>Public Restroom</td>
<td>-1.61</td>
<td>0.34</td>
</tr>
<tr>
<td>Work Place</td>
<td>-1.30</td>
<td>0.31</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Shared</th>
<th>Reference : Not Shared</th>
<th>812.9</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shared</td>
<td>1.03</td>
<td>0.31</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Retention</th>
<th>Reference : Unspecified</th>
<th>818.3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Week</td>
<td>-0.27</td>
<td>0.34</td>
</tr>
<tr>
<td>Forever</td>
<td>0.32</td>
<td>0.28</td>
</tr>
<tr>
<td>Until Satisfied</td>
<td>1.42</td>
<td>0.35</td>
</tr>
<tr>
<td>Year</td>
<td>0.57</td>
<td>0.29</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Location + Retention</th>
<th>Reference : Library + Until satisfied</th>
<th>825.2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Work Place : Unspecified</td>
<td>-4.50</td>
<td>1.54</td>
</tr>
<tr>
<td>Public restroom : Unspecified</td>
<td>-8.87</td>
<td>1.96</td>
</tr>
<tr>
<td>Friends House : Unspecified</td>
<td>0.39</td>
<td>0.95</td>
</tr>
<tr>
<td>Home : Unspecified</td>
<td>-5.75</td>
<td>1.62</td>
</tr>
<tr>
<td>Department Store : Forever</td>
<td>1.98</td>
<td>1.00</td>
</tr>
<tr>
<td>Public restroom : forever</td>
<td>-7.16</td>
<td>1.56</td>
</tr>
<tr>
<td>Home : Forever</td>
<td>-2.84</td>
<td>1.18</td>
</tr>
<tr>
<td>Department Store : Week</td>
<td>-0.82</td>
<td>1.13</td>
</tr>
<tr>
<td>Public restroom : Week</td>
<td>-8.34</td>
<td>1.87</td>
</tr>
<tr>
<td>Work Place : Year</td>
<td>-3.16</td>
<td>1.11</td>
</tr>
<tr>
<td>Home : Year</td>
<td>-3.24</td>
<td>1.29</td>
</tr>
</tbody>
</table>

Figure 4.6: GLMM Regression model for Allow/Deny
Figure 4.7: GLMM Regression Summary of BIC values for the Allow/Deny model

<table>
<thead>
<tr>
<th>Factor</th>
<th>BIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>User Perceived Benefit</td>
<td>612.9</td>
</tr>
<tr>
<td>Data Type + User Perceived Benefit</td>
<td>626</td>
</tr>
<tr>
<td>User Perceived Benefit + Retention</td>
<td>654</td>
</tr>
<tr>
<td>Data Type + User Benefit</td>
<td>789</td>
</tr>
<tr>
<td>User Benefit</td>
<td>787.5</td>
</tr>
<tr>
<td>Location + Data Type</td>
<td>804.5</td>
</tr>
<tr>
<td>Happening Today</td>
<td>806.5</td>
</tr>
<tr>
<td>DataType</td>
<td>808</td>
</tr>
<tr>
<td>Location</td>
<td>808.8</td>
</tr>
<tr>
<td>Shared</td>
<td>812.9</td>
</tr>
<tr>
<td>Retention Time</td>
<td>818.3</td>
</tr>
<tr>
<td>Location + Retention Time</td>
<td>825</td>
</tr>
</tbody>
</table>
Chapter 5

Comparison of Machine Learning Approaches

In this chapter, we conducted an experiment on the collected data from the replication study in chapter 4. This phase aims to study how Machine Learning algorithms behave using different techniques from the original study. The results helped us evaluate the impact of different feature interactions on our target variables. It also aided us to build a strong hypothesis for our final proposed solution and collect various feedback to help users better assess their privacy risks.

5.1 Methodology and Procedures

The results collected from section 4.3 helped us understand how different factors impacted users’ privacy perceptions in different IoT locations. The GLMM model we used enabled us to classify location and sensor types, as shown in Section 4.1. These new feature categorisations would be helpful when creating new factors to provide a better ML pattern building. The main goal of this approach is to build a system capable of mimicking real-world scenar-
ios where users have control over their shared data sustainably.

5.1.1 Data Preparation

After exploring all the features used in our replication study, we decided to clean the data further and eliminate factors that users do not have control over. Factors such as user perceived benefits and retention time are hypothetical and may depend on the authority managing them. To that end, we decided to take another route and only include data type, location, and privacy level. Concerning the target variable, we decided to start with a binary classification where our model would be able to predict if a user should enter or not a specific location. Based on the location feature, we were able to classify locations as low, medium, and High. Room Scores were designed based on users’ comfort level results from the previous study as follows:

- **High**: Self-governing locations where users expressed high concerns or did not trust their governing authority, such as Department Stores and Public Restrooms.

- **Medium**: Locations not accessible to the general public or may make users uncomfortable when sharing their data, such as Workplace and Friend house.

- **Low**: Location accessible to the general public or location where users feel comfortable sharing their data, such as at home and library.

We also engineered a new Sensor Score feature that transforms data types into a numerical score from one to four based on their sensitivity level. Sensor score classification was derived from figure 4.3, with a sensor score of four being the most sensitive sensor capturing identifiable information and a sensor score of one being the less sensitive sensor capturing motion or environmental patterns. Here is the classification of sensors from the most sensitive to the least sensitive sensor: Biometric, Video, Tracking, and temperature.
Privacy levels were based on Dr. Westin’s privacy indexes [248, 148] as it was built on users’ privacy attitudes and concerns. Dr. Westin categorized users’ privacy levels as follow:

- Privacy Fundamentalist: High privacy concerned individuals that reject consumer benefits claims and seek more robust privacy regulation measures
- Privacy Pragmatist: Medium privacy concerns individuals that weigh the benefits of various consumer opportunities versus how practical their privacy is in a particular scenario
- Privacy Unconcerned: Low privacy concerned individuals that do not have enough background on privacy concepts and are willing to share their private data with any organization.

Figure 5.1 represents the privacy level distribution collected from our replication study data, whereas Figure 5.2 summarizes all the features used in our approach and their respective levels.

![Figure 5.1: Privacy Level Distribution](image)
5.1.2 Machine Learning

To develop our models, we used Python as a programming language [200]. Python libraries offer a wide range of services, from Data analytics and visualization to preprogrammed ML models that are easy to implement and tune [200, 201].

To execute our approach, we decided to use Google Collaboratory (Colab), a product from Google research. Collab is an online platform, built on Jupyter Notebooks, used to execute arbitrary python code on the cloud with no setup or installation requirement [65]. Colab performs well with ML problems and is especially suited for small data sets as it does not require extensive resources [65]. First, as shown in Figure 5.3, we decided to test three ML models: Support Vector Machine (SVM), Logistic Regression (LR), and Decision Trees (DT) to assess the best approach for our binary classification. We utilized the NumPy library to support multidimensional arrays, matrices, and high-level mathematical functions, Pandas for structured data analysis, and Scikit-learn for classification and regression models.

To evaluate the performance of a supervised ML model we can use different mathematical measures built around confusion matrices. Most researchers nowadays focus on how well a classifier can correctly predict classes when testing the data [225]. To assess different characteristics of our approach we are going to use the following measurements:

- **Accuracy**: "Approximates how effective the algorithm is by showing the probability..."
of the true value of the class label; in other words, it assesses the overall effectiveness of the algorithm” [225] But it does not necessarily indicate how well the algorithm performed as classed may be imbalanced. Formula: \[ \text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN} \]

- **Precision**: "Estimates the predictive value of a label, either positive or negative, depending on the class for which it is calculated; in other words, it assesses the predictive power of the algorithm” [225]. Formula: \[ \text{Precision} = \frac{TP}{TP + FP} \]

- **Recall**: "approximates the probability of the positive (negative) label being true; in other words, it assesses the effectiveness of the algorithm on a single class.” Formula: \[ \text{Recall} = \frac{TP}{TP + FN} \] [225]

- **F1-score**: Is the Harmonic mean between precision and recall [74]. Formula : \[ \text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \]

- **AUC (Area Under the Curve)**: The Receiver Operating Characteristics (ROC) represents the tradeoff between benefits (True Positive Rate) and Cost (False positive rate). Spackman [227, 96] was considered one of the first researchers that used AUC to evaluate the performance of different Machine Learning models. AUC has been considered one of the statistical measurements to provide a better approximation of the performance of Machine Learning Models [122, 55, 96].
5.2 Machine Learning Results

This section summarizes the results obtained from training our model using the factors and interactions created in section 5.1. Different Machine Learning models were trained to predict users’ decisions to allow or deny entry to a specific location. The first step was to check the data distribution between different features and their interaction with the target variable (Allow and Deny). Figure 5.4 illustrates a sample of different factor level combinations in the dataset used to build our model. All features were tested against the target variable to confirm our results and hypothesis built in Section 4.3. Figure 5.5 summarizes some of the results obtained from our dataset depicting the percentage distribution. The data visualization obtained will help us further understand the general privacy perception of users and what might alter their privacy decisions. Privacy fundamentalists exhibited the highest deny rate of 25%, while privacy unconcerned exhibited the low deny rate of 5%. On the other hand, the privacy pragmatist Deny/Allow rate was almost balanced. Another observation from this data distribution was that biometric sensors and Video showed a high denial rate of 22% and 15%, respectively.

We first tested each ML model using the KFold cross-validation approach to evaluate
our models. We tested different K-Fold split methods to try to figure out what would be the optimum threshold for our model approach. The 5-split approach (n=5) performed the best compared to other approaches, as shown in figure 5.6. This particular split yielded 75% accuracy compared to 73% for ten splits and 58% for two splits. The five split approach would be similar to having 80% training data and 20% test data that we will use later on to confirm our results. We also run ten iterations each time to ensure our trained data will perform well regardless of where the split occurred.

Figure 5.7 illustrates ten tests we run for the five K-fold approach where each iteration = Kfold (Five splits) * Range (From one to ten). The dot in our first iteration represents an outlier (noise) where the accuracy score was noticeably different from the rest. We can also notice from our figure that iteration six to iteration ten had more Variance as we had more
After finding the best distribution between the trained data and the testing data, we decided to test the performance of different algorithms on each predictive class. We studied precision, recall, and F-1 score related to each target variable for SVM, LR, and DT. Figure 5.8 shows an example of the classification report of the DT model containing a description of the predictive score of the allow/deny decisions. This detailed report distribution will help us understand how well our overall model performed and how balanced our distribution is among different matrices. Each predictive class across all measurements showed an even distribution varying at a rate of 0.03 at most.

Upon ensuring each class distribution is evenly matched among different performance
measures, we compared the overall performance of our models using 90% and 80% training data. Figure 5.9 describes the results obtained when comparing DT, LR, and SVM performance using different testing data sizes. Based on the accuracy, the Decision Tree classifier performed better, with an overall accuracy of 77%, than Logistic Regression with 75% and SVM with 70%. The F1-Score also supports the accuracy results by showing that DT outperforms LR (0.754) and SVM (0.692) classifiers with a score of 0.773. The accuracy evaluation shows that all models performed better with a smaller set of training data yielding an increase of 1 to 3% percent across all performance measurements. Figure 5.10 summarizes the performance of each model individually compared to others.

It is important to note that Accuracy, Precision, Recall, and F1-Score are performance measurement techniques used to evaluate the performance of ML models. Still, they might not be an excellent indicator for choosing the best Classifier [190]. To cope with this problem, we needed to study if the slight variance between the results was statistically significant. The paired T-Test is a standard statistical hypothesis testing approach for evaluating the mean performance of different models developed by Dietrich [86]. According to the Student’s T-Test, we can compute the T-Statistics with a k-1 degree of freedom between 2 sets of classifiers [84]. After acquiring the T-Score we can compute the P Value and compare it to the significance level Alpha. If the P Value is smaller than Alpha, we can reject the null hypothesis that the two models are performing equally the same.
To address the shortcoming of different T-Test approaches, we decided to use the Paired-TTest-X2CV [86] using an alpha threshold of 0.05. We run a T-Test on SVM, LR, and DT using K-Fold cross-validation and a null hypothesis stating that the performance of the models is significantly different. Upon comparing LR and SVM, we found out that $p (0.005) < \alpha (=0.05)$, which enabled us to reject the null hypothesis and conclude that the two algorithms are significantly different (LR better than SVM). We also found the same results when comparing DT and SVM, where DT outperformed the SVM classifier with a p-value of 0.007. When comparing DT and LR, we found out that $p = 0.38$. Since $P > \alpha (=0.05)$, we cannot reject the null hypothesis and may conclude that the performance of the two algorithms is not significantly different. Since the performance of LR and DT are the same, we decided to use the AUC score, which was found to be more efficient than accuracy [122] to choose between classification models. Based on AUC scores, we can note that DT outperforms LR with an AUC score of 0.77.

### Table 5.9: Training data results

<table>
<thead>
<tr>
<th>Classifier</th>
<th>90% Training Data</th>
<th>80% Training Data</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Accuracy</td>
<td>precision</td>
</tr>
<tr>
<td>Decision Tree Classifier</td>
<td>0.7581</td>
<td>0.7586</td>
</tr>
<tr>
<td>Logistic Regression Classifier</td>
<td>0.7419</td>
<td>0.7471</td>
</tr>
<tr>
<td>SVM Classifier</td>
<td>0.6935</td>
<td>0.7181</td>
</tr>
</tbody>
</table>

5.3 Discussion

The primary purpose of our first experiment was to conduct a test on our collected data (Replication Study) to see how well the selected features can contribute in predicting an individual decision to allow or deny entry to a specific IoT location. To accomplish this task, we first studied the impact of individual factors on users’ comfort levels, as shown in figure 4.3. Using the obtained result as a classification standard, we built a GLMM model in Section 98.
4.3 to help us understand different factor interactions. Our data distribution results indicate that users classify each feature based on the risk associated with them. The results collected will help us create different ranking systems for different factors. Using fewer feature levels will help our ML model create more pertinent patterns and make better predictions. Based on the data distribution, we were able to classify the most important factor interaction that may potentially impact users’ privacy precision when making a decision.

The main issue with the Naeini et al. [179] approach was that some of the factors used are not realistically achievable. Factors such as retention time, sharing details, or user benefits cannot be obtainable in real-life scenarios. Our goal was to use factors that users could have access to or control over when visualizing a specific location. To achieve this objective, we cleaned our data to only include privacy level, data Type, and location. We also engineered additional features to help our model create a more robust pattern when making a prediction, as explained in section 5.1.1. Our data distribution among different factors and their targeted
output was balanced and confirmed our previous observations. Achieving an equitable data distribution in ML is essential as it helps attain a higher accuracy prediction. The performance measurements indicate that our models performed well with a smaller data set and lesser features compared to the original approach. We achieved an accuracy of 77.42% with the Decision Tree model compared to 79% accuracy for Naeini approach that used a data set 10 times bigger than ours and more feature combinations.

The main objective is to create an application that could mimic real-world examples and help users better understand their environment and the risks associated when sharing their personal data. This experiment gained us a better understanding of users’ privacy needs and was the driver for creating a better suitable approach. Examining our results enabled us to design a better approach to mitigate user privacy fatigue. Another issue in Naeini’s research is that their experiment only included one data type at a time, which means that each location can only include a single IoT sensor.

Based on our observations, and to enhance user privacy experience, we decided to bring the following improvements:

- Consider only factors that users have control over to limit user interaction with our application while creating new features around them. The goal is to efficiently assist users to avoid decision exhaustion state.

- Build a simpler survey to collect the necessary data and avoid user information overload. Our previous survey was exhaustive and contained many questions.

- Build a balanced survey that will contain an even question distribution between different factor interactions.

- Recruit a broader range of users from different backgrounds and privacy knowledge. Our previous survey was demographically tied to campus students resulting in a sampling bias.
• Evaluate different coding techniques to enhance our results. In our replication study, we had to follow the same coding methods used by Naeini et al. [179] to be genuinely able to compare our results with theirs.

• Shift from a binary classification (Allow/Deny) to a recommender system with different privacy risk levels (High Risk, Medium Risk, Low Risk). We believe it is better for users to use our predictive model as a recommendation tool instead of deciding for them. Allowing or denying an entry may bring even more confusion to users and deepen their privacy fatigue phenomenon. Using a recommender system will facilitate decision-making while keeping users in control of their actions.
Chapter 6

A New Survey of User’s Privacy Preferences

In this chapter, we present the steps we overtook to design PPM (Privacy Preference Manager), a ML recommender system, based on the feedback of our preliminary testing phase in section 5.3. We first showcase how we constructed the new survey, explain how we used the cleaned data to build our recommender system, and finally, all the techniques used to improve our ML model for optimal performance.

6.1 PPM Survey

To achieve our primary goal of creating plausible scenarios that can mimic real-world IoT environments, we had to investigate the main factors that users had control over. chapter 5 enabled us to gather the necessary feedback to build a simple survey with only the essential data to be captured. In the next section, we will describe all the steps that enabled us to reach this objective.
6.1.1 Creating The Survey

To create a balanced survey, we evaluated all possible factor interactions and distributed them evenly among all scenarios. When creating the study, we based the factor distribution on Location containing five different levels and Data Type containing four different levels, as shown in Figure 6.8. We also had to include different Data Type combinations to give the user the freedom to visit locations with various IoT sensors. We created five different Data type interactions (Up to three combinations at most) to account for interactions that can impact users’ privacy preferences. Randomly combining different factor levels may result in unbalanced data distribution. To cope with this challenge, we selected each factor level individually and included all logical combinations. Figure 6.1 illustrates all factor interactions created and showcases how we evenly distributed all features among all survey versions for an equitable data collection.

Based on the distribution of the factors shown in Figure 6.1, we created nine realistic scenarios and five different versions to diversify the possible outcome of our vignette study.
Those 45 scenarios were crafted to mimic real examples in the simplest way possible not to overwhelm participants with unnecessary information. To make our scenarios reliable, we introduced factors in the same order to avoid confusion and redundancy. Within those vignettes, we used three different factors (Data-Type, Device-Type, and Location) that impact user privacy preferences when interacting with IoT environments. Data-Type refers to the type of data collected and includes four different levels: Biometric, Video, Tracking, and Temperature. Location refers to the place at which the data is collected and includes five different levels: Library, Public Restroom, Friend’s House, Department Store, and Workplace. Device type refers to the device utilized to collect the data and includes the following levels: Video Camera, Fingerprint Scanner, Facial Recognition, Temperature Sensor, Motion Sensor, and Smart Badge. Note that Device-Type was used to make our scenarios more realistic and will not be used as a factor in building our Machine Learning model. The following example is a scenario from version one: “You are at work. Your company has Iris scanners and cameras deployed around the building. Your Biometric Data and Video recordings are used for employee’s identification and monitoring purposes.” Each version of the survey includes two parts:

- **Privacy preference level:** Before going through the scenarios in each version, participants were asked to answer three questions based on Westin Privacy Segmentation Index [248]. This approach helped us categorize users based on their privacy attitudes. We modified the questions to fit our IoT approach as follows: **Q1:** Users have lost all control over how personal information is collected and managed by IoT devices (smart devices). **Q2:** Most IoT devices (smart devices) handle the personal information they collect about users in a proper and confidential way. **Q3:** Existing laws and organizational practices provide a reasonable level of privacy protection for IoT (smart devices) users. All those questions were coded using a Four-point Likert scale based on Westin’s research (Strongly Agree, Somewhat Agree, Somewhat Disagree,
Strongly Disagree). In section 6.2 we will explain how to categorize users based on their responses.

- **Sample Survey Scenarios:** Participants were asked to complete nine different scenarios containing different factor combinations. In each scenario, we asked our participants two questions: The first one, if given a choice, how likely would they agree or disagree with data collection in specific conditions. In the second part, we included data storage and data retention information to study a possible shift in participants’ perceptions. Both questions were coded using a five-point Likert scale from “Strongly Agree” to “Strongly Disagree.”

### 6.1.2 Recruiting Participants

Based on previous participants’ recruitment challenges, we decided to use Prolific to hire participants and collect the necessary data to build our model. Prolific is a platform that helps researchers find a targeted audience and recruit participants to participate in the survey [16]. Participants recruited via Prolific were fully anonymous as they were assigned a unique identification number. The Prolific platform enabled us to choose the area of expertise and specific demographics (such as minimum age and language proficiency) of the recruited participants. People of all ages, genders, and backgrounds were welcome to join our study as long as they were over 18 years old and fluent in English. Additionally, the Prolific platform does not store any survey data on their servers but instead enables us to share our Google survey link and collect the data on our end. All data collected through our Google Forms would be kept on Google servers until the study is completed and deleted once the final model is satisfactory. We decided to recruit 250 participants (Four times the size of our previous experiment) and assign 50 participants to each version for equitability purposes. Each participant was paid $1.60 to complete a 10-minute survey. The paid amount was
6.2 Data Cleaning, Coding and Engineering

All five versions were created using Google forms as it presents a convenient way to craft the scenarios and store the answers on an excel sheet for further processing. We were able to gather 229 participants across all versions, each exposed to nine different scenarios for a total of 2062 data points. We had to remove 21 participants from our survey for various reasons, such as a low completion time of two minutes, while the average was around five minutes. Their responses show a lack of effort and inconsistency towards the majority of the questions they had to answer.

After gathering all the necessary data from our survey, we had to combine all versions and distribute all factors accordingly. Our first step was to figure out the privacy level of each of our participants based on Westin’s coding technique [248] as follows: Based on Q1, any participant that responded ”Strongly Agree” or ”Somewhat Agree” is considered privacy-concerned. Concerning Q2 and Q3, participants who answered ”Strongly Disagree” or ”Somewhat Disagree” are considered privacy-concerned. We next use those parameters to categorize participants into three categories:

- **Privacy Fundamentalist**: Participants who were Privacy concerned on every question answered.

- **Privacy Unconcerned**: Participants who did not show Privacy concerns on any question answered.

- **Privacy Pragmatist**: Participants who showed Privacy concerns on some questions and no Privacy concerns on others.
Figure 6.2 shows 34% Privacy Fundamentalist, 58% Privacy Pragmatist, and 8% Privacy unconcerned, which aligns with Westin's theory [248].

Our next step was to find the best coding distribution for our recommender system based on the Five points Likert scale. As explained, our recommender System’s goal is to guide participants in making a suitable decision that meets their Privacy expectations. Coding the recommendations properly is essential as each distribution can yield different outputs. Our system is built on three recommendation levels: High Risk, Medium Risk, and Low Risk but the data collected from our audience was based on a five-point scale. To tackle this problem, we had to analyze the data distribution for all versions, as shown in Figure 6.3, which will help us eventually find the best approach. We next tried four different distributions to cover all possible combinations. Our first approach had an uneven distribution where medium risk was dominant with a 73% overall distribution. This cannot be a viable approach as most of our predictions will turn out to be Medium Risk, which may further confuse our users. Approach three had a minimal Low-Risk distribution of 11%, and approach Four had a High-risk distribution of 16%. Approach two Showcased the most balanced distribution (Figure 6.4) with a low-risk rate of 41%, a Medium risk rate of 21%, and a High-risk rate of 38%. We further confirmed this approach by calculating the mean of the coded scale (Strongly Agree = 5, Agree = 4, Neutral = 3, Disagree = 2, and strongly disagree = 1) and ended up with a mean of 3 and an approximate standard deviation of 1.3 confirming that "Neutral" is a point of reference.

To minimize user interaction with our system, we chose only to capture three factors that greatly impacted users’ privacy perception and can be controlled by users. As explained in Section 6.4, we needed to find ways to engineer more features. Acquiring additional features enables our ML model to build better patterns and perform better with its predictions. We introduced Sensor Score, which classifies Data Types based on their sensitivity level. We further created Location Type, which categorizes Locations as follows: Public location
One of the primary implementations of the PPM approach was enabling users to choose more than one Data Type at a time. The issue with this concept was dealing with null values in our data set, knowing that some locations may have one or multiple sensors depending on the scenario. One should also note that ML models do not handle null variables well [149].
When dealing with unknown variables, the best course of action is to translate the Data type features (DataType1, DataType2, DataType3) into one feature that genuinely represents the weight associated with each Data type combination. Based on our previous research, we already know that people associate their privacy risk with how sensitive sensors are; thus, we built a new feature around the Sensor Score. To determine the right weight for each Data Type combination, we first analyzed their GLMM interaction to see how users perceive the risk associated with each combination. As shown in Figure 6.6, it is clear that each time we add a data type, the risk perceived by users augments based on how sensitive the sensors added are. The next step was to find the best weight formula for each data type combination in a manner that does not unbalance any Data type interaction. This step was conducted in a trial-and-error fashion where multiple data type weights were tested to satisfy our following formula:

\[ \text{DatatypesScore} = \text{Max.SensorScore} + (\text{DataType2.SensorScore} \times \text{DataType2.Weight}) + \]
(\text{DataType3.SensorScore} \times \text{DataType3.Weight})

With Max.SensorScore representing the Highest sensor score among all 3 Data Types in a scenario. DataType2 and DataType3 represent the remaining two sensors. The null score represents scenarios with less than 3 Data Types. Figure 6.7 summarizes the data type sensor score and their associated weight.

<table>
<thead>
<tr>
<th>Ref: Biometric</th>
<th>Estimate</th>
<th>Std Error</th>
<th>Z-Value</th>
<th>P-Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Biometric + Video</td>
<td>-0.45</td>
<td>0.21</td>
<td>-2.23</td>
<td>0.02</td>
</tr>
<tr>
<td>Biometric + Tracking</td>
<td>-0.39</td>
<td>0.2</td>
<td>-1.92</td>
<td>0.053</td>
</tr>
<tr>
<td>Biometric + Temperature</td>
<td>-0.07</td>
<td>0.27</td>
<td>-0.26</td>
<td>0.79</td>
</tr>
<tr>
<td>Biometric + Video + Tracking</td>
<td>-0.72</td>
<td>0.24</td>
<td>-2.98</td>
<td>0.0028</td>
</tr>
<tr>
<td>Biometric + Video + Temperature</td>
<td>-0.51</td>
<td>0.23</td>
<td>-2.46</td>
<td>0.08</td>
</tr>
</tbody>
</table>

Figure 6.6: GLMM Data Type Interactions.

<table>
<thead>
<tr>
<th>Sensor Score</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Biometric</td>
<td>4</td>
</tr>
<tr>
<td>Video</td>
<td>3</td>
</tr>
<tr>
<td>Tracking</td>
<td>2</td>
</tr>
<tr>
<td>Temperature</td>
<td>1</td>
</tr>
<tr>
<td>Null</td>
<td>0</td>
</tr>
</tbody>
</table>

Figure 6.7: Sensor Score and associated weights.

Our results show a proper Data Type score distribution among all possible combinations while respecting the weight associated with each Data Type individually. This approach will make it easier to drop any Null value in our dataset without jeopardizing the interaction between different Datatype values. After Cleaning all the data and getting suitable coding scales through the engineered features showcased in Figure 6.8, we can start building our ML model and test how well our dataset performs using different ML techniques.
6.3 Machine Learning Approach

This section explains in detail all the ML models and boosting approaches we applied to our final dataset to maximize our final model performance.

6.3.1 Design

Unlike our previous experiment, we used Jupyter notebook [15] as we had a larger Dataset and needed access to additional resources. We also decided to broaden our horizon and apply more classifiers in an attempt to find the best fit for our final approach. As explained in Section 6.4, using multiple Models and boosting techniques enable us to maximize our final model performance. To the best of our knowledge, no available dataset or research has attempted to recommend to users the risk associated with interacting with an IoT environment. Therefore, it is important to test all possible techniques and explore all avenues to improve our model performance. Based on all the knowledge we acquired and the feedback we collected from our previous experiments, we decided the following:

- Use our original dataset to train the following models: SVM, LR, DT, RF, and XG-Boost without applying any engineered feature stated in Section 6.2. This means the only features used would be Privacy Level, Location, and the three instances of Datatype.

<table>
<thead>
<tr>
<th>Features</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Privacy Level</td>
<td>Fundamentalist, Pragmatist, Unconcerned</td>
</tr>
<tr>
<td>Data Type</td>
<td>Biometric, Video, Tracking, Temperature</td>
</tr>
<tr>
<td>Location</td>
<td>Work, Library, Friend's House, Department Store, Public Restroom</td>
</tr>
<tr>
<td>Sensor Score</td>
<td>4 (Most sensitive), 3, 2, 1 (Least sensitive)</td>
</tr>
<tr>
<td>Location Type</td>
<td>Public (Department Store, Library), Semi-Public (Work, Friend's House), Private (Restroom)</td>
</tr>
<tr>
<td>Environment Score</td>
<td>High, Medium, Low</td>
</tr>
<tr>
<td>Datatypes Score</td>
<td>Calculated using the formula in Section 6.2</td>
</tr>
</tbody>
</table>

Figure 6.8: A summary of Feature Levels.
• Analyze the feature interaction with our target variable to check for inconsistencies or redundancies. Data distribution is essential when constructing our models and can help us find anomalies (Unbalanced datasets).

• If necessary, apply data augmentation to reduce the gap between data samples, which may not provide sufficient information to the ML model to build strong patterns.

• Introduce the newly engineered features: Sensor Score, Location Type, Environment Score, and DataType Score among models.

• Use Feature Selection to enhance our model potentially. This step may help in selecting the relevant subsets and eliminate weaker classifiers to improve our predictive accuracy. We will specifically use RFE as it performs well with small datasets classification problems.

• Use the Min-Max normalization technique that may improve data quality and reduce the domination of high numerical feature values.

### 6.3.2 Results

This section summarizes the results obtained from applying multiple techniques stated in the previous section to improve the performance of our ML models. Applying and testing different approaches is the key to showing a gradual ML enhancement as no comparable research can be set as a reference point. Our first step was to test multiple Machine Learning algorithms on our original dataset that did not include any engineered features and analyze its results to check for any data distribution discrepancies. Our preliminary attempt using a five K-Fold cross-validation technique exhibited an accuracy of 55% for SVM using a linear kernel, 56.5% for Decision Trees and Random Forest, and 57% for both Logistic Regression and XGBoost. Standard Deviation among all models varied between 0.015 and
0.024, showing consistencies among different splits. Upon checking the data distribution against our target Variable, we noticed disparities with some feature level distributions, as shown in Figure 6.9.

In this example, we can see that Video is evenly distributed between Medium Risk and High Risk, which indicates that it will be difficult for our trained model to differentiate between them and make the correct prediction. As stated in section 3.4, the best way to deal with unbalanced small data sets is to target weak classifiers with random oversampling. Our next step was to apply augmentation on weak classifiers until the specific distribution was adjusted. It can be seen from Figure 6.10 using targeted augmentation enabled an even distribution among different feature levels. To further test our hypothesis, we wanted to study its impact on the accuracy of our trained model. For testing purposes, we decided to choose Logistic Regression as a standard model to compare the evolution in performance using a five K-fold cross-validation approach. The trained model exhibited an increase in

Figure 6.9: Unbalanced Data Example.
performance of 5.1% (From 57% to 62.1%) with a standard deviation of 0.013 among all splits.

Our next step was to study the effect of adding the newly engineered features created in section 6.2 on our trained model. Adding new features might help our ML model develop new patterns and increase its performance and prediction capabilities. Figure 6.11 displays the increase in accuracy at different states when each engineered feature is introduced. We report a slight increase in accuracy on the augmented dataset from 62.1% to 62.8% when all features are used. Regarding Feature Selection, we decided to use Recursive Feature Elimination as it is a good approach to apply to small classification problems [123]. Applying RFE to the Augmented dataset presented a slight increase in Logistic Regression accuracy, varying between 61.8% and 62.7% when selecting the best feature combination. We observed how accuracy fluctuated using different feature selection combinations. We also applied
Min-Max scaling to see its effect when our data is normalized. We note a slight increase in accuracy from 61.2% to 61.5% when Min-Max is applied to Logistic Regression.

After individually testing all methods stated in the previous section, we combined them and tested their effect on multiple classifiers. We report Accuracy, Precision, Recall, and AUC score for each of our Five ML models using 80% training Data and 20% testing data. We notice that based on accuracy, precision, and recall matrices, the Decision Tree classifier outperformed any other model exhibiting an accuracy of 67%, a precision of 66.6%, and a recall of 66.9%. Based on the AUC score, the XGBoost classifier showed a better score of 0.786 compared to other models. Figure 6.12 summarizes our results using different measurement matrices, and Figure 6.13 draws a visual comparison between all trained models. We also studied the execution time of each of our five models. Our evaluation process indicates that the Decision Tree classifier had the fastest execution time of 4 milliseconds, and SVM had the slowest execution time of 690 milliseconds. As explained in section 5.2, Accuracy, Precision, Recall, and F1-scores are performance measurement techniques that do not clearly evaluate the performance of ML models. To cope with this issue, we should study if the small variance between the results is statistically significant among different models using Dietrich T-test technique. Upon running the T-Test on Random Forest and SVM, we
can conclude that since $p (0.09) > \alpha (=0.05)$, we cannot reject the null hypothesis and may conclude that the performance of the two algorithms is not significantly different. The T-Test was run on all possible algorithm combinations; however, we could not find any statistical significance in performance among all algorithms.

<table>
<thead>
<tr>
<th>Model</th>
<th>Accuracy</th>
<th>Precision</th>
<th>Recall</th>
<th>F1-Score</th>
<th>AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model-2: XGB Classifier</td>
<td>0.66440</td>
<td>0.67006</td>
<td>0.66440</td>
<td>0.65268</td>
<td>0.78571</td>
</tr>
<tr>
<td>Model-1: RandomForest</td>
<td>0.65758</td>
<td>0.66730</td>
<td>0.65758</td>
<td>0.65116</td>
<td>0.79484</td>
</tr>
<tr>
<td>Model-5: LogisticRegression Classifier</td>
<td>0.64225</td>
<td>0.61849</td>
<td>0.64225</td>
<td>0.61697</td>
<td>0.78020</td>
</tr>
<tr>
<td>Model-3: DecisionTree Classifier</td>
<td>0.66951</td>
<td>0.66626</td>
<td>0.66951</td>
<td>0.65782</td>
<td>0.77977</td>
</tr>
<tr>
<td>Model-4: SVM Classifier</td>
<td>0.62692</td>
<td>0.62605</td>
<td>0.62692</td>
<td>0.62312</td>
<td>0.74771</td>
</tr>
</tbody>
</table>

Figure 6.12: PPM Model Comparison.

Figure 6.13: PPM Model Visualization.
6.3.3 Discussion

This experiment was built on the feedback collected from the two studies conducted in Chapter 4 and Chapter 5. Chapter 4 research enabled us to study users feeling concerning data collection in different IoT locations and helped us in evaluating the contribution of various factors that may impact user privacy choices. The result obtained helped us design a more transparent and informative IoT privacy approach that may help users cope with the Privacy Fatigue phenomenon. Chapter 4 helped us test our preliminary hypothesis (Obtain results comparable to Naeini et al. research [179]) by using only factors that users have control over. The result obtained further assisted us in building a simpler survey and establishing a better classification approach to cope with the dilemma of decision making. Our final approach backbone was built on a simpler system that satisfies all the feedback and takeaways collected from our previous research while guiding our users through dealing with IoT environments in a seamless way.

One of the main issues we encountered with our final experiment was having some feature levels evenly distributed, which caused data distortion. Upon further data analysis, we noticed that some scenarios were a bit ambiguous and did not specifically describe our intended IoT environment vision. The following is an example of a misleading scenario used in version four: “You are at a public restroom. Some people in the restroom are using their smartphone camera to video record a graffiti drawn on the wall”. Most of our participants exhibited a Low-Risk target even though this scenario intended to expose a high sensitivity sensor in a private location. We can hypothesize that people are familiar with using cell phones in sensitive locations without the intention of exposing others’ personal data. Crafting a scenario might seem like a simple task. Still, we noticed that users’ perspectives and visions should have been analyzed more thoroughly when building IoT scenarios. Augmentation was targeted around particular feature levels in some IoT locations where the described scenarios did not meet the intended design and caused data inconsistencies.
Our original data set only included 2061 data points and three features to work with, which pushed us to engineer new factors to help our ML create new dependencies and cope with the issues of having a small data set. Adding new features helped us extract more information from categorizations approaches and slightly increase our model performance. Feature selection results using RFE did not exhibit any efficiency enhancements to our model because our input features are limited, and using fewer features might degrade our results. Usually, Feature selection is used to reduce redundancies between factor combinations and eliminate noise within a big dataset using many features. We also noticed that normalization did not affect our model performance. This stems from the fact that our dataset values were coded in a categorical manner; thus, there was no need for scaling before modeling.

After applying all boosting techniques to our models, we recorded Accuracy, Precision, Recall, F1, AUC score, and execution time to try to select the best model for IoTPP. Based on Accuracy, which is one of the evaluation matrices one may use to compare between models, we report that performance varied between 62.3% and 67%. To truly be able to choose one model over another, we had to study if the change in performance was statistically significant or not. Using the Dietrich T-Score method, we proved that all models performed the same, and no statistical improvement was found. Although no statistical significance was recorded, we based our results on the AUC score, which was used in multiple research [96, 122, 55] and proved to be more efficient than Accuracy in choosing between classification models. Based on AUC scores XGBoost classifier performed the best among other models, with a score of 0.786 and an execution time of 250 Milliseconds. We will be using XGBoost as our final model to build the IoTPP web application.
Chapter 7

IoTPP Web Application

7.1 Web App: Overview

In this section, we introduce IoTPP Web App, a user privacy tool that provides recommendations to users when visiting IoT Locations. IoTPP, a web implementation of PPM, relies on the trained model obtained from our previous chapter to guide users with a tailored privacy recommendation based on their privacy level. IoTPP Web App was designed to test the accuracy of PPM and gather more data to build a larger dataset and optimize our ML approach. Through IoTPP web app, participants would be able to interact with different IoT sensors in multiple IoT locations. Based on the feature combinations, our system would be able to make a recommendation as follows: Low Risk, Medium Risk, and High Risk. Users can choose from five locations (Department Store, Restroom, Workplace, Library, and Friend’s House) using four different Data Types (Biometric, Video, Tracking, and Temperature) and three sensors combination at most. A drop-down menu will be provided for users to choose their desired Location and Data Type for a straightforward visualization approach. After gathering minimal data from our participants, our system will create newly designed features and apply them to our trained model to make a suitable recommendation. Upon testing
our recommender tool, participants will have to use The NASA TLX scale [20] to assess their mental workload and help us study the effectiveness of our privacy tool in mitigating the privacy fatigue phenomenon. We will also use the SUS questionnaire [59] to assess the usability of IoTPP and further enhance our design.

7.2 Design and Methodology

IoTPP Web App was built to transparently help users deal with their privacy preferences while respecting their anonymity. When designing our Web App, users’ choices and privacy perceptions were considered the core of our approach. We also tried to seamlessly guide our users through our Web App and provide them with only the necessary information for a pleasant experience. IoTPP was designed using the Flask framework [22], as it represents a convenient way to build web applications using python. Flask is implemented on Werkzeug, a comprehensive WSGI (Web Server Gateway Interface) web application library, and Jinja2 libraries to create HTML templates [22]. IoTPP was hosted using Pythonanywhere [9], an online IDE and Web hosting platform-based python programming language. We used the Prolific platform [16] to recruit Participants and test our recommender system. Participants recruited via Prolific were fully anonymous. Each participant was assigned a unique identification number to be used as their ID in our Web App. Prolific platform do not store any data on their servers but rather let the researchers share their Web application links and collect the data on their own. Data collected through the IoTPP Web App will be stored in a secure database using MySQL services [12].

When interacting with IoTPP Web App, participants must first enter their unique Prolific ID to keep their activity anonymous, as described in Figure 7.1. The collected ID number was only used for analysis purposes and was not associated with participants’ personal data. Users will then be redirected to a page to assess their privacy level. Users can select from
three different propositions that better reflect their privacy perceptions to help the system aggregate the level of privacy tailored to their needs, as shown in Figure 7.2. Privacy levels built on Westin’s model are used as a feature in our trained model to make a suitable recommendation for a particular scenario. Upon choosing the privacy level tailored to their personalities, users will be redirected to an information tab to get familiar with different locations and sensor categories (Figure 7.3). Previous studies showed that users’ lack of knowledge was a driver of privacy leaks and information disclosure [135]. For that purpose, we explicitly added the categorization approach we built in Chapter 4 to help our participants understand the risk associated when interacting with different features. Users are redirected to our main privacy lobby after familiarizing themselves with different sensor and location levels. Figure 7.4 illustrates how participants can create various IoT scenario combinations in a drop-down menu fashion. Users may choose a location of their liking and up to three IoT sensors at once. Our design needed to be minimal not to overwhelm participants and make it easier for them to visualize themselves in different IoT environments. Once the scenario is created, users can use our recommender system to get a recommendation built upon our trained model. As it can be seen from Figure 7.5 IoTPP displays the desired scenario along with the Privacy Risk recommendation associated with it. We used different color palettes for each recommendation to indicate each risk level’s importance.

When designing IoTPP, our purpose was to test the accuracy of our recommender system and enable users to disagree with our recommendation if it did not suit their privacy expectations. The goal of shifting from a binary classification to a recommender system was to guide users instead of forcing them into a specific decision. To achieve this goal, we decided to collect user feedback in two ways:

- If the recommendation is suitable, users are redirected to a new tab, as shown in Figure 7.6, where he/she can repeat the experiment with a different feature combination.
Figure 7.1: IoTPP Homepage: Collecting users’ IDs (ID numbers are not associated with personal data)

Figure 7.2: IoTPP privacy level assessment page: Select a statement that best describes a user privacy perception.

• If the recommendation is unsuitable, we redirect them to a new window to collect their desired Privacy risk. In Figure 7.7, we showcase how we collect the user Privacy Risk feedback along with a comment tab to gather more information when needed. We then store all the data in a database to further enhance our dataset and build a robust future model.

The Web APP was built based on feedback from previous experiments only to capture the necessary information and make it easy for users to navigate through IoTPP without frustrations. Our main goal was to give control back to users while guiding them through privacy recommendations. We decided to recruit 30 participants of all ages, genders, and backgrounds as long as they were over 18 years old and fluent in English. Upon completion of
the experiment, each participant was compensated $2.63 based on an estimated completion time of 15 minutes. The paid amount is based on a fair hourly rate of $8 per hour calculated on the website. Each participant was asked to interact with the Web App twice, fill out a NASA-TLX questionnaire, and respond to the SUS scale questions. We used a Google form to guide our participants through the entire experiment process. On a final note, we asked our participants if they encountered errors while using the IoTPP web App.
Figure 7.5: IoTPP recommendation page: Providing users with a privacy risk assessment and scenario visualisation.

Figure 7.6: Suitable Recommendation Page: The privacy recommendation aligns with users' privacy needs.

Figure 7.7: Collecting Users’ Feedback when the recommendation is unsuitable.
7.3 Evaluation

7.3.1 NASA TLX

NASA TLX (Task Load Index) is a subjective assessment tool used to assess a user’s workload when interacting with human-machine interface systems [114]. NASA TLX was developed by NASA ARC (Ames research center) and quickly became a gold standard in multiple applications for measuring perceived workload when accomplishing a task or immediately afterward [113]. NASA-TLX consists of six subscales representing different measurement clusters that reflect each factor’s contribution to the workload of each activity [114]. The scale accounts for Mental Demand, Physical Demand, Temporal Demand, Frustration, Effort, and Performance [113]. In our experiment, we will be using the Raw NASA-TLX, which excludes the weighting step of NASA-TLX and is proven to produce highly correlated results compared to the original approach [215, 113, 105]. To interpret the scores obtained from our NASA TLX, we will be using Hancock at al.[110] approach, as illustrated in Figure 7.8. Upon task completion, users were redirected using our Google Form to complete the NASA-TLX scale and upload their results to our google servers. We were able to hire 30 participants through the prolific platform and measured the average of each subscale separately, as depicted in Figure 7.9. Our results varied between 14.5 for Physical Demand and 30.5 for Effort, with an average of 22.9 among all subclasses.

7.3.2 System Usability Scale

To evaluate the usability of IoTPP Web App, we decided to use the System Usability Scale (SUS). The SUS scale, also called the ”Quick and Dirty” survey, developed by Brooke in 1996 [59], provides an easy and convenient way to assess the usability of a product [46]. The SUS scale, built on a Five-point Likert scale, is a flexible technique able to assess the usability of a wide range of technological interfaces, from novel hardware platforms to in-
<table>
<thead>
<tr>
<th>Workload</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low</td>
<td>0-9</td>
</tr>
<tr>
<td>Medium</td>
<td>10-29</td>
</tr>
<tr>
<td>Somewhat high</td>
<td>30-49</td>
</tr>
<tr>
<td>High</td>
<td>50-79</td>
</tr>
<tr>
<td>Very high</td>
<td>80-100</td>
</tr>
</tbody>
</table>

Figure 7.8: The Interpretation Score of NASA TLX Scale [197, 110]

![Nasa RTLX](image)

Figure 7.9: Subscale Results of NASA RTLX.
The System Usability Scale also represents a straightforward score calculation easily understandable by a wide range of users with no experience in human factor analysis [46]. Research has shown that a score above 70 illustrates a usable product, with better products scoring in the high 70s to the upper 80s [197, 46]. It is worth mentioning that products scoring below 50 are considered unacceptable and products with a score between 50 and 70 are judged to be marginal [46]. Figure 7.10 shows a SUS score breakdown that describes a tool’s usability in terms of acceptability. To evaluate participants’ satisfaction, users were prompted to partake in the SU Scale presented in our google form. We used the same method presented by Brooke in [59] to calculate the usability scores. After gathering all the data from the ten different questions among 30 participants, we ended with a final SUS score of 73.91 and a standard deviation of 16.3. We recorded a low individual score of 22.5 and a highest individual score of 95. We also report that 70% of our participants scored above 70 (acceptable range), 23% scored between 50 and 70 (marginal range), and 7% scored below 50 (unacceptable range). Figure 7.11 summarizes the SUS score for all 30 participants.
7.4 Observations, Limitations, and Future Work

Upon analyzing the overall results of the NASA RTLX scale and based on Hancock et al. [110] approach, we can state that our users’ perceived workload is "Medium," as it falls under the range between 10 and 29. Those results are encouraging as it illustrates that our IoTPP web App was perceived as easy to use, and users were not fatigued or burdened when interacting with the Web App. The highest value recorded among all subclasses was Effort, with a score of 30.5, which falls under a "Somewhat High" workload. Effort indicates the amount of hard work (mentally and physically) users had to achieve to complete the required task [113, 114]. We hypothesize that this stems from the fact that 35.5% of participants encountered internal server errors when testing IoTPP. This constructive feedback will push us to develop our web app further and use better Host services in the future. Due to our limited set of participants, we cannot claim that our IoTPP Web App was able to fully mitigate the concept of Privacy Fatigue. We also realized that other research [217, 32] had used a differ-
ent NASA RTLX approach by comparing the results of different IoT approaches. In future work, we will ask participants to accomplish the same tasks without using our IoTPP tool. IoT simulators such as Cooja Contiki [4] can be used to set up IoT scenarios for our participants. Their workload will be recorded through the NASA TLX scale and compared to our Web App results. This approach will allow us to use the T-Test to state if the difference in workload is statistically significant between both methods.

Concerning our usability test, our result indicates that we achieved the level of usability a tool would hope to reach. Based on our Score (73.91), we can claim that IoTPP usability is satisfactory and falls under the good usability rating score, as shown in Figure 7.10. We observed that 70% of our participants regarded IoTPP as ”Good” or better ,and 23% evaluated IoTPP as ”OK”. Our Sample size of 30 participants exceeds the recommended number of participants needed to evaluate usability efficiently [35, 236]. It should be noted that the SUS score can not be used in isolation to make an absolute judgment about a product but rather mixed with other usability measurement matrices such as effectiveness and efficiency [89, 237, 10]. Future work will incorporate those measurements by setting up multiple tasks (Web App Vs. Simulator) that participants would have to complete. Although IoTPP achieves an acceptable SU Score, we seek to improve its usability through studying and applying state-of-the-art research to further enhance user UI experience.

IoTPP web app was also designed to measure the accuracy of our recommendation system. We report that based on user input and assuming that all users tested our web app only twice, 37 out of 60 (61.66%) recommendations suited users’ privacy preferences. We will harness this feedback to improve our PPM approach and enhance our Dataset. Future development should take into consideration user-specific choices when the recommendation does not satisfy their expectations. Those preferences should be automatically applied and updated each time the user interacts with the same IoT space. Future work should also incorporate a more realistic approach that is community-driven. In other words, users can report
all discovered sensors tied to a real location on the map and anonymously share their experience with the community and system. This crowd sourcing approach is key to populating a map containing sensors where IoTPP can operate autonomously and make a recommendation based on user privacy level and location. Our current design was an attempt to test and validate our proposed solution in an effort to collect more data and user feedback to improve our trained model, mitigate user privacy fatigue, and evaluate IoTPP usability.
Chapter 8

Conclusion

This thesis addresses how feasible it is to guide users better manage their privacy preferences through restoring users’ trust in IoT spaces. The purpose of this research is to empower users’ complete control over their surroundings without burdening them with a multitude of choices to make. While looking into the challenges of IoT privacy in the literature review, we discovered that the human factor side is a complex subject that needs a flexible solution to fit each identity differently. Our research also revealed that users’ privacy preferences constantly shift, unveiling a disparity between privacy attitudes and actual behaviors. To address those challenges, we had to set up goals to be achieved throughout our research.

To address the first goal, we had to conduct a replication study in order to examine users’ comfort levels concerning data collection in different IoT locations. Our replication study investigation revealed that users are uncomfortable sharing identifiable data tied to their persona. This preliminary study conducted five years apart from the original research yielded different results, especially toward smartphones, a mini IoT system encompassing various sensors capturing information continuously. Our Statistical analysis toward comfort level enabled us to discover the factors that most impacted users’ privacy perceptions when exposed to a range of IoT devices, as shown in Figure 4.3. This preliminary study demonstrated
that being comfortable with sharing sensitive information does not systematically mean users are willing to allow data collection when given a choice.

Our second goal was to use the data collected to categorize the impact of data gathering by different types of IoT sensors and study their impact on users’ privacy perceptions. We used the Generalized Mixed Effect Model (GLMM) to construct our mathematical analysis and study the impact of different factor interactions and their dependencies. As shown in Figures 4.6 and 4.7, our GLMM approach enabled us to review the contribution of different factors that may impact user privacy choices and decisions. The collected results allowed us to categorize different features based on their impact on users’ privacy choices. Our BIC results Towards comfort level and Allow/Deny decisions enabled us to measure the impact of Data Type and Location on user’s Privacy decisions. Our GLMM model estimates entitled us to classify the importance of each data type and location level based on their sensitivity level. Our results signal that users’ privacy perception shifts when they control their environment.

Based on the categorization results, we decided to test if it was feasible to predict users’ Allow/deny decisions using only factors that users have control over. Our third goal was to evaluate different Machine Learning approaches to assess their potential effect on data gathering. This preliminary experiment attempted to mimic real-world environments using fewer features and new categorization techniques for better ML pattern building. We tested our approach using three known classifiers and based our analysis using different statistical measures from the original study for better comprehensibility. Our results indicated that we could achieve comparable performances to the original study, using a smaller dataset and fewer features. This experiment gained us a better understanding of users’ privacy needs and was the driver for creating an informative and transparent approach. The knowledge collected from this evaluation enabled us to introduce new improvements to better guide users manage the risk associated with their data collection.

To address our fourth goal, we introduce PPM, a Machine learning recommender system
built on the enhancements collected from the previous experiments. We start by building a simpler survey that collects only factors that users have control over while creating new features around them to avoid information overload. We evaluated different coding techniques to enhance our results and found the best approach that satisfies users’ privacy needs. We also provided a shift from a binary classification to a privacy risk-level assessment approach to facilitate decision-making and restore user control and trust. Our ML testing phase indicated that we could predict users’ privacy risks based on their location, sensor combination, and privacy Level. Although no comparable results were found in the literature review, we presented all enhancement techniques applied to get the level of performance we achieved.

The feeling of vulnerability combined with users not having the choice of filtering their sensitive data was the driver to build IoTPP, a web application capable of guiding users through their privacy choices. IoTPP, a Web-based version of the Privacy Preference Manager, was designed to achieve our fifth goal and provide users with a tailored privacy recommendation before interacting with IoT spaces. IoTPP allowed users to choose from different locations and sensor combinations using an easy visualization process. Our Web App was also built to help users apprehend the ML prediction and allow them to share their experiences anonymously.

IoTPP enabled us to test the accuracy of our proposed solution, evaluate users’ workload (privacy fatigue), assess the usability of our privacy tool, and gather more data to build a larger dataset and optimize our ML approach. The results collected indicate that:

- 61.66% of the recommendations produced by PPM satisfied the participants’ privacy expectations.
- We achieved a Medium user workload state with an average NASA TLX score of 22.9.
- IoTPP Web App was usable with an average System Usability Score of 73.91 across all participants. 70% of the participants rated IoTPP as “Good” or better and 23%
rated it as "OK".

Besides the future research directions stated in Section 7.4, we plan to provide an Android version of IoTPP to improve the application accessibility and build a larger community. We also plan to further develop our App’s functionality to better assist users in understanding the ML prediction through visualization techniques and user experience.
Bibliography


[106] Isabelle Guyon and André Elisseeff. An introduction to variable and feature selection. 


[109] Jiawei Han, Jian Pei, and Micheline Kamber. *Data mining: concepts and techniques*. Elsevier, 2011.


[240] Sandra Wachter. N o r m a t i v e c h a l l e n g e s o f i d e n t i f i c a t i o n i n t h e i n t e r n e t o f t h i n g s : P r i v a c y , p r o f i l i n g , d i s c r i m i n a t i o n , a n d t h e g d p r . *Computer law & security review*, 34(3):436–449, 2018.


