Machine Learning Techniques for Identification using Mobile and Social Media Data

Beatrice M. Perez Mila de la Roca

A dissertation submitted in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

of

University College London.

Department of Security and Crime Science

University College London

September 8, 2020

I, Beatrice M. Perez Mila de la Roca, confirm that the work presented in this thesis is my own. Where information has been derived from other sources, I confirm that this has been indicated in the work.

Abstract

Networked access and mobile devices provide near constant data generation and collection. Users, environments, applications, each generate different types of data; from the voluntarily provided data posted in social networks to data collected by sensors on mobile devices, it is becoming trivial to access big data caches. Processing sufficiently large amounts of data results in inferences that can be characterized as privacy invasive. In order to address privacy risks we must understand the limits of the data exploring relationships between variables and how the user is reflected in them.

In this dissertation we look at data collected from social networks and sensors to identify some aspect of the user or their surroundings. In particular, we find that from social media metadata we identify individual user accounts and from the magnetic field readings we identify both the (unique) cellphone device owned by the user and their course-grained location. In each project we collect real-world datasets and apply supervised learning techniques, particularly multi-class classification algorithms to test our hypotheses. We use both leave-one-out cross validation as well as k-fold cross validation to reduce any bias in the results. Throughout the dissertation we find that unprotected data reveals sensitive information about users. Each chapter also contains a discussion about possible obfuscation techniques or countermeasures and their effectiveness with regards to the conclusions we present. Overall our results show that deriving information about users is attainable and, with each of these results, users would have limited if any indication that any type of analysis was taking place.

Impact Statement

Data is being collected at an unprecedented rate. More than 100 networked devices are connected every second [1]; each of which, report on some facet of the life of its owner. In this dissertation we explore the identification risk of two types of data: sensor data and social media data.

The greatest impact of this work can be found in the research methods and the general methodology we present. In terms of algorithms, we show an implementation of the divide-and-conquer design paradigm for a Random Forest classifier allowing the algorithm to process successfully tens of thousands of classes; then, also algorithmically but in the field of time series classification and analysis, we present a classifier that takes into account the form (*i.e.*, the shape or outline) of a signal and uses it to determine the best fit of an unknown unlabeled observation. In terms of applications, we propose a methodology to analyze metadata and sensor information deriving from them unique sets of identifiers that link back to user accounts, in the case of metadata, and mobile devices, in the case of sensor readings. Finally, we present a proof-of-concept project for a coarse-grained localization algorithm that relies on potentially crowdsourced sensor readings for accuracy.

Outside of academia, the empirical results we present can be used to inform public policy and discourse. To achieve this, the most difficult task is making the relevant information accessible to the right people. Having the work published in competitive journals and presenting the work in conferences and workshops to experts on the field raises the profile of our conclusions, potentially making them more accessible to public servants and members of governing bodies interested in the topic. Realistically, the expectation is that the impact and relevance of this work

will become more evident in the coming years once it has been accepted by the discipline and validated through the works that carry on from our conclusions. For the moment, engaging in collaborations both within the UK and abroad in the topics presented in this dissertation can help set the tone in the ongoing discussion that is the topic of data analysis and privacy.

Acknowledgements

It is impossible to reach this milestone without the support, inspiration, and encouragement of a million people. Here I can only acknowledge a few but I am grateful to all. First to my family: to the ones given and the ones chosen. To my parents for teaching me to think and to handle each turn in the road with a steadfast commitment to a true plan. To my sisters and grandmother for making me believe that I could do this, even when I was almost sure that I couldn't; and my brother for providing the playful reminder that if things were easy they wouldn't be worthwhile. To my other sisters for listening to my rants and showing nothing but understanding.

Second, to all the friends along the way. To my academic siblings and the extended members of the research group: Abhinav, Ben, Mariflor, and Victor. Your conversations, contributions, and companionship have made this experience incredibly rewarding. To the dozens of amazing flatmates that were recruited, in the course of my research, to participate in crazy experiments! The number of awkward moments in the name of science will be the source of good stories for the rest of our lives. Your support has been invaluable.

Finally to my mentors, both official and informal. To my supervisors, Mirco Musolesi and Gianluca Stringhini: I have learned (though I suspect it is an ongoing process) the intricacies of research and the surrounding fluff. Ultimately, your efforts have led me here, to graduation. And for this, I will be forever grateful. To my managers and collaborators, you too have helped in my professional development and the skills and lessons learned will stay with me throughout my career.

And to the few that fit into all of these categories: I could not have done this without you. Thank you all.

Contents

1 Introduction **15**

1.1 Overview 15

1.1.1 Digital Traces and Identification 16

1.1.2 Digital Traces and Privacy 17

1.2 Motivation 18

1.3 Research Questions 19

1.4 Structure and Contributions 20

1.5 Scope 22

1.6 Publications 23

2 Literature Review **24**

2.1 Definition of Privacy 24

2.2 Attacks on User Privacy 26

2.3 Basic Notions on Privacy 28

2.4 Privacy Mechanisms in Social Media and Mobile Devices 30

2.4.1 Security in Mobile Devices 31

2.4.2 Privacy in Online Social Networks 32

2.5 Social Networks and the Risk of Identification 34

2.5.1 Identification Through Text 34

2.5.2 Identification from the Network 36

2.5.3 Linking Information Across Datasets 38

2.6 Metadata and Identification 39

2.6.1 Biometrics: User Authentication from Physical Properties . 39

2.6.2 Disruption: Identifying Online Misbehavior 42
2.7 Identification of Mobile Devices 45
2.7.1 Content-Based Identification 45
2.7.2 Hardware-Based Identification 46
2.8 Identification of Places: Localization 48
2.8.1 Model-Based Localization 48
2.8.2 Fingerprint-Based Localization 49
2.9 Contribution with respect to the state of the art 51
2.9.1 Account Identification 51
2.9.2 Device Fingerprinting 52
2.9.3 Localization 52

3 Machine Learning: an overview **53**
3.1 Multinomial Logistic Regression 54
3.2 Random Forests 54
3.3 k Nearest Neighbors 55
3.4 Artificial Neural Networks 56
3.4.1 Convolutional Neural Networks 57
3.4.2 Siamese Networks 57

4 Identification of User Accounts from Twitter Metadata **59**
4.1 Motivation 62
4.1.1 Formal Definition of the Study 62
4.1.2 Attack Model 63
4.2 Methods 64
4.2.1 Metadata and the case of Twitter 64
4.2.2 Feature Selection 64
4.2.3 Implementation of the Classifiers 65
4.2.4 Obfuscation and Re-Identification 65
4.2.5 Inference Methods 66
4.3 Experimental Settings 67

4.3.1 Dataset . . . 67
4.3.2 Ethics Considerations . . . 67
4.3.3 Experimental Variables . . . 68
4.4 Results . . . 70
4.4.1 Identification . . . 70
4.4.2 Obfuscation . . . 74
4.4.3 Execution Time . . . 75
4.5 Discussion and Limitations . . . 76
4.6 Summary . . . 77

5 Device Identification from Magnetic Field Emissions 78
5.1 Background: Privacy and Electromagnetism . . . 81
5.2 The Magnetometer Sensor . . . 82
5.2.1 The Hall Effect . . . 83
5.2.2 Sensor Calibration . . . 84
5.2.3 Software Interface . . . 85
5.3 Overview of the Attacks . . . 86
5.3.1 Malware Attack: Identification Through the Analysis of the *Bias* reported by the sensor . . . 86
5.3.2 Physical Proximity Attack: Identification Through Readings Collected Through an External Device . . . 87
5.4 Description of the Identification Model . . . 88
5.5 Malware Attack: Identification through the Bias Readings . . . 89
5.5.1 Ethics . . . 90
5.5.2 General Characteristics of the Dataset . . . 90
5.5.3 Classification Task . . . 92
5.5.4 Feature Generation . . . 92
5.5.5 Results . . . 93
5.5.6 Limitations and Countermeasures . . . 100
5.6 Physical Proximity Attack: Identifying One Device from Another . 101
5.6.1 Experiment Setup and Data Collection . . . 101

5.6.2 Feature Generation . . . 102
5.6.3 Results . . . 104
5.6.4 Limitations and Countermeasures . . . 105
5.7 Discussion . . . 106
5.8 Summary . . . 106

6 Location Inference from Magnetic Field Data 108
6.1 Preliminaries and Motivation . . . 111
6.1.1 Key Concepts . . . 111
6.1.2 Motivation . . . 113
6.2 Methodology . . . 114
6.2.1 Techniques for Feature Extraction . . . 114
6.2.2 Criteria for Assessing Prediction Models . . . 120
6.3 Dataset . . . 121
6.3.1 Data Collection . . . 121
6.3.2 Description of the Dataset . . . 122
6.3.3 Similarity Between Measurements . . . 124
6.4 Results . . . 124
6.4.1 All Places, All Devices Evaluation . . . 124
6.4.2 Leave-a-Place-Out Evaluation . . . 125
6.4.3 Leave-a-Device-Out Evaluation . . . 129
6.4.4 Discussion . . . 132
6.5 Summary . . . 133

7 Conclusions 135
7.1 Summary of Contributions . . . 135
7.2 Discussion . . . 138
7.3 Limitations . . . 139
7.4 Future Work . . . 140
7.5 Outlook . . . 142

Bibliography 143

List of Figures

4.1 Change in accuracy for a single feature combination and increasing users. 68
4.2 Performance of the top 20% of combinations per classifier for increasing observations per user. 68
4.3 Averaged model accuracy for logarithmic-step user size increase. . . 73
4.4 Performance of the most popular features for increasing tuple size. . 74
4.5 Change in predictive accuracy per classification algorithm for increasing percentage of input data obfuscation. 74
4.6 Mean execution time as a function of features. 75
4.7 F-score for increasing intermediate sample sizes. 75

5.1 Measuring the Magnetic Field from a MEMS device. The flow of current (I) through a conductor polarizes the material. The resulting voltage between opposite sides is known as the Hall Voltage, from which you can derive the magnetic induction B. 82
5.2 Effect of hard-iron and soft-iron distortion on magnetic field readings [2]. 85
5.3 The magnetometer collects 3D readings of the magnetic field in the vicinity of the device. The same axes are used to eliminate the signal emitted by the device. 87
5.4 Screenshot of the main activity of the application. 91
5.5 Magnitude of the *bias* for a subset of users. 91
5.6 Determining the impact of the number of observations per device for 175 devices. 94

5.7 Change in classification F-Score for 20 users, 100 readings per user, and 500ms interval increase between consecutive measurements. . . 95
5.8 Influence of battery state on the generated magnetic field 96
5.9 Cross validation results for the identification of four different Nexus 6 devices. 98
5.10 Distribution of the values of the *bias* for each axis. 99
5.11 Magnetic field associated with four smartphones. 103
5.12 Classification F-Score for increasing duration of training signal. . . 104

6.1 Overview of the zero-permission attack that leverages the magnetometer of mobile devices to capture magnetic field readings and then infer the location-type of the place where the user is currently situated from the magnetic signature of that location. 110
6.2 The visual representation of a *shapelet*. In the figure, each sequence corresponds to an 80 second time-series present in at least 90% of the observations for each location-type. 118
6.3 Histogram of distances of within-class observations. 122
6.4 Full-Signal Matching: Class accuracy for Dynamic Time Warping proximity based classification. 125
6.5 Statistical Descriptors: Class Accuracy for XGB Classifier. 126
6.6 Automated Features: Classification Class Accuracy for 4-Layers CNN. 126
6.7 Shapelets: Class Accuracy for RF Classifier. 127
6.8 Leave-a-Place-Out, Combined Feature Set: Class Accuracy for RF Classifier. 127
6.9 Class accuracy for Euclidean Distance proximity based classification. 129
6.10 Class Accuracy for RF classifier. 130
6.11 Classification Class Accuracy for 3-Layers CNN. 130
6.12 Class Accuracy for XGB Classifier. 131
6.13 Leave-a-Device-Out, Combined Feature Set: Class Accuracy for RF Classifier. 131

List of Tables

4.1 Description of relevant data fields. 63

4.2 KNN classification accuracy using ten observations per user of two features follower count and friend count for input. We ran each experiment for an increasing number of users u. 67

4.3 Entropy calculation for feature list. 71

4.4 KNN Classification using dynamic features. 72

4.5 RF Classification using dynamic features. 72

4.6 Accuracy of the top combination for n number of inputs for the KNN classifier. 72

4.7 Accuracy of the top combination for n number of inputs for the RF classifier. 72

4.8 Accuracy of the top combination for n number of inputs for the MLR classifier. 73

5.1 Starting from the left, column one lists the two types of magnetic field events. The middle columns contain the size and description of the response. Finally, the column on the far right contains the units corresponding to the values reported. 82

5.2 Precision and Accuracy for the classification of 175 devices. In the table, s denotes the number of readings or samples included in the training set. 94

5.3 Description of relevant data fields. 101

6.1 Leave-a-Place-Out. 125

6.2 Leave-a-Place-Out. 126
6.3 Leave-a-Place-Out. 126
6.4 Leave-a-Place-Out. 127
6.5 Leave-a-Place-Out: Shapelets and Statistical Descriptors Combined. 127
6.6 Leave-a-Place-Out: Frequency domain. 127
6.7 Leave-a-Device-Out. 129
6.8 Leave-a-Device-Out. 130
6.9 Leave-a-Device-Out. 130
6.10 Leave-a-Device-Out. 131
6.11 Leave-a-Device-Out: Shapelets and Statistical Descriptors Combined. 131
6.12 Leave-a-Device-Out: Frequency domain. 132

Chapter 1

Introduction

On the Internet, nobody knows you are a dog.

Peter Steiner

1.1 Overview

Interaction with digital devices is the backdrop of modern society. Smart TVs and refrigerators, network connected toothbrushes and scales, navigation systems, satellite radios, access tokens, loyalty cards, social media and personal electronic devices are just a few examples of the technologies that are constantly collecting and generating data about users and their environment. Individuals are often unaware, unqualified or unwilling to disable this ever-present machinery of collection. Despite the privacy risks they might bring, and perhaps because it provides useful services to ultimately enhance user's experiences, location services allow navigation apps to guide users to new places or through changing traffic conditions, open (free) WiFi access points reduce cellular data consumption, motion sensors in a phone are legitimately used for gaming. Keeping track of sensors and access to data is an expensive mental task that users find difficult to justify [3].

Companies have taken on the responsibility of incorporating the necessary (digital) protections to their products. The security platform for both mobile operating systems and online social networks, for example, provides sandbox environments for third-party applications and controlled access to data generated through the use of the product [4, 5]. Nonetheless, decades of high-profile data

breaches [6, 7, 8], (state level) cyber-attacks [9, 10], high-impact malware such as the ransomware epidemic that infested the English National Health Services [11], as well as targeted education campaigns by both government and social enterprises [12], have led to stronger legislation surrounding technology and the appropriate use of data. Overall, regulations set basic standards across providers, address the question of legitimate access, set boundaries on the geographic dissemination of data, and give special provisions for storage, analysis, and use of personal data [13]. While this all accounts for progress, it is not enough. Data collected by the different sensors in personal devices, as well as the records generated from our interactions with technology in general are often not considered *invasive* and are therefore not covered by the protections of the many privacy regulations in effect today [14]. To some extent this is justified: on their own and with only superficial analysis, these records do not reveal anything of importance about any single user. However, once we realize that the data about an individual is actually part of an aggregated dataset, which includes all users and when we consider that the period of collection spans for years, if not decades, we then realize that the analysis of such a rich dataset reveals a great deal about the behaviour of an individual and their environment.

The data collection platforms we focus on, and which are expected to grow in use, are comprised of multi-modal sensor platforms available in personal (mobile) devices and social media data that contains real-time reports by people on the ground. All of this information combined provides data on both individual users and aggregate social behaviors; from the user's perspective, this data is commonly referred to as the user's digital trace [15, 16, 17].

1.1.1 Digital Traces and Identification

Many of the services available on the Web require, if not the natural identity, some form of unique identifier where each user receives or can access specialized content in the form of access to data they posted or purchased, recommendations based on their past behavior, connection to the digital version of the services they subscribe to (*e.g.*, banking, utilities), etc. This is analogous to the physical world where "an individual becomes a person when he or she has a recognizable identity"[18].

Identification is the process of one-to-many matching [19]. In the physical world, this happens naturally: we assign to an individual a name (*i.e.*, label) and a set of characteristics. The better we know a person, the more and more distinct are the descriptors we use. Similarly, in the digital world, the label depends on the service or the group being studied but the characteristics are derived from the digital traces relating to each entity.

1.1.2 Digital Traces and Privacy

Privacy is a human right [20]. Privacy was famously defined by Thomas Cooley in the late 19th century as "the right to be left alone" [21]. Under common law, each individual has "the right of determining to what extent his thought, sentiments and emotions shall be communicated to others" [22]. This is the general principle: we have a right to control what is ours, but more than that, this right is not merely applicable to property nor is it limited to certain data fields relating to information about individuals. We have a right to: private and family life, home, and communications; the protection of personal data; freedom of thought, conscience, and religion; freedom of expression and information; freedom to conduct a business without intrusion and interference; the right to an effective remedy and to a fair trial (i.e., compensation for damages as determined by a court) [23], which combined constitute our privacy.

Personal electronic devices provide a window into our life. In 2019 there were 19.4 billion active network-connected devices [1]. Each of these devices will contribute to the digital trace of the owner. Deriving behavioral traits from data without explicit permission is more than a simple invasion of privacy [15]. Without proper oversight, this information might be used to make determinations about people (and their abilities) without their consent. For example, our shopping habits might be used to predict our financial trustworthiness and be used in the case where we would like to take out a loan, analyzing our driving skills might be used as a factor of consideration for the premium of car insurance, or our job performance might be affected by inferences drawn from our social engagement calendars, just to name a few.

Mobile Devices and Social Media platforms are two sources for extracting information about users. Processing video or images results in the identification of individuals depicted in the media (which has been used to ban patrons from gambling establishments) [24]; similarly, a set of images from the same photographer can be used to reveal the hometown of the photographer [25]; and the processing of written text can lead to the approximate age and gender of its author [26] and their mood at the time of the post [27].

Most data protection legislation includes statements about the analysis and storage of personal information. In the UK, the data protection legislation states that "personal data should be collected for a specic purpose and should be adequate, relevant and not excessive in relation to the purposes for which it is collected and processed" [28]. Our point of contention is that personal information is a potentially poor approximation for what the law is supposed to cover. In a world where "even the limited information of partial profiles may be sufficient for abuse by inference on specific features only" [29], we must look at all types of data and what they reveal.

In this dissertation we will focus on the problem of identification considering the specific examples of Twitter data (in particular metadata associated to the posts of an account) and magnetometer data from mobile phones. We believe that this thesis considers key examples that are representative of a large set of identification problems based on the analysis of personal digital traces.

1.2 Motivation

The whole of the dissertation presents problems through the lens of applied work. In Chapter 4, we use Twitter to discuss what in the literature is referred to as opportunistic data collection [30]. Communication channels, software applications, financial transactions, these are only a few examples of services that attach to their primary content information about the record generated. Unlike the primary content, the storage, processing, and sharing of metadata is unregulated and is often at the core of the business model of "free" online services. The purpose of Chapter 4

is to explore whether behavioral traits and descriptors contained in metadata are reliable features for identification and argue that if it is, then opportunistic data should be treated with the same care as more sensitive data. Twitter was the best source of information. It provided ground truth and access to millions of homogeneous users (we removed extreme accounts).

In Chapter 5, we use cell phones to study a universal method for electronic device identification. The Internet of Things and "Smart" Environments are, and will continue to become, ubiquitous. In order to protect and integrate new technology into our life, it is vital that we are able to first identify distinct devices. Cell phones are a good litmus test. Off-the-shelf, they are sensing platforms already integrated into society. They provide a wide range of software and hardware configurations and through the use of apps allow for a viable data collection process.

Finally in Chapter 6 we focus on location data and address two challenges: privacy for users and an inexpensive coarse-grained localization method for applications. In terms of privacy, location data is extremely sensitive. Location monitoring can reveal social preferences, state of mind, and predict where a user might be in the future. Investigating whether sensors on mobile phones are leaking sensitive information is, to our perspective, important. The insight is that the magnetic field is a naturally-occurring phenomena that reacts to human activity. Identifying and measuring patterns in the resulting local field might give some information about the location.

1.3 Research Questions

More specifically, in this dissertation we address the following research questions. First, considering the unique patterns of user behavior captured in social media interactions, we ask: *is it possible to identify an individual from a set of metadata fields from a randomly selected set of Twitter user accounts?*

Then, we direct our attention to mobile devices focusing on a specific sensor embedded in today's mobile phone. For the second project, we leverage the relationship between electricity and magnetism. The basic premise of this study is that

the electric current that flows through circuits inevitably generates a magnetic field. From this, the research question is then: given a set of devices, *are the differences in the characteristics of the radiated emissions between each device sufficient to uniquely identify each one of them?*

Finally, continuing from the previous work, we use magnetic field readings from the phone's sensor and focus on a coarse-grained localization problem. Magnetism is a prevalent force in nature that is sensitive to man-made structures, electric (and electronic) devices, and human activities. Given the widespread availability of magnetic field sensors, *can we identify the current location-type*[1] *of a user from low-power sensor readings present in their phones?*

Following the unifying theme of identification and privacy, in addition to the questions listed above, we will also consider the implications of these scenarios and discuss the potential countermeasures to each one of them.

1.4 Structure and Contributions

In the following we will discuss the structure and contributions of this dissertation in detail. In Chapter 2 we discuss the state of the art in identification from online social networks and mobile sensor data.

Throughout Chapter 4 we will look at online social networks and in particular at the metadata contained in each message. We will use Twitter as a case study to quantify the uniqueness of the metadata in relation to user account identity. We will also consider the effectiveness of potential obfuscation strategies. In this chapter, we define metadata as any information that relates to a post. In the case of Twitter, metadata contains information about the message being posted (*e.g.*, the time of the post) and the account from which it is posted (*e.g.*, the number of friends linked to the account), both of which are embedded in the post. Previous research has mainly been focused on the problem of the identification of a user from the content of a message. In reality, the footprint of the message is much smaller than the metadata

[1]The term location-type will be formally defined in Chapter 6. For now, think of the location-type as the category to which a distinct place belongs. As an example, one way to categorize UCL would be as the type *University*. Some other categories can be: parks, bridges, subway stations, coffee shops, etc.

it generates, and despite the volume of information, it is often still categorized as non-sensitive. In this chapter, we analyze atomic fields in the metadata of each tweet and systematically combine them in an effort to classify new tweets as belonging to a particular user account. We apply three supervised-learning classification algorithms that abstract from the features provided to predict the account identifier of unseen (unlabeled) posts. Using Random Forests, k-Nearest Neighbors, and Multinomial Logistic Regressions we demonstrate that we are able to identify any user in a group of 10,000 with 96.7% accuracy. Moreover, if we broaden the scope of our search and consider the 10 most likely candidates, we increase the accuracy of the model to 99.22%. We also found that data obfuscation is hard and ineffective for this type of data: even after perturbing 60% of the training data, it is still possible to classify users with an accuracy greater than 95%. These results have strong implications in terms of the design of metadata obfuscation strategies, not only for Twitter, but more generally, for other social media platforms with similar circumstances.

Then, in Chapter 5, we consider sensor data from cellphones. Specifically, we show how magnetic field emissions can be used to generate fingerprints that are unique to each device. Previous works on device identification rely on specific characteristics that vary with the settings and components available on a device. This, in turn, limits the number of devices on which any single approach is effective. By contrast, all electronic devices emit a magnetic field, which is accessible internally through the Application Programming Interface (API) or externally through a sensor placed in proximity to the device. In this project, we conducted an in-the-wild study over a period of four months and collected mobile sensor data from 175 devices. In our experiments we observed that the electromagnetic field reported by the magnetometer identifies devices with an accuracy of 98.9%. Furthermore, we show that even if the sensor was removed from the device or access to it was discontinued, identification would still be possible from a secondary device in close proximity to the target. Our findings suggest that the magnetic field emitted by smartphones is unique and fingerprinting devices based on this feature can be performed without any interaction from users.

In Chapter 6 we present techniques for inferring the location-type from magnetometer readings. Location data is particularly invasive when considered through the lens of privacy. Indeed, location information extracted from mobile devices reveals our routine, significant places, and interests, just to name a few. Given the sensitivity of this information, location information (*i.e.*, GPS and network location) is protected by mobile operating systems and users have control over which applications can access it. We argue that applications may still infer coarse-grain location information by using alternative sensors that are available in off-the-shelf mobile devices and do not require any permissions from the users. In this chapter, we present a zero-permission location inference attack where each location is identified on the geomagnetic characteristics of its environment. We analyze over 90 hours of time-series magnetic field data collected with 5 devices from 110 distinct locations throughout London and consider four different feature extraction techniques in the classification. We present our results using two evaluation criteria: leave-a-place-out which simulates identifying an unknown location from within a set of known categories and leave-a-device-out, which simulates identifying known locations from unknown devices.

Chapter 7 concludes this dissertation summarizing the key findings, limitations of the methods and in general of the work, and proposing some projects where our work could be used as the basis to tackle open research problems.

1.5 Scope

Identification and authentication are broad topics. By choice we focus on specific application problems. Even if the proposed techniques and methodologies can be applied to a broader set of problems we do not make any claims in terms of generilizability. Whilst we developed mobile applications for some of these tasks, we do not create tools or systems for identification. Moreover, in each chapter, we also discuss countermeasures against the proposed attack, but as you will see, we base our attacks on intrinsic properties of the technology employed and, as such, discussions around the countermeasures are mostly strategies to mitigate risk by shifting

the responsibility towards the user.

1.6 Publications

The work presented in this dissertation has been presented as separate projects at different venues.

- The work on Twitter metadata was presented at the 12th AAAI International Conference on Web and Social Media (ICWSM'18) and published in its proceedings [31].
- The work on device identification was presented at the 12th ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec'19) and published in its proceedings [32].
- The work on location-type inference is submitted for publication at the 18th ACM International Conference on Mobile Systems, Applications, and Services (MobiSys'20).

Chapter 2

Literature Review

Thematically, the literature presented in this chapter is split in two parts: first a discussion on privacy — its definition, attacks and the privacy protections available in mobile devices and social networks; and a second part which is more detailed with respect to the areas that will be discussed in the dissertation. In this second part, we begin with a discussion of the use of metadata for identification and classification, followed by the tools and techniques available for device fingerprinting, and finally an overview of the technology deployed for the localization of users and entities.

2.1 Definition of Privacy

Privacy is universal, present in all cultures and all times, though its expression is different depending on the social context of a people [33, 34].

The concept of privacy and technology's relationship to it is constantly being explored and tested. Novelists, lawyers, psychologists, sociologists and now computer scientists have shown an interest in this area. Indeed, many different areas of human life are influenced by it. On one hand, people argue that the institution of privacy promotes a state of 'social hypocrisy' and that complete transparency would move society to a state where there would be no shame as people would realize that behaviors which cause shame, and might therefore be considered private, are common to all. On the other end of the spectrum, proponents suggest that privacy is a necessary condition for love, trust and human relations [35, 36, 37]. It provides psychological protections allowing a person to express ideas and be understood by

the right people in the right context.

In developing a unifying concept of privacy, the literature suggests two approaches. First, a definition-based approach where each variation of the definition will cover a different facet of what privacy is and, second, an experience-like approach where the subjective value of privacy is observed and its attributes derived [33].

Definitions of privacy are varied and provide a focus on different aspects of life. Some regard privacy as a right where the control of information disclosure lies with the person to which it refers; others see privacy as a facade that stands for control (akin to private property) over information, personal identity or physical access.

One point of convergence seems to be that privacy is a necessary condition for freedom in the sense that the very act of observing someone might inadvertently cause them to adjust their behavior. In a free and democratic society, we should all have the right to behave as we will until and unless there is a greater social concern that compels us to override some behavior.

Privacy, then, provides a sphere of protection that works in three dimensions: first, in our ability to maintain a public self in as much as there is a private self, which is protected; second, a psychological protection against harm (or ridicule) that may come from the judgment of others; and, finally, in the formation of new relationships where we allow others access to compartmentalized expressions of ourselves as we develop bonds and align interests when working towards a common goal [24, 33].

Technology, and its facility to transmit and store information, has the power to disrupt the boundaries of privacy from one generation to the next. The Internet, being the archetype dissemination platform, provides real time access to information that can be processed by anyone who views it. Moreover, the threats to privacy are compounded when we consider that digital technologies allow a would be 'observer' to follow a person without respite often overlooking the sensitive nature of what can be revealed [35]. The protection of information privacy is, therefore, the

ongoing, iterable, adversarial task of adding obstacles in the path of those who aim to collect or process information (or make inferences) that are not in keeping with the rights and well-being of an individual [38].

2.2 Attacks on User Privacy

The goal of any adversary in terms of privacy is to increase their knowledge of the entity they pursue [39]. This model applies to transmission channels where an attacker might be interested in identifying either the sender or the receiver or instead be interested in learning something about the message; it applies to static datasets that have been collected, anonymized, and are subsequently released where an attacker might be interested in identifying a record or learning something new about a collection of them — the field of differential privacy has looked at the requirements that must be fulfilled for a static dataset to be truly anonymous; or, alternatively, the model works in a dynamic environment where access to information requires some form of interaction between the adversary and its target, and where the type of information available is bounded by the resources of both parties. The focus of this dissertation is on the latter. We are interested in exploring what can be learned from imperfect data and making inferences about the user. For every case, we specify the resources (*i.e.*, data, sensors, access) the adversary must have at their disposal.

There are two main types of information an adversary might be after: *(i)* the identity of the entity to which the data relates (*i.e.*, identification or identity disclosure) or *(ii)* a set of attributes that relate to that entity (*i.e.*, attribute disclosure) [40, 41, 42].

In this context identity can be defined as "any subset of attribute values of an individual person which sufficiently identifies this individual within any set of persons" [40]. Identity is most often associated with human beings however we would like to point out that commercial and governmental organizations, electronic devices, and generated "bureaucratic" personas (*e.g.*, bank accounts, customer loyalty records, utilities records) all have this property of identity: something that makes them distinguishable from others like it.

Previous work on identification includes both identification of users across datasets and profiling users within the same (even anonymized) data. In de Montjoye *et al.* [41], the authors consider financial records, specifically card transactions, from an anonymized dataset and builds a profile for each user. In general, they are able to accurately assign new transactions to an individual with 90% accuracy. They also explore the impact of gender and income on the accuracy of this classification. Their model, built on historical records, extracts patterns from the time of the transaction, the shop, and value range of the expenses and aggregates purchases in terms of price range and shop category to further understand uniqueness in this context. They find that obfuscating features are of little consequence when trying to protect users' privacy.

There are different applications for the same technology. Whilst [41] builds profiles for users, the techniques developed for identification purposes are used elsewhere for provenance assurance. In their paper Garcia-Romero and Espy-Wilson [42], use a single audio recording clip to determine whether the entirety of the clip was recorded using the same device or whether some part of it had been tampered with and inserted from a separate source into the original. The authors look at microphones and landlines and are able to successfully separate the two categories and determine which device made the measurement. Similarly, there is an ever-growing body of work on biometric identification and authentication. We discuss the characteristics of this particular type of identification in Section 2.6.

Identity is not however the only relevant information that can be learned. Identity is most useful when the target is known, *i.e.*, when we have additional sources of information and the *identity* becomes the key between what is known and what is to be revealed. However, when the entity is unknown, there is valuable information to be learned from the attributes that are revealed by a group. There is a one-to-many relationship between entities and traits [19]. Any entity is described by a collection of traits and each trait reveals some information about the entity. Attribute disclosure is the task of mining or revealing traits that relate to an entity [43]. It is a different task to identity disclosure, not a lesser one [40]. The application of such

research may vary from marketing (*i.e.*, uncovering patterns in aggregated data) to profiling (*i.e.*, uncovering characteristics of a single individual).

In their work, Cumby *et al.* explore the possibility of predicting the shopping list of a user from historical transactional data [44]. The authors use the available purchases made by one individual for the two preceding years to build a personalized model that populates the shopping list for the current visit for that customer. While the motivation of this paper is commercial (*i.e.*, the aim of the paper is to maximize revenue for the store by providing a positive shopping experience), the fact that they predict customers' preferences (thereby increasing the revenue of the store) reveals the shopping behavior (*i.e.*, a trait) of the user.

Using the sensing platform available in smartphones, Weiss *et al.* use the three-dimensional changes in acceleration recorded by the phone to determine the weight, height and biological sex of the user carrying it [45]. In their experiments, they collect data from approximately 70 individuals and apply different supervised learning techniques to make predictions. They show that predicting these traits is possible with an accuracy ranging between 71% (for sex) and 85% (for height). And, while these attributes could be measured from a device they can also be inferred from text as is shown by Rao *et al.* in [26] where they derive the age, gender and political orientation of the author of a tweet.

Ultimately, attacks on privacy originate from different sources of information (*e.g.*, historical records, sensors and public information) and different types of data (textual, numeric measurements, transactions) and result in either the identity or the traits of an entity being revealed, which in turn may be contrary to the physical, financial or psychological well-being of an individual. As such, these risks cannot be overlooked.

2.3 Basic Notions on Privacy

The gold standard for security was proposed in 1977 as the *ad omnia* principle which states that anything that can be learned about a respondent from a statistical database should be learnable without access to the database [30]. Though originally

intended for databases, the *ad omnia* principle has also been extended to cryptography and network security. In this dissertation, the commodity we are studying is data: collecting, analyzing, and sharing data. A privacy-oriented analysis of a dataset will start from existing notions; primarily, statistical database security, differential privacy, and privacy preserving data mining.

A (statistical) database is a collection of records with the ability to create and return relevant subsets from queries posted to the system. The work on privacy for statistical databases was motivated by a need to provide statistical (*i.e.*, aggregate) information about a population without revealing sensitive information about individuals [46, 47]. The techniques can be broadly classified into query restriction (*e.g.*, frequency and sample overlap between queries) and data perturbation (*e.g.*, adding noise to the output of a query, replacing the values in the database with a sample from the same distribution). The problem in presenting data as aggregates is that the literature shows that accurate statistics and record privacy are not mutually attainable [46, 48].

Differential privacy approaches the same problem from the opposite perspective. The problem with the *ad omnia* principle was that a useful database should reveal information in aggregate, information that was not known before. Differential privacy is based on the principle that inserting or removing individual observations should have no substantial effect on the statistics derived from a database [30]. The literature presents different algorithms that are differentially private [49], but in the end, rather than being a firm guarantee, the goal of diferential privacy is to minimize the risk for any single user of being singled out in a dataset.

In the real world, the benefits derived from big data analysis have led the research community to work a situation where the release of some information is inevitable [30]. The purpose of privacy preserving data mining is to look at the analysis of a database and determine the disclosure risks for individuals from it. Agrawal *et al.* study privacy preserving data mining from the point of view of modeling. First, in [47] the authors look at the reliability of models constructed from perturbed (or obfuscated) datasets using decision trees. And then, in [50] they ex-

plore the best algorithms to use in the perturbation process. Verykios *et al.* [51] published a survey on the methods and propose a taxonomy for privacy-preserving algorithms in different contexts.

The privacy implications of continuous online access surpass the boundaries of data storage and analysis [47]. In this chapter we discuss how the literature is relevant to the task of identification: from privacy in payment systems and transactions in Section 2.2 to the leaking of information when combining multiple datasets in Section 2.5.3 and location privacy in pervasive computing in Section 2.8.

2.4 Privacy Mechanisms in Social Media and Mobile Devices

"Privacy is measured by the information gain of an observer" [39]. The scope of this work is limited to the information that a passive observer may extract from a system (*i.e.*, the use of a system in the manner for which it was intended). Our interest is open data. The premise is that this information was released by its owner in a specific context and for a particular purpose. In this work, we explore what unintended information this data may reveal about the users. Furthermore, we concentrate on online social networks and mobile devices as the sources from which we obtain the data for our analysis. With more than 7 billion network-capable devices in use and more than 2 billion Facebook users alone, privacy risks from these widely adopted technologies have universal implications.

The most widely adopted privacy-protection method in both social networks and mobile devices relies on the use of Access Control Lists (ACL). The protections offered by ACLs rest upon regulating the interaction between entities (or resources). Once a request has been initiated, the operating system checks the origin, destination and type of interaction on a table associated to the object of the request. Each entry in the table describes the level of interaction allowed with all other entities. For the request to be granted, the requesting entity must have an entry in the table for the operation to be performed [52]. In the case of mobiles, the requesting entity can be an application installed on the phone or a script running on a browser

whereas the object of the request might be a component of the phone, a peripheral or one of the other applications installed. In the case of social media, the requesting entity might be another user (through an account) or an advertising company whereas the object of the request might be the fields of data associated to a profile.

2.4.1 Security in Mobile Devices

Security and privacy are closely related. Strong security measures are needed to guarantee that the privacy protections in place are adequate. For mobile devices, a secure platform has to take into account physical loss of the device, unauthorized access through any of the radios (*i.e.*, Bluetooth, WiFi, network card), and a wide range of development standards in the software installed on the phone (from the operating system which they control to perhaps inexperienced developers promoting their apps); all while maintaining uninterrupted service and adequate protection to the information stored on the device and accessible through it.

Android and iOS have converged to similar security platforms. First, each application runs in its own environment (*i.e.*, sandbox); they both also have secure chips where the most sensitive information is stored (*e.g.*, payments cards through NFC, biometrics for authentication mechanisms, encrypted data); and finally, they both regulate interactions between system resources and applications through a permission platform where the most relevant resources are ultimately placed under user control [53, 54, 4].

The level of importance of a resource, and by extension its protection under the permission system, depends on its ability to adversely impact either user experience, the device itself or any of the other applications installed in it [52]. Permissions are enforced by the operating system at runtime and, while permissions are usually granted at install-time, users have the ability to enable and disable access at any point. Once the application is running, if the appropriate permissions are not granted, the operation fails [54].

One of the key challenges in designing a privacy-preserving system based on permissions is the balance between usability and protection. It is inherent to a permission-based system that the use cases be defined a priory [52]. One might ar-

gue that the most secure solution would be to place a protection around all possible interactions through the use of permissions. This however is contrary to a positive user experience. Mobile devices are part of everyday life. They are fully integrated into the daily activities of the owners. Placing all responsibility on users (as they would be ultimately responsible for granting or revoking permissions) would compromise this synergy. On the other hand, having the OS block resources (*i.e.*, by not providing Application Programming Interface (API) support) would hinder the ability of these personal assistants to provide timely and relevant information to their users.

2.4.2 Privacy in Online Social Networks

In 2010, 48% of Americans participated in at least one online social network with the global statistic being around the same [55]. Today, that number is up. As of June 2019, 7 in 10 Americans use social networks to access news and entertainment and maintain their social relationships [56].

Online Social Networks (OSN) were created as a means to manage and maintain past, present and future social relations. In some instances, people could replicate their real social networks and share news, pictures, opinions and videos with existing friends; alternatively, users could establish new connections with other users that share hobbies, interests and other relationships. The ability to share and search through profiles along with the ability to communicate are core features offered by OSN [57, 5].

Posting and sharing information is necessary to foster online communities. However, even as part of such open environments, privacy is still a vital service for the protection of users and the continued success of the online platform. While it is true that the privacy and security requirements of OSN vary depending on the purpose of the network and the expectation of its users, there are three main areas that should be considered when evaluating the privacy protections of a service: identity, personal space and communications [5].

First, the service providers should put safeguards in place to protect the identity of its users. Whether this means making sure that unique identifiers are not

traceable back to the physical individual in an anonymous service, or that personal information is only available to those authorized by users to see it, the identity of a user must be safeguarded against the threats of stalking, slander, spamming or phishing [5].

Second, it is the responsibility of the network provider to protect the environment that a user has created for themselves and their established relations, what would be equivalent to the personal space of a user in a face-to-face relationship. This translates in the digital realm to, among other things, the context of the information that was shared. Taken out of context, simple statements could be unduly harmful to the reputation and prospects of a user. Given that one aspect of privacy is to be understood by the right people in the right context, service providers are also responsible for ensuring that unauthorized parties are not able to gain access to a user's personal space.

Finally, most OSN also provide a private communication mechanism between users. Aside from any visibility expectations of public posts, whenever a service offers communication between users, the reasonable assumption made by users is that, when in private communication, only the members engaged in the conversation have access to its content. Once again, guaranteeing this privacy requirement is in the best interest of the service as it might be liable for violations or its reputation damaged by a scandal stemming from this.

In reality, though, while these three areas are basic requirements for a privacy-preserving system, OSN handle sensitive information in the form of followers or friends (*i.e.*, the social graph of a user) and interests uncovered through clicks and search results. As discussed in [58] this information can be used to infer anything from habits and personality traits to health conditions that, if accessed by malicious entities, can cause real harm to members of the network.

Unlike the case of mobile devices where Android is an open source operating system, online social networks have grown on proprietary software. Our understanding of privacy problems depend on the perception of users measured against their settings, interactions with the different platforms through their APIs and scan-

dals, like that of Cambridge Analytica, where private information about users was revealed to third parties.

To date, the privacy controls that have been introduced are incomplete. While it is true that it is inherently difficult to design a permission-based system for users with a wide range of technological expertise [52], studies have shown that the controls in place are oftentimes lacking in terms of safeguarding against a dedicated adversary [59] but most importantly, they are difficult to use and are either not adopted or misinterpreted by the majority of users [55, 57, 58, 60]. In the case of Facebook, a user's inability to successfully enforce their privacy preferences was severe enough to warrant a settlement between the company and the Federal Trade Commission (FTC) on the basis of "deceptive privacy changes" [61].

In short, privacy in an online social network is a balance between the social aspect of the network (*i.e.*, the ability to discover new connections) and the confidential expectation of users on different forms of communication. OSN platforms have yet to converge into a system that supports their data-as-asset business model while protecting the privacy of their members, which results in a wealth of information about users being openly available to the public or to unauthorized third-parties. Finally, this rocky path of privacy exploration has highlighted the tensions between user preferences and the power asymmetry between platforms and users, which results in companies being able to manipulate, to some degree, the cognitive biases in users leaving their information open to exploitation [57].

2.5 Social Networks and the Risk of Identification

With the amount of information posted to each site, social network platforms hold vast amounts of data on their users. In this subsection, we discuss the risks in terms of both attribute and identity disclosure from two types of data available to providers: the text-based content of a post and the network structure of the data.

2.5.1 Identification Through Text

Understanding identity through the written word is a topic of interest in law, history, literature, intelligence, among others [62]. Stylometry is the statistical analysis of

text and has been used to study the authors and styles of Shakespeare, the Bible, and the Federalist Papers [63]. Commonly, stylometric signatures contain markers such as word length, letter usage, punctuation, readability indexes, sentence count, average sentence length, and many others [64, 65]. These features can be combined statistically using for example, Bayesian statistics or through one of the many classification algorithms described in machine learning literature [62, 66, 67].

In addition to the authorship recognition task [67, 68, 69, 70], or perhaps inspired by it, the use of user-generated text has given way for several inferences about authors. In [71], Perito *et al.* use usernames as a means to link profiles over multiple social networks. In their work, they develop a measure of entropy between the usernames associated to each account. They use Markov-chains and Levenshtein distance to capture the similarity between strings containing usernames and ultimately, predict the likelihood of two usernames (from different platforms) resolving to the same physical person.

In [72], Juola *et al.* design an experiment to test the potential of style as a form of continuous authentication. In particular, they measure the keyboard-based behavior of 79 people and collected through repeated tasks over the course of a working week. They find that 1,000 characters are sufficient to uniquely identify each user with an accuracy of 79.3%. In addition to identity, they are able to infer the personality of the author (using the Myers-Briggs Personality Inventory), their gender (76.6% accuracy), and the dominant handedness (96% accuracy).

Using similar methods, Afroz *et al.* apply stylometry to analyze anonymous text in order to identify users that might be posting from different accounts [65]. In their work, they adapt language specific authorship features to handle the jargon and style of underground internet forums. They were able to successfully build classifiers, which included features extracted from the usernames of different accounts, to find (with different levels of confidence) whether different accounts map to the same user.

In terms of large-scale authorship, Narayanan *et al.* [73] present a study where they look at 2.4 million posts taken from 100,000 possible authors and attempt to

find the correct author of some unattributed text. Their results show that in 20% of the cases the authors are uniquely identifiable; and, if they expand their task to reduce the candidate pool of authors, they find that in 35% of cases they can have the correct author be in the top 20 guesses. It is worth noting that in their work they only focus on the same type of text (*i.e.*, blog posts) and under the assumption that authors have not actively tried to obfuscate their style. In this work, Narayanan *et al.* highlight the problem of deanonymization as a threat to free speech. They endorse the US Supreme court ruling that "anonymity is a shield from the tyranny of the majority, a means to protect unpopular individuals from retaliation ... at the hand of an intolerant society" [74] and point to how data processing can potentially pose a real risk to individuals. Brennan and Greenstadt in their work [64] offer a response to this concern; they show how stylometric techniques perform in the face of adversarial attacks. They focus their attention on two categories: obfuscation attacks and imitation attacks. In their work, they recruit participants and assign tasks that simulate writers attempting to hide their real identity. While the experiments they perform are limited in scope due to the number of participants, they find that stylography based authorship techniques are not able to withstand deception.

2.5.2 Identification from the Network

When studied as a graph, an online social network (OSN) consists of edges, nodes, and the attributes associated to each of these elements; usually, nodes are individuals and edges represent friend relationships or flows of information between them [15, 43, 75, 76]. OSN providers are responsible for protecting the privacy of their users, a task which is closely related to their business stability. At the same time, as centralized entities, they provide attackers with a single point of failure placing themselves at the center of potential data breaches [15]. Moreover, and as companies that provide free services to their clients, their primary source of revenue is the information they control. Data released either publicly or to third parties contains random identifiers in place of names or other account IDs [77, 78]. This process known as naive anonymization promises to preserve all the privacy attributes in the network however, this guarantee is valid only as so far as an adversary has "no infor-

mation about individuals in the original graph" [75]. In practice, as we will discuss in the rest of this section, adversaries have multiple methods to obtain auxiliary information.

The literature describes both active and passive privacy attacks constructed around the structure of the network. In an active attack, the adversary has the ability to create links and nodes in the network before the anonymized dataset is released. These will be used as seeds from which the rest of the nodes will be re-identified [59, 79]. In their work, Backstorm *et al.* [79] show that an attacker needs only $O(\sqrt{log\ n})$ seeds to be able to re-identify targeted nodes in any anonymized network. In their experiments, out of 4.4 million nodes and 7 deanonymization seeds, they compromised 2400 edges. While the authros in [79] look at networks as static entities, in [77] Ding *et al.* propose looking at sequential releases of data to deanonymize nodes and edges. Similar to [79], the threat model assumes that the adversary has full knowledge of seed nodes and in this case, the auxiliary information comes from the different snapshots of the network.

Different from the attacks described above, a passive attack is characterized by an adversary who only has observable information of the network. Neighborhood attacks, described by Zhou and Pei in [15, 80], demonstrate the privacy risk to users when a network is anonymized but the adversary has information about the user and their connections. Matching entities and their connections is sufficient to reveal the target. Similarly, Zhelva *et al.* demonstrated the risk of having both public and private profiles in the network [43]. In their paper, some nodes are labeled and some are not (*i.e.*, labeled nodes represent public profiles) but the relationship between nodes is always visible. The privacy risk in this circumstance is that the edges of the graph are potentially sensitive attributes that users with private profiles did not wish disclosed. Similar to [76] presented by Backstorm *et al.*, Narayanan *et al.* take the method presented in [59] and apply it to a dataset from a "big data" competition. They find that by looking at seed reidentification as a combinatorial optimization problem they are able to reidentify 64.7% of nodes and submit the winning entry to a link prediction contest with social network data. Instead of placing seeds in the

network before it is anonymized, they gather auxiliary information from a labeled second crawl. They combine deanonymization with link prediction techniques in order to carry out the identification task.

2.5.3 Linking Information Across Datasets

Linking back to the physical identity of an individual is only one way in which databases can overlap. In general, *linkage attack* is the term used to describe situations in which two or more datasets, a primary data source in combination with one or more databases containing auxiliary information, are joined to disclose traits that would otherwise be unknown [30]. Sweeney [81] presented a privacy protection model that looks at the attributes associated with each record and states that a dataset is k-anonymous if and only if there are k records that respond to each query. In later papers, Machanavajjhala [82] *et al.* and Li *et al.* [39] show that, in certain circumstances, k-anonymity is not a strong enough guarantee to prevent re-identification and propose prior knowledge (of a respondant) and lack of diversity within classes as two conditions that allow for individuals to be re-identified from seemingly anonymous datasets. From an applied perspective, and once again focusing on the relationship between digital and physical identity, there has been indeed several projects that link the two.

The work presented by Wondracek *et al.* [83] combines information from a history stealing attack[1] with membership lists obtained from social networks. The authors use combined group membership to uniquely identify users. In a static dataset they identified 42.06% of users had unique group membership combinations and in a real-world scenario they traced 1,207 users (12.1% of their participants) from the social network Xing.

In [84], Jain *et al.* deploy an identity resolution system that looks at the public attributes in Facebook and Twitter and links accounts across both networks. They divide their task in two: identity search and identity matching and use both profile attributes (defined by the user) and network attributes (defined by the user's con-

[1] History Stealing is a known attach in which a malicious website can extract the browsing history of a visitor [83]

nections). Overall, they collected data from 543 Facebook users and were able to successfully match 39% of those to a respective Twitter account.

2.6 Metadata and Identification

Research-driven legislation has effectively secured Personally Identifiable Information (PII) and has placed financial and criminal penalties on entities that exploit these traits. However, little attention has been focused on the role of metadata. We define metadata as information that describes some primary content. Often perceived as being devoid of meaning, metadata is more readily available than the source to which it refers. Perhaps because of this, metadata could present a significant risk to user privacy.

In this section, we discuss how metadata is used exclusively or in combination with other forms of data as a means to single out elements in a set.

2.6.1 Biometrics: User Authentication from Physical Properties

The field of authentication, and in particular research aimed at biometrics, seeks as its primary purpose to find unique identifiers for individuals that are inexpensive, unobtrusive and with a low false-positive rate [19, 24, 28]. In essence, they condense individuals into metadata (*i.e.*, information about them) for the purpose of identification which is then used to determine whether a user has legitimate access to a system.

Biometrics have become a common authentication method for smartphones and laptops alike. In 2013, Apple released the iPhone 5S with its first generation touchID module [85] with the faster second generation deployed only two years later in the iPhone 6S [86]. While fingerprint scanners had been used as early as 2004 to unlock phones, Apple's release triggered a change in the market. Since then, we have seen a growing trend: applications (financial and otherwise) that use fingerprint recognition to authenticate users in their service.

In order to be classified as a biometric, a descriptor must fulfill certain characteristics. The trait must be universal, distinctive, permanent, accessible and acceptable, while the system used for collection and authentication must be efficient

and robust [28, 18]. Universal refers to the recurrence of the trait in the population. A true biometric must be generally shared by all individuals (*i.e.*, it must be a textbook characteristic of the group). Distinctive relates to the ability to differentiate members of the group from the proposed trait. It is not necessary for a biometric to be *unique* but it must be unlikely that two individuals will share the value attributed to the descriptor. Permanence speaks to the invariant nature of the descriptor over time. Similar to the property of distinctiveness, it is not necessary for a biometric candidate to be unchanged throughout a persons lifetime but usually biometrics will remain stable for decades. Finally, in terms of traits, acceptable and accessible are closely related. For a trait to be adopted as a biometric it must be acceptable by users, *i.e.*, people need to be comfortable in sharing the trait; and lastly, the trait must be quantifiable (*i.e.*, easy to measure) and the collection process must not be invasive (*i.e.*, easy to collect).

Once a biometric trait has been identified an authentication system is designed around it. A successful system will be efficient and robust. It must be hard for an attacker to impersonate a user. Practically, this could involve measuring additional characteristics along with the biometric (*e.g.*, temperature, heart rate). Finally, the system must be timely. The time-to-authenticate beginning with detection and collection all the way to access must be reasonable for the intended application [87].

Having discussed the requirements, and perhaps due to their use in mobile devices, we can immediately identify fingerprints and face geometry as good features for authentication. However, more broadly there are two types of features that can fall within the category of biometrics: physiological descriptors and behavioral descriptors. Physiological descriptors that fulfill the conditions described in [28] to be characterized as biometrics are obtained from "patterns in our genetic makeup" [88]. Current and proposed physiological descriptors are: fingerprints, iris scans, hand geometry, facial recognition, ear shape, skin patterns and luminescence, body odour, among others [28]. Alternatively, the second source of inputs are those derived from our behavior. Behavioral biometrics come from patterns in the way we complete everyday tasks [88]. Some examples of traits derived from behavior

include penmanship (handwriting analysis), keystroke analysis, typing rhythm, gait analysis, and voice recognition [18, 28].

The benefits of using biometric authentication systems are many. From the users' perspective they greatly reduce the mental load imposed by the myriad of authentication systems that we encounter every day. For a user, biometrics are permanent, their appeal lies in that they cannot be lost, stolen or forgotten. For companies and system administrators, these guarantees place most of the responsibility on the user (biometrics are not transferable [2]; by extension when a user is authenticated the verifier can be reasonably certain that the request is legitimate) and they don't incur in the expense of having to issue tokens or, in fact, replace them [28].

On the other hand, the privacy risks associated with biometrics are higher: by definition, one of the characteristics of biometrics is permanence. Any information lost (or released) through a data breach implies that the user will be linked, implicated, or identified for decades after the incident has happened. Moreover, the types of inferences that can be drawn from biometrics might be much more compromising. Whereas a password might reveal place of work or a (relevant) historical fact, an iris scan for example may reveal primary private identifiers such as health conditions and age [90]. Similarly, biometrics have been used in the past to look for patterns in"natural populations, in searching for change in bodily and psychological parameters over time and for grounding racial classification" [18]. This is why whether physiological or behavioral, human biometric information is protected under privacy legislation [28].

Independent from discussions on the appropriateness of this data for everyday authentication tasks, the academic community has explored what continuous authentication systems would require and what they would achieve. One motivation for this type of work is a vulnerability that remains to be addressed: once unlocked (and for as long as the screen remains active) there is no available way to guarantee that the person that unlocked the phone is the one that has continued to use it. In

[2] *Transferable* in this context means that the person requesting access can convince the verifier that they hold a credential without revealing any other information [89]. As biometrics are both inherent and unique to a person, under normal circumstances only the authorized entity will authenticate.

practice, some applications are released with their own authentication method that gets triggered when the app is launched, but there is a case to be made for a system wide service. In [91] the authors present the advantages and hurdles towards developing a stronger authentication system with the aid of keystroke characterization. More encompassing though is [92] where Patel *et al.* present a survey of the methods and techniques that would be required for an active authentication system. They present the forward facing camera as the sensor required to measure facial geometry (a physiological biometric); the gyroscope and accelerometer to determine gait (a behavioral biometric); and the touch screen and orientation sensor for touch gestures and hand movements (both behavioral biometrics) [92].

2.6.2 Disruption: Identifying Online Misbehavior

As metadata reveals behavioral patterns, one common application of behavioral classification is identifying entities that break the norm. To this end we focus on two types of attacks: the identification of problematic content and the identification of malicious accounts. For the remainder of the section, we will explore different examples of how metadata is key to solving these problems.

2.6.2.1 Clickbait and other forms of abusive behavior

For all the positive things about the Internet, it also seems to have amplified different forms of abusive or malicious behavior. The past few years have shown an increase in the number of instances and the severity of each event for things like cyber-bullying, offensive language, hate speech and sarcasm [93]. In this section we present some studies that show the usefulness of metadata as a tool to identify and hopefully prevent this kind of behavior.

In their paper, Founta *et al.* look at classifying anti-social behavior not only from the text of the message that was posted but also from the online behavior and relationships of the account that originated each post. They combine metadata fields extracted from the text of the posts — Tweets (*e.g.*, number of emoticons, the number of words completely capitalized), the user's behavior on the network (*e.g.*, popularity, the number of subscribed lists, the number of likes), and the social

network of the author (*e.g.*, reciprocity between user and followers, clustering tendencies of the user and followers) as well as other forms of textual descriptors to predict whether an account was engaged in abusive behavior and more particularly the type of behavior it exhibited [93].

Similarly, in mariconti2018coordinated they combine the textual transcript of a video along with still images associated with it with the metadata available from the post to build an ensemble classifier and predict the likelihood that a video will be targeted by a raid originated at some other social network. In their analysis the authors compare classifiers for each of the three data-type inputs and find that in their experiments, the classifiers using metadata consistently outperform the other data type. Ultimately, following their proposal, content providers (their experiments look at YouTube but others may work as well) could do a risk assessment of the videos they host and offer preventive or protective services to at-risk posts.

Finally, metadata has also proved useful in the detection of YouTube videos that aim to trick users into following their link. This is a problem known as "clickbait" where the posts are designed to mislead viewers towards the content so as to increase the number of views and from that the revenue they generate from advertisements. In a paper presented by Zannettou *et al.* researchers use the thumbnail, the headline and the tags associated to each video posted on the platform as input to a deep learning classifier that detects these type of posts [94].

2.6.2.2 Identification of Fake Accounts

Fake accounts consume resources (*e.g.*, network bandwidth, memory, computations from search queries) that can otherwise be made available to legitimate users or alternatively not used at all reducing the overall expense of the service. This, in addition to the social load that spammers dump on the network: they saturate user attention by sending spam emails, they abuse user privacy by having unwanted participants in the social network, or what is even more invasive, spam accounts exhibiting human-like behavior may befriend real users making these fake accounts more difficult to identify [95, 96].

Metadata has become a core component of the services offered by OSNs. Twit-

ter, for instance, provides information about the users mentioned in a post, the number of times a message was re-tweeted, when a document was uploaded and the number of interactions of a user with the system, just to name a few. This is not merely extra information: users rely on it to measure the credibility of an account [97] and much of the previous research in fighting spam accounts relies on metadata for detection [95, 98].

Metadata also has been used to differentiate between organically acquired (*i.e.*, legitimate) and paid (*i.e.*, fake but different from spam accounts) followers. Fake followers are used to artificially augment the popularity of one account with the intent to secure any range of benefits from fame and glory (*i.e.*, a proxy to social status) to lucrative marketing contracts by becoming social media *influencers*. Shen *et al.* [99], uses the the ratio of follower count and following count, the percentage of bi-directional friends, the ratio of the original posts, proportion of nighttime posts, all of which are derived exclusively from the account metadata, as well as the diversity of topics posted by each account to separate both types of followers. Similarly, in [100] Cresci *et al.* also look at the problem of identifying fake followers on micro-blogging services. In their work they look at Twitter and combine elements of the content of posts with metadata of the user accounts to build a classifier that separates legitimate followers from fake ones. In terms of metadata, their work focuses on the number of friends, the number of tweets, and the ratio between the number of friends and the number of followers.

Finally, in [101] and following from previous work on fake followers, the authors focus on identifying accounts that correspond to fake influencers. With respect to online social media, an influencer is a user that, because of their pre-existing connections to (a great number of) other accounts, they provide marketing services to different companies (*i.e.*, these correspond to the accounts that want to secure marketing contracts discussed above). Zenonos *et al.* develop a set of 10 features with which they are able to separate real and fake influencers. In their work, they also find that using network descriptors, particularly that of degree centrality, they are able to achieve their goal. Degree centrality is a measure of the importance of a

node in a network. In their work, they calculate the centrality of a potential influencer by finding an average of the centrality of the nodes around it. As a measure of the connectivity of a node, this descriptor is metadata of the account for which it was computed.

Having seen that metadata is a valuable resource, the remaining sections will discuss the relevance of mobile devices to users' privacy. In the next section we argue that given the proximity between mobile devices and users and their typical one-to-one correspondence, identifying a mobile device is often equivalent to identifying the individual that owns it. However, due to the abundance of methods and types of data, we discuss smartphone identification in the next section.

2.7 Identification of Mobile Devices

Previous work on device fingerprinting can be grouped into two parts: identification through the content available on the device and identification through its physical characteristics.

2.7.1 Content-Based Identification

Given the personal nature of devices, we expect the user-specific content of a phone (e.g., photos, applications, contacts, music) to be significantly different from most if not all others. In [102] Quattrone et al. present an interesting approach combining both content (i.e., system settings, language settings, etc.) and physical descriptions (i.e., internal and external memory size, device manufacturer and model) to generate their signature. Using similar types of auxiliary information in [103] the authors expose different information leaks. One of these reveals the identity of the user through the monitoring of network data usage statistics of some key applications.

Other studies focus on the identification of users using lists of applications installed on the phones. Both [104] and [105] find that the list of applications installed is highly discriminative in terms of devices and that even if, instead of app names, the classification was based on the app categories, it would still reveal some partial information about the user.

In [106] the authors take a different approach: they collect all possible public

resources available in iOS devices and build their classifiers with a compilation of these features. They find, for example, that the top 50 most frequently played songs can be used to uniquely identify a device with 94% accuracy. The problem they present is that, while fingerprinting is possible (and accurate), it is also based on user behavior, which changes over time. Overall, their main finding is that the unprotected information available from Apple devices generates unique signatures from which mobile phones can be identified.

Finally, in [107] the authors combine a set of descriptors from the browser, system and hardware attributes as well as some behavioral characteristics of the user. The paper was originally intended to corroborate the standardized settings of the web browser. The authors also find that by combining different attributes they are able to generate a fingerprint for identification.

2.7.2 Hardware-Based Identification

The concept of manufacturing variability has been presented before in its application as a means for identification and authentication. From a theoretical point of view, in [108] the authors present the properties and applications of silicon-based physically unclonable functions (PUFs). In practice, the viability of the method is implemented by Lee *et al.* in [109] where their system exploits the timing delays of transistors and wires in individual circuit boards to compute cryptographic keys using PUFs. The uniqueness of each device adds the required randomness in the key generating process and the secret is known only to the device.

Moving away from cryptography, silicon imperfections can be found in every electronic circuit available. Naturally, an interesting topic of study is the effect of imperfections on the ever present smartphone. The array of sensors accessible to mobile devices gives an important added perspective. It not only allows the exploitation of internal differences for identification, but it also adds a dimension in the form of environmental interactions.

In [110] they present two approaches for identification. The authors show that combining microphone and speaker systems and computing the distortion of a control signal results in accurate classification of the device; however, for this

approach they need to request two permissions, namely `RECORD_AUDIO` and `MODIFY_AUDIO_SETTINGS`. In the same paper they introduce a second form of cyber attack: identification through the calibration error of the accelerometer. They require no permissions to collect readings from the sensor. They created a web application and ask participants to leave their devices stationary and facing up while the data collection takes place. The authors find that it is possible to identify all devices with an accuracy of 58.7%. Imperfections introduced in the manufacturing process make each device unique; moreover, this uniqueness can be quantified and used for identification. However, the experimental design of this paper calls for all users to perform the same task. As countermeasures, they propose two methods to reduce the usefulness of their attack: the calibration of each sensor at production time to eliminate the variability in the readings and the introduction of some random value to prevent the recognition of a baseline state in the device.

Other projects have also looked at identification from sensor imperfections and biases. In [111] the authors present results using the manufacturing imperfections in the accelerometer for identification. Similarly, in [112] the authors use the imperfections found in the magnetometer to identify devices. These two papers rely on identifying one state in the device and finding the difference between the known state and the measurement to compute the influence of the imperfection thereby identifying each phone. In [113] the authors use silicon-based imperfections of network interface cards to successfully identify devices through the passive analysis of the radio signals they transmit. Lastly, in [114], Kohno *et al.* show that remote identification is possible and viable without permission or any modification to the target device and across different types of devices. They rely on clock skews to generate the fingerprint but, in their approach, there needs to be an established connection between attacker and target. This requires some access to the target device, which might not always be feasible.

2.8 Identification of Places: Localization

Thus far, we have contextualized identification of accounts within the field of behavioral biometrics and the emission of a magnetic field as a hardware-based identifier of each phone. In addition to identity (or uniqueness) of an entity there is privacy-relevant information that can be derived from crowd sourcing sensor readings. In this section, we place the work presented in Chapter 6 within the literature of localization. Localization is the term used to describe the process of placing an entity in a known frame of reference [115], in other words, the identification of place. From an autonomous system perspective, and according to the taxonomy presented by Thrun *et al.*, we are addressing a global localization problem in a dynamic environment where we can only passively monitor users (*i.e.*, we have no control over their future behavior) [115].

In this context, localization of an autonomous agent can only be carried out through sensor readings, either by matching current readings against an up-to-date map or through geometric calculations that determine the unknown position of the agent against a known marker. In the following sections, we discuss each of these approaches with respect to indoor and outdoor localization.

2.8.1 Model-Based Localization

Model-based techniques are comprised of methods where the movement of the user is calculated from a sensor, including GPS, as well as cellular and WiFi signals. Model-Based techniques are by far the most common and widespread techniques used for localization. Global Navigation Satellite Systems (GNSS) can currently localize users with an accuracy of centimeters [116]. Assisted GPS (A-GPS) was developed to add robustness to the system in urban environments where line of sight to satellites is hindered by buildings or environmental conditions such as weather or pollution making GPS unreliable [117]. Alternatively, mobile operating systems give developers the option to localize users based exclusively on cellular radio signal strength, while less accurate than GPS, it provides a low-power alternative to GNSS. The position of the user is inferred by extracting features from a signal and comparing them to an established ground truth. As an example, network signal

strength is one of the most commonly used features with the actual position of the user computed from the triangulation between multiple cellular base stations and the signal received from each station by the device [118].

These methods can be applied to indoor environments as well. Localization from WiFi access points using received signal strength has been found to be accurate enough to track users inside buildings with the capability of distinguishing between adjacent rooms [119, 120, 121]. These methods only require WiFi signal strength and the layouts of the building so as to localize a user. The primary limitation of these methods is that the signal strength calculation is typically carried out on the device whereby continuous tracking drains the resources of a phone [120].

Alternatively, model-based techniques are also used as a means of error correction for dead-reckoning schemes. Urban dead-reckoning is the process by which a user's location is tracked by measuring the side-effects of motion, usually through the use of the compass and the gyroscope. It has been shown that urban dead reckoning can accumulate errors of up to 100 meters in 6 minutes of collected data [122]. Indoor localization methods often combine dead reckoning with model-based localization to maintain acceptable tracking accuracy [123, 124].

The error correction mechanism is obtained from a variety of sources. For example, Haverinen and Kemppainen use GPS readings to mark the entrance of the building and again collect active GPS readings from open areas or windows to validate and correct their location [125]. Wang *et al.* collect magnetic field readings along with gyration and acceleration information from different users to find landmarks and correct the error for individual users [122].

2.8.2 Fingerprint-Based Localization

Fingerprint-based techniques include all methods where a site (or any other geographical location) is surveyed in order to build a map, which is then used to pinpoint the location of a user.

Fingerprint-based techniques are costly with respect to collection and maintenance of the maps. Active areas of research in this field are mainly centered on reducing the overall financial cost necessary for building maps primarily through

leveraging crowd-sourcing techniques where data is passively collected by a large population [126]. In the following, we detail the approaches specific to the type of environment, whether it is indoor or outdoor.

Indoor Environments Localization in an indoor environment is linked to a map with a high level of detail. Even for small spaces this presents a challenge in terms of the volume of data that must be available for the task. In fact, it is important to consider that sensor values might vary with altitude as well as with horizontal displacement. Furthermore, selecting a sensor that provides consistent differences in enclosed environments increases the complexity of the task. Potential solutions to these problems are manual collection of ambient readings or strategic placement of location beacons (and sensors) throughout the area of study under consideration [122, 124, 127, 128].

Chung *et al.* use the magnetic field to determine the position of a user in a corridor. They manually collect magnetic field readings every 60 cm and find that localization is accurate to 1.64 m for 90% of the test observations [127]. They also test their system in elevators and in the atrium of a building and find that there are measurable differences across these locations. Following the work by Chung *et al.*, Haverinen and Kemppainen test whether the difference in magnetic field readings can be used to locate an agent across a larger area (in their work, they test four buildings). They find that these readings have low variability over time while being spatially distinct and build a localization system based on them [125]. Carrillo *et al.* [129] focus on the same problem, indoor localization from a map, and improve the accuracy of their system by using all three components of the magnetic field reading. In [130], Galvan *et al.* combine three passive sensors the microphone, the light sensor and the magnetometer to estimate the location of a user in a structure. Wang *et al.* use the magnetometer and the light sensor to localize an individual within a structure and they go a step beyond and test their system in open-plan environments like parking lots and shopping centers and still obtain good results [131]. Wang *et al.* show that magnetic field readings can be used to discriminate between activities such as standing or walking. In particular, they use changes in the magnetic field to

identify when a user is moving in an escalator [122]. Finally, Ashraf *et al.* [132] combine the accelerometer and the magnetometer to determine the level of activity of the user and proceed to extract from the magnetic field readings the pattern formed by the time-series observation. Again, using a fingerprint-based map they are able to localize a user in one of six buildings in their campus.

Outdoor Environments Outdoor localization approaches are classified based on the technique used to establish the map. One common fingerprinting method from outdoor environments is matching the visual cues obtained from the camera of the phone to a map of geo-tagged images [118].

Narain *et al.* use the combination of the motion and position sensors available in smartphones (primarily the accelerometer and gyroscope) to infer the trajectory of a moving car [133]. In particular, they propose a method to reconstruct the route taken by the car based on the physical characteristics of the roads (*i.e.*, speed, bumps, stop signs and curvature), the junctions between roads and the angle recorded for each turn. With this information, they are able to match the estimated candidate trajectory to the map of the (known) city using OpenStreetMap and estimate the actual route taken by the user. Similarly to our work, the authors propose a zero-permission attack to determine location information about users from motion sensors. However, we focus on a different localization task that aims to identify the places that users have visited solely from the magnetometer sensor instead of the trajectories they have taken. Moreover, we employ an event-based approach that identifies events and matches events to locations thereby reducing the need for up-to-date maps.

2.9 Contribution with respect to the state of the art

2.9.1 Account Identification

Having explored the use and characteristics of biometrics as well as the identification by means of behavioral traits, we propose to look at metadata to determine whether account (or user) information is latently contained in it. In Chapter 4 we present a method that follows a stratified combinatorial approach to look for fea-

tures that might differentiate between user accounts in Twitter. Our claim is that the way we interact with technology is unique to an individual. We argue that it is possible to extract some information from the metadata that is generated when we access such systems.

2.9.2 Device Fingerprinting

Proceeding from the work in account identification, we then turn our attention to the problem of device fingerprinting. Similar to the work on accelerometers and audio systems presented in [111] we show that the physical features intrinsic to a device exist and can be used for identification. However, in contrast to these works, it is harder to develop countermeasures that obfuscate the magnetic field. Moreover, by using the magnetic field, we are developing a technique that can potentially characterize any electronic device.

2.9.3 Localization

To the best of our knowledge, the work presented in Chapter 6 is the first that explicitly maps location-types in terms of their environmental characteristics. We take into consideration features like the structural similarity between buildings, the isolation of repeated events, the size and distribution of people and crowds, and any similarity across the tasks undertaken at each location. We combine these characteristics to generate a *magnetic signature* for each location-type, which we use to identify new locations in the city. The signatures for each location-type are composed of a set of shapelets as defined in [134, 135, 136, 137, 138]. The classification of a new observation into a location-type is a function of the likelihood of a shapelet being present in that observation. We propose a novel, low-cost, and passive methodology for location inference. In particular, we present supervised classification algorithms that take as input the time-series (observations) for all classes and predict the probability and label (*i.e.*, location-type) of a new observation.

Chapter 3

Machine Learning: an overview

The methods found in all chapters of this dissertation hinge upon machine learning and the idea that one set of parameters can be used to predict or reveal additional information about a single feature or a group of features. In particular, we focus on two paradigms: unsupervised and supervised learning.

In the following chapters, unsupervised learning is used in the process of discovery. More generally, this learning paradigm takes in a set of M observations $\{(\mathbf{X}_1),(\mathbf{X}_2),\ldots,(\mathbf{X}_M)\}$ of $p-dimensional$ vectors. The goal is to infer properties about the density distribution across dimensions and observations. In unsupervised learning, the value of p is often much higher than than what it would be in supervised learning and therefore things like cluster analysis from different variables and dimensionality reduction are very common [139].

The core conclusions of the dissertation use supervised learning to sustain them. Supervised learning starts with the premise of a dataset consisting of a known number of labeled groups as well as several examples of observations that belong to each group. The goal is that given some training data (S)

$$S = \{(\mathbf{X}_1, y_1),(\mathbf{X}_2, y_2),\ldots,(\mathbf{X}_m, y_m)\},$$

we infer a function f_s such that

$$f_S(\mathbf{X}_i) \approx y_i.$$

Developing f_s from the know dataset makes it so that future unlabeled data

$$S' = \{(\mathbf{X}_{m+1}),(\mathbf{X}_{m+2}),\ldots\},$$

we can assign the appropriate y. When $y \in \{-1,+1\}$ this task is known as classification. If instead, $y \in IR$ the task is known as regression.

A learning algorithm is the mapping $S \rightarrow f_S$. Throughout the dissertation there are a few examples of learning algorithms, the rest of the chapter is intended to provide background into the most used of these algorithms.

3.1 Multinomial Logistic Regression

The logistic regression is a nonlinear model specifically designed for binary dependent variables [140]. The probability of the outcome is given by

$$Pr(Y = 1|X_1, X_2, \ldots, X_k) = F(\beta_0 + \beta_1 X_1 + \cdots + \beta_k X_k),$$

where $X_1 \ldots X_k$ are the different regressors (or variables) and F, the population regression function, is given by the logistic cumulative distribution function.

Multinomial Logistic Regression (MLR) is the multi-class adaptation of this binary prediction method. Multi-class methods are needed when the outcome includes three or more possibilities. For these instances, the logistic regression is built in a one-vs-rest algorithm where each class (one at a time) is assigned as the positive instance and all other classes as negative. The final prediction for each new observation is given by

$$\hat{y} = argmax f_C(x),$$

where $f_C(x)$ is a vector that contains the prediction of each model for all C classes of the x being evaluated. In general, and as we will see in Chapter 4, the primary disadvantage of using MLR is that it is not scalable to the number of output classes.

3.2 Random Forests

Decision trees are the basic building blocks for random forests. A decision tree is a classification algorithm based on a binary search tree where each node specifies a test on one of the attributes available to the system. The nodes are constructed in such a way that the most informative[1] features (*i.e.*, tests) are higher up in the tree.

[1] There are two common ways to rank features in terms of the information they contain, the Gini Coefficient and Shannon's Entropy.

The leaf nodes in the tree will correspond to the outcome variable. Therefore an observation will be classified once it has traversed the tree and assigned to the class that corresponds to the leaf in its path [141, 142].

There are several factors that support the rise in popularity of decision trees. First, once trained, every tree can be written out as a set of logical rules (following the *if* ... *then* principle). These make the model interpretable removing the appearance of a black box. Then, in principle, decision trees can accept a wide variety of input data. While this depends on the implementation of one particular algorithm, as long as there is a test for the data, it can be included in a node. Similarly, because of their construction decision trees are robust to noise. Features that are not informative with respect to the prediction variable will be automatically excluded from the nodes; having outlying values in the training data will also have limited impact in the outcome of the tests [139].

Random forests were presented as a way to reduce the variance[2] of the classification tree [143, 139]. This method takes a large number of independent trees and obtains a class vote from each one. Individual results are then averaged and in its simplest implementation, a majority vote determines the final prediction for each observation. Based on the idea of bagging, random forests are also effective in reducing the likelihood of a model over-fitting a particular training set.

3.3 k Nearest Neighbors

k-Nearest Neighbors (KNN) is an example of a *lazy* learning method. Lazy learning or instance learning refers to those algorithms that generalize from the training observations only after receiving a test observation. In other words, lazy learners work on local weighted functions to make a prediction whereas eager learners compute a global approximation for the target function and simply evaluate test cases.

[2]Two terms that are relevant to random forests are boosting and bagging. Boosting is the idea that combining predictors yields better performance. While this is counter intuitive in people, it has been shown that learners will specialize in different aspects of the problem and often the best solution is found when multiple learners are combined. Bagging is instead the concept of 'bootstrap aggregating'. Bootstrapping is the idea of sampling with replacement. Two of its primary benefits are that since it generates slightly different input data for different learners it will first allow trees to specialize in different aspects (*i.e.*, priming the trees for boosting); and then it will reduce the variance of the classifier. [141]

The main practical difficulties with instance learning are that first the processing is done during testing (*i.e.*, each test observation is compared against all training observations) and second choosing an appropriate function to define what is 'close' or 'related' is not trivial (this becomes particularly important in high-dimensional space when proximity in two features might be inverted).

As a classifier, KNN assigns to each new observation x_n the label that corresponds to the most common value of a window of size k as defined by some distance metric[3]. Formally, the model behaves as follows

$$\hat{f}(x_n) = argmax \sum_{n=1}^{k} \delta(v, f(x_i))$$

where $v \in \{v_1, v_2, \cdots, v_s\}$ belongs to the finite set of labels present in the training set, δ is the distance function, and k is the hyperparameter that indicates the number of neighbors under consideration.

The choice of k has a definite impact on the performance of the algorithm. A small k makes the algorithm sensitive to noise; make it too large the grouping (or spacing) of the training data has an impact on the accuracy, points that are further away are less relevant to the new observation [139, 144].

3.4 Artificial Neural Networks

An artificial neural network is a mathematical model designed to imitate how the neurons in our brain operate, resulting in the multi-layer function learning from data. In essence, input signals from a previous layer are adjusted and passed along to the next layer, and the signals of the last layer determine the output of the network. Usually, for more complex problems, multiple layers of neurons are stacked in order to generate a prediction.

[3] A distance metric is a function that has three characteristics: first it takes two inputs and returns a scalar (typically we thing in three dimensions but this is just as applicable to more), second it is symmetric (*i.e.*, not affected by the direction of the change), and third it must follow the triangle inequality that states that between three points a, b and c the distance from a to b plus the distance from b to c should be greater than or equal to the distance from a to c. Typically the most common choice for metric is the Euclidean distance as it is easy to generalize to multiple dimensions. [142].

In terms of supervised learning, the procedure involves collecting a dataset and selecting appropriate features; normalizing these inputs; splitting the data into training, testing and validation sets; training the model; and finally testing on unseen observations. Using an artificial neural network for supervised learning requires an aditional step: designing the architecture for the network; which involves determining the depth, width, and activation functions of the model. The parameters of each layer can be learned and adjusted to achieve a better solution based on some loss functions. By only employing differentiable layers, these parameters can be effectively learned with back-propagation [141], which recursively propagates the gradient backwards starting from the loss functions from the last layer to the first input layer.

3.4.1 Convolutional Neural Networks

One of the tasks that led to the development of new architectures was that of identifying objects in an image. The human mind can identify as belonging to the same category an object under an amazing array of transformations. The hypothesis leading to Convolutional Neural Networks (CNN) is that grouping nearby pixels will benefit the object recognition process [145]. The success of this method made CNNs the method of choice for computer vision problems, moreover it has become the specialized architecture when processing data with localized spatial correlations.

As the name indicates, CNNs use the *convolution* operation instead of the general *matrix multiplication*. CNNs belong under the umbrella term of Deep Learning. A deep neural network is characterized by relatively more layers of neurons.

3.4.2 Siamese Networks

For classification problems, neural networks and deep learning models are inspired by the human brain and the way people acquire concepts. One major difference between the two is that while human learning can happen reliably with as little as 2 or 3 observations, artificial (deep) networks need thousands or tens of thousands of examples to achieve the same performance [4] [146, 147, 148].

[4] Typically, the penalty for using a reduced data set is that the network will overfit the training examples and underperform on test observations. [146]

To mitigate this problem, researchers introduced *Siamese Networks*, a model where two networks, with the same architecture and matching weights, are joined at the last layer. The additional constraints allow the network to not only distinguish between different classes but to group within class similarities resulting in overall better performance. In general, in addition to the classification loss, an additional contrastive loss [149] is introduced between the latent representations of two inputs. This loss encourages the similarity of hidden features of like pairs while maximizing the dissimilarity between the features of unlike pairs.

Siamese Networks work under the assumption that generic knowledge from known classes can be incorporated in the definition of new classes (even if the categories are unseen). In other words, Siamese Networks use the discriminative features from the cross-entropy loss and add to it the hidden features from the new class.

Chapter 4

Identification of User Accounts from Twitter Metadata

Platforms like Facebook, Flickr, and Reddit allow users to share links, documents, images, videos, and thoughts. Data has become the newest form of currency and analyzing data is both a business and an academic endeavor. Regardless of the reason for the release of a specific dataset, it is likely that a dataset containing descriptive information about accounts and posts becomes readily available. In this work, we aim to understand what this descriptive information reveals about users. When online social networks (OSNs) were first introduced, privacy was not a major concern for users and therefore not a priority for service providers. With time, however, privacy concerns have risen: users started to consider the implications of the information they share [57, 150] and in response OSN platforms have introduced coarse controls for users to manage their data [58]. Indeed, this concern is heightened by the fact that this descriptive information can be actively analyzed and mined for a variety of purposes, often beyond the original design goals of the platforms. For example, information collected for targeted advertisement might be used to understand political and religious inclinations of a user. The problem is also exacerbated by the fact that often these datasets might be publicly released either as part of a campaign or through information leaks.

Previous work shows that the content of a message posted on an OSN platform reveals a wealth of information about its author. Through text analysis, it

is possible to derive age, gender, and political orientation of individuals [26]; the general mood of groups [151] and the mood of individuals [27]. Image analysis reveals, for example, the place a photo was taken [152], the place of residence of the photographer [25], or even the relationship status of two individuals [153]. If we look at mobility data from location-based social networks, the check-in behavior of users can tell us their cultural background [154] or identify users uniquely in a crowd [155]. Finally, even if an attacker only had access to anonymized datasets, by looking at the structure of the network someone may be able to re-identify users [59]. Most if not all of these conclusions could be considered privacy-invasive by users, and therefore the content is what most service providers are starting to protect. However, access control lists are not sufficient. We argue that the behavioral information contained in the metadata is just as informative.

Metadata has become a core component of the services offered by OSNs. For example, Twitter provides information on users mentioned in a post, the number of times a message was re-tweeted, when a document was uploaded, and the number of interactions of a user with the system, just to name a few. These are not merely extra information: users rely on these to measure the credibility of an account [156] and much of the previous research in fighting social spam relies on account metadata for detection [98, 95].

In this chapter, we present an in-depth analysis of the identification risk posed by metadata to a user account. We treat identification as a classification problem and use supervised learning algorithms to build *behavioral signatures* for each of the users. Our analysis is based on metadata associated to micro-blog services like Twitter: each tweet contains the metadata of the post as well as that of the account from which it was posted. However, it is worth noting that the methods presented in this work are generic and can be applied to a variety of social media platforms with similar characteristics in terms of metadata. In that sense, Twitter should be considered only as a case study, but the methods proposed in this chapter are of broad applicability. The proposed techniques can be used in several practical scenarios such as when the identifier of an account changes over time, when a single

user creates multiple accounts, or in the detection of legitimate accounts that have been hijacked by malicious users.

In security, there are at least two areas that look at identity from opposite perspectives: on one hand research in authentication looks for methods that, while unobtrusive and usable, consistently identify users with low false positive rates [92]; and, on the other hand, work on obfuscation and differential privacy aims to find ways by which we can preserve an individual's right to privacy by making information about them indistinguishable in a set [39]. This study is relevant to both: we claim that in the same way that our behavior in the physical world is used to identify us [157, 97, 158], the interactions of a user with a system, as represented by the metadata generated during the account creation and its subsequent use, can be used for identification. If this is true, and metadata can in fact represent us and as it is seldom if ever protected, it constitutes a risk for users' privacy. Our goal is therefore, to determine if the information contained in users' metadata is sufficient to fingerprint an account. Our contributions can be summarized as follows:

- We develop and test strategies for user identification through the analysis of metadata through state-of-the-art machine learning algorithms, namely Multinomial Logistic Regression (MLR) [159], Random Forest (RF) [143], and K-Nearest Neighbors (KNN) [160].

- We provide a performance evaluation of different classifiers for multi-class identification problems, considering a variety of dimensions and in particular the characteristics of the training set used for the classification task.

- We assess the effectiveness of two obfuscation techniques in terms of their ability of hiding the identity of an account from which a message was posted.

We demonstrate that through the application of supervised algorithms, we are able to identify 1 user in a group of 10,000 with approximately 96.7% accuracy. If we broaden the scope of our search and consider the best 10 candidates from a group of 10,000 users, we are able to achieve a 99.22% accuracy.

4.1 Motivation

4.1.1 Formal Definition of the Study

We consider a set of users

$$\mathbf{U} = \{u_1, u_2, \ldots, u_k, \ldots, u_M\}.$$

Each user u_i is characterized by a finite set of features

$$\mathbf{X^{u_k}} = \{x^{u_k}{}_1, x^{u_k}{}_2, \ldots, x^{u_k}{}_R\}.$$

In other words, we consider M users and each user is represented by means of R features. Our goal is to map this set of users to a set of identities

$$\mathbf{I} = \{i_1, i_2, \ldots, i_k, \ldots, i_M\}.$$

We assume that each user u_k maps to a unique identity i_l.

Identification is framed in terms of a classification problem: in the training phase, we build a model with a known dataset; in our case, we consider a dataset in which a user u_k as characterized by features X_{u_k}, extracted from the user's profile, is assigned an identity i_l. Then, in the classification phase, we assign to each user $\hat{u}_k$, for which we assume that the identity is unknown, a set of probabilities over all the possible identities.

More formally, for each user $\hat{u}_k$ (i.e., our test observation) the output of the model is a vector of the form

$$\mathbf{P_I}(\hat{u}_k) = \{p_{i_1}(\hat{u}_k), p_{i_2}(\hat{u}_k), \ldots, p_{i_M}(\hat{u}_k)\},$$

where $p_{i_1}(\hat{u}_k)$ is the probability that the test observation related to $\hat{u}_k$ will be assigned to identity i_1 and so on. We assume a *closed* system where all the observations will be assigned to users in $\mathbf{I}$, and therefore, $\sum_{i_l \in \mathbf{I}} p_{i_l}(\hat{u}_k) = 1$. Finally, the identity assigned to the observation will be the one that corresponds to $argmax_{i_l}\{p_{i_l}(\hat{u}_k)\}$.

Two users with features having the same values are indistinguishable. Moreover, we would like to point out that the values of each feature can be static (constant) over time or dynamic (variable) over time. An example of static feature is the

Table 4.1: Description of relevant data fields.

Feature	Description
Account creation	UTC time stamp of the account creation time.
Favourites count	The number of tweets that have been marked as 'favorites' of this account.
Follower count	The number of users that are following this account.
Friend count	The number of users this account is following.
Geo enabled	(boolean) Indicates whether tweets from this account is geo-tagged.
Listed count	The number of public lists that include the account.
Post time stamp	UTC time of day stamp at which the post was published.
Statuses count	The number of tweets posted by this account.
Verified	(boolean) Indicates that Twitter has checked the identity of the user that owns this account.

account creation time. An example of dynamic feature is the number of followers of the user at the time the tweet was posted. Please also note that, from a practical point of view, our objective is to ascertain the identity of a user in the test set. In the case of a malicious user, whose identity has been modified over time, we assume that the 'real' identity is the one that is found in the training set. In this way, our method could also be used to group users with very similar characteristics and perhaps conclude that they belong to the same identity.

4.1.2 Attack Model

The goal of the study is to understand if it is possible to correctly identify an account given a series of features extracted from the available metadata. In our evaluation, as discussed before, the input of the classifier is a set of new (unseen) tweets. We refer to a successful prediction of the account identity as a *hit* and an unsuccessful one as a *miss*. We assume that the attacker is able to access the metadata of tweets from a group of users together with their identities (i.e., the training set); and, that the new tweets belong to one of the users in the training set.

We present the likelihood of success of an identification attack where the adversary's ultimate goal is to identify a user from a set given this knowledge about the set of accounts. To achieve this, we answer this question: *Is it possible to identify an individual from a set of metadata fields from a randomly selected set of Twitter user accounts?*

4.2 Methods

4.2.1 Metadata and the case of Twitter

We define metadata as the information available pertaining to a Twitter post. This is information that describes the context on which the post was shared. Apart from the 140 character message, each tweet contains about 144 fields of metadata. Each of these fields provides additional information about: the account from which it was posted; the post (e.g. time, number of views); other tweets contained within the message; various entities (e.g. hashtags, URLs, etc); and the information of any users directly mentioned in it. From these features (in this work we will use features, fields, inputs to refer to each of the characteristics available from the metadata) we created combinations from a selection of 14 fields as a basis for the classifiers.

4.2.2 Feature Selection

Feature selection methods can be essentially grouped in three classes following the classification proposed by Liu and Yu in [161]: the filter approach that ranks features based on some statistical characteristics of the population; the wrapper approach that creates a rank based on metrics derived from the measurement algorithm; and a group of hybrid methods which combine the previous two approaches. In the same paper, the authors claim that the wrapper method is guaranteed to find the optimal combination of inputs for classification based on the selected criteria. Three years later, in [160], Huang *et al.* provided validation by experimentally showing that for classification, the wrapper approach results in the best possible performance in terms of accuracy for each algorithm.

From the three proposed, the only method that allows for fair comparison between different algorithms is the wrapper method. Since it guarantees optimal feature combination on a per algorithm basis, it eliminates any bias in the analysis due to poor selection. Ultimately, we conducted a comprehensive stratified search over the feature space and obtained a ranking per level for each of the algorithms. Here, a level corresponds to the number of features used as input for the classifier and we will use n to denote it. In the first level, where $n = 1$ we looked at the predictive

power of each of the 14 features individually; for $n = 2$ we looked at all combinations of pairs of features, and so on. We use the term combinations to describe any group of n un-ordered features throughout this chapter.

The features selected were those that describe the user account and were not under direct control of the user with the exception of the account ID which was excluded as it was used as ground truth (*i.e.*, the label) of each observation. As an example, the field describing the users' profile background color was not included in the feature list while number of friends and number of posts were. Table 4.1 contains a description of the fields selected.

4.2.3 Implementation of the Classifiers

We consider three state-of-the-art classification methods: Multinomial Logistic Regression (MLR) [159], Random Forest (RF) [143], and K-Nearest Neighbors (KNN) [160]. Each of these algorithms follow a different method to make the recommendation and they are all popular within the community.

We use the implementation of the algorithms provided by sci-kit learn [162], a Python library. The optimization of the internal parameters for each classifier was conducted as a combination of best practices in the field and experimental results in which we used the cross-validated grid search capability offered by scikit-learn. In summary, we calculated the value of the parameters for each classifier as follows. For KNN, we consider the single closest value based on the Euclidean distance between the observations; for RF, we chose entropy as the function to measure the effectiveness of the split of the data in a node; finally, for MLR, we selected the limited-memory implementation of the Broyden-Fletcher-Goldfarb-Shanno (*LM-BFGS*) optimizer as the value to optimize [163].

4.2.4 Obfuscation and Re-Identification

Obfuscation can only be understood in the context of data sharing. The goal of obfuscation is to protect the private individual fields of a dataset by providing only the result of some function computed over these fields [164]. To succeed, the possibilities are either to obfuscate the data or develop algorithms that protect it [165]. We

reasoned that an attacker will have access to any number of publicly available algorithms. This is outside of our control. However, we could manipulate the granularity of the information made available. Our task is to determine whether doing so is an effective way of protecting user privacy, particularly when obfuscated metadata is released.

In this work we focus on two classic obfuscation methods: data randomization and data anonymization [47, 166, 167]. Data anonymization is the process by which the values of a column are grouped into categories and each reading is replaced by an index of its corresponding category. Data randomization, on the other hand, is a technique that alters the values of a subset of the data points in each column according to some pre-determined function. We use rounding as the function to be applied to the data points. For each of the values that were altered, we rounded to one less than the most significant value (*i.e.*, 1,592 would be 1,600 while 31 would be 30). We measured the level of protection awarded by randomization by recording the accuracy of the predictions as we increased the number of obfuscated data points in increments of 10% until we reached full anonymization (*i.e.*, 100% randomization) of the training set.

4.2.5 Inference Methods

Statistical inference is the process by which we generalize from a sample a characteristic of the population. Bootstrapping is a computational method that allows us to make inferences without making any assumptions about the distribution of the data and without the need of formulas to describe the sampling process. With bootstrapping we assume that each sample is the population and then aggregate the result from a large number of runs (anywhere between 50 and 1,000 times depending on the statistic being drawn) [168]. In this study we are primarily interested in the precision and accuracy of each classifier as a measure of their ability to predict the correct user given a tweet. The results we present are an average over 200 repetitions of each experiment. In each experiment, the input data was randomly split between training and testing sets using a 7:3 proportion, which is a typical setting in evaluation of machine learning algorithms.

Table 4.2: KNN classification accuracy using ten observations per user of two features follower count and friend count for input. We ran each experiment for an increasing number of users u.

u	top result	top 5
10	94.283 ($\pm$0.696)	98.933 ($\pm$0.255)
100	86.146 ($\pm$0.316)	96.770 ($\pm$0.143)
1,000	70.348 ($\pm$0.112)	90.867 ($\pm$0.076)
10,000	47.639 ($\pm$0.039)	76.071 ($\pm$0.029)
100,000	28.091 ($\pm$0.089)	55.438 ($\pm$0.192)

4.3 Experimental Settings

4.3.1 Dataset

For data collection, we used the Twitter Streaming Public API [169]. Our population is a random[1] sample of the tweets posted between October 2015 and January 2016 (inclusive). During this period we collected approximately 151,215,987 tweets corresponding 11,668,319 users. However, for the results presented here we considered only users for which we collected more than 200 tweets. Our final dataset contains tweets generated by 5,412,693 users.

4.3.2 Ethics Considerations

Twitter is the perfect medium for this work. On one hand, users posting on Twitter have a very low expectation of privacy: it is in the nature of the platform to be open and to reach the widest audience possible. Tweets must always be associated with a valid account and, since the company does not collect demographic information about users upon registration, the accounts are not inherently linked to the physical identity of the users. Both these factors reduce but do not eliminate any ethical concerns that may arise from this work. Nonetheless, we submitted this project for IRB approval and proceeded with their support.

[1]It is worth noting that since we use the public Twitter API we do not have control on the sampling process. Having said that, an attacker will most probably access the same type of information. A typical use case is the release of data set of tweets usually obtained in the same way.

Figure 4.1: Change in accuracy for a single feature combination and increasing users.

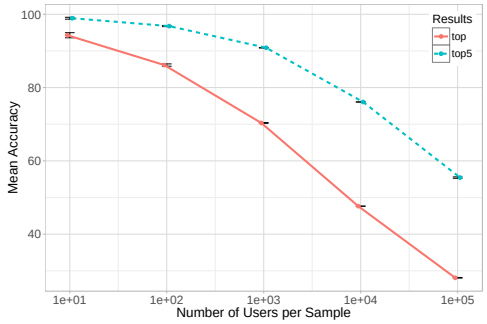
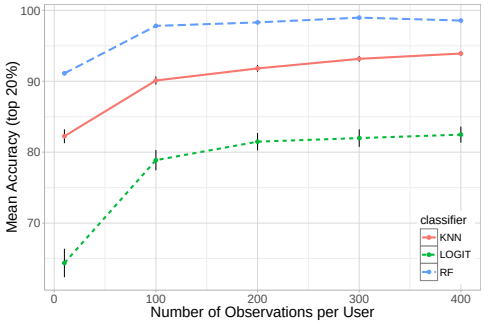


Figure 4.2: Performance of the top 20% of combinations per classifier for increasing observations per user.



4.3.3 Experimental Variables

We identified two variables that regardless of the features would influence the accuracy of the classifiers: the number of users considered in each experiment (which is equivalent to the number of classes in the classifier) and the number of observations per user. In this section we discuss both factors.

4.3.3.1 Number of Users

As an attack, guessing is only viable for smaller user pools: indeed, there is a 1:10 probability of randomly classifying a tweet correctly for a user pool made up of 10 users, whereas there is a 1:10,000 probability in a pool of 10,000 users. The likelihood of guessing correctly is inversely proportional to the number of users. Therefore, the first task was to compare and describe the way in which increasing the number of users affects the accuracy of the classifiers. We evaluate each algorithm (*i.e.*, MLR, RF, KNN) on a specific configuration of training data in order to obtain a trained model. We trained models for all feature combinations however, we present only the results for the best combination of parameters for each classifier.

We first analyze the impact of the number of classes. In Figure 4.1 we present with fixed parameters the effect of increasing only the number of outputs for each of the classifiers. Each model was built with two input features (*i.e.*, $n = 2$ where the features are number of friends and number of followers) and 10 observations per user. Some of the results we present are independent of the underlying classification

algorithm. For these instances, we present results using only KNN. As will be shown later, KNN shows the best performance in terms of prediction and resource consumption.

Figure 4.1 shows that the loss in accuracy is at worst linear. In a group of 100,000 users, the number of friends and of followers was sufficient to identify 30% of the users (around 30,000 times better than random). However, while the accuracy of the classification gradually declines, there is a dramatic increase in the cost (in terms of time) when building the model. As we will discuss in the last part of the Results section, the greatest obstacle with multi-class classification is the number of classes included in the model.

4.3.3.2 Number of Entries per User

The next variable we consider is the number of observations per user per model. Our objective is to visualize the relationship between accuracy and the number of tweets to set a minimum value for the rest of the study.

To set this parameter, we fixed the number of input features at $n = 2$ and $u = 1,000$ then we ran models with 10, 100, 200, 300, and 400 tweets per user. Figure 4.2 shows the aggregated results for the 20% most accurate feature combinations over 400 iterations. As the figure shows, ten entries is not enough to build robust models. For all classifiers we see that the behavior is almost asymptotic. There is a significant increase in accuracy as the number of observations per user reaches 100. However, for all subsequent observations, the variation is less apparent. Each of the points in the graph contains the confidence interval associated with the measurement however, for RF the largest error is 0.2. It is worth noting that given the number of observations per user needed for robustness, with our data set we could only scale as far as 10,000 users.

Using the bootstrapping method for inferences, going forward, we repeated each experiment 200 times. Each time with different configurations in terms of users and observations (*i.e.*, tweets) per user. By standardizing the number of observations per user at 200 tweets, we preempt two problems: first, by forcing all users to have the same number of tweets we reduce the likelihood of any user not

having some entries in the training set; and second, it prevents our results from being biased towards any user with a disproportionate number of observations. This might be considered as potentially artificial, but we believe it represents a realistic baseline for evaluating this attack.

4.4 Results

We present results in three areas: accuracy of classification, the resilience of each classifier in terms of their prediction accuracy when using obfuscated input data, and execution time of each algorithm and a method for improving their performance.

4.4.1 Identification

In building and comparing models we are interested in the effects and interactions of the number of variables used for building the models which we denote n and the number of output classes which we refer to as u.

We define accuracy as the correct assignment of an observation to a user. In a multi-class algorithm, the predicted result for each observation is a vector containing the likelihood of that observation belonging to each of the users present in the model. A prediction is considered correct if the user assigned by the algorithm corresponds to the true account ID. In the rest of the chapter, we report aggregated results with 95% confidence intervals. In our analysis, the account creation time was found to be highly discriminative between users. We present results with the dynamic features, defined below, then we present our findings with the account creation time, and finally, we give a comprehensive analysis of the task of identification given combinations of both static and dynamic features for three different classifiers.

4.4.1.1 Static Attribute: Account Creation Time

Twitter includes the Account Creation Time (ACT) in every published tweet. The time stamp represents the moment in which the account was registered, and as such, this value is constant for all tweets from the same account. Each time stamp is composed of six fields: day, month, year, hour, minute, and second of the account creation.

For perfect classification (i.e., uniqueness), we look for a variable whose value is constant within a class but distinct across classes (as explained before, each user in is a class). The account creation time is the very example of one such variable. We tested the full ACT using KNN and found that, even for 10,000 users, classifying on this feature resulted in 99.98% accuracy considering 200 runs. Nonetheless, the ACT is particularly interesting because while it represents one unique moment in time (and thus infinite and highly entropic), it is composed of periodic fields with a limited range of possible values. As we see in Table 4.3 individually, each of the fields has at least a 75% drop in entropy as compared to the combined ACT. Since full knowledge of the ACT can be considered as the trivial case for our classification problem, in the following sections we will consider the contribution of each field of the account creation time separately.

Table 4.3: Entropy calculation for feature list.

feature	entropy
ACT	20925
statuses count	16132
follower count	13097
favorites count	12994
friend count	11725
listed count	6619
second	5907
minute	5907
day	4949
hour	4543
month	3581
year	2947
post time	1789
geo enabled	0995
verified	0211

4.4.1.2 Dynamic Attributes

By dynamic attributes we mean all those attributes that are likely to change over time. From Table 4.1 these are the counts for: friends, followers, lists, statuses, and favorites, as well as, a categorical representation of the time stamp based on the hour of each post.

Table 4.4 presents the two best performing combinations in terms of accuracy for each of the values we consider for n and u. In 10,000 users there is a 92%

Table 4.4: KNN Classification using dynamic features.

u	n	features	accuracy
10,000	3	friend, follower, listed count	92.499 (±0.0008)
		friend, follower, favorite count	91.158 (±0.0006)
	2	friend, follower count	83.721 (±0.0005)
		friend, favorite count	78.547 (±0.0006)
1,000	3	friend, follower, listed count	95.702 (±0.0037)
		friend, follower, favorite count	93.474 (±0.0015)
	2	friend, follower count	91.565 (±0.0026)
		friend, listed count	89.904 (±0.0028)
100	3	friend, follower, listed count	98.088 (±0.0037)
		friend, follower, favorite count	97.425 (±0.0058)
	2	friend, follower count	97.099 (±0.0051)
		friend, listed count	95.938 (±0.0073)
10	3	friend, follower, favorite count	99.790 (±0.0014)
		friend, favorites, listed count	99.722 (±0.0015)
	2	friend, favorites count	99.639 (±0.0016)
		follower, friend count	99.483 (±0.0022)

Table 4.5: RF Classification using dynamic features.

u	n	features	accuracy
10,000	3	friend, follower, favorite count	94.408 (±0.0008)
		friend, follower, status count	94.216 (±0.0006)
	2	friend, follower count	81.105 (±0.0005)
		friend, favorite count	75.704 (±0.0006)
1,000	3	friend, follower, favorite count	96.982 (±0.0008)
		friend, follower, status count	96.701 (±0.0008)
	2	friend, follower count	90.889 (±0.003)
		friend, favorite count	89.271 (±0.004)
100	3	friend, follower, favorite count	99.286 (±0.0014)
		friend, listed, favorite count	99.149 (±0.0017)
	2	friend, follower count	97.690 (±0.0029)
		listed, friend count	97.275 (±0.00363)
10	3	friend, listed, favorite count	99.942 (±0.0005)
		follower, favorites, friend count	99.930 (±0.00061)
	2	friend, listed count	99.885 (±0.0008)
		follower, friend count	99.776 (±0.0013)

Table 4.6: Accuracy of the top combination for n number of inputs for the KNN classifier.

u	n	features	accuracy(%)
10,000	3	day, minute, second	96.737(±0.019)
	2	follower, friend count	83.719(±0.021)
	1	friend count	14.612(±0.036)
1,000	3	listed count, minute, second	99.648(±0.022)
	2	friend, listed count	92.809(±0.050)
	1	friend count	40.151(±0.089)
100	3	month, minute, second	100.00 (±0.000)
	2	minute, second	98.836(±0.101)
	1	friend count	78.650(±0.330)
10	3	month, minute, second	100.00 (±0.000)
	2	month, minute	100.00 (±0.000)
	1	friend count	96.428(±0.312)

Table 4.7: Accuracy of the top combination for n number of inputs for the RF classifier.

u	n	features	accuracy(%)
10,000	3	listed count, day, second	94.234(±0.022)
	2	friend count, minute	81.352(±0.347)
	1	friend count	23.958(±0.089)
1,000	3	friend count, minute, second	99.881(±0.008)
	2	friend count, second	97.28(±0.032)
	1	friend count	49.538(±0.086)
100	3	day, minute, second	100.00 (±0.000)
	2	friend count, minute	99.595(±0.023)
	1	friend count	81.489(±0.299)
10	3	day, minute, second	100.00 (±0.000)
	2	day, second	100.00 (±0.000)
	1	friend count	96.858(±0.256)

chance of finding the correct account given the number of friends, followers and the number of times an account has been listed. Table 4.5 presents similar results for the RF algorithm. Even without the ACT, we are able to achieve 94.41% accuracy in a group of 10,000 users. These results are directly linked to the behavior of an account and are obtained from a multi-class model.

4.4.1.3 Combining Static and Dynamic Attributes

As we expected from our feature selection, the top combinations per value of n inputs are different per classifier. Tables 4.6, 4.7, 4.8 show the accuracy and the error obtained for the best performing pair of features aggregated over all runs for the three classifiers.

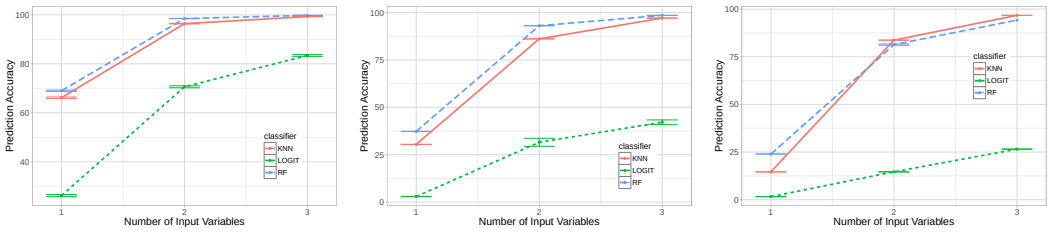
We can see that the least accurate predictions are those derived by means of the MLR algorithm. Then, probably for its robustness against noise, RF performs best for the smaller user-groups, but it is KNN that has the upper hand for the case where $u = 10,000$.

u	n	features	accuracy(%)
10,000	3	day, minute, second	96.060(±0.060)
	2	day, minute	29.571(±5.11)
	1	second	2.241(±0.080)
1,000	3	hour, minute, second	99.329(±0.060)
	2	day, minute	61.77(±5.11)
	1	hour	3.105(±0.080)
100	3	day, minute, second	100.00 (±0.000)
	2	second, minute	98.494(±0.116)
	1	second	26.303(±0.536)
10	3	day, minute, second	100.00 (±0.000)
	2	day, minute	100.00 (±0.000)
	1	second	94.129(±0.702)

Table 4.8: Accuracy of the top combination for n number of inputs for the MLR classifier.

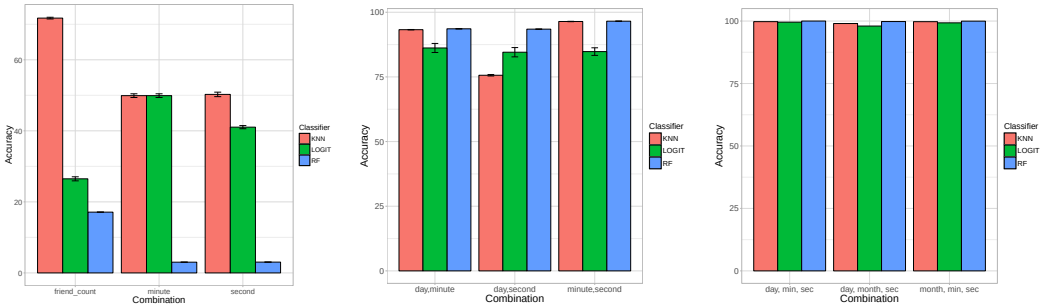
In general, the classification task gets incrementally more challenging as the number of users increase. We are able to achieve a 90% accuracy over all the classifier with respect to a 0.01% baseline offered by the random case. If we consider the 10 most likely output candidates (i.e. the top-10 classes), for the 10,000 user group there is a 99.22% probability of finding the user.

Finally, we looked at how the best performing combinations across all algorithms behaved in models for each classifier. The top three combinations per value of n are presented in Figures 4.4a, 4.4b, 4.4c. For the same value of u as we increase n the accuracy increases.



(a) $u = 100$ **(b)** $u = 1,000$ **(c)** $u = 10,000$

Figure 4.3: Averaged model accuracy for logarithmic-step user size increase.



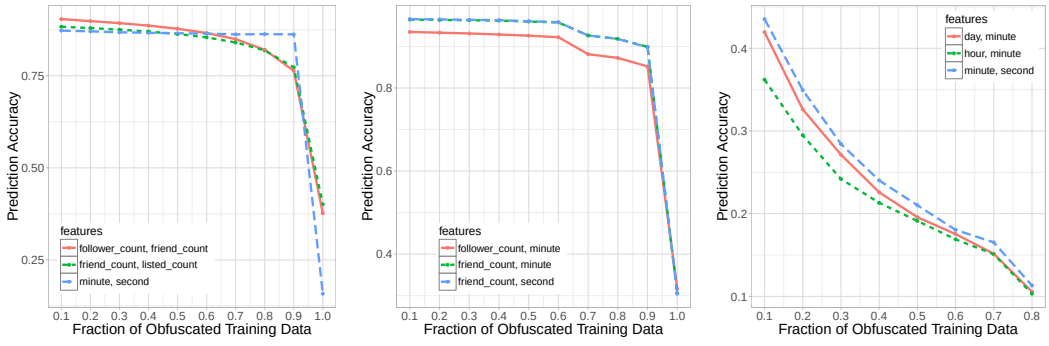
(a) $n = 1$ **(b)** $n = 2$ **(c)** $n = 3$

Figure 4.4: Performance of the most popular features for increasing tuple size.

4.4.2 Obfuscation

The final analysis looks at the effects of data anonymization and data randomization techniques on the proposed methodology. As we state in the Method section we start with the original data set then apply a rounding algorithm to change the values of each reading. The number of readings that pass through the algorithm increase in steps of 10% from no anonymization to 100% perturbation where we show full randomization. To test obfuscation, we selected the 3 most accurate combinations of features for $n = 2$ and $u = 1,000$ for each of the classification methods.

While we are not working with geospatial data, we find that similar to [170] the level of protection awarded by perturbation is not very significant until we get to 100% randomization. Figures 4.5a, 4.5b, and 4.5c show how each algorithm performs with an increasing number of obfuscated points. RF is the best performing, all three combinations stay stable for a longer period of time and gives the best re-



(a) KNN **(b)** RF **(c)** MLR

Figure 4.5: Change in predictive accuracy per classification algorithm for increasing percentage of input data obfuscation.

Figure 4.6: Mean execution time as a function of features.

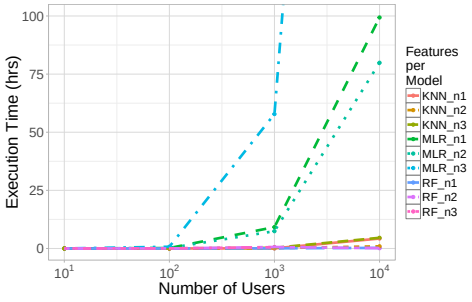
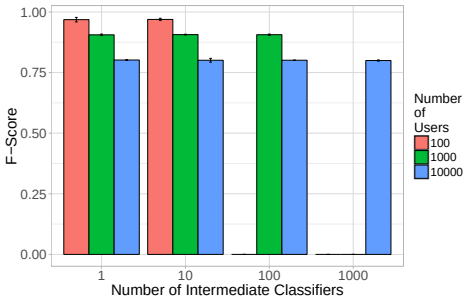


Figure 4.7: F-score for increasing intermediate sample sizes.



sult with data anonymity. MLR is the most sensitive of the three. Even with 20% randomization, there is a steep decrease in terms of prediction accuracy.

4.4.3 Execution Time

To compare the performance of the classifiers in terms of execution time we used a dedicated server with eight core Intel Xeon E5-2630 processors with 192GB DDR4 RAM running at 2,133MHz. For the implementation of the algorithms, we used Python 2.7 and Sci-kit learn release 0.17.1.

Figure 4.6 shows execution time as a function of the number of output classes in each model. Note that the performance gap between MLR and the other two is significant. While KNN and RF show a linear increase over the number of users, the rate of change for MLR is much more rapid. At $u = 1,000$ and $n = 3$, for example, MLR is 105 times slower than RF and 210 times slower than KNN. The performance bottleneck for multi-class classifiers is the number of output classes. Finding a viable solution is fundamental for this project and for the general applicability of the method. To address this, we implemented a *divide and conquer* algorithm where we get intermediate predictions over smaller subsets of classes and build one final model with the (intermediate) results. This method allows for faster execution and lower memory requirements for each model and parallel execution resulting in further performance enhancements. Figure 4.7 shows the effectiveness of the proposed method for $u = 100, 1000,$ and 10000. We also observe that the accuracy is independent from the number of subsets.

4.5 Discussion and Limitations

We have presented a comparison of three classification methods in terms of accuracy and execution time. We reported their performance with and without input obfuscation. Overall, KNN provides the best trade-off in terms of accuracy and execution time with both the original and the obfuscated data sets. We tested each of the algorithms following the wrapper method by increasing the number of input features in each model in steps of one. The results, summarized in Tables 4.6, 4.7 and 4.8, show that the metadata can be effectively exploited to classify individuals inside a group of 10,000 randomly selected Twitter users. While both KNN and RF are similar in terms of accuracy and execution, MLR is consistently outperformed. Moreover, as shown in Figures 4.3a, 4.3b, and 4.3c, as the number of output classes increases, the difference in performance becomes more apparent. It is also important to note that, as shown in Table 4.8, the accuracy exhibited by MLR depends entirely on the six constituent features of the account creation time. Finally, the performance of MLR is the most sensitive to obfuscation as shown in Figure 4.5c the rounding algorithm results in a monotonic drop in accuracy for the best performing combinations.

One of the challenges in this work was the scalability of the classification algorithms. A key challenge in designing this type of algorithm is performance in terms of the scalability of its parameters. We found that while both the number of input features and the number of output classes have a detrimental impact on performance, the bottleneck can be identified in the latter. To address this we designed an ensemble-like algorithm that subsets the total number of users considered and creates smaller trained models using fewer output classes and then, combine them. Figure 4.7 shows that the results obtained from this method are reliable. We ran comparisons for $u = 100, 1000$, and 10000 using KNN and found that the precision and the recall of each model are equivalent to the ones obtained from a single model. This proves the scalabiliy of the proposed approach beyond the number of users available in our test data set.

4.6 Summary

In this chapter, we have used Twitter as a case study to quantify the uniqueness of the association between metadata and user identity, devising techniques for user identification and related obfuscation strategies. We have tested the performance of three state-of-the-art machine learning algorithms, MLR, KNN and RF using a corpus of 5 million Twitter users. KNN provides the best performance in terms of accuracy for an increasing number of users and obfuscated data. We demonstrated that through this algorithm, we are able to identify 1 user in a group of 10,000 with approximately 96.7% accuracy. Moreover, if we broaden the scope of our search and consider the best 10 candidates from a group of 10,000 users, we achieve a 99.22% accuracy. We also demonstrated that obfuscation strategies are ineffective: after perturbing 60% of the training data, it is possible to classify users with an accuracy greater than 95%.

We believe that this work will contribute to raising awareness of the privacy risks associated to metadata. It is worth underlining that, even if we focused on Twitter for the experimental evaluation, the methods described in this work can be applied to a vast class of platforms and systems that generate metadata with similar characteristics. This problem is particularly relevant given the increasing number of organizations that release open data with metadata associated to it or the popularity of social platforms that offer APIs to access their data, which is often accompanied by metadata.

Chapter 5

Device Identification from Magnetic Field Emissions

Smartphones are equipped with a wide range of sensors that measure a variety of physical quantities including light, humidity, orientation, acceleration, pressure, proximity, and location [171]. The information collected is used to enhance user experience (e.g., automatic screen brightness adjustment and energy saving mode during phone calls), to improve services (e.g., location-based services like Uber and calendar reminders based on travel time), and also as a commodity that benefits third parties (e.g., the use of microphones in mobile devices to respond to ultrasonic beacons in order to measure the audience of an advertisement [172]). In many instances, identifying a device or a user is trivial and expected (particularly for services that require registration) but for unknown third parties (i.e., secondary data collectors that gain access to the data without users being explicitly aware of their activities) or for applications where the user expects to remain anonymous (e.g., applications for gaming, weather, and the news), many of these sensors potentially expose the user's private information [173].

The magnetometer is the sensor responsible for measuring the magnetic field in the environment of the phone. It is commonly available in all platforms and is usually combined in the same chip with the sensors for linear acceleration and gyration. Apple, with the release of the iPhone 3GS in 2009, added magnetometers to the sensor array available on their devices [174]. The Android operating system

opened API support that same year with the release of Cupcake (Version 1.5) [175]. The magnetometer provides a new means to effectively fingerprint mobile devices in a manner that is *transparent* to users. Accessing magnetometer readings requires no permission declaration on either platform and the fact that the magnetic field is radiated means that it can be measured from outside the phone.

In this chapter, we show that measurements of the magnetic field can be used for identification. Every electronic device generates a magnetic field. In smartphones, it varies depending on the components active in the phone, the tolerance level of each component, the degradation of the manufacturing materials on the phone, and the topology of the underlying circuit. Each cell phone contains hundreds of thousands of transistors alone and every component in the phone is unique. Previous work shows that two seemingly identical components have unavoidable differences that arise from the manufacturing process [108]. This variability has been leveraged for identification of sensors and other integrated circuits [113, 109]. We present two identification attacks that rely on magnetic fields: a *malware-based* approach that can be carried out by any application installed on a device that contains a magnetometer and a *proximity-based* attack that can identify any device inside the range of the external sensor measuring the magnetic field.

In the malware attack, the device on which the app is installed is both the source and the instrument of collection of the magnetic field. To demonstrate the feasibility of the attack, we released an app on the Play Store. Our final dataset comprises 175 devices and a minimum of 10,000 readings per device. The application collects readings from the magnetometer and transmits them back to our servers for analysis. We use these readings to generate *fingerprints* with which we identify devices. We show that selecting 1,000 randomly spaced readings is sufficient to identify one device from a group of 175 with an accuracy of 98.9%. To mitigate the risk to users, we propose adding sensor readings to the permission platform. While this would not directly affect the generation of the fingerprint, it would at least serve as a warning system to users.

However, even after protecting the readings with the new permission, bypass-

ing this safeguard is still possible: we present a second attack vector where the magnetic field readings are collected from a device in close proximity to the target. Here, the requirements (i.e., the sensor and the software that collects the readings) have shifted to the attacker and the victim has no possible means of detection.

This work is not the first in considering magnetic emissions in the context of privacy and security. Previous work shows that electromagnetic leaks can be used to determine the instruction being executed by a processor [176] and to extract data from a system through the manipulation of memory access [177, 178] or through write instructions to the hard drive [179]. However, *this is the first work to use magnetic fields as a way to achieve identification*. There are some *practical* difficulties related to device fingerprinting in general. First, the fingerprinting accuracy attained with a specific set of features is likely dependent on the total number of devices present in the dataset. Second, the market diversity for mobile devices with all the possible vendors, carriers, and designs makes it difficult to find a single universal feature for identification. We address these by conducting a user study with 175 devices over a period of four months.

In this chapter, we propose three key contributions. First, we investigate and characterize the emission and collection of magnetic fields from a large set of off-the-shelf mobile devices, discussing how readings can be used to generate fingerprints and later be used in an identification attack. In second place, we present two identification attacks based on magnetic field emissions that enable adversaries to track devices remotely. Tracking can happen without the knowledge and consent of the user because of the lack of permissions covering sensor readings, particularly the electromagnetic field, on electronic devices. Effectively, this means that attacks of this nature can be carried out by anyone, at any time, and they are as of now undetectable. Furthermore, we present, for each attack the challenges and open research questions regarding the countermeasures available and our opinion as to why addressing these will not result in the fingerprint being undetectable. Finally, we present a methodology to process sensor readings and through the use of two machine learning algorithms, k-Nearest Neighbors (KNN) [144] and Random

Forests (RF) [143], build models to distinguish between devices. We show that, by means of this approach, we are able to achieve correct identification of the devices with up to 98.9% accuracy in 1,000 readings.

5.1 Background: Privacy and Electromagnetism

These next two chapters present different applications of the side channel information leaked through magnetic field readings. In this chapter, we present the ability of an attacker to identify a specific device and in Chapter 6 we explore how the same readings can be used to infer the location type of a user. While we are the first to present these applications, we are not the first to look at the relationship between electricity and magnetism in terms of security and privacy.

In [179] Biedermann *et al.* use an off-the-shelf magnetometer to measure the magnetic field that is emitted by the movement of the pointer to different addresses of the disk. Following their analysis, it is possible to deduce which instruction is currently being executed by the processor without requiring either direct access or digital interaction with the machine. Simillarly, Zajic *et al.* test the boundaries of EM memory leakage in terms of distance and present the relationship of memory location and transmission strenght [178]. Following the same line of research, Callan *et al.* propose a measurement that estimates the amount of information leaked by different instructions and the general characteristics of the command (*e.g.*, distance to memory and integer division operations) that originated the signal [176].

Moving slightly into the realm of mobile devices, Guri *et al.* present a method for information exchange between a desktop and a cellphone over radio signals generated by the manipulation of the multi-channel memory architecture. Transmitting data this way happens is possible through electromagnetic emissions, GSM, UMTS or LTE technologies. While operable over up to 30 meters, this proposal requires malware installed on the (air gapped) desktop and the appropriate software on the mobile device to receive it [177]. Most recently, in [180] the authors leverage magnetic field signals to intercept the data being written to a laptop's hard drive. Most of the proposed attacks require dedicated software on the target and the mobile device

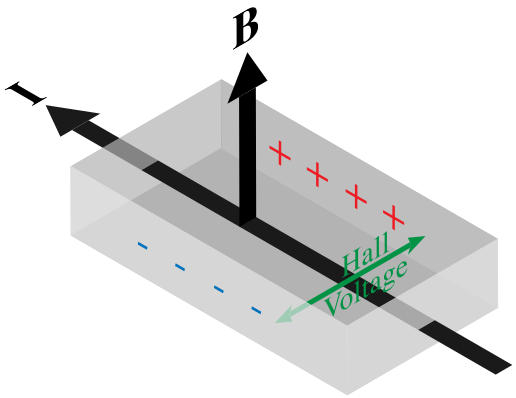


Figure 5.1: Measuring the Magnetic Field from a MEMS device. The flow of current (I) through a conductor polarizes the material. The resulting voltage between opposite sides is known as the Hall Voltage, from which you can derive the magnetic induction B.

Table 5.1: Starting from the left, column one lists the two types of magnetic field events. The middle columns contain the size and description of the response. Finally, the column on the far right contains the units corresponding to the values reported.

Sensor Call	Response	Description	Units
TYPE_MAGNETIC_FIELD	SensorEvent.values[0]	Geomagnetic field strength: x axis	μT
	SensorEvent.values[1]	Geomagnetic field strength: y axis	
	SensorEvent.values[2]	Geomagnetic field strength: z axis	
TYPE_MAGNETIC_FIELD_UNCALIBRATED	SensorEvent.values[0]	Geomagnetic field strength (without hard iron calibration): x axis	μT
	SensorEvent.values[1]	Geomagnetic field strength(without hard iron calibration): y axis	
	SensorEvent.values[2]	Geomagnetic field strength(without hard iron calibration): z axis	
	SensorEvent.values[3]	Iron bias estimation: x axis	
	SensorEvent.values[4]	Iron bias estimation: y axis	
	SensorEvent.values[5]	Iron bias estimation: z axis	

is used only as a measurement instrument.

In a clear break from these papers, the analysis we carry out on magnetic field signals is not an indication of a memory access or a particular instruction. Instead, by understanding variability we measure uniqueness and by using signal degradation and interference we pinpoint events. To the best of our knowledge we are the first to attempt either one of these applications.

5.2 The Magnetometer Sensor

The popularity and use of magnetic sensors have exploded in the last decades and, while there are different technologies capable of measuring magnetic fields, the most common distribution for mobile devices is the Micro-ElectroMechanical Sys-

tems (MEMSs) [181]. MEMS is the name given to batch fabrication techniques that allow for the combination of miniaturized (typically in the range of micrometers to millimeters) mechanical and electrical systems that generate a response in the macro scale. These silicon-based systems are composed of mechanical structures, sensors, actuators, and electronics all in the same chip [182].

5.2.1 The Hall Effect

Android classifies the magnetometer as a position sensor [171]. In many devices it is, along with the accelerometer and the gyroscope, contained in a 9-axis MEMS chip, where each sensor provides data over 3 axes [183, 184]. In order to measure the magnitude of the magnetic field, the chip uses a Hall Effect transducer. As shown in Figure 5.1, in a semi conductive material, such as silicone, there is a direct relationship between the magnetic field B and the current I. Depending on its orientation, the magnetic field will pull the current to one side of the surface causing the plate to be charged. The resulting potential difference is labeled as the Hall Voltage [185, 186].

$$F = q_0 E + q_0 v \times B \tag{5.1}$$

The magnitude of the force is given by the Lorentz Force Equation (Equation (5.1)) which states that when an electron moves the force it experiences is a function of its charge, the direction in which it is moving, and the orientation of the magnetic field it is moving through. In the equation, F is the force, E the electric field, v the velocity of the charge, B the magnetic field, and q_0 the magnitude of the charge.

Over time, the force (F) is canceled out through a balance achieved between the the magnetic force pushing the charged particles in one direction and the electric force pushing them back to the center of the plate. At which point,

$$V_H = -wvB \tag{5.2}$$

where the magnetic field becomes a linear function of the Hall Voltage (V_H), the thickness of the semi conductive material in the direction parallel to the current (w),

and the velocity of electrons traveling through the sensor (v) [187].

In reality, the measure is an approximation: while we are theoretically interested in the value of the magnetic field (H) measured in amperes per meter (A/m), the sensor actually measures the magnetic induction (B) measured in Tesla (T). Equation (5.3) shows the relationship between the magnetic field and the magnetic induction. The factor μ in the equation corresponds to the magnetic permeability of the material. While this value varies depending on the medium through which the wave propagates, we assume that the medium is similar for all mobile devices (i.e., plastic, air, and other electronics). We assume that the value of μ is constant for all the readings and we use H as a proxy of B [188].

$$B = \mu H \tag{5.3}$$

5.2.2 Sensor Calibration

The reading reported by the magnetometer is modeled by Equation 5.4.

$$y_m = T_m m + h_m + \varepsilon \tag{5.4}$$

where T_m is a matrix representing, among others, the soft-iron distortion, m is the true magnetic field vector, h_m is a bias vector dominated by the hard-iron distortion, and y_m is the measurement vector in three dimensions [184].

Contrary to other sensors which can be calibrated using a correction formula determined at manufacturing time, the magnetometer requires as part of the calibration the magnetic field disturbances caused by ferromagnetic materials on the PCB [189, 190]. The practical implication is that two of the terms in Equation 5.4 are computed in real time.

The two main types of disturbances to the measurements collected from the magnetometer are known as soft-iron and hard-iron interference. The soft-iron effect is caused by materials that have no intrinsic magnetic field and are attached to the magnetometers' reference frame [184]. Metals like iron and nickle produce soft iron distortions when, for example, the geomagnetic field induces a load on them.

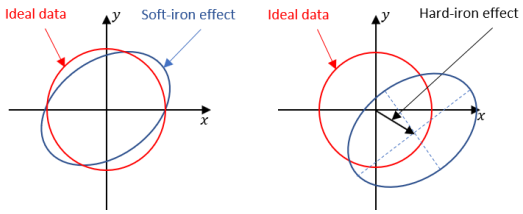


Figure 5.2: Effect of hard-iron and soft-iron distortion on magnetic field readings [2].

The hard-iron effect is instead caused by materials that are permanently magnetized (*i.e.*, materials that generate their own field). In a mobile device, the sensor rotates together with the other (magnetized) components and the hard-iron distortion corresponds to an additive vector to the true readings.

In an environment free from interference, rotating the sensor in an axis perpendicular to the surface of the circuit and plotting the measurements results in a circle centered at $(0,0)$. In Figure 5.2, this ideal data is depicted by the red circle where the radius of the circle is the magnitude of the magnetic field. The soft-iron effect, shown on the left, causes the circle to become an ellipse. The hard-iron effect, on the right, displaces the center of the circle (or the ellipse) in either of the axes. Typically, hard-iron and soft-iron effects, collectively known as the *bias*, occur together and correcting the distortion requires to first remove the hard-iron effect (and centering the ellipse at $(0,0)$) and then correcting the distortion to the circle [189]. From a computational perspective, different authors have proposed a variety of methods from maximum likelihood estimation [191] to numerical methods employing gradient descent [184] and adaptive measures using the accelerometer and gyroscope present on the same chip [190], calibration for magnetometers is an active area of research.

5.2.3 Software Interface

The Android Sensor Stack controls the access to the magnetometer [192]. It has seven levels of abstraction that go from the low-level hardware component to the high-level software application. The lower layers are implemented by the hardware manufacturers and include drivers and library interfaces to the sensor. The higher levels are determined by Google and are made available for developers to use in

their applications through the Software Development Kit (SDK). The API allows an application to register a sensor and it can be programmed to generate a response upon a value change event. Table 5.1 describes the data structure used by Android to represent the readings of the magnetic field. The magnetic sensor has the ability to output both the raw sensor readings and a calibrated measure for internal interference. The calibrated reading is the combination of the uncalibrated reading and some bias. This bias is the internal magnetic field generated by the phone. The two types of sensors on the left-most column along with the values and the differences between them will be discussed in more detail in Section 5.5.4.

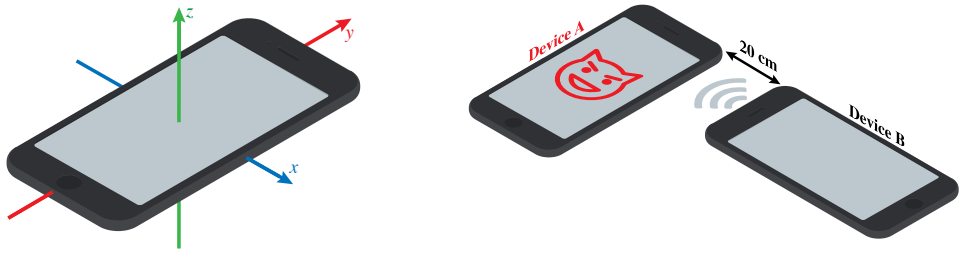
5.3 Overview of the Attacks

We consider an adversary who aims to establish the identity of a device without directly requesting the Unique Device Identifier (UID) provided by the platform. Android protects the UID through the telephony permission, which is classified as a dangerous permission, and the attacker may not want to alert the user as to their intention [193]. The methodology we propose results in a strong component of the identity (with an accuracy of over 90%) being revealed to attackers without the consent of the user. We assume that the adversary can either have an application installed on the target device (*malware attack*) or get in close physical proximity to it (*physical proximity attack*) and that the attacker will have access to the collected readings.

5.3.1 Malware Attack: Identification Through the Analysis of the *Bias* reported by the sensor

We first consider a malware attack where the primary goal of the adversary is to maximize the number of devices to fingerprint. The underlying assumption is that each phone has both a magnetometer and an app that collects the readings and transmits them back to the attacker. The app gets installed on the phone through the usual means: through social engineering, drive-by download, or as a hidden task in a legitimate application.

The magnetic induction (B) reported by the Android SDK is measured in the



(a) The position of the 3D axes are fixed with respect to the phone.

(b) A well placed sensor will capture the emissions of proximate devices.

Figure 5.3: The magnetometer collects 3D readings of the magnetic field in the vicinity of the device. The same axes are used to eliminate the signal emitted by the device.

3D coordinate plane. Figure 5.3a shows the collection axes with respect to all Android devices. The Hardware Abstraction Layer (HAL) interface provided by Android gives developers six values corresponding to the raw sensor readings and the device's internal bias each reported in three dimensions.

In this first attack we use the magnetic field emitted by the phone (i.e., the *bias*) as input features to the models. Section 5.5 provides an in-depth analysis of the attack, including a description of our use of each of the fields in Table 5.1.

5.3.2 Physical Proximity Attack: Identification Through Readings Collected Through an External Device

The first attack is in some sense an *internal* attack entirely software-based. The sensor, while present in the device, is not being used to assess the environment in which it operates but rather to report a value that comes as the outcome of a calibration equation. In other words, there is no evaluation of external phenomena, but only of the intrinsic characteristics of the MEMS-based sensor. Instead, in the second attack, the sensor is used to sense the radiation coming from the device to be identified, i.e., it can be seen an *external* attack. Figure 5.3b presents the second attack scenario. We have two devices: a target or victim and an adversary or attacker. The requirement is that the attacker has a magnetometer and the software with which to collect and process the readings.

The challenge is to extract a set of characteristics that describes the behavior

of the signal emitted by the targeted device from the environmental signal collected by the attacker (which we assume we have). The deployment of the second attack is described in Section 5.6 and the features selected for identification are discussed in detail in Section 5.6.2.

5.4 Description of the Identification Model

To show the feasibility of the method we propose, we follow three steps: data collection; feature definition and analysis; and, identification. Each step will be discussed in detail in the remainder of the chapter.

To carry out identification, we use supervised classification algorithms. We are interested in two pieces of information: a unique identifier for each device which will be used as ground truth (i.e., the label of each observation) and the measure of the magnetic field emitted by each device (i.e., the separable features).
For each device d_i in our dataset $\mathbf{D}$ of size $|\mathbf{D}| = M$

$$\mathbf{D} = \{d_1, d_2, \ldots, d_i, \ldots, d_M\},$$

we have a set of L independent readings

$$\mathbf{R}^{d_i} = \{\mathbf{r}_1^{d_i}, \mathbf{r}_2^{d_i}, \ldots, \mathbf{r}_j^{d_i}, \ldots, \mathbf{r}_L^{d_i}\},$$

with i being the unique identifier of each device and, therefore, the label assigned to the appropriate reading during classification. Each reading $r_j^{d_i}$ for device d_i is composed of N features on which the classifier is built.

$$\mathbf{F}(\mathbf{R}_j^{d_i}) = \{f_1(\mathbf{R}_j^{d_i}), f_2(\mathbf{R}_j^{d_i}), \ldots, f_k(\mathbf{R}_j^{d_i}), \ldots, f_N(\mathbf{R}_j^{d_i})\}.$$

The model trained on $\mathbf{F}(\mathbf{R}_j^{d_i})$ is then evaluated on unseen unlabeled observations. We will use d_{unk} to denote a reading taken from an unknown device. This will be the case for all the readings used for testing.

The outcome of each test $\mathbf{R}_j^{d_{unk}}$ is a conditional probability distribution given by $Pr(d_i|\mathbf{F}(\mathbf{R}_j^{d_{unk}}))$ for each device d_i in $\mathbf{D}$. Finally, we use the optimal decision rule to determine the device $\hat{d}$ to which $\mathbf{R}_j^{d_{unk}}$ belongs to

$$\hat{d} = argmax_{d_i} Pr(d_{unk} = d_i|\mathbf{F}(\mathbf{R}_j^{d_{unk}})).$$

If the predicted device $\hat{d}$ matches the ground truth (i.e., the true label of the reading), which we have for the testing set, we consider the test a success and classify it a *hit*. If alternatively, the observation is misclassified, then it becomes a *miss*.

To summarize, we can formulate our research question as follows: *given a set of devices, are the differences in the characteristics of the radiated emissions between each device sufficient to uniquely identify each one of them?*

5.5 Malware Attack: Identification through the Bias Readings

The first set of results we present are with respect to the malware attack, i.e., an attacker that wishes to remotely identify devices such as phones and tablets. The main challenge in remotely identifying devices is finding a weakness that can be accessed across all models and all manufacturers. Like for previous studies, the success of the malware attack depends on the device having the appropriate hardware. Given the popularity and widespread use of the magnetometer, we accept this as a reasonable constraint. To collect sensor information, an attacker only requires access to the API. The most natural way for data to be collected and transmitted is through an application. In order to use it they would need to request network access permission to send information. Alternatively, as it was shown in [110], the attacker might gain access to the readings through a browser application and collect them that way. We chose to develop an application to access `TYPE_MAGNETIC_FIELD_UNCALIBRATED` and collect the 3D readings from the phone's magnetometer. From the six readings we obtain from each sensor event (see Table 5.1), we focus on those that correspond to the internal bias of the phone. The values for the *bias* are constant for a session and we will show that only a few measurements are sufficient to generate a robust signature. Moreover, once generated, we observed that the signature is stable over time and therefore the frequency with which the sensor must be sampled to maintain accuracy is reduced to a bi-monthly event.

The main implication of this attack resides in the ability of service providers

(and other interested parties) to track users across multiple devices by linking and distinguishing different devices through log-in events;alternatively, it can be used to recognize the same device when the application has been removed then reinstalled, or to link multiple accounts that connect from the same device.

5.5.1 Ethics

While we are only identifying devices, the typically exclusive relationship between owner and device (as well as their proximity) implies we are indirectly identifying users. In addition to sensor readings, we collected basic demographic information as well as unique device identifiers that could be traced back to individuals. For these reasons, we submitted this project to our institution's review board and received ethical approval for this work.

Each participant had access to an overview of the project as well as details on the type of data that would be collected throughout the study. Additionally, the information sheet and the informed consent form are available on the project's website as well as withdrawal procedures and a contact email in case of any follow-up questions.

5.5.2 General Characteristics of the Dataset

To carry out the attack, we developed *MyMagneticField*, an app distributed through the Google Play Store. Figure 5.4 presents a screenshot of the main *activity* (following the Android terminology) of the app. In exchange for their participation, users were able to see (i) the magnetic field displayed on the screen (both the 3D readings and the combined magnitude); (ii) an electronic compass; and (iii) a heat map showing the intensity of the magnetic field at the locations of the data points.

The app was installed by 315 users in 15 countries. We identified 41 manufacturers with 61% of all devices belonging to the top 5 most popular brands (i.e., Samsung, Xiaomi, Huawei, Motorola, and Lenovo) and a total of 187 distinct models. The magnetometers can all be traced back to 14 vendors, the most popular being Yamaha Corporation. We also found that the same phone manufacturer may have more than one company supplying their sensors.

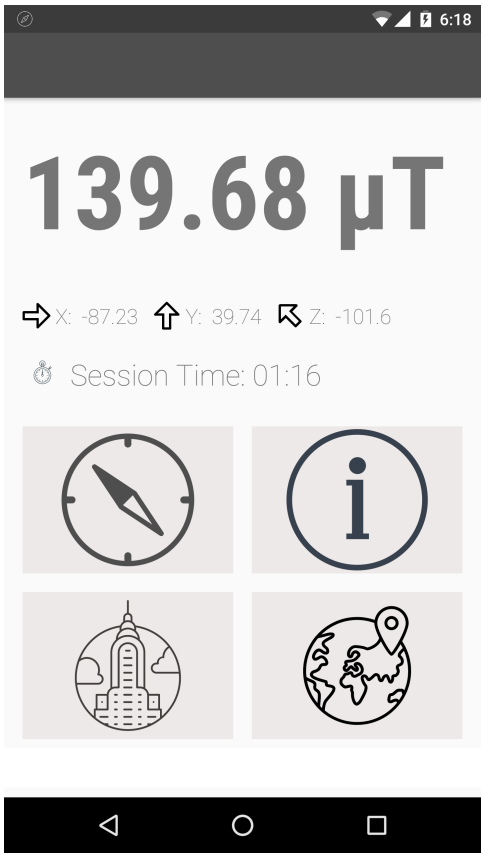


Figure 5.4: Screenshot of the main activity of the application.

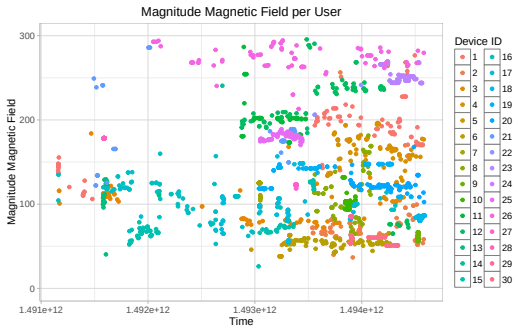


Figure 5.5: Magnitude of the *bias* for a subset of users.

From the dataset, we discarded all devices for which we had less than 33 minutes 20 seconds of data collection (the equivalent of 10,000 readings) and all those users that installed the application on phones that did not have a magnetometer. We also deleted entries for which the magnetic readings were all recorded as $0\mu T$. The final dataset contains readings from 175 devices.

5.5.3 Classification Task

Figure 5.5 provides a high-level view of the distribution of the magnitude of the magnetic field for 30 devices collected over time. We observe that the classes are not easily separable: the values are clustered together, often overlapping. The behavior displayed suggests that, while the emissions of each device are not random, boundary-based algorithms are not likely to be effective. We select Random Forest (RF) [143] and k-Nearest Neighbors (KNN) [144] for the classification task. We chose RF for its robust behavior to noise and KNN for its distance-based selection of the predicted class.

We use the term model to refer to the evaluation of an algorithm with specific training data (i.e., a classifier with a defined dataset). The parameters required for each algorithm were chosen as a combination of best practices and empirical results through the use of the grid search algorithm in scikit-learn [162]. KNN requires two parameters to be set: the number of neighboring points to be considered in the class assignment of the unlabeled data (i.e., the value of k) and the definition of the distance metric that links any two points. In our KNN models, we use Euclidean distance as the metric between two points as recommended in [194] and $k = 1$ as the optimal value reported by the grid search.

5.5.4 Feature Generation

The sensor stack introduced in Section 5.2 provides a list of variables developers can use to access each of the sensors (with their corresponding data streams) available in the device. There are two possible values that may be used to access magnetometer readings: `TYPE_MAGNETIC_FIELD` and `TYPE_MAGNETIC_FIELD_UNCALIBRATED`. Equation 4.2 shows the relationship between observations collected using each of these values. In the equation below, each element is a vector that contains features in three axes: x, y, and z. $\mathbf{R}_{unC}$ contains the raw readings collected from the *uncalibrated* variable and $\mathbf{R}_C$ corresponds to radiation readings external to the phone. The values for $\mathbf{R}_{IB}$ represent the internal bias of the phone, in other words, the magnetic field emitted by the device that is expected to interfere with the environmental measure.

$$\mathbf{R}_{unC} = \mathbf{R}_C + \mathbf{R}_{IB} \tag{5.5}$$

The physical interpretation of Equation 4.2 confirms the feasibility of the attack: each device does, in fact, generate a field that is calculated at run-time which interferes with the environmental reading. Essentially, the specification states that the raw sensor readings from `TYPE_MAGNETIC_FIELD_UNCALIBRATED` are the environmental magnetic field as well as the noise introduced by the device itself (i.e., the internal magnetic field of each phone), which they refer to as bias.

The correction formula for each sensor is provided by the device manufacturer. However, we know from the SDK that the bias correction takes into account three factors: the internal temperature of the phone, the internal interference generated by electrical components in the device (i.e., hard-iron calibration), and the internal interference created by any shielding materials contained in the device (i.e., soft-iron calibration) [175]. In a tri-axial MEMS magnetic sensor, the corrections, like the measurements, are reported in three dimensions. Table 5.1 gives the API's description of the values returned by the sensor and the difference between the calibrated and uncalibrated readings.

In all the experiments presented in this section, we use the values of the *bias* reported in three axes, x, y, and z with respect to each device, as the input features for each of the models. We treat measurements as independent data points (i.e., we do not consider in our analysis the temporal relationship between consecutive measurements). As we will show in the following sections, given the high accuracy of the attack, there was no need for more complex manipulation of the input data. Rather than being detrimental to the proposed idea, the simplicity of the protocol shows that this method is effective for identification.

5.5.5 Results

The magnetic field radiated by each device generates a unique signature that can be accessed through the sensor stack by the bias values reported in `TYPE_MAGNETIC_FIELD_UNCALIBRATED`. In Table 5.2, the values presented

s	classifier	precision	accuracy
10,000	KNN	0.995 (±0.019)	0.995 (±0.0001)
	RF	0.995 (±0.016)	0.995 (±0.0001)
1,000	KNN	0.989 (±0.002)	0.989 (±0.0004)
	RF	0.988 (±0.002)	0.988 (±0.0003)
100	KNN	0.964 (±0.0005)	0.964 (±0.001)
	RF	0.957 (±0.0004)	0.957 (±0.002)
10	KNN	0.888 (±0.0001)	0.871 (±0.011)
	RF	0.869 (±0.0001)	0.840 (±0.015)

Table 5.2: Precision and Accuracy for the classification of 175 devices. In the table, *s* denotes the number of readings or samples included in the training set.

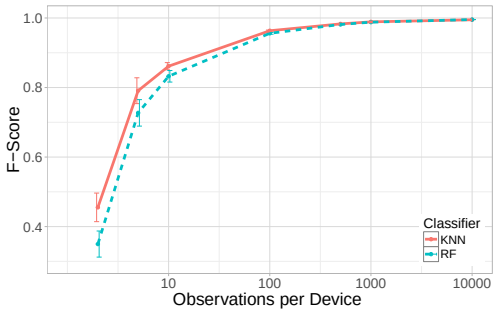


Figure 5.6: Determining the impact of the number of observations per device for 175 devices.

are an aggregation over 60 runs of each model. The left-most column (*s*) contains the number of measurements per device used to train each model. Figure 5.6 shows the influence of the number of observations included in the training set for each device. For completeness, we also ran the experiments following a 10-fold cross-validation scheme for each model and found the results to have no significant difference to the ones presented using aggregated randomized sampling. Combined, these results show that the accuracy does not reflect models that are overfitted but rather a true identifier contained in the data.

In terms of the success of the attack, Table 5.2 shows that for the same number of users, higher accuracy requires an increase in the number of observations per user. In Section 6.3 we discussed that all valid users have at least 10,000 observations. As mentioned above, each experiment was conducted 60 times and the results are an average over all the runs. For each user (in every iteration), all observations (ordered by time) are split into a 70/30 proportion for training and testing data respectively. The observations used in the algorithm are chosen randomly from all data points collected for each user.

5.5.5.1 Sampling Rate: Interval Between Readings

In mobile devices, the normal frequency range of the emitted magnetic field is between 200 Hz and 2.4 kHz [195]. From the strictly theoretical analysis based on Shannon's Sampling Theorem [196], the sampling frequency for our work should

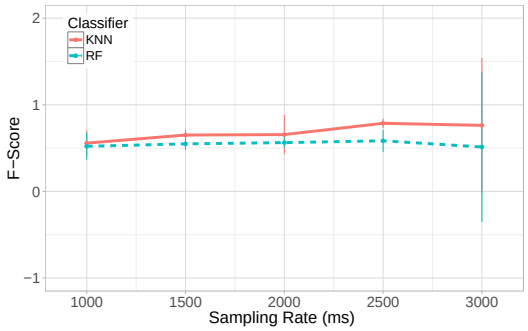
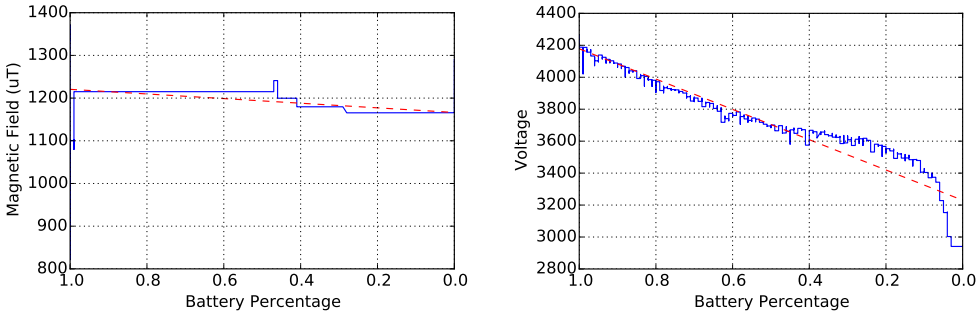


Figure 5.7: Change in classification F-Score for 20 users, 100 readings per user, and 500ms interval increase between consecutive measurements.

be between 400 Hz and 4.8 kHz. At worst, the sensor should report measures every 2.5 milliseconds (ms); at best, samples should be collected every 0.208 ms. However, this range is unattainable for two reasons: on one hand the usability of the application (and the undetectable nature of the attack) and on the other, the sampling framework provided by Android in the API.

There is a hard limit on the sampling rate for all sensors in a device fixed at 1 kHz [183]. This is lower than the upper boundary of our range at 12 kHz. Moreover, even at 1 kHz, the measurements are not guaranteed to happen at fixed intervals. To obtain readings, the framework requires a `SensorManager` to be initialized with the sampling rate. This sampling rate is interpreted by the OS as a flexible request. Android returns readings with an actual frequency anywhere between 90 and 220% of the requested value. The variability is added to accommodate for the sensor's operating conditions and any discrepancies with the CPU clock [183]. Therefore, even if the theoretical rate was within the Android-allowed parameters it is still impossible to enforce a fixed time span between measurements in the system.

In terms of usability, higher sampling rates translate into a steep decrease in the battery life of the device. During beta testing, many participants uninstalled the app because of the high battery consumption. Keeping the same settings in the full deployment would amount to low retention rates and slow dissemination, both of which are contrary to the purpose of the study. We can conclude that uncharacteristically draining the resources of the device could signal malicious behavior to the victim. Any such flag would limit the effectiveness of the attack. Potential targets



(a) Magnetic field as a function of battery consumption for a Motorola Nexus 6.

(b) Voltage presented as a function of battery consumption for a Motorola Nexus 6 device.

Figure 5.8: Influence of battery state on the generated magnetic field

would uninstall the app and attackers would not get any data. In our study, for reasons of recruitment and retention, we used Android's `SENSOR_DELAY_NORMAL`, which probes the sensor for new readings (approximately) every 200 *ms*.

In practice, our experiments show that the value of the *bias* remains nearly constant until the sensor is re calibrated. Moreover, we are not attempting to reconstruct the original signal. Instead, we either directly use the values obtained or derive features from a sampled version of the signal. We test this by manually generating a dataset where we control the interval between readings. Figure 5.7 shows the change in classification as we increase the latency. We construct different datasets for intervals 500 *ms* apart between 1 and 3 *seconds*. We run both classifiers training on 100 readings for 20 randomly selected users. We observe that down-sampling does not adversely influence our results.

5.5.5.2 Battery Consumption: The Influence of Voltage on Electromagnetic Emissions

Given the results presented in Table 5.2, one legitimate concern might be that, when measuring the internal magnetic field, the state of the battery is driving the measurement. In other words, the battery status might heavily influence the identification results, making them unreliable. To rule this possibility out, we now look at the influence of battery discharge on the magnetic emission of the device.

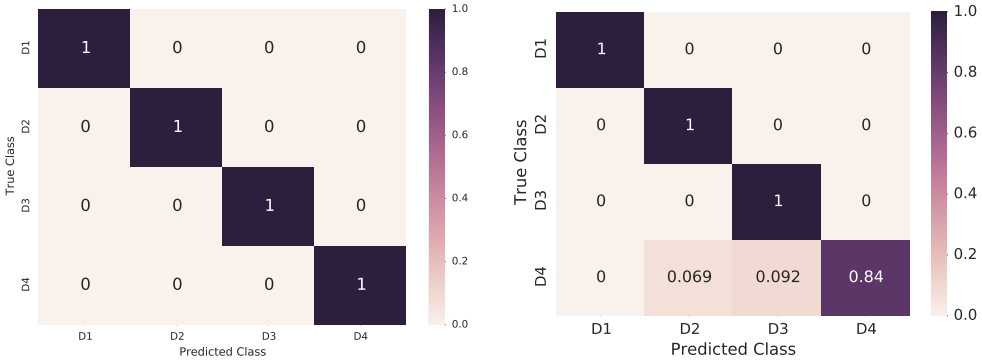
As discussed in Section 5.2, the magnetic field emitted by the phone is a

byproduct of the current flowing through a device at any given time. However, measuring the current is not trivial and prone to error. To test the effect of battery discharge, we performed a controlled experiment on 10 phones where we measured both the battery levels and the radiation emitted by the phone. All of the measurements were collected at the same location and similar circumstances. Each device was set to stream an 8-hour video with the screen brightness and volume at maximum settings. Each experiment took on average 5 hours.

Figures 5.8a and 5.8b show an example of the typical behavior observed. In Figure 5.8a the magnetic field is presented as a function of the battery percentage. Comparing these readings to those presented in Section 5.6.1, we can see that battery depletion serves as a proxy for time and, as evidenced by the trendlines shown in red, the two variables are nearly independent. In contrast, Figure 5.8b shows that the voltage and the battery depletion have a very strong correlation with a coefficient of 0.9994. The physical explanation for this is that, as the battery of the phone is used, the voltage in the battery falls. Once the voltage is not sufficient to sustain the current that is required for the operation of the device, the phone shuts down with the battery depleted. The relationship between voltage, current and resistance is given by Ohm's Law as $V = IR$ where V is the voltage, I is the current, and R the resistance. Both the empirical evidence and the physical explanation attest to the fact that in using magnetic field readings, we are not identifying the battery level of the phone.

5.5.5.3 Unique-in-a-line: Differentiating Between Similar Devices in a Semi-Controlled Environment

The dataset described in Section 6.3 contains readings from devices of the same brand and indeed the same model. However, given that we do not achieve perfect classification, it is important to understand whether the values provided by the API are the same across a model or whether they depend on each device. Moreover, from Section 5.5.5.2 we learned that the magnetic field depends also on the resources in use, and therefore it would be equally as important to determine whether we can distinguish devices executing the same tasks.



(a) Confusion matrix for device identification for co-located devices running simultaneously.

(b) Confusion matrix for the identification of each phone at different locations through the city.

Figure 5.9: Cross validation results for the identification of four different Nexus 6 devices.

To test these scenarios we created a second application that triggers a sequence of events over the span of 45 seconds. The sequence includes playing a video from the phone's internal memory, a controlled-step increase and decrease of the screen's brightness, and the calculation of two Fibonacci numbers. Our aim was to standardize the behavior of the phone when collecting measurements. We tested this on four Nexus6 devices purchased at the same time, running the application simultaneously, at the same location. We ran the application 10 times on each phone and computed a leave-one-session-out cross validation scheme where we train on nine iterations and test on the remaining one (for each phone). We repeat this 10 times, leaving a different session out each time, then aggregate the results.

In this experiment we achieved perfect classification. Every test value was assigned to the correct device. The magnetic field emitted by a phone is not hard-coded into all devices of the same model and while it varies with the level of activity of a phone, different phones will exhibit different values.

5.5.5.4 Impact of Geography: Understanding the Spatial Sensitivity of the Measurements

Another external factor that may influence the results is the location of the device at the time of the measurement. In attempting to identify each device, we may be inadvertently fixating on its environment. One form of environmental noise is the presence of significant sources of radiation (e.g., transmission lines, electric

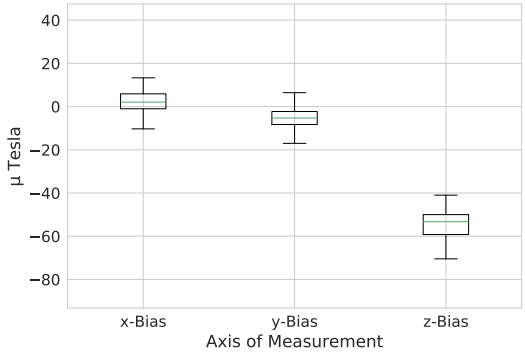


Figure 5.10: Distribution of the values of the *bias* for each axis.

substations, thunderstorms, etc.) close to the measurement device [197, 198].

Magnetic fields that result from natural events are in the range of micro(μ) and mili(m) Tesla (T). For human-generated emissions, the World Health Organization suggests a maximum dose of 2 T with a recommended occupational exposure of no more than 200 mT [199]. Each country has its own regulations but, as the values we find are in the range of μT, both natural and man-made events have the potential of affecting our results. We define geographic independence as the ability to correctly classify a device using readings from a previously unobserved location.

To test for geographic independence we used the application described in Section 5.5.5.3 and collected readings at 10 different locations in a city. The locations include public transport (buses and subways), coffee shops, university laboratories, restaurants, stores, residential buildings, etc. We aggregate these results in a leave-a-place-out cross validation scheme where we used 9 locations in the training set and the remaining location for testing. This is repeated until every location has been used for testing. With an accuracy of 96%, the results in Figure 5.9b show that the internal bias of the phones is not controlled by the environment.

5.5.5.5 Robustness Over Time: Exploring the Stability of the Value of the magnetic field over 3 months.

Finally, we provide results to support the claim for the robustness of the method over time. We investigate the temporal validity of the signature of a device, i.e., the time interval during which the model can be used for the classification task after its

initial training.

Using the application described in Section 5.5.5.3 we present measurements from a single device at intervals of 12 hours over a period of 90 days. The boxplots shown in Figure 5.10 show that while the values are not identical, the variance within each axis is low. With these results we are confident that the signature of a device will remain constant over the period of one week.

5.5.6 Limitations and Countermeasures

The primary limitation of this attack is that the *bias* can only be accessed from within each device. On one hand, any malicious app can access this data, as there are no permissions required for the magnetometer; but on the other, as we will show in Section 5.6, this information cannot be collected from a secondary device. In our opinion, this is not sufficient to discount the validity of this attack primarily because when possible, identification can be accomplished without the knowledge and consent of the user.

We have shown that the internal magnetic field obtained from the Android sensor interface serves as an adequate proxy for identity. It provides over 90% accuracy in identification without alerting the user to any suspicious behavior. In case this becomes a widespread attack, and therefore a real hindrance for The Open Handset Alliance[1] it could be counteracted by protecting the sensor via Android's permission platform. Having the sensor available but under the responsibility of the user is a good compromise and it is in agreement with the precedent set by the platform [200, 201, 202]. The decision to introduce permissions for thes sensor in the Android API an attack based on similar principals but carried out *externally* would still be possible.

In the next section we will discuss how, by placing a secondary device (i.e., an external measurement instrument) within range of the target, identification remains feasible even when internal access to the sensor is not possible.

[1]The Open Handset Alliance (OHA) is the group of device manufacturers, software developers, and telecommunications providers responsible for promoting the Android operating system.

Table 5.3: Description of relevant data fields.

Feature	Description
Maximum/Minimum	(time) the maximum/minimum amplitude of the signal.
Mean/Median	(time) Measures of the central tendencies of the signal.
Variance	(time) The square of the difference between each element in the signal and the mean.
Mean/Median Absolute Deviation	(time) The average of the deviation of each element in the signal to the mean/median.
Skewness/Kurtosis	(time) A measure of the symmetry of the signal and its outlying values.
Average Peak Duration	(time) The average length of the intervals corresponding to a local maxima in the signal.
Power Spectral Density	(frequency) Frequency corresponding to the maximum value of the power spectral density.
Spectral Centroid	(frequency) A measure of the center of mass of a signal calculated as the weighted average of the frequencies present.
Frequency Maximum	(frequency) The frequency corresponding to the spectral maximum.
Noise Power	(frequency) Derived from the power spectral density, it is one descriptor of the noise in the signal.

5.6 Physical Proximity Attack: Identifying One Device from Another

Successful attacks are contingent on access to the target system. In many cases, network connectivity is enough to gain access, execute the exploit, and retrieve the response. It also allows for an increase in the number of potential targets and offers some measure of protection for the attacker. However, the lack of physical presence also translates into diminished control. The results presented in Section 5.5.5 place the hardware and software constraints on the targets and require users to install an application (or another form of malware) on a phone that has the appropriate sensor. Once the application is running successfully, an attacker can collect information and identify devices simultaneously. However, the novelty of our attack is not that we identify devices through sensor readings (as if to imply that the sensors are the source of the signature) instead, what we propose is that each device emits magnetic radiation that can be collected by a magnetometer and used to generate a unique signature. In this second attack, we shift hardware and software requirements to the attacker and accept physical proximity as the limiting constraint. The implication is that now any electronic device is identifiable; all that is necessary is for the attacker to have a standard phone with a magnetometer and the software to conduct the attack. In this section we present the identification results for the attack illustrated in Figure 5.3b.

5.6.1 Experiment Setup and Data Collection

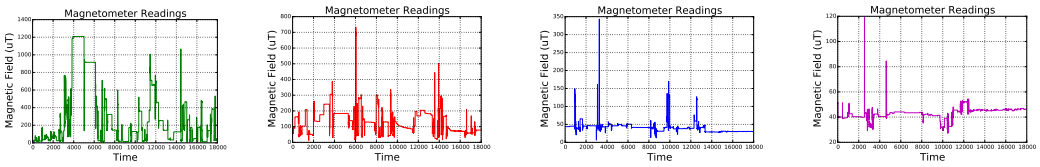
In this scenario we have an adversary (d_A) who collects readings from a victim (d_V), the target of the identification attack. This second dataset was all collected in the

same location over a period of two weeks from a group of voluntary participants selected through convenience sampling. Each participant interacted with their own device and the collection device was the same for all measurements. The parameters of the collection were more restrictive than those in the first attack, for instance, for the duration of the collection, participants were (mostly) stationary and their devices were continuously in use. Each participant was briefed at the beginning of the session with the procedure and objectives. The participants were told that the content of their data was not the object of the study, only the emissions of their device. They were each told that the distance between the two devices should never exceed a palm width (approximately 20 cm) and that d_A should remain immobile in a flat surface (i.e., resting on a table). After giving each of them some time to answer questions, they received no further input from the researcher. In practical terms, this means that there is a fair expectation of noise in the collected readings. The participants had no restrictions on the applications being used. They were told that they should maintain continuous interaction (i.e., their task was to use their phone without allowing it to enter standby mode). Participants responded in different ways, some watched videos, others listened to music, responded to emails, used chat applications, played games, read e-books, etc. We assume that the data collected reflects the range of resource consumption and behavior that would be normally associated with each device. All measurements were completed with the same instance of d_A and the settings of the device were adjusted to the highest resolution available (i.e., a sampling rate of 10 Hz).

Each of the participants used their device continuously providing us data for a period of 30 min. Overall we collected data from 30 participants that resulted in 4 different mobile platforms (i.e., Android, Apple, Blackberry, Windows), 7 brands, and 22 distinct models.

5.6.2 Feature Generation

We have already shown that the internal magnetic field can be used for identification. The next task is to determine whether the unique internal field can be extracted from the environmental magnetic field collected from a secondary device. Differ-



(a) iPhone SE. **(b)** iPhone SE. **(c)** Galaxy S5. **(d)** Galaxy Core Prime.

Figure 5.11: Magnetic field associated with four smartphones.

ent from the analysis conducted in Section 5.5.5 the measurements collected by d_A do not consist solely of the magnetic field of d_V. Instead, the values represent the magnetic field around d_A, which *includes* the emissions of d_V. A second major difference is that while in Section 5.5.5 the magnetic field was measured in three orthogonal axes of fixed position and orientation with respect to the device. When we look at the environmental magnetic field, the frame of reference is variable and the axes with respect to the Earth are constantly changing. Hence, for a valid comparison, we transform the three-dimensional readings into a single value through the magnitude (norm) of the readings which is calculated as follows:

$$\|\mathbf{m}\| = \sqrt{\sum_{i=1}^{n} m_i^2} \tag{5.6}$$

where $\mathbf{m}$ is the vector containing the (x, y, z) measurements of each reading.

Extracting a specific signature from the environmental magnetic field is a challenging problem. Any environmental reading will contain at least one source of white noise, our planet's naturally occurring magnetic field. If the measurement is recorded in an urban setting, then an additional source of noise are the ferromagnetic materials in the vicinity of the device primarily those used for a building's structural integrity [203]. Regardless of the source, the noise is variable and in the order of μT well within our range of interest. However, given the universality of the noise any identification attack must, in order to be successful, be able to overcome these challenges and find the signature of a device.

Figures 5.11a, 5.11b, 5.11c, and 5.11d show the signal as collected from four instances of d_V. Each graph corresponds to a different device, the first two represent the same model (an iPhone SE), and the last two are both Samsung devices, one

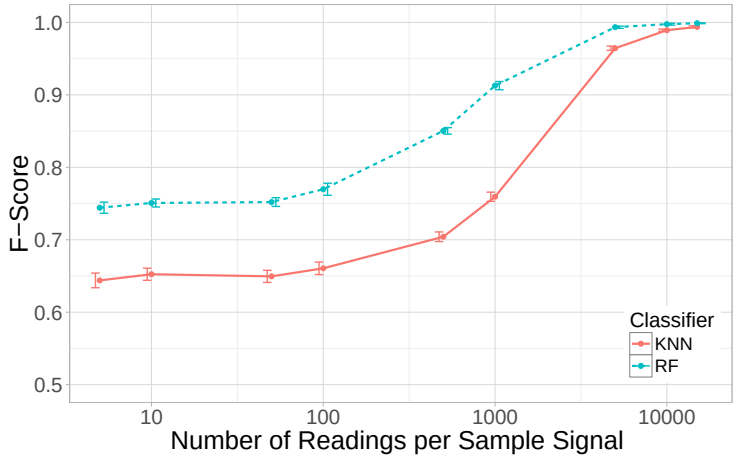


Figure 5.12: Classification F-Score for increasing duration of training signal.

low-range (Galaxy Core Prime) and the other relatively high-range (Galaxy S5). As evidenced by the figure, in the second dataset, we place no constraints on the manufacturing company or the operating system. In addition to Android, we collected measurements from Apple, Blackberry and Microsoft devices as well as collecting measurements for different devices of the same model and manufacturer. The results presented later in this chapter seem to indicate that this method is applicable to all mobile devices. We would also like to underline another difference: whereas in the previous attack we consider each reading as a discrete signal, now we interpret these observations as time series. Indeed, in the malware attack, we had direct access to the phone's magnetic field. This is no longer the case. Now, we must extract the phone's field from the environmental measurements and, in order to achieve this, we include in our analysis the relationship between adjacent observations.

We extract the features described in Table 5.3. Following the recommendation in [204], we rank the features according to the accuracy exhibited in each classifier and build the models in accordance with the rank. Each 30 minute signal is divided in two: the first 20 minutes are used for training and the remaining 10 for testing.

5.6.3 Results

The signal of interest is the magnitude of the magnetic field as measured in the environment of d_A. For each device we considered a signal sampled at 10 Hz comprised of 18,000 measurements (i.e., 10 measures per second for 30 minutes). For each

device, we sampled the signal for different time-spans and computed the features described in Section 5.6.2.

Figure 5.12 summarizes the main results. We ranked each of the features in Table 5.3 and used the three best performing descriptors for the final models. Using the signal's minimum, maximum, and median values and a training signal duration of 16.7 *min* we achieve 99.9% accuracy with RF and 99.0% with KNN. Overall, the attack is viable and results in accurate identification. If however, we reduce the period for data collection, we observe that training time is approximately halved to 8.3 *min* and, at the same time, we are still able to achieve 99.4% for RF and 96.7% for KNN. It is worth underlining that the random baseline for classification is around 3% for both methods.

5.6.4 Limitations and Countermeasures

The primary limitation of this attack is the proximity required between d_A and d_V. This limits the number of devices that can be identified at any time and it puts physical demands on the attacker that are simply not there in the malware attack. This however, is a limitation of our resources. For this experiments we were using Hall Effect Sensors to measure magnetic field. A more sensitive sensor, a fluxgate magnetometer for example, would undoubtedly increase the range of the measurement. However, the resources available for this project did not allow us to empirically test the range of the fluxgate sensor. With respect to countermeasures, legislation already regulates the emissions from a device to prevent, among other things, interference across devices. Manufacturers already incorporate shielding into their electronics' designs.

While the magnetic field is a single physical (vectorial) quantity measured by one sensor, it is actually the result of an intricate interaction of components, material degradation, and user behavior just to name a few. An effective countermeasure for this attack would be a cover that offers full shielding for the device. However, if the price of completely shielding a phone undermines any of the users' expectations of mobile devices (e.g., lightweight, multi-purpose, connected, and efficient), the consumers might prefer to risk the attack rather than to incur in the penalty of the

protection.

5.7 Discussion

The techniques presented in this chapter can also be used to forge the identity of devices. Indeed if an attacker has access to the device as in the first type of attack (malware), it would be sufficient to record and reproduce this magnetic field (assuming that the sensor API would be intercepted). In the proximity attack, it is harder to recreate the signature of a device as the model is not build directly on the magnetic field but rather on a features set derived from an external measurement of that signal. We believe this is an interesting area to be explored in the future.

Another interesting aspect is that the magnetic field is an inherent characteristic of all electronic devices. The privacy challenges it presents arise from the physical aspects of the technology and as such finding effective countermeasures that nullify the attack is a hard problem. For this reason, we believe that this work also poses potentially interesting challenges for the materials and signal analysis communities in general. Continuing this same line of research, future work involves taking this in-the-wild empirical study and testing the actual boundaries of the assumptions (i.e., noise, temperature, signal strength, etc.) under laboratory conditions so as to determine the optimal parameters of the attack and any further applications of it.

5.8 Summary

We have presented the feasibility of identifying devices from the magnetic field they emit. We have considered two types of attack, one internal, based on the magnetic field both from within a device, as reported by the Android API, and one external, based on the readings collected from a secondary device in close proximity to it.

With respect to the internal attack, we have discussed the classification results using two supervised learning algorithms, KNN and RF, over two different datasets and we have shown that we are able to identify one device in a group of 175 and approximately 1.6 minutes of data with a precision of 98.9% This holds an advantage of approximately 165 times over the precision exhibited by the random classification baseline where the probability of success is around 0.6%. Even if

this internal attack is limited to those devices that have a magnetometer and which have the data collection application installed, the lack of permissions associated to collecting readings from the sensor makes this attack invisible to users.

As far as the external (proximity) attack is concerned, we have shown that from a distance of approximately 20 *cm* a Hall Effect sensor is sufficient to collect the signal from the victim and generate the fingerprint. Using three features in the temporal domain, we have achieved a maximum accuracy of 99.9% by using a 16 minute signal for training.

Chapter 6

Location Inference from Magnetic Field Data

Mobile phones are equipped with a variety of sensors used by applications to obtain a user's contextual information (*e.g.*, location, humidity, acceleration, and network connectivity) to support a variety of applications and services [205]. However, the sensors embedded in smartphones have also unintentionally become the source of information leaks that might adversely impact the privacy of their owners. Indeed, information extracted through sensors can be used to identify users or devices [111, 32, 206], and infer their behavioral patterns, interests, personal preferences [207], and even their health condition [208]. The fact that information can automatically be extracted by means of passive sensors is generally perceived negatively by users [209, 210, 211, 212].

In order to mitigate privacy risks, mobile operating systems have adopted a permission-based paradigm where applications must request access to any of the protected resources on the phone, including camera, microphone, location, and contact list. Each permission is flagged as sensitive depending on the invasive nature of the resource and the importance of the data it might reveal; therefore, the permission to answer phone calls is more restricted as compared to the permission that accesses the currently connected WiFi access point. Users can choose to disable system-wide access to a certain type of information, or control it at the application level where access to some information can be revoked [213, 214, 215].

Location is one of the information types that is protected by such permission systems. Location data has been shown to be sensitive for users as it can be used to track and profile individuals (*e.g.*, for advertising purposes). In particular, it can be used to infer a user's identity [216, 217]; it can reveal a user's significant locations and points-of-interests (*e.g.*, home location, work location, morning coffee shop) along with their transportation routine and use them to predict trajectory and future locations [218]; and, finally, it can be used to detect the general behavior of a single user and group users based on the similarity across their interests [219].

The restrictions placed on location data have led to the development of alternative methods that aim to derive location from sensors that are not protected under permission systems. Currently, any Android and iOS application can access the gyroscope, accelerometer, and magnetometer sensors without requiring the user's permission. These methods generally combine previously acquired environmental information (*e.g.*, produced by surveying a location) with real-time sensor readings to ascertain the user's current position [220]. In particular, Wang *et al.* combine accelerometer and gyroscope measurements in order to localize users [122] and Chen *et al.* combine WiFi signal strength with FM radio signals to determine a users' location [221].

In this chapter, we show how it is possible leverage the magnetometer, which is not protectedy by any permission, in order to infer the location-type that corresponds to the current position of a user. Effectively, we present a *zero-permission localization attack* based exclusively on sensor data which exploits the readings captured by the magnetometer, present in most smartphones today, to infer a type of location from their magnetic signature. We are not the first to address the challenge of sensor-based localization. However, where previous studies focus on mapping locations to pinpoint position, we present a method that reduces the need for surveying the target locations (*i.e.*, locations where localization will take place) at the cost of reduced granularity.

We present four different methods for time-series analysis and classification: (i) full signal matching, (ii) statistical descriptors, (iii) automated feature extraction,

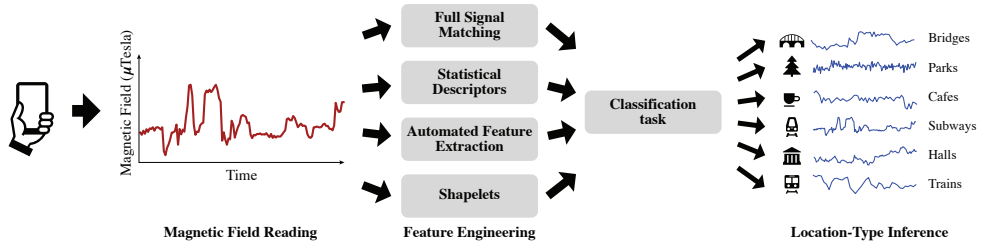


Figure 6.1: Overview of the zero-permission attack that leverages the magnetometer of mobile devices to capture magnetic field readings and then infer the location-type of the place where the user is currently situated from the magnetic signature of that location.

and (iv) shapelet analysis both in the time and frequency domains. More specifically, we compare the two traditional methods of full signal pattern matching and statistical features with two novel methods, namely automated feature extraction using convolutional neural networks through One Shot Learning [147] and shapelets, a method that extracts repeated patterns observed in the form (or the outline) of a time series [137]. An overview of the method is presented in Figure 6.1.

In order to evaluate our approach we collect a labeled dataset, used as ground truth, by conducting an in-the-wild measurement study and sampling the magnetic field at different locations over a metropolitan area. We collect readings from off-the-shelf smartphones at ten different types of locations: long-distance train stations, urban train stations (*i.e.*, subways), parks, bridges, coffee shops, halls, laundromats, bus stops, parking lots, and gyms. The places were chosen as alternatives for three groups: indoor environments, outdoor environments, and environments that contain clearly distinguishable events (*e.g.*, trains passing by). For each of the 10 location-types, we collect 10 minutes of magnetic field readings sampled at 1 Hz from five different phones at 11 locations. The entire dataset therefore consists of a total of 91 hours of readings. We evaluate the prediction performance of our four methods by first identifying a location-type from the magnetic readings of an unknown location and second, by identifying a location-type from the magnetic readings of an unknown device.

We propose three major contributions as the outcome of this work. First, we collected over 91 hours of labeled magnetic field readings from various locations

across a major metropolitan area. We show that location-types can be inferred from magnetic field readings available without any system permissions across a wide range of smartphones. Then, we compare the performance of four methods for time-series classification in identifying location-types, introducing two novel methods based on automated feature extraction using convolutional neural networks through One Shot Learning and a classification method based on shapelets. We perform an in-depth analysis of the choice of the values of the parameters for each method, providing a methodology for their selection. Finally, through a proof-of-concept implementation and an in-the-wild study across different locations, we demonstrate the feasibility of the location identification attack.

6.1 Preliminaries and Motivation

In this section we introduce the key concepts that will be in use throughout the chapter and present the main motivation for using the magnetometer in localization.

6.1.1 Key Concepts

In writing this work, we take ownership of some words and create new concepts. We define them as follows:

Location-Type. As defined in [115], localization is the problem of ascertaining the position of an agent relative to a map. In this work, we address the problem of localization from a novel perspective, where our primary interest is to understand the environmental similarities between places. Location-type is the name we assign to the top level of a two-tier hierarchy (the lower level being the distinct places). Locations are grouped based on common environmental characteristics including, building structure and materials, human movement patterns, and events (defined later in this section).

Magnetic Field. The magnetic field is the combination of geomagnetic and electromagnetic phenomena. Geomagnetic fields describe the naturally occurring (magnetic) field emitted by the planet's core and the local variations caused by ferromagnetic materials such as iron, cobalt, or nickel [128, 125]. As an example, the steel structure of a building and the movement of metal objects (*e.g.*, a car or a train) will

distort the planet's field by generating a non-electronic signal. On the other hand, electromagnetism results from the flow of an electric current. Electromagnetic fields can be measured from electrical power sources or electronic appliances and hand-held devices. The magnetometer (or digital compass) present in a smartphone is the sensor that measures the magnetic field in the vicinity of that phone. As such, the magnetic field measurement represents a single combined reading that unifies *geo* and *electro* magnetism. The sensor reports each measurement in 3D, *i.e.*, once completed, the observation at each location consists of three time series (one per dimension) of the same lenght and matching timestamps. In our analysis, however, we collapse all dimensions into a single time series using the norm of the vector with element i defined as (x_i, y_i, z_i) where x, y, and z are the time series corresponding to the three spatial dimensions. This reduction is necessary when collecting data in a real-world settings and the motion of the sensor cannot be controlled. In this work, the orientation of the phone's axes is continuously changing with respect to Earth [125]. Taking the norm of the vector that describes each measurement allows us to compare observations.

Time-Series. A time-series is an ordered set of values (*i.e.*, the magnetic field) sampled at a fixed interval that, together, form a single observation or reading [222]. The length of the time series is equal to the number of values available in that observation. Subsets or samples of a time-series with more than one element (which will be associated to shapelets later) are time-series observations in their own right, with the distinction that the length of the subset must be no greater than the length of the originating observation(s).

Events. We define events as the dynamic extension of landmarks for time-series data. While a landmark represents a set of structural characteristics of an environment (*e.g.*, the corner of a building, the presences of a fountain, the electrical wiring of a room), an event represents a transient occurrence (*e.g.*, the movement of a train or car, riding on an escalator, or walking by a person). Unlike landmarks, which are unique to a place, events can be identified in places described by a location-type. In our analysis, we make the assumption that locations that belong to the same type

are characterized by similar events, which will be associated to specific *shapes* in the time-series of the magnetic field.

6.1.2 Motivation

Information about a user's physical location provides private insights about the users themselves. Applications installed on the users' smartphones can access location data provided by location services using the GPS receiver and the cellular network through permission-based controls offered by the operating system. It is left to the user's discretion to grant location access (and revoke it) to any application that requests it [213, 214, 215].

On the other hand, side-channel, zero-permission attacks have been exploited to infer users' location using alternate approaches without requesting access to the corresponding GPS and cellular permissions. They rely on constructing, for each target location, a map of the environmental characteristics measured by the sensors embedded in the smartphone. These sensors, whose access does not require any specific permissions, include primarily the magnetometer, the accelerometer and the gyroscope [223, 224, 133, 225, 226]. In practice, given a target location (*e.g.*, an area or a building), a map can be built by recording sets of sensor measurements (*e.g.*, acceleration from the accelerometer and/or magnetic field from the magnetometer) that are associated to precise coordinates within the location being surveyed. The literature shows that magnetic field readings recorded at target locations are different depending on the coordinates at which they have been taken [127]. This implies that it is possible to build maps from the readings collected from mobile sensors. Any attacker can then leverage this information to track users within the confines of the map; that is, at any point they can determine the exact coordinates of the user within that area or building. They achieve this by comparing new readings against values in a map. While essential for localization, maps are limited in their application.

Collecting and maintaining accurate maps for every target location is difficult and requires high quality survey data. In particular, it requires the attacker to select a target location and to survey its environmental characteristics assigning each

measurement to a location on a grid. The solution we propose is to group distinct locations into *location-types* and extract common *events* that fully describe each location-type. This provides a cost-efficient technique to determine a user's location eliminating the need to survey all possible sites for given location-type. As such, an attacker (or alternatively, a service) would be able to derive coarse-grained location data without collecting information about any specific place.

Contextually, this work contributes towards the development of a new technology. As opposed to the GPS module, the magnetometer is a low-power sensor and accessing readings from a magnetometer does not require any notification to users (or permissions) in either Android or iOS. Moreover, once the information is processed it can be used to protect the privacy of users: while the GPS provides detailed location information (accurate to centimeters of the true position), a generic localization method based on magnetic field could potentially be used to verify that a user is within the premises of a certain type of place or in a specific environment without disclosing exact information. Both marketing and verification applications could obtain the information they need without being invasive with regards to user location.

6.2 Methodology

In this section we detail the different approaches that we use in our methodology to predict location-types from magnetic field readings collected from off-the-shelf smartphones. In particular, our methodology addresses two problems: (1) classification from time-series; and (2) deriving location-types from magnetic field readings.

6.2.1 Techniques for Feature Extraction

Measurements over time provide a more complete view of reality, they allow us to uncover relationships between consecutive measurements. We propose that collecting and processing longitudinal magnetic field readings will prove to be useful in the task of localization. However, time-series classification, clustering, and prediction are identified in the literature as hard problems. Challenges in time-series analysis include: depending on the length of the time series, the resource consumption

(*i.e.*, computational complexity) of the method; the definition of a similarity metric in data that is noisy and prone to outlying values and shifts; and finally, the high-dimensionality of each observation. An in-depth discussion of these challenges can be found in [138, 222, 227, 228]. In this work we compare four methods for time-series analysis: (i) full signal matching; (ii) statistical descriptors; (iii) automated feature extraction through the use of neural networks; and (iv) classification through the extraction of shapelets, in both time and frequency domains. The details of these methods are discussed in the remainder of this section.

6.2.1.1 Full Signal Matching

Pattern matching is a method for time-series classification where each test sample is matched against all labelled training samples. The test sample is then assigned to the label of the closest training sample (*i.e.*, with the shortest distance to it). Consequently, this approach requires a pairwise distance calculation from every signal in the testing set to every signal in the training set, which might become intractable as the number of signals grows. In this analysis, we used four different distance measures for performing classification task: Dynamic Time Warping (DTW), Euclidean Distance, Cosine Distance, and Bhattacharyya Distance [229]. We selected these distance measures as they extract different information from the corresponding input signals: DTW involves both signals and tries to find the best fit between them in a cross-correlation computation, the Euclidean Distance measures the magnitude of the separation between the sampled signals, the Cosine Distance measures the angular separation between the two vectors being compared, and finally, the Bhattacharyya Distance captures the divergence between the probability distributions of each of the signals.

One of the key limitations of this approach is that it is sensitive to small variations between the two signals, including differences in the mean for matching signals, which could negatively impact the classification performance. In contrast, we aim at identifying similarities between observations collected in noisy environments. Therefore, in the context of our application, we are expecting this method to have poor prediction accuracy. Nevertheless, we use it as the baseline metric for

our performance evaluation.

6.2.1.2 Statistical Descriptors

The second approach we consider consists in the extraction of representative statistical descriptors (i.e., features) from the time-series. The extracted features are used to represent key characteristics of the data. It is worth noting that the classification performance using these features depends on the degree to which the sampled data represents the population of interest. Good statistical models require a sufficient sample size.

We selected eight commonly used statistical features: the median, the amplitude, the energy of the signal, the magnitude and frequency of the natural frequency, the spectral centroid, and the magnitude and frequency of maximum power. Note that we sub-sample the input data and compute these features in both the time and frequency domain. It is worth noting that other alternative features might be extracted.

Also for ensuring the replicability of the experiments, we used the open source versions of three state-of-the-art classification algorithms provided by the scikit-learn library [162] to construct our prediction models: k-Nearest Neighbors (kNN) [144], Random Forests (RF) [143], Extreme Gradient Boost (XGB) [230]. kNN is one of the most popular and effective unsupervised learning algorithms, whereas RF and XGB are widely used for their interpretability and performance, respectively.

6.2.1.3 Automated Feature Extraction

Algorithms based on neural networks remove the burden of manual extraction of features in order to train prediction models [231]. In particular, in this study, we use Siamese Networks [147] — a specific type of neural network architecture that aims to learn to differentiate between two inputs rather than classifying the inputs into given classes. This network architecture is comprised of two identical neural networks each taking one of the two inputs and the last layers of the two networks are then fed to a contrastive loss function to compute the similarity between the inputs [232]. Two neural networks are identical if they have the same configuration

in terms of parameters and weights. Since the weights across the two networks are shared, there are fewer parameters to train, which in turn means that less amount of training data is required. This also reduces the chance of overfitting. Since our input is in the form of a time series, we use 1-D convolutional layers to extract patterns from the data, which are passed from the feed-forward layers to obtain the results for the output layer that is used for computing similarity. More specifically, our network consists of C convolutional layers (CNN layers) and a feed-forward layer that maps the features (extracted by the CNN layers) to the output layer with 100 nodes (*i.e.*, the number of final features to be extracted). We considered values of $C \in [2,3,4]$. We do not consider higher values of C given the size of the training set under consideration.

6.2.1.4 Shapelets (in Time Domain)

Shapelets are small local patterns in a time-series that are highly predictive of a class [136, 138]. In the context of our work, each shapelet corresponds to an *event* that is found in at least 90% of the observations belonging to a certain *location-type* (with 11 distinct locations and 5 devices we require an event to be present in 50 observations before a candidate is considered a shapelet). Our assumption is that for each class there exists at least one shapelet contained in all observations of that class, and we consider shapelets of different sizes where longer shapelets are more valuable in terms of class separation. In Figure 6.2, we introduce the visual representation of an event. These shapelets correspond to one example per class with a length of 80 seconds.

One benefit of adopting this technique is that the analysis of candidate shapelets is independent for each class. This means that the length of a shapelet is also a feature we are considering in the input and that the maximum length is determined on a per class basis (*i.e.*, one way to distinguish between classes that may have similar events is the duration of that event). As an example, the passing of a long-distance train and a subway are similar events but long-distance trains are longer which might correspond to longer shapelets.

The methodological contribution of this work incorporates shapelets into prob-

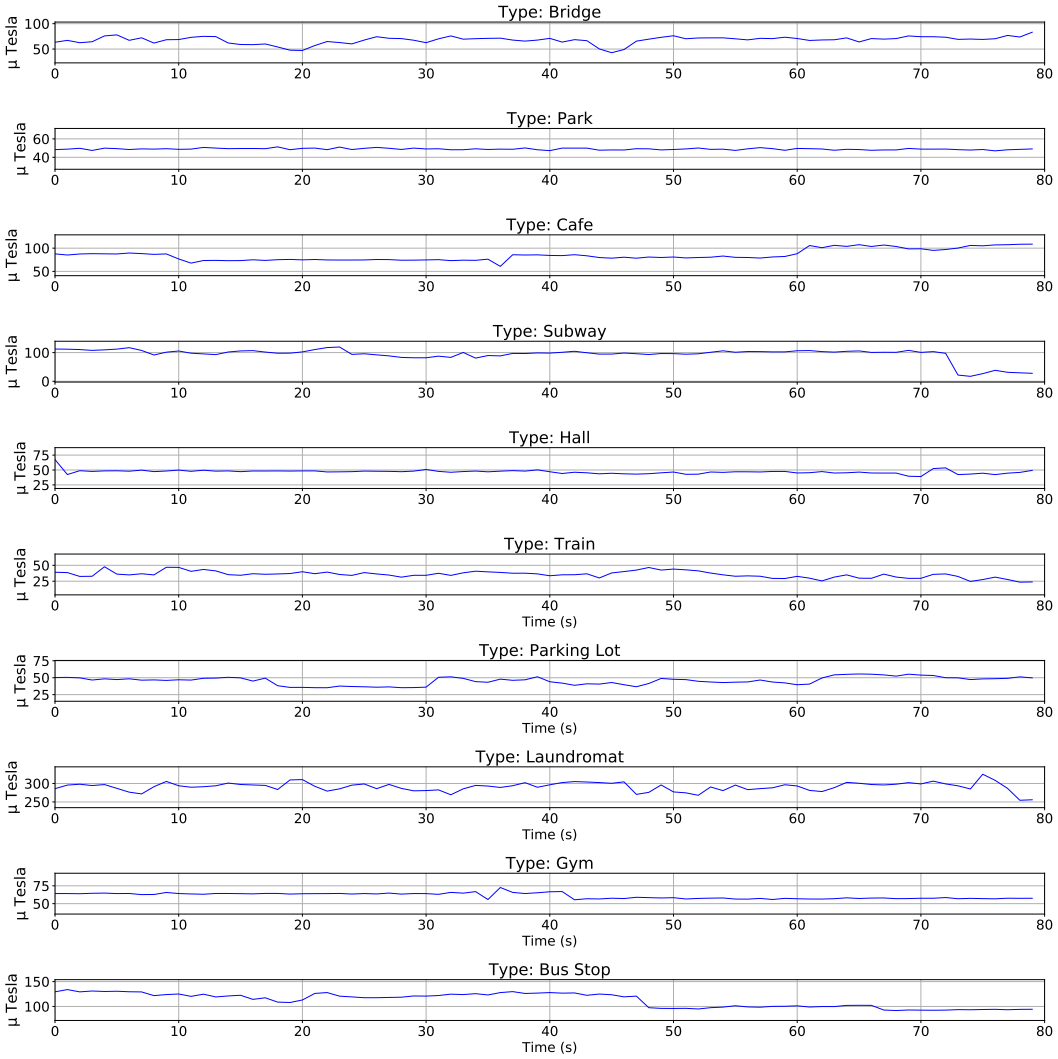


Figure 6.2: The visual representation of a *shapelet*. In the figure, each sequence corresponds to an 80 second time-series present in at least 90% of the observations for each location-type.

abilistic classification algorithms. Our method takes labeled time series observations and returns the location-type of an unlabeled observation. The algorithm is divided in three steps: first, it extracts shapelets from the training observations (to build a shapelet dictionary); then, uses the training data to describe each class in terms of the discovered shapelets; and finally, it classifies test observations into the known classes. Conceptually, extracting the representative shapelets for each class is straightforward: for each sub-sequence of any duration and beginning at any starting point, we identify the sub-sequence(s) present in distinct locations that belong to that class.

Given a certain application domain, the first step is to select a range of lengths for which a shapelet has a semantic meaning (and a step size to traverse this range). In our study, we select shapelet lengths ranging from 20 seconds to 2 minutes in steps of 20 seconds. Then, for each length in the range, we create a bag that contains all possible sub-sequences from all the training observations for a single class. Secondly, we cluster all the elements in the bag. As there can be a variable number of shapelets (*i.e.*, events) in the different bags, we use hierarchical clustering to create links between the samples. Then, we inspect each cluster to determine the number of distinct locations present in that cluster. We only accept clusters where the minimum number of locations is met. Finally, for each of the valid clusters, we compute the medoid of the elements in that cluster. The signal that corresponds to the medoid becomes the shapelet added to the dictionary. We repeat this process for all lengths in the range.

Having extracted all representative shapelets (*i.e.*, for all lengths and classes), the second part of the algorithm consists in characterizing each class with respect to the shapelet dictionary. However, since the shapelets are time-series, we cannot use them as raw inputs for training the models. Furthermore, since their size and number may vary from one class to another, this causes a symmetry problem between the training and testing sets. This characterization step addresses both these problems by creating equal length samples from both training and testing data and computing the distance from each sample to each of the shapelets. In the training set, the distances are associated with the label of the class to which the sample belongs, whereas in the testing set this becomes the prediction variable. Each distance in the newly created matrix represents minimum value between a windowed segment in the sample and each shapelet.

The last step of the algorithm is the classification of unknown signals. Similar to the classification with the statistical features, we use *k*NN, RF, and XGB for the prediction of a location-type.

6.2.1.5 Shapelets (in Frequency Domain)

It is common practice in signal processing to study signals in the frequency domain [233]. Following the same procedure described in Section 6.2.1.4, we generate a bag of shapelets (with all the possible sub-signals from the training set) and transform each one to the frequency spectra before computing the correlation between all signals. Doing the piecewise transformation before extracting the shapelets might result in better classification if the sources of the signal are monotonic. We integrate the frequency analysis method, known as the Generalized Correlation Coefficient (GCC) [234] with the shapelet-based classifier and compare the accuracy of both methods.

We follow the same clustering procedure and apply hierarchical clustering to the distance matrix generated using GCC. The shapelet dictionary is extracted from the inspection of each cluster. During classification, we use GCC to compute the correspondence between each of the elements in the shapelet dictionary and the signal used as input. This process is repeated (separately) for the observations in the test set and evaluated using the same algorithms.

6.2.2 Criteria for Assessing Prediction Models

In this section we discuss the three approaches we used for evaluating the performance of the different methods to predict location-types.

6.2.2.1 All Places, All Devices

This evaluation approach aims at answering the question as to whether the models are capable to correctly classify a known location from a known device. In the data, we take the 10-minute readings collected from each device and divide it into ten 1-minute segments. We then proceed to a 70/30 cross evaluation by randomly selecting 3 segments for the testing set and the remaining 7 for the training set, i.e., there is no overlapping between training and test sets. We repeat this process and average over 30 iterations using the analysis methods detailed in Section 6.2.1. Under this evaluation criterion, we assume that we have (previous) data from the device being tested at the location from which the new observation is tested.

6.2.2.2 Leave-a-Place-Out

This evaluation approach aims at examining the location-type prediction performance from magnetic field readings belonging to locations that have not been seen by the model but belonging to the existing set of location-types. For instance, in this approach, we aim to classify a magnetic field reading that belongs to a train station but in a station different to the ones present in the training set. With this approach, we want to investigate whether the need to map each location can be eliminated and, thus, the location-type can be determined based on its characteristics. In order to carry out this type of evaluation, we remove from each location-type a single distinct location at a time. We use the removed location as the test set and all the readings from the remaining locations in the training set. This process is repeated until each location is assigned once to the testing set.

6.2.2.3 Leave-a-Device-Out

In this final evaluation approach, we are interested in determining whether the models can predict the location-type from the reading of a new device (*i.e.*, an unknown user). To carry out this evaluation, we select all readings from one device and use them in the testing set. The training set is then composed of all readings of the remaining devices. We use cross validation to evaluate the performance of the method.

6.3 Dataset

The data collection process was carried out in three stages: (*i*) application development, (*ii*) data collection, and finally (*iii*) analysis and classification.

6.3.1 Data Collection

Our hypothesis in this project is that similar locations will be characterized by similar magnetic fields. We are attempting to determine whether a fingerprint exists for a particular location-type and use it to predict the class of unknown locations. This task requires on-site data collection at a number of locations across the city. The measurement devices must therefore be mobile and, if possible, able to be crowdsourced. For their sensors and ubiquity, we use cell phones to collect magnetic field

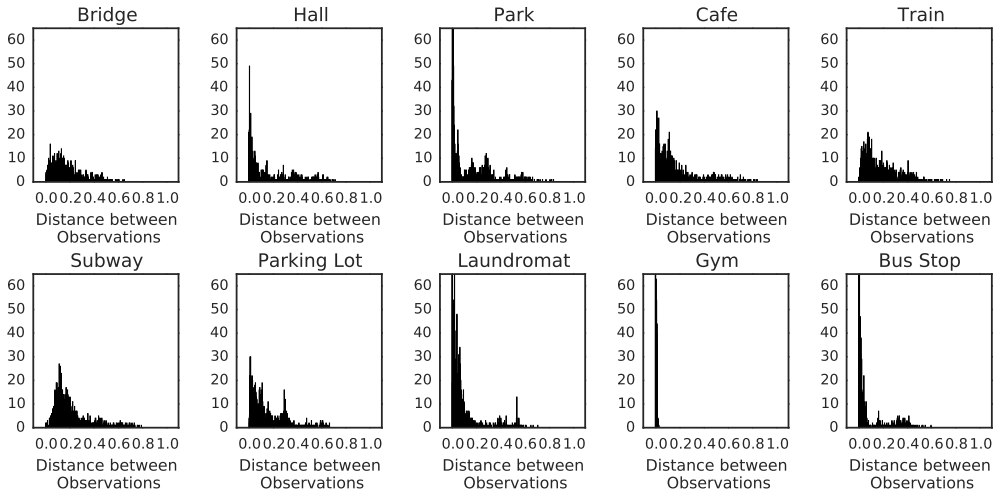


Figure 6.3: Histogram of distances of within-class observations.

readings at each location.

We developed two applications, for Android and iPhone, and used 14 devices to collect measurements (6 Android phones, 8 iPhones). Both applications collect the three dimensional magnetic field readings along with location, linear acceleration, user activity (for Android phones), and phone identifiers. In the evaluation (*i.e.*, testing) of each model we only considered the five devices for which we had data at all locations.

6.3.2 Description of the Dataset

The final dataset that we collected consists of 550 magnetic-field readings: we collected the readings from 5 off-the-shelf smartphones at 110 different locations (10 location-types and 11 locations per type allowing us to train on at least 10 locations for each location-type). Each magnetic reading contains at least 10 minutes of data sampled at 1 Hz. In our case, we are not interested in reconstructing the function and sampling at this rate allows us to record changes without overtaxing either the battery nor the other resources of the phone. We designed a collection methodology such that the ways the data that was collected mimic realistic everyday life situations. The volunteers were instructed to hold one phone at a time while walking around the location as they would normally do for 10 minutes. The samples (even those for the same locations) were taken at different times of the day over a period

of three months by different volunteers in an attempt to get measurements to capture the inherent characteristics of a location rather than any bias introduced by the volunteers.

The types of places were chosen in order to have different examples for one of three groups. The first group is composed of indoor environments; the categories of places in this group are halls, gyms, laundromats, and coffee shops. We define halls as internal spaces with high ceilings with hollow central spaces and columns separating the space. In the case of coffee shops, we observe that for chain establishments (*i.e.*, franchised locations), different locations have similar layouts, appliances, building structures, and users conform to similar behaviors. Therefore, we selected one chain of coffee shops in the city with at least 11 distinct locations. Laundromats and gyms are similar in the sense that there would be a high concentration of machines (electric and otherwise) performing periodic tasks (*e.g.*, a treadmill, a rowing machine, a washing machine). Thus, we made measurements to reflect these machines and the general ambience of each location.

The second group is comprised of outdoor environments. As proof-of-concept examples we considered parks, bus stops, and bridges. Parks were chosen as example of non-built open spaces. In terms of territorial expansion, the smallest of the parks we visited had an area of 19 acres, while the biggest an area approximately 350 acres. Bridges are usually instead located close to other buildings or built infrastructure. The bridges visited are all of the same approximate length (the difference between shortest and longest structure is 112 meters) but vary in terms of use (*i.e.*, we sampled bridges with both pedestrian and automotive traffic) and the intensity of that traffic as well as bridge design.

Finally, the third group contains subway stations characterized by deep underground structures (as well as the moving trains) and long-distance train stations (*i.e.*, above-ground, open areas). Also in this group we include parking lots were all locations visited can be described as underground structures (some with several parking levels) characterized by pillars, cars, low ceilings, and little or no pedestrian activity. Locations, excluding the parks, are contained in an area of 9.7 mi^2. In any

case, parks are contained within the greater metropolitan area of the city taken into consideration in this study.

6.3.3 Similarity Between Measurements

In Section 6.2 and again in Section 6.4 we describe the method used to separate one class from another and our rate of success in accomplishing this task. However, as part of exploration of the dataset, we now focus on the similarity between the observations of the same class. Figure 6.3 shows a histogram of the distance matrix between the observations. From the plots we can see that most of the observations within a class are similar to each other. The class that corresponds to Gym, for example, has the most similar observations (using cosine distance as a metric). This indicates that separating the distinct locations is particularly hard in this case. On the other hand, accurate classification and clustering require both minimizing distances within a class and maximizing distances between classes. Given the proximity of the observations, low classification scores may indicate that the similarity between classes makes this separation a difficult task.

6.4 Results

In this section we present the evaluation of the four time-series classification methods, which we discussed in Section 6.2.1, constructed to infer the location-types using magnetic field readings. As discussed in Section 6.2.2, we evaluate each method using three criteria: (*i*) all places, all devices, (*ii*) leave-a-place-out, and (*iii*) leave-a-device-out.

6.4.1 All Places, All Devices Evaluation

This evaluation corresponds to a scenario where the classifier has been trained with data originating from the device being tested at the location in question (*i.e.*, we have labeled data for all devices at every location). The separation between training and testing occurs over the temporal component of each observation. We convert each observation into ten 1 minute segments and randomly select 3 segments to form the testing set. The remaining 7 segments become the training set. We repeat this process 30 times to reduce the impact of the random selection in the results.

This evaluation criterion results in 100% accuracy for location-type identification from all devices.

In this scenario, the attack requires having labeled data from all devices at each target location (*i.e.*, building a map per device with all locations). In a deployed system this would require having real-time labeled data to use as the basis of the map. Because of the pervasiveness of mobile devices this could be possible however, this would be the equivalent of the brute-force approach. In the following sections we present two scenarios where we explore how to generalize the method for new devices and places.

6.4.2 Leave-a-Place-Out Evaluation

In this evaluation scenario, we use the data of a single location (from all devices) as the test set and the data for the remaining 109 places as train set. We repeat this until we use the data for each of the 110 places for testing, then aggregate the results of all iterations to determine the overall performance.

Table 6.1: Leave-a-Place-Out.

Full Signal Matching	
Distance Measure	*Accuracy*
DTW	**0.1862**
Euclidean	0.1750
Cosine	0.1713
Bhattacharyya	0.1005

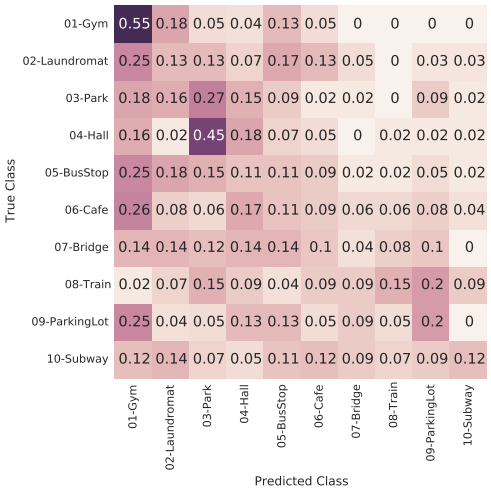


Figure 6.4: Full-Signal Matching: Class accuracy for Dynamic Time Warping proximity based classification.

In Table 6.1, 6.2, 6.3, and 6.4 we present the average accuracy for each classifier constructed for the feature extraction methods and evaluated using leave-a-place-out evalutation criterion (discussed in Section 6.2.2). Overall, our results

Table 6.2: Leave-a-Place-Out.

Statistical Descriptors	
Classifiers	*Accuracy*
*k*NN	0.1170
RF	0.2040
XGB	**0.2100**

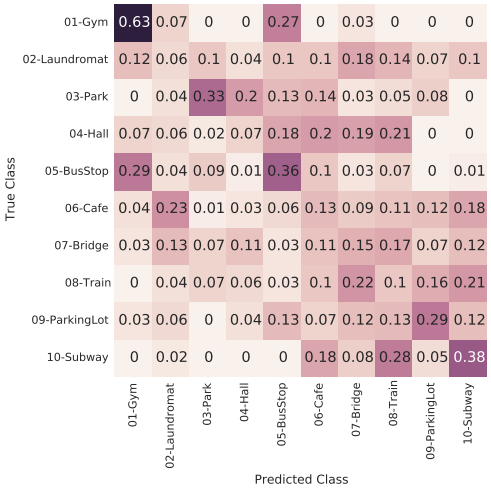


Figure 6.5: Statistical Descriptors: Class Accuracy for XGB Classifier.

Table 6.3: Leave-a-Place-Out.

Automated Feature Extraction	
CNN Layers	*Accuracy*
2	0.0901
3	0.0907
4	**0.0902**

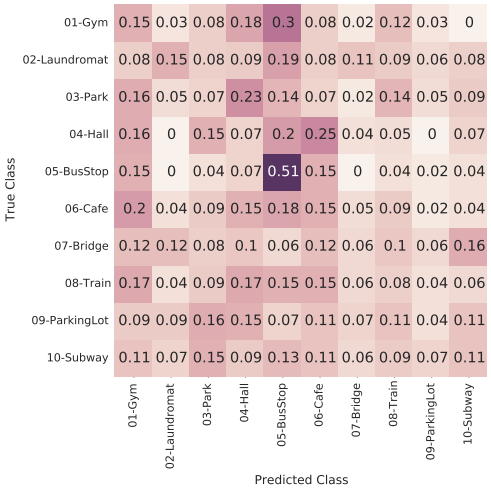


Figure 6.6: Automated Features: Classification Class Accuracy for 4-Layers CNN.

Table 6.4: Leave-a-Place-Out.

Shapelets	
Classifiers	*Accuracy*
*k*NN	0.2690
RF	**0.4045**
XGB	0.3920

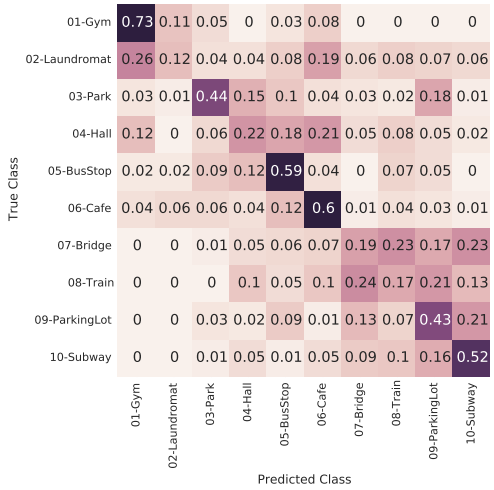


Figure 6.7: Shapelets: Class Accuracy for RF Classifier.

Table 6.5: Leave-a-Place-Out: Shapelets and Statistical Descriptors Combined.

Combined Feature Set	
Classifiers	*Accuracy*
*k*NN	0.1860
RF	**0.3907**
XGB	0.3442

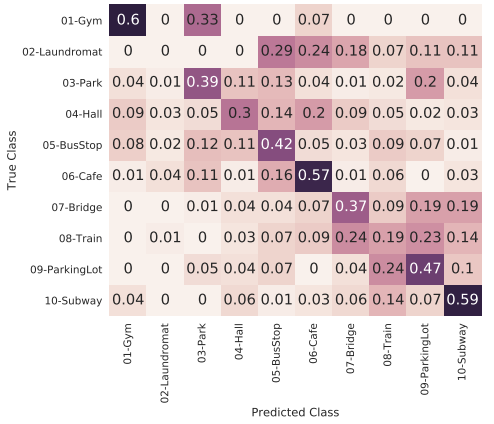


Figure 6.8: Leave-a-Place-Out, Combined Feature Set: Class Accuracy for RF Classifier.

Table 6.6: Leave-a-Place-Out: Frequency domain.

Frequency Domain			
	Shapelets	*Statistical Descriptors*	*Combined*
*k*NN	0.1654	0.2066	0.2066
RF	**0.1779**	**0.3333**	0.2447
XGB	0.1667	0.3152	**0.2915**

show that the classifier constructed with shapelets outperforms all others. The methods based on full signal matching (Table 6.1) and convolutional neural networks (Table 6.3) exhibit approximately random behavior in terms of their performance in predicting location-type. One important note is that we are not creating an absolute rank of the feature-extraction and classification methods. Our intent is to show that, for some applications, data aquisition is a real challenge (as is the case in this study). For this and similar studies, traditional signal processing methods can be outperformed by the method we propose: a combined shapelet classifier. As the results show, even with limited data, there is a significant increase in classification performance.

In particular, the classifiers based on statistical descriptors and shapelets are optimized using three algorithms (*i.e.*, *k*NN, RF, and XGB). Our results show that both RF and XGB achieve the highest accuracy (with a negligible difference between them), whereas, *k*NN has the worst performance. On the other hand, full signal matching classification is optimized through four different distance metrics (*i.e.*, DTW, Euclidean, Cosine, and Bhattacharya distances). We find that DTW is the measure that performs best. Finally, we optimize automated feature classification for different number of convolution layers (*i.e.*, 2, 3, and 4 layers with 4, 8, and 16 filters, respectively). The results for this optimization show that the model with 3 and 4 convolution layers achieve the best accuracy.

We further investigate whether the two best methods (*i.e.*, statistical descriptors and shapelets) extract and exploit different information from the data. To this end, we construct a new method that combines both of these feature sets as input. If both methods extract the same information, then the accuracy of the combined model should not improve. We compare the results of the new combined feature classifier against statistical descriptors and shapelets used as baselines. We construct the new combined feature set method by using the *k*NN, RF, and XGB classifiers.

In order to better understand the performance of all methods, in Figures 6.4, 6.5, 6.6, 6.7 and 6.8 we present the confusion matrices for the best performing parameters for each of the five methods (*i.e.*, combined features and the four methods

presented in Section 6.2.1). In each matrix, the average of the diagonal values corresponds to the overall accuracy of the indicated classifier.

Table 6.6 shows the results for the Leave-a-Place-Out analysis using the generalized correlation coefficient. As we can see, computing the statistical descriptors in the frequency domain results in higher classification accuracy. In contrast, classification based on shapelets in the frequency domain results in less than half the accuracy obtained in the time domain.

6.4.3 Leave-a-Device-Out Evaluation

The readings collected by mobile devices have some error due to the quality of the sensors, their age, and usage [112]. Each device has the software necessary to compensate for the error and provide measurements close to the actual value. As such, this error-correction enables us to crowd-source the data collection in order to process and extract more robust descriptors for each *location-type* as we would be able to observe the same places under different environmental settings. We evaluate our methods using a leave-one-device-out scenario as discussed in Section 6.2.2.

Table 6.7: Leave-a-Device-Out.

Full Signal Matching	
Distance Measure	*Accuracy*
DTW	0.1852
Euclidean	**0.2407**
Cosine	0.1481
Bhattacharyya	0.1204

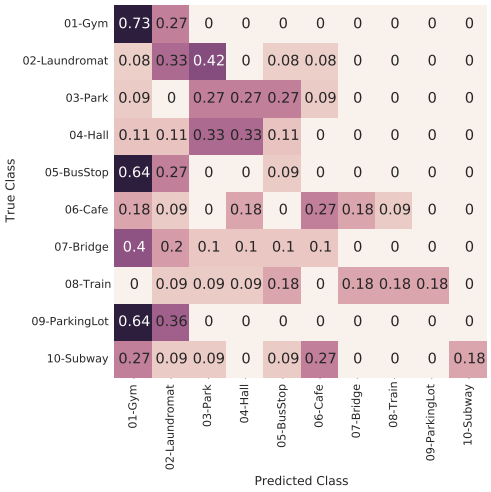


Figure 6.9: Class accuracy for Euclidean Distance proximity based classification.

In Tables 6.7, 6.8, 6.9 and 6.10 we present the average accuracy for the four classification approaches. Our results demonstrate that classifiers based on statistical descriptors and shapelets outperform the others (*i.e.*, full signal matching and

Table 6.8: Leave-a-Device-Out.

Statistical Descriptors	
Classifiers	*Accuracy*
*k*NN	0.1776
RF	**0.2998**
XGB	0.2926

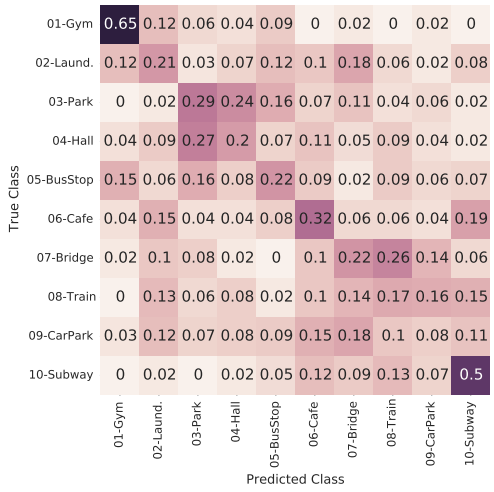


Figure 6.10: Class Accuracy for RF classifier.

Table 6.9: Leave-a-Device-Out.

Automated Feature Extraction	
CNN Layers	*Accuracy*
2	0.1025
3	**0.1291**
4	0.1274

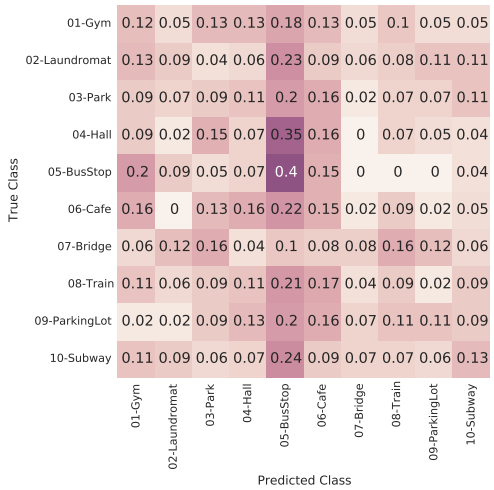


Figure 6.11: Classification Class Accuracy for 3-Layers CNN.

automated feature extraction). Note that this result is consistent with the analysis in Section 6.4.2.

Similar to the leave-a-place-out scenario, we optimized the methods based on statistical descriptors and shapelets using the same three classifiers (*i.e.*, *k*NN, RF, and XGB). Our results remain consistent and show that both RF and XGB achieve the highest accuracy (with a negligible difference between them). Full signal matching classification is optimized through four different distance metrics (*i.e.*, DTW, Euclidean, Cosine, and Bhattacharya distances), and we found that the method that

Table 6.10: Leave-a-Device-Out.

Shapelets	
Classifiers	*Accuracy*
*k*NN	0.2425
RF	0.3552
XGB	**0.3593**

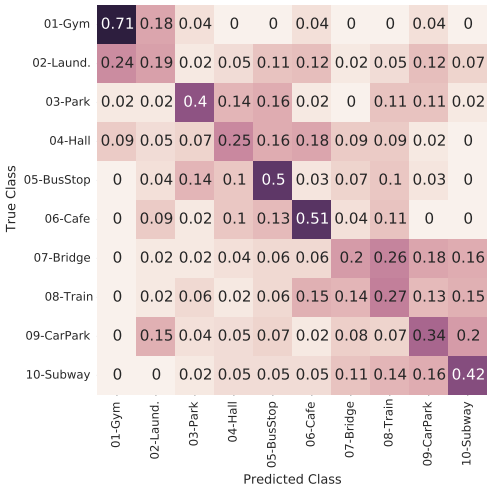


Figure 6.12: Class Accuracy for XGB Classifier.

Table 6.11: Leave-a-Device-Out: Shapelets and Statistical Descriptors Combined.

Combined Feature Set	
Classifiers	*Accuracy*
*k*NN	0.1776
RF	**0.4256**
XGB	0.4200

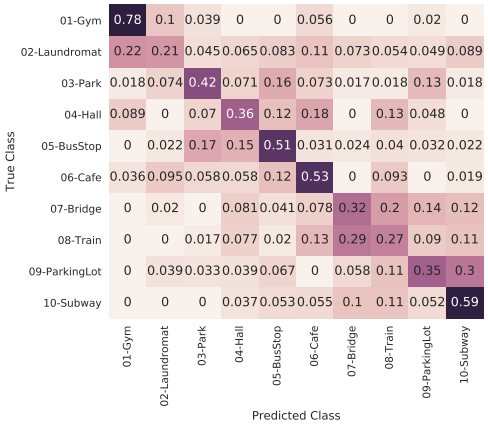


Figure 6.13: Leave-a-Device-Out, Combined Feature Set: Class Accuracy for RF Classifier.

works best uses Euclidean distance as its metric. Finally, we optimize the automated features classifier for different number of convolution layers (*i.e.*, 2, 3, and 4 layers with 4, 8, and 16 filters respectively). The results are again consistent with the previous analysis to show that the model with 3 and 4 convolution layers achieve the best accuracy.

As discussed earlier in Section 6.4.2, we investigate whether the two best methods (*i.e.*, statistical descriptors and shapelets) extract and exploit different information from the magnetic field readings. We construct a new classifier that combines both feature sets as input and compares the results against statistical descriptors and

Table 6.12: Leave-a-Device-Out: Frequency domain.

Frequency Domain			
	Shapelets	*Statistical Descriptors*	*Combined*
*k*NN	0.1750	0.1701	0.1640
RF	**0.2030**	0.3095	0.2634
XGB	0.1675	**0.3147**	**0.2790**

shapelets based methods (used as baselines). The evaluation results show that the classifier based on the combined features achieves an accuracy improvement of 7% as compared to the best of the previous methods. The increase in performance by the combined method can be attributed to the fact that the statistical descriptors exclude the outlying values from a dataset, whereas they are included by the shapelets in the temporal patterns [222]. Consequently, this shows the consistency of the two methods (*i.e.*, shapelets and statistical descriptors) for extracting complementary information from the time-series data. In Figures 6.9, 6.10, 6.11, 6.12 and 6.13 we present the confusion matrices for the best performing models for each of the five methods (the four methods presented in Section 6.2.1 and that based on the combination of the best two features).

Finally, Table 6.12 presents the results for the *GCC* analysis in the Leave-a-Device-Out evaluation. As compared to the Leave-a-Place-Out evaluation, the pattern remains the same: the classification accuracy using statistical descriptors improves while the accuracy using shapelets decreases.

6.4.4 Discussion

Shapelet-based classification derives time-series features that contain information about the form (or shape) of a segment of the observation and its outlying values. There is a two-fold bennefit to this method: first, shapelets make the classification algorithm explainable, and second, they provide a mechanism where the classification can run on the target device.

Explainability is addressed by considering that there might be a one-to-one correspondence between shapelets and events. It would therefore be possible to add to the shapelet dictionary a label to describe each event. This however requires a more exhaustive data collection process and the detailed manual labeling of each

reading and along with the variations that are allowed for each event. A data collection project of such magnitude, while achievable, is beyond the scope of this project.

In terms of resources, an application characterized by a disproportionate use of memory (used to store readings) and battery (for network transmission continuous observations) might look suspicious. In contrast, a system where shapelets are computed offline and transferred to the phone for classification will be practically unnoticeable to the user. Indeed, after training, classification in itself is a resource-efficient task. In our implementation, once computed, all shapelets (*i.e.*, the features) required approximately 50 kB of storage.

One of the contributions of this work is to study the form of a signal as a valuable source of information. In the time domain, the form corresponds to events or activities with a specific periodicity: a train moving towards the sensor or a person walking by. This information is lost when the signal undergoes the frequency transform, correspondingly the accuracy when classifying using shapelets drops significantly between the time and the frequency domain. This loss of meaning then extends into the combined feature set. Where before the two types of input where complementary and the accuracy improved from their combined information, the combination in the frequency domain results in a negative contribution making the classes less separable and reducing the overall accuracy.

6.5 Summary

In this chapter we have presented a zero-permission attack based on the use of the magnetometer and considered a variety of methods for identifying location-types from their magnetic signature. In particular, we have discussed two novel effective methods based on automated feature extraction using convolutional neural networks through One Shot Learning and shapelets (in time and frequency domains). We have discussed methodologies for selecting the parameters of these techniques.

In order to perform the experimental evaluation of the proposed attack, we performed an in-the-wild measurement study. We collected over 91 hours of data

from 10 location-types across a major metropolitan area and present our results from a set of 110 distinct locations. The collection times at each location varied in terms of time of day, day of the week, and, for some locations, the readings between phones differ by up to a month. We expect that by allowing such variation, we mitigated the impact of external influences on the generalizability of our results. We have shown that we are able to identify the type of a place using devices that were not used for collecting the data in the first place achieving accuracy equal to 40% against a random baseline of approximately 10% for both of them. These results show that there is an apparent privacy risk in terms of location inference associated to the possibility of accessing the magnetometer readings without permission.

Chapter 7

Conclusions

The emphasis of this dissertation is on empirical work. Each chapter presents a proof-of-concept study of an idea that can be generalized either in its methods, as is the case of both metadata (Chapter 4) and localization (Chapter 6), or in its application, as is the case with the traits used for device fingerprinting in Chapter 5. In every chapter we have focused on data that is accessible for collection either through public APIs or through the use of a sensing device. Our conclusions suggest that data generated by users poses a risk of identification for the user or their environment. It is our hope that these findings may contribute to a broader discussion about the appropriate use of data and its associated risks particularly by companies that employ practices that have come to be known as surveillance capitalism [235, 236].

7.1 Summary of Contributions

Each chapter presents a different facet of privacy and identification, now, we present the contributions of the dissertation, as a whole, in terms of the research questions addressed in this thesis.

We first looked at the metadata information available with each message posted to Twitter. We used the descriptors (or characteristics) of individual accounts captured in the metadata to distinguish between one account and another. We found that using four numerical fields were sufficient to generate fingerprints for each account, to the extent that we could distinguish one in 10,000 accounts with an accuracy of 96%. Practically, we showed that metadata is useful for identification.

This conclusion can potentially be extended to other services that record a user's interactions and behavior. Metadata is interesting in several respects. The nature of the information makes it so that it does not always fall under the purview of data protection legislation[1] but it is nonetheless an integral component of services offered by different providers. Similarly, users are aware that this type of information is collected, but are generally unaware of the types of inferences that can be drawn from it (phone records, for example, are a good example of users being aware that companies keep logs of call duration and timestamps but have limited knowledge of what this can say about them. Such as, for example, the place they live, where they shop, etc.).

The focus of the following chapter was on the problem of device fingerprinting. The aim was to find an identifier that would be universal, in other words, an identifier that would be independent of the presence of a specific component in a device as well as its manufacturer and model. In the second project, we looked at the magnetic field generated by individual devices as a possible source of identification. The implications of such an identifier were heightened by the fact that, for the first time, identification could be carried out without having network access to the target device. We found that using the magnetic field we could generate a robust fingerprint which could achieve (in our dataset) an accuracy of 98%. Sensor readings are characterized as low-risk and are therefore not protected under the permission-based architecture of mobile operating systems. Moreover, even if it were, identification would still be feasible and impossible to detect.

The main work of the dissertation is complete with the localization problem presented in Chapter 6. In this project, we make use of the magnetic field not for identifying a device but a place, a location. Artifacts (*e.g.*, refrigerators, coffee ma-

[1]Data protection legislation is only applicable to the processing of personal data of a natural person, a right which they loose upon their death. Personal data are defined as the "any information which are related to an identified or identifiable natural person. [13]". While a broad interpretation of the law is suggested to mean protection of data such as the one analyzed in this dissertation in practice the law is mostly applied to personally identifiable information like name, birthday, and credit card details to name a few and other sensitive information such as genetic, biometric and health data, racial and ethnic origin, political opinions, religious or ideological convictions or trade union membership. [237].

chines), people, and structures (*e.g.*, support columns that contain beams) generate a distortion of the magnetic field in their vicinity. With the magnetometer contained in each phone, we can match distinct locations to the likelihood of one or more changes present at a similar locations. We tried to find commonalities across places where the same activities (characterized by a particular distortion) might take place. The work we present shows that the magnetometer can partially reveal the environment on which the user is located (as opposed to their exact location).

We believe that the contribution of this work is methodological in nature. Indeed, the techniques presented can be applied to different and larger datasets. In fact, as future work, we plan to study the robustness and generalizability of the proposed methods by repeating the experiments for additional types of locations and different cities. It would be interesting to see if the similarities across location-types hold despite the baseline change in the magnetic field. Another interesting dimension to be investigated further is the stationarity of the magnetic signatures over time for the same location, for example over several months or years.

The sensitivity of the information that can be extracted from location data is potentially very high. Having even partial information about location makes it possible for an attacker to predict where a user will be in the future (derived from locations they have been to in the past), which might result in real personal risk to the user, it might also reveal personal traits of the individual, as might be their preferred locations or religious, political, or professional affiliations when you cross correlate this information with other datasets. While the ability to understand location without the need to map is interesting and useful (and can be used to protect, as would be the case if magnetic field localization were to be used instead of other more accurate location sensors, or violate a user's privacy, in case it were to be used as a tool for following and tracking the owner) we present a methodology that is new, readily accessible, and primed for discussion. Once again, the data and the principles used in this work do not rely on a security flaw that can be "patched" but on measurements of the environment, and as such, it bears consideration.

7.2 Discussion

One broader point of this dissertation was to consider some aspects of the reality of data privacy concerns in today's technological landscape. In every chapter, we have looked exclusively at data that is easy to access and difficult to protect. Data generated by individuals reflect some aspect of that person. Accessing this data is simple and deriving valuable information from it is a matter of persistence and resources. In our view, it is unreasonable to think we can place restrictions on every possible type of data. Even if we did, in some cases as was shown in Chapter 5, it is still possible for a dedicated attacker to learn something about the user. Where then, does this leave us? In Section 2.1 we discussed that privacy is essential in life as we know it, and this author agrees: without privacy, trust and therefore friendship, as well as many aspects of human relationships, would be unattainable. This is a discussion (along with many others) on the impact of technological developments on society. One suggestion might be to emphasize the importance of respect in a culture where everything can be known. We should discourage any practice that reduces people to objects to be studied or perhaps more realistically, financially exploited.

One practical conclusion derived from this work relates to the ease of access to potentially sensitive data. Permission-based security as defined in [52] has been adopted by major mobile operating systems [214, 215, 213]. This provides users with personalized controls in the trade-off between functionality and privacy. It allows developers to request access to whatever functionality in the device they might see fit and allows users the right to deny, revoke, or restrict access to the resources of their devices. There is, even in this model, the risk of 'function creep'. Function creep is the term used to describe the practice of using legitimately-collected data for an ulterior purpose [18]. While some form of function creep is almost expected (*e.g.*, when platforms use data for advertisement), this practice causes mistrust in the system and represents an ethical breach. The work presented in Chapters 4, 5, and 6 of this dissertation fall within this problematic definition. From a user's perspective, a legitimate use of metadata would be to, for example, identify spam

accounts as was demonstrated in [95, 98]. They would not however, expect to be re-identified from secondary information, specially when the data was intended to be anonymized. The same argument might be used in the work presented in Chapter 6. Navigation and gaming are two areas where the use of the magnetometer is central to the purpose of the task. Inferring any type of location information might be, at first, unexpected.

7.3 Limitations

Starting from the most general, the limitations of the work presented in this dissertation can be categorized as being either methodological or practical. Methodological limitations include those that we accept when we chose the types of analysis. From supervised learning, we inherit one of the most significant limitations: in our results, we are always comparing against classes that are known (*i.e.*, the classes for which we have observations). Using this approach we lack the flexibility of saying that an observation belongs to an unknown class. Similarly, the features used for the classifiers are not guaranteed to be optimal for the task at hand. This is particularly true for the localization problem.

Practical limitations are those that are related to the characteristics of the datasets and the observations contained in them. In terms of dataset, the key limitation is perhaps sample size. The number of observations and classes in our datasets is always limited (even for the Twitter study where the dataset included millions of users). For this reason it was not possible to study experimentally the validity and scalability of the proposed methods with very large datasets. Our strategy for robustness relied on randomizing the data collection process as an attempt to minimize the likelihood of constraining results to a particular set of circumstances. Any significantly larger dataset would require repeating the experiments to check whether the hypothesis hold.

This limitation extends to the observations in the dataset. Measurement instruments are not perfect and the data collection process happened organically. This makes our results useful and applicable to the technology currently available but it

also means that better instruments might lead to different results. Furthermore, all readings were collected at a particular point in time and under some specific conditions. We have a reasonable expectation that measurements at other times will be similar however, it is not possible to conclude that this is generally guaranteed.

7.4 Future Work

The work presented in this dissertation is intended as a stepping stone in a much greater body of knowledge. The objectives outlined in each chapter describe the scope and methods used in each piece of research. Individually, these projects contribute only small increments however, each of them opens new avenues of exploration. This section presents a summary of ideas for future research that follow from the contributions presented in each chapter of this dissertation.

Attribute Disclosure from Metadata In terms of privacy implications, attribute disclosure gives some avenue for further studies. If metadata is distinct enough to identify unique accounts then, it is sufficiently informative to potentially disclose traits that might be shared across several accounts. Similar to the works discussed in Chapter 2 on stylometry, analysis conducted on metadata and combined with other sources of information could leak private attributes of the user (*e.g.*, employment status, travel patterns, relationships).

Missing Link. Another open problem that the work in this dissertation might contribute towards resides in identifying the same entity across different datasets. This problem, also known as entity resolution or de-duplication, would be greatly simplified if there was a permanent unique identifier present across several datasets [238]. There is ample literature in the area of device fingerprinting to suggest that sensor readings include small variations that can lead to the identification of the measurement device used during collection [110, 111, 112, 206]. The work presented in Chapter 5 could lead to finding an underlying commonality across all sensors present on the same device. If it were to be found, this features could lead to the unification of independent measurements (from different datasets) to the same entity.

Automatic Device Recognition. The work presented in Chapter 5 suggests that each device generates a distinct magnetic field. In that chapter, we made the assumption that a single device is present during the measurement process. Building on this work it would be interesting to see whether it is possible to build a network of co-located (or frequently interacting) devices. This would require understanding how to distinguish between the interfering signals of multiple devices and the range of the emitted signal.

The privacy risk for users in this case is not in the identification of their personal device but rather, in the potential ability of devices identifying each other. As an example, this could lead to an undetectable mapping of all the devices that belong to the same individual. Or more generally, devices that repeatedly 'meet' which would indicate a some form of interaction between the owners (*e.g.*, coworkers, friends, commuters). In practice, applications of the resulting device networks would be useful for both police investigations and commercial enterprises in their understanding of customers.

Bio-Electric Fields. Following the work presented on Chapters 5 and 6, in addition to the device and the environment, one other distinct entity that might contribute with its own magnetic field are the users themselves. In Section 2.6.1 we discuss the characteristics required for a trait to become a biometric and the traits that have become 'mainstream' in this area.

Cells in the body are contained in fluids which are typically electrically neutral. Body fluids become charged in the presence of conductive ions required for different cell functions. Neurons, for example, use sodium, potassium, and chloride while the combination of elements like sodium, potassium, and calcium are more prevalent in the heart muscle [239]. In general, modern medical techniques measure bioelectric phenomena with electrodes strategically placed in the surface of the body however the magnetic field generated by any of these processes can be measured with a magnetometer [239, 240, 241]. Given the results obtained for electrical devices, it would be interesting to see whether the biomagnetic field is distinguishable from an electronic one and whether it can be used as a biometric.

7.5 Outlook

There is no way to stop the progress of technology nor would we want to. What then is the way forward once we realize that anonymization of data might not be the correct avenue in our pursuit of privacy? This author agrees with Zhang *et al.* in [17] when they state that it is very reasonable to give control of the data to the users. This has also been the legal trend. The benefits that technology brings to our lives is definitely worthwhile however the insidiousness displayed by some entities when sharing and analyzing data that was shared in good faith leaves a deep seed of mistrust in users. This type of behavior should continue to be discouraged through both the free market, by exposing data breaches and misuses, and state-level regulations to specifically prevent behaviors that may be out of the hands of consumers (*e.g.*, company mergers and acquisitions). Research in privacy like that presented through the dissertation aims to bring to light new aspects that have been overlooked with the intent of enabling those with the power to make decisions with the knowledge necessary to carry out their duty.

Bibliography

[1] State of the IoT 2018: Number of IoT devices now at 7B Market accelerating. https://iot-analytics.com/state-of-the-iot-update-q1-q2-2018-number-of-iot-devices-now-7b/, August 2018.

[2] MathWorks. Magnetometer Calibration Coefficients. https://www.mathworks.com/help/fusion/ref/magcal.html, 2020. Accessed: 2020-08-27.

[3] Anne Adams and Martina Angela Sasse. Users Are Not the Enemy. *Communications of the ACM*, 42(12):41–46, 1999.

[4] Tielei Wang, Kangjie Lu, Long Lu, Simon Chung, and Wenke Lee. Jekyll on iOS: When Benign Apps Become Evil. USENIX Security '13, Washington, D.C., 2013.

[5] Chi Zhang, Jinyuan Sun, Xiaoyan Zhu, and Yuguang Fang. Privacy and Security for Online Social Networks: Challenges and Opportunities. *IEEE Network*, 24(4):13–18, 2010.

[6] Cambridge Analytica Files. https://www.theguardian.com/news/series/cambridge-analytica-files, March 2018. Accessed: 2019-12-20.

[7] Lance Bonner. Cyber Risk: How the 2011 Sony Data Breach and the Need for Cyber Risk Insurance Policies Should Direct the Federal Response to Ris-

ing Data Breaches. *Washington University Journal of Law & Policy*, 40:257, 2012.

[8] Mary J Culnan and Cynthia Clark Williams. How Ethics Can Enhance Organizational Privacy: Lessons from the Choicepoint and TJX Data Breaches. *MIS Quarterly*, pages 673–687, 2009.

[9] John Markoff. Before the Gunfire, Cyberattacks. *New York Times*, 12:27–28, 2008.

[10] David P Fidler. Was Stuxnet an Act of War? Decoding a Cyberattack. *IEEE Security & Privacy*, 9(4):56–59, 2011.

[11] Roger Collier. NHS ransomware attack spreads worldwide, 2017.

[12] Department for Business, Energy and Industrial Strategy. Cyber Streetwise: open for business, 2014.

[13] European Commission. General Data Protection Regulation (GDPR). `https://gdpr-info.eu/`, 2018. Accessed: 2019-12-20.

[14] General Services Administration. GSA Directive CIO P 2180.1. `https://www.gsa.gov/reference/gsa-privacy-program/rules-and-policies-protecting-pii-privacy-act`, October 2014. Accessed: 2019-12-20.

[15] Prateek Joshi and C-C Jay Kuo. Security and Privacy in Online Social Networks: A Survey. ICME '11, Barcelona, Spain, 2011.

[16] Monica Tentori and Jesus Favela. Activity-Aware Computing for Healthcare. *IEEE Pervasive Computing*, 7(2):51–57, 2008.

[17] Daqing Zhang, Bin Guo, Bin Li, and Zhiwen Yu. Extracting Social and Community Intelligence from Digital Footprints: An Emerging Research Area. UIC '10, Xi'an, China, 2010.

[18] Emilio Mordini and Sonia Massari. Body, Biometrics and Identity. *Bioethics*, 22(9):488–498, 2008.

[19] Anil K Jain, Ruud Bolle, and Sharath Pankanti. *Biometrics: personal identification in networked society*. Springer Science & Business Media, 2006.

[20] E Justice. Brandeis: Olmstead v United States, 277 US 438, 478, 72 L, 1928.

[21] Thomas McIntyre Cooley. *A Treatise on the Law of Torts, Or the Wrongs which Arise Independently of Contract*. Callaghan, 1906.

[22] Samuel D. Warren and Louis D. Brandeis. The Right to Privacy. *Harvard Law Review*, 4(5):193–220, 1890.

[23] General Data Protection Regulation. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the Protection of Natural Persons with Regard to the Processing of Personal Data and on the Free Movement of such Data, and Repealing Directive 95/46. *Official Journal of the European Union (OJ)*, 59:1–88, 2016.

[24] Anton Alterman. A Piece of Yourself: Ethical Issues in Biometric Identification. *Ethics and Information Technology*, 5(3):139–150, 2003.

[25] Kazem Jahanbakhsh, Valerie King, and Gholamali C. Shoja. They Know Where You Live! *CoRR*, abs/1202.3504, 2012.

[26] Delip Rao, David Yarowsky, Abhishek Shreevats, and Manaswi Gupta. Classifying Latent User Attributes in Twitter. SMUC '10, Toronto, ON, Canada, 2010.

[27] Jie Tang, Yuan Zhang, Jimeng Sun, Jinhao Rao, Wenjing Yu, Yiran Chen, and A. C. M. Fong. Quantitative Study of Individual Emotional States in Social Networks. *IEEE Transactions on Affective Computing*, 3(2):132–144, April 2012.

[28] Vasilios Zorkadis and P Donos. On Biometrics-based Authentication and Identification from a Privacy-protection Perspective: Deriving Privacy-enhancing Requirements. *Information Management & Computer Security*, 12(1):125–137, 2004.

[29] Jeremy Blackburn, Emiliano De Cristofaro, Michael Sirivianos, and Thorsten Strufe. Cybersafety in Modern Online Social Networks (Dagstuhl Reports 17372). *Dagstuhl Reports*, 7(9):47–61, 2018.

[30] Cynthia Dwork. A firm foundation for private data analysis. *Communications of the ACM*, page 8695, 2011.

[31] Beatrice Perez, Mirco Musolesi, and Gianluca Stringhini. You Are Your Metadata: Identification and Obfuscation of Social Media Users Using Metadata Information. ICWSM '18, Stanford, CA, 2018.

[32] Beatrice Perez, Marco Musolesi, and Gianluca Stringhini. Fatal Attraction: Identifying Mobile Devices Through Electromagnetic Emissions. WiSec '19, Miami, Florida, 2019.

[33] Ferdinand Schoeman. Privacy: Philosophical Dimensions. *American Philosophical Quarterly*, 21(3):199–213, 1984.

[34] Alan F Westin. *Privacy and Freedom*. Bodley Head, 1967.

[35] Stanley I. Benn. *Privacy, freedom, and respect for persons*, page 223244. Cambridge University Press, 1984.

[36] Charles Fried. Privacy: A Moral Annalysis. *Yale Law Journal*, 77:21, 1968.

[37] Robert S. Gerstein. Intimacy and Privacy. *Ethics*, 89(1):76–81, 1978.

[38] Luciano Floridi. The Ontological Interpretation of Informational Privacy. *Ethics and Information Technology*, 7(4):185–200, 2005.

[39] Ninghui Li, Tiancheng Li, and Suresh Venkatasubramanian. t-Closeness: Privacy Beyond k-Anonymity and l-Diversity. In *ICDE '07*, Istanbul, Turkey, 2007.

[40] Vicenç Torra. *Data privacy: Foundations, new developments and the big data challenge*. Springer, 2017.

[41] Yves-Alexandre de Montjoye, Laura Radaelli, Vivek Kumar Singh, and Alex "Sandy" Pentland. Unique in the Shopping Mall: On the Reidentifiability of Credit Card Metadata. *Science*, 347(6221):536–539, 2015.

[42] Daniel Garcia-Romero and Carol Y Espy-Wilson. Automatic Acquisition Device Identification from Speech Recordings. ICASSP '10, Dallas, TX, USA, 2010.

[43] Elena Zheleva and Lise Getoor. To Join or Not to Join: The Illusion of Privacy in Social Networks with Mixed Public and Private User Profiles. WWW '09, pages 531–540, Madrid, Spain, 2009.

[44] Chad Cumby, Andrew Fano, Rayid Ghani, and Marko Krema. Predicting Customer Shopping Lists from Point-of-sale Purchase Data. SIGKDD '04, Seattle, Washington, USA, 2004.

[45] Gary M Weiss and Jeffrey W Lockhart. Identifying User Traits by Mining Smart Phone Accelerometer Data. SensorKDD '11, San Diego, CA, USA, 2011.

[46] Joseph F Traub, Yechiam Yemini, and H Woźniakowski. The statistical security of a statistical database. *ACM Transactions on Database Systems (TODS)*, pages 672–679, 1984.

[47] Rakesh Agrawal and Ramakrishnan Srikant. Privacy-preserving Data Mining. In *SIGMOD '00*, Dallas, Texas, USA, 2000.

[48] Dorothy E Denning and Jan Schlörer. A fast procedure for finding a tracker in a statistical database. *ACM Transactions on Database Systems (TODS)*, pages 88–102, 1980.

[49] Cynthia Dwork. Differential privacy: A survey of results. In *TAMC '08*, Xi'an, China, 2008.

[50] Dakshi Agrawal and Charu C Aggarwal. On the design and quantification of privacy preserving data mining algorithms. In *SIGMOD-SIGACT-SIGART '01*, 2001.

[51] Vassilios S Verykios, Elisa Bertino, Igor Nai Fovino, Loredana Parasiliti Provenza, Yucel Saygin, and Yannis Theodoridis. State-of-the-art in privacy preserving data mining. *ACM Sigmod Record*, 33(1):50–57, 2004.

[52] David Barrera, H Güneş Kayacik, Paul C Van Oorschot, and Anil Somayaji. A Methodology for Empirical Analysis of Permission-based Security Models and its Application to Android. CCS '10, Chicago, IL, USA, 2010.

[53] Charlie Miller. Mobile Attacks and Defense. *IEEE Security & Privacy*, 9(4):68–70, 2011.

[54] Asaf Shabtai, Yuval Fledel, Uri Kanonov, Yuval Elovici, Shlomi Dolev, and Chanan Glezer. Google Android: A Comprehensive Security Assessment. *IEEE Security & Privacy*, 8(2):35–44, 2010.

[55] Yabing Liu, Krishna P. Gummadi, Balachander Krishnamurthy, and Alan Mislove. Analyzing Facebook Privacy Settings: User Expectations vs. Reality. IMC '11, Berlin, Germany, 2011.

[56] Social Media Fact Sheet. `https://www.pewinternet.org/fact-sheet/social-media/`, June 2019.

[57] Fred Stutzman, Ralph Gross, and Alessandro Acquisti. Silent Listeners: The Evolution of Privacy and Disclosure on Facebook. *Journal of Privacy and Confidentiality*, 4(2):2, 2013.

[58] Emiliano De Cristofaro, Claudio Soriente, Gene Tsudik, and Andrew Williams. Hummingbird: Privacy at the Time of Twitter. In *SP '12*, San Francisco, CA, USA, 2012.

[59] Arvind Narayanan and Vitaly Shmatikov. De-anonymizing Social Networks. In *SP '09*, Oakland, CA, USA, 2009.

[60] Joseph Bonneau, Jonathan Anderson, and Luke Church. Privacy Suites: Shared Privacy for Social Networks. In *SOUPS '09*, Mountain View, CA, USA, 2009.

[61] Facebook Settles FTC Charges That It Deceived Consumers By Failing To Keep Privacy Promises. `https://www.ftc.gov/news-events/press-releases/2011/11/facebook-settles-ftc-charges-it-deceived-consumers-failing-keep`, November 2011.

[62] David Madigan, Alexander Genkin, David D Lewis, Shlomo Argamon, Dmitriy Fradkin, and Li Ye. Author Identification on the Large Scale. CSNA '05, 2005.

[63] Frederick Mosteller and David L. Wallace. Inference in an Authorship Problem. *Journal of the American Statistical Association*, 58(302):275–309, 1963.

[64] Michael Robert Brennan and Rachel Greenstadt. Practical Attacks Against Authorship Recognition Techniques. IAAI '09, Pasadena, CA, USA, 2009.

[65] Sadia Afroz, Aylin Caliskan Islam, Ariel Stolerman, Rachel Greenstadt, and Damon McCoy. Doppelgänger Finder: Taking Stylometry to the Underground. SP '14, San Jose, CA, USA, 2014.

[66] Efstathios Stamatatos, Nikos Fakotakis, and George Kokkinakis. Automatic Text Categorization in Terms of Genre and Author. *Computational Linguistics*, 26(4):471–495, 2000.

[67] Matthew L Jockers and Daniela M Witten. A Comparative Study of Machine Learning Methods for Authorship Attribution. *Literary and Linguistic Computing*, 25(2):215–223, 2010.

[68] Patrick Juola, John Sofko, and Patrick Brennan. A Prototype for Authorship Attribution Studies. *Literary and Linguistic Computing*, 21(2):169–178, 2006.

[69] Efstathios Stamatatos. A Survey of Modern Authorship Attribution Methods. *Journal of the American Society for information Science and Technology*, 60(3):538–556, 2009.

[70] Moshe Koppel, Jonathan Schler, and Shlomo Argamon. Computational Methods in Authorship Attribution. *Journal of the American Society for Information Science and Technology*, 60(1):9–26, 2009.

[71] Daniele Perito, Claude Castelluccia, Mohamed Ali Kaafar, and Pere Manils. How Unique and Traceable Are Usernames? PETS '11, Waterloo, ON, Canada, 2011.

[72] Patrick Juola, John I Noecker, Ariel Stolerman, Michael V Ryan, Patrick Brennan, and Rachel Greenstadt. Keyboard-Behavior-Based Authentication. *IT Professional*, 15(4):8–11, 2013.

[73] Arvind Narayanan, Hristo Paskov, Neil Zhenqiang Gong, John Bethencourt, Emil Stefanov, Eui Chul Richard Shin, and Dawn Song. On the Feasibility of Internet-Scale Author Identification. SP '12, San Francisco, CA, USA, 2012.

[74] Supreme Court of the United States. McIntyre vs. Ohio Elections Commission, 514 U.S. 334, 1995.

[75] Michael Hay, Gerome Miklau, David Jensen, Don Towsley, and Philipp Weis. Resisting Structural Re-identification in Anonymized Social Networks. *Proceedings of the VLDB Endowment*, 1(1):102–114, 2008.

[76] Arvind Narayanan, Elaine Shi, and Benjamin IP Rubinstein. Link Prediction by De-anonymization: How We Won the Kaggle Social Network Challenge. IJCNN '11, San Jose, CA, USA, 2011.

[77] Xuan Ding, Lan Zhang, Zhiguo Wan, and Ming Gu. De-anonymizing Dynamic Social Networks. GLOBECOM '11, Houston, TX, USA, 2011.

[78] Jianwei Qian, Xiang-Yang Li, Chunhong Zhang, and Linlin Chen. De-anonymizing Social Networks and Inferring Private Attributes Using Knowledge Graphs. INFOCOM '16, San Francisco, CA, USA, 2016.

[79] Lars Backstrom, Cynthia Dwork, and Jon Kleinberg. Wherefore Art Thou R3579x?: Anonymized Social Networks, Hidden Patterns, and Structural Steganography. WWW '07, Banff, Alberta, Canada, 2007.

[80] Bin Zhou and Jian Pei. Preserving Privacy in Social Networks Against Neighborhood Attacks. volume 8 of *ICDE '08*, Cancun, Mexico, 2008.

[81] Latanya Sweeney. k-anonymity: A model for Protecting Privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(5):557–570, October 2002.

[82] Ashwin Machanavajjhala, Daniel Kifer, Johannes Gehrke, and Muthuramakrishnan Venkitasubramaniam. l-diversity: Privacy beyond k-anonymity. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 1(1):3–es, 2007.

[83] Gilbert Wondracek, Thorsten Holz, Engin Kirda, and Christopher Kruegel. A Practical Attack to De-anonymize Social Network Users. SP '10, Oakland, CA, USA, 2010.

[84] Paridhi Jain, Ponnurangam Kumaraguru, and Anupam Joshi. @ i seek'fb. me': Identifying Users Across Multiple Online Social Networks. WWW '13, Rio de Janeiro, Brazil, 2013.

[85] Apple Announces iPhone 5s. https://www.apple.com/uk/newsroom/2013/09/10Apple-Announces-iPhone-5s-The-Most-Forward-Thinking-Smartphone-in-the-World, September 2013.

[86] Is the iPhone 6s Touch ID 2 sensor too fast? https://www.imore.com/touch-id-2-too-fast, December 2015.

[87] BBC News. Eye scanner project is scrapped. http://news.bbc.co.uk/1/hi/england/tyne/3652638.stm, September 2004. Accessed: 2019-12-20.

[88] Heather Crawford, Karen Renaud, and Tim Storer. A Framework for Continuous, Transparent Mobile Device Authentication. *Computers & Security*, 39:127 – 136, 2013.

[89] Marina Blanton and William Hudelson. Biometric-Based Non-transferable Anonymous Credentials. ICICS '09, Beijing, China, 2009.

[90] Stanley Thompson and Randy Kardon. The Argyll Robertson Pupil. *Journal of Neuro-Ophthalmology*, 26(2):134–138, 2006.

[91] Alen Peacock, Xian Ke, and Matt Wilkerson. Typing Patterns: A Key to User Identification. *IEEE Security Privacy*, 2(5):40–47, 2004.

[92] Vishal M. Patel, Rama Chellappa, Deepak Chandra, and Brandon Barbello. Continuous User Authentication on Mobile Devices: Recent Progress and Remaining Challenges. *IEEE Signal Processing Magazine*, 33(4):49–61, July 2016.

[93] Antigoni Maria Founta, Despoina Chatzakou, Nicolas Kourtellis, Jeremy Blackburn, Athena Vakali, and Ilias Leontiadis. A Unified Deep Learning Architecture for Abuse Detection. WebSci '19, Boston, Massachusetts, USA, 2019.

[94] Savvas Zannettou, Sotirios Chatzis, Kostantinos Papadamou, and Michael Sirivianos. The Good, the Bad and the Bait: Detecting and Characterizing Clickbait on YouTube. SPW '18, San Francisco, CA, USA, 2018.

[95] Gianluca Stringhini, Christopher Kruegel, and Giovanni Vigna. Detecting Spammers on Social Networks. In *ACSAC '10*, Austin, Texas, USA, 2010.

[96] Zhi Yang, Christo Wilson, Xiao Wang, Tingting Gao, Ben Y. Zhao, and Yafei Dai. Uncovering Social Network Sybils in the Wild. *ACM Transactions on Knowledge Discovery from Data*, 8(1):2:1–2:29, 2014.

[97] Cheng Bo, Lan Zhang, Xiang-Yang Li, Qiuyuan Huang, and Yu Wang. SilentSense: Silent User Identification via Touch and Movement Behavioral Biometrics. In *MobiCom '13*, Miami, FL, USA, 2013.

[98] Fabricio Benevenuto, Gabriel Magno, Tiago Rodrigues, and Virgilio Almeida. Detecting Spammers on Twitter. CEAS '10, Redmond, Washington, USA, 2010.

[99] Yi Shen, Jianjun Yu, Kejun Dong, and Kai Nan. Automatic Fake Followers Detection in Chinese Micro-blogging System. PAKDD '14, Tainan, Taiwan, 2014.

[100] Stefano Cresci, Roberto Di Pietro, Marinella Petrocchi, Angelo Spognardi, and Maurizio Tesconi. A Fake Follower Story: Improving Fake Accounts Detection on Twitter. *IIT-CNR, Tech. Rep. TR-03*, 2014.

[101] Savvas Zenonos, Andreas Tsirtsis, and Nicolas Tsapatsoulis. Twitter Influencers or Cheated Buyers? In *DASC/PiCom/DataCom/CyberSciTech '18*, 2018.

[102] Anthony Quattrone, Tanusri Bhattacharya, Lars Kulik, Egemen Tanin, and James Bailey. Is This You?: Identifying a Mobile User Using Only Diagnostic Features. In *MUM '14*, Melbourne, Victoria, Australia, 2014.

[103] Xiaoyong Zhou, Soteris Demetriou, Dongjing He, Muhammad Naveed, Xiaorui Pan, XiaoFeng Wang, Carl A. Gunter, and Klara Nahrstedt. Identity, Location, Disease and More: Inferring Your Secrets from Android Public Resources. In *CCS '13*, Berlin, Germany, 2013.

[104] Jagdish Prasad Achara, Gergely Acs, and Claude Castelluccia. On the Unicity of Smartphone Applications. WPES '15, Denver, Colorado, USA, 2015.

[105] Remo Manuel Frey, Runhua Xu, and Alexander Ilic. A lightweight User Tracking Method for App Providers. In *CF '16*, Como, Italy, 2016.

[106] Andreas Kurtz, Hugo Gascon, Tobias Becker, Konrad Rieck, and Felix Freiling. Fingerprinting Mobile Devices using Personalized Configurations. *Proceedings on Privacy Enhancing Technologies*, pages 4–19, 2016.

[107] Thomas Hupperich, Davide Maiorca, Marc Kührer, Thorsten Holz, and Giorgio Giacinto. On the Robustness of Mobile Device Fingerprinting: Can Mobile Users Escape Modern Web-Tracking Mechanisms? In *ACSAC '15*, Los Angeles, CA, USA, 2015.

[108] Ingrid Verbauwhede and Roel Maes. Physically Unclonable Functions: Manufacturing Variability As an Unclonable Device Identifier. In *GLSVLSI '11*, Lausanne, Switzerland, 2011.

[109] Jae W. Lee, Daihyun Lim, Blaise Gassend, G. Edward Suh, Marten van Dijk, and Srinivas Devadas. A Technique to Build a Secret Key in Integrated Circuits for Identification and Authentication Applications. In *VLSI '04*, Honolulu, HI, USA, 2004.

[110] Hristo Bojinov, Yan Michalevsky, Gabi Nakibly, and Dan Boneh. Mobile Device Identification Via Sensor Fingerprinting. *arXiv preprint arXiv:1408.1416*, 2014.

[111] Sanorita Dey, Nirupam Roy, Wenyuan Xu, Romit Roy Choudhury, and Srihari Nelakuditi. AccelPrint: Imperfections of Accelerometers Make Smartphones Trackable. In *NDSS '14*, San Diego, CA, USA, 2014.

[112] Gianmarco Baldini, Franc Dimc, Roman Kamnik, Gary Steri, Raimondo Giuliani, and Claudio Gentile. Identification of Mobile Phones using the Built-in Magnetometers Stimulated by Motion Patterns. *Sensors*, 17(4):783, 2017.

[113] Vladimir Brik, Suman Banerjee, Marco Gruteser, and Sangho Oh. Wireless Device Identification with Radiometric Signatures. In *MobiCom '08*, San Francisco, CA, USA, 2008.

[114] Tadayoshi Kohno, Andre Broido, and Kimberly C. Claffy. Remote Physical Device Fingerprinting. *IEEE Transactions on Dependable and Secure Computing*, 2(2):93–108, April 2005.

[115] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic Robotics*. MIT press, 2005.

[116] Galileo Navigation. `https://www.esa.int/Our_Activities/Navigation/Galileo/What_is_Galileo`, 2018. Accessed: 2019-12-20.

[117] Goran M Djuknic and Robert E Richton. Geolocation and Assisted GPS. *IEEE Computer*, (2):123–125, 2001.

[118] Quoc Duy Vo and Pradipta De. A Survey of Fingerprint-Based Outdoor Localization. *IEEE Communications Surveys Tutorials*, 18(1):491–506, 2016.

[119] Ning Chang, Rashid Rashidzadeh, and Majid Ahmadi. Robust Indoor Positioning using Differential WiFi Access Points. *IEEE Transactions on Consumer Electronics*, 56(3):1860–1867, 2010.

[120] Krishna Chintalapudi, Anand Padmanabha Iyer, and Venkata N. Padmanabhan. Indoor Localization Without the Pain. In *Mobicom '10*, Chicago, IL, USA, 2010.

[121] Jie Yang and Yingying Chen. Indoor Localization Using Improved RSS-Based Lateration Methods. In *GLOBECOM '09*, Honolulu, HI, USA, 2009.

[122] He Wang, Souvik Sen, Ahmed Elgohary, Moustafa Farid, Moustafa Youssef, and Romit Roy Choudhury. No Need to War-drive: Unsupervised Indoor Localization. In *MobiSys '12*, Low Wood Bay, UK, 2012.

[123] Oliver Woodman and Robert Harle. Pedestrian Localisation for Indoor Environments. In *UbiComp '08*, Seoul, Korea, 2008.

[124] Chenshu Wu, Zheng Yang, Yunhao Liu, and Wei Xi. WILL: Wireless Indoor Localization without Site Survey. *IEEE Transactions on Parallel and Distributed Systems*, 24(4):839–848, 2013.

[125] Janne Haverinen and Anssi Kemppainen. Global Indoor Self-localization Based on the Ambient Magnetic Field. *Robotics and Autonomous Systems*, 57(10):1028–1035, 2009.

[126] Anshul Rai, Krishna Kant Chintalapudi, Venkata N. Padmanabhan, and Rijurekha Sen. Zee: Zero-effort Crowdsourcing for Indoor Localization. In *Mobicom '12*, Istanbul, Turkey, 2012.

[127] Jaewoo Chung, Matt Donahoe, Chris Schmandt, Ig-Jae Kim, Pedram Razavai, and Micaela Wiseman. Indoor Location Sensing Using Geomagnetism. In *MobiSys '11*, Bethesda, MD, USA, 2011.

[128] Brandon Gozick, Kalyan P. Subbu, Ram Dantu, and Tomyo Maeshiro. Magnetic Maps for Indoor Navigation. *IEEE Transactions on Instrumentation and Measurement*, 60(12):3883–3891, 2011.

[129] Daniel Carrillo, Victoria Moreno, Benito Úbeda, and Antonio F Skarmeta. Magicfinger: 3d magnetic fingerprints for indoor location. *Sensors*, 15(7):17168–17194, 2015.

[130] Carlos E Galván-Tejada, Juan Pablo García-Vázquez, Jorge I Galván-Tejada, J Rubén Delgado-Contreras, and Ramon F Brena. Infrastructure-less indoor localization using the microphone, magnetometer and light sensor of a smartphone. *Sensors*, 15(8):20355–20372, 2015.

[131] Qu Wang, Haiyong Luo, Aidong Men, Fang Zhao, and Yan Huang. An infrastructure-free indoor localization algorithm for smartphones. *Sensors*, 18(10):3317, 2018.

[132] Imran Ashraf, Soojung Hur, and Yongwan Park. BLocate: A building identification scheme in GPS denied environments using smartphone sensors. *Sensors*, 18(11):3862, 2018.

[133] Sashank Narain, Triet D. Vo-Huu, Kenneth Block, and Guevara Noubir. Inferring User Routes and Locations Using Zero-Permission Mobile Sensors. In *SP '16*, San Jose, CA, USA, 2016.

[134] Josif Grabocka, Nicolas Schilling, Martin Wistuba, and Lars Schmidt-Thieme. Learning Time Series Shapelets. In *KDD '14*, New York, NY, USA, 2014.

[135] Jon Hills, Jason Lines, Edgaras Baranauskas, James Mapp, and Anthony Bagnall. Classification of Time Series by Shapelet Transformation. *Data Mining and Knowledge Discovery*, 28(4):851–881, 2014.

[136] Abdullah Mueen, Eamonn Keogh, and Neal Young. Logical-Shapelets: An Expressive Primitive for Time Series Classification. In *KDD '11*, San Diego, CA, USA, 2011.

[137] Lexiang Ye and Eamonn Keogh. Time Series Shapelets: A New Primitive for Data Mining. In *KDD '09*, Paris, France, 2009.

[138] Jesin Zakaria, Abdullah Mueen, and Eamonn Keogh. Clustering Time Series using Unsupervised Shapelets. In *ICDM '12*, Brussels, Belgium, 2012.

[139] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The elements of statistical learning*. Springer series in statistics New York, 2001.

[140] James H Stock and Mark W Watson. *Introduction to Econometrics*. Pearson, 2015.

[141] Stephen Marsland. *Machine learning: an algorithmic perspective*. CRC press, 2015.

[142] Tom M Mitchell. *Machine learning*. McGraw Hill, 1997.

[143] Leo Breiman. Random Forests. *Machine Learning*, 45(1):5–32, 2001.

[144] Thomas Cover and Peter Hart. Nearest Neighbor Pattern Classification. *IEEE Transactions on Information Theory*, 13(1):21–27, January 1967.

[145] Eli Stevens, Luca Antiga, and Thomas Viehmann. *Deep Learning with PyTorch*. Manning, 2020.

[146] Li Fe-Fei, Fergus, and Perona. A bayesian approach to unsupervised one-shot learning of object categories. In *ICCV '03*, Nice, France, 2003.

[147] Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. Siamese neural networks for one-shot image recognition. In *ICML '15*, Lille, France, 2015.

[148] Brenden Lake, Ruslan Salakhutdinov, Jason Gross, and Joshua Tenenbaum. One shot learning of simple visual concepts. In *CogSci '11*, Boston, MA, USA, 2011.

[149] Jane Bromley, Isabelle Guyon, Yann LeCun, Eduard Säckinger, and Roopak Shah. Signature verification using a" siamese" time delay neural network. In *NIPS '94*, Denver, CO, USA, 1994.

[150] Lee Humphreys, Phillipa Gill, and Balachander Krishnamurthy. How Much Is Too Much? Privacy Issues on Twitter. In *ICA '10*, Singapore, 2010.

[151] Johan Bollen, Huina Mao, and Alberto Pepe. Modeling Public Mood and Emotion: Twitter Sentiment and Socio-economic Phenomena. In *ICWSM '11*, Barcelona, Spain, 2011.

[152] James Hays and Alexei Efros. IM2GPS: Estimating Geographic Information from a Single Image. In *CVPR '08*, Anchorage, AK, USA, 2008.

[153] Yan Shoshitaishvili, Christopher Kruegel, and Giovanni Vigna. Portrait of a Privacy Invasion. *Proceedings on Privacy Enhancing Technologies*, 2015(1):41–60, 2015.

[154] Thiago H. Silva, Pedro O. S. Vaz de Melo, Jussara M. Almeida, Mirco Musolesi, and Antonio A. F. Loureiro. You are What you Eat (and Drink): Identifying Cultural Boundaries by Analyzing Food & Drink Habits in Foursquare. In *ICWSM '14*, Ann Arbor, MI, USA, 2014.

[155] Luca Rossi and Mirco Musolesi. It's the Way You Check-in: Identifying Users in Location-based Social Networks. In *COSN '14*, Dublin, Ireland, 2014.

[156] Gang Wang, Manish Mohanlal, Christo Wilson, Xiao Wang, Miriam Metzger, Haitao Zheng, and Ben Y Zhao. Social Turing Tests: Crowdsourcing Sybil Detection. In *NDSS '13*, San Diego, CA, USA, 2013.

[157] Kyle O. Bailey, James S. Okolica, and Gilbert L. Peterson. User Identification and Authentication Using Multi-modal Behavioral Biometrics. *Computers & Security*, 43:77–89, 2014.

[158] Liang Wang and Xin Geng. *Behavioral Biometrics for Human Identification: Intelligent Applications*. IGI Global, 2009.

[159] Christopher M Bishop. *Pattern Recognition and Machine Learning*. Springer, New York, 2001.

[160] Chenn-Jung Huang, Dian-Xiu Yang, and Yi-Ta Chuang. Application of Wrapper Approach and Composite Classifier to the Stock Trend Prediction. *Expert Systems with Applications*, 34(4):2870–2878, 2008.

[161] Huan Liu and Lei Yu. Toward Integrating Feature Selection Algorithms for Classification and Clustering. *IEEE Transactions on Knowledge and Data Engineering*, 17(4):491–502, April 2005.

[162] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12(Oct):2825–2830, 2011.

[163] Dong C. Liu and Jorge Nocedal. On the Limited Memory BFGS Method for Large Scale Optimization. *Mathematical Programming*, 45(1):503–528, 1989.

[164] Miranda Mowbray, Siani Pearson, and Yun Shen. Enhancing Privacy in Cloud Computing Via Policy-based Obfuscation. *The Journal of Supercomputing*, 61(2):267–291, 2012.

[165] Majid Bashir Malik, M. Asger Ghazi, and Rashid Ali. Privacy Preserving Data Mining Techniques: Current Scenario and Future Prospects. In *ICCCT '12*, Chennai, India, 2012.

[166] David E. Bakken, Rupa Parameswaran, Douglas M. Blough, Andy A. Franz, and T. J. Palmer. Data Obfuscation: Anonymity and Desensitization of Usable Data Sets. *IEEE Security Privacy*, 2(6):34–41, 2004.

[167] Huseyin Polat and Wenliang Du. Privacy-Preserving Collaborative Filtering Using Randomized Perturbation Techniques. ICMD '03, Melbourne, FL, USA, 2003.

[168] Christopher Mooney and Robert Duval. *Bootstrapping: A nonparametric approach to statistical inference*. Sage, 1993.

[169] Twitter, Inc. Twitter REST Public API documentation. `https://developer.twitter.com/en/docs/tweets/search/api-reference/get-search-tweets`, 2018. Accessed: 2019-12-20.

[170] Daiyong Quan, Lihua Yin, and Yunchuan Guo. Enhancing the Trajectory Privacy with Laplace Mechanism. Trustcom/BigDataSE/ISPA '15, Helsinki, Finland, 2015.

[171] Sensors Overview: Android Open Source Project. `https://developer.android.com/guide/topics/sensors/sensors_overview.html`, 2018. Accessed: 2019-12-20.

[172] Vasilios Mavroudis, Shuang Hao, Yanick Fratantonio, Federico Maggi, Christopher Kruegel, and Giovanni Vigna. On the Privacy and Security of the Ultrasound Ecosystem. *Proceedings on Privacy Enhancing Technologies*, 2017(2):95–112, 2017.

[173] Lukasz Olejnik. Report on Sensors APIs: privacy and transparency perspective. W3C Invited Expert, April 2016.

[174] Alasdair Allan. *Basic sensors in IOS: Programming the accelerometer, gyroscope, and more*. O'Reilly Media, 2011.

[175] Sensor Types: Android Open Source Project. `https://source.android.com/devices/sensors/sensor-types`, 2018. Accessed: 2019-12-20.

[176] Robert Callan, Alenka Zajic, and Milos Prvulovic. A Practical Methodology for Measuring the Side-Channel Signal Available to the Attacker for Instruction-Level Events. In *MICRO '14*, Cambridge, United Kingdom, 2014.

[177] Mordechai Guri, Assaf Kachlon, Ofer Hasson, Gabi Kedma, Yisroel Mirsky, and Yuval Elovici. GSMem: Data Exfiltration from Air-Gapped Computers

over GSM Frequencies. In *USENIX Security '15*, Washington, DC, USA, 2015.

[178] Alenka Zajic and Milos Prvulovic. Experimental Demonstration of Electromagnetic Information Leakage from Modern Processor-memory Systems. *IEEE Transactions on Electromagnetic Compatibility*, 56(4):885–893, 2014.

[179] Sebastian Biedermann, Stefan Katzenbeisser, and Jakub Szefer. *Hard Drive Side-Channel Attacks Using Smartphone Magnetic Field Sensors*, pages 489–496. Springer Berlin Heidelberg, Berlin, Heidelberg, 2015.

[180] Nikolay Matyunin, Jakub Szefer, Sebastian Biedermann, and Stefan Katzenbeisser. Covert Channels using Mobile Device's Magnetic Field Sensors. In *ASP-DAC '16*, Macau, China, 2016.

[181] Yongyao Cai, Yang Zhao, Xianfeng Ding, and James Fennelly. Magnetometer basics for mobile phone applications. Technical report, MEMSIC, 2012.

[182] PRIME Faraday Technology Watch 2002: An Introduction to MEMS. `http://www.lboro.ac.uk/microsites/mechman/research/ipm-ktn/pdf/Technology_review/an-introduction-to-mems.pdf`, 2018. Accessed: 2019-12-20.

[183] HAL Interface: Android Open Source Project. `https://source.android.com/devices/sensors/hal-interface`, 2018. Accessed: 2019-12-20.

[184] Konstantinos Papafotis and Paul P Sotiriadis. Computationally efficient calibration algorithm for three-axis accelerometer and magnetometer. In *MOCAST '19*, 2019.

[185] Chavdar Roumenin. Microsensors for magnetic fields. In *MEMS: a practical guide to design, analysis, and applications*, pages 453–521. Springer, 2006.

[186] W. Cai, J. Chan, and D. Garmire. 3-axes mems hall-effect sensor. In *2011 IEEE Sensors Applications Symposium*, San Antonio, TX, USA, 2011.

[187] Edward Ramsden. *Hall-effect sensors: theory and application*. Elsevier, 2011.

[188] Anne Perrin and Martine Souques. *Electromagnetic Fields, Environment and Health*. Springer Science & Business Media, 2013.

[189] Talat Ozyagcilar. Calibrating an ecompass in the presence of hard and soft-iron interference. *Freescale Semiconductor Ltd*, pages 1–17, 2012.

[190] Andrew Spielvogel and Louis Whitcomb. A stable adaptive observer for hard-iron and soft-iron bias calibration and compensation for two-axis magnetometers: Theory and experimental evaluation. *IEEE Robotics and Automation Letters*, 5(2):1295–1302, 2020.

[191] José Fernandes Vasconcelos, G Elkaim, Carlos Silvestre, P Oliveira, and Bruno Cardeira. Geometric approach to strapdown magnetometer calibration in sensor frame. *IEEE Transactions on Aerospace and Electronic systems*, 47(2):1293–1306, 2011.

[192] Sensor Stack: Android Open Source Project. `https://source.android.com/devices/sensors/sensor-stack`, 2018. Accessed: 2019-12-20.

[193] Panagiotis Andriotis, Martina Angela Sasse, and Gianluca Stringhini. Permissions Snapshots: Assessing Users' Adaptation to the Android Runtime Permission Model. In *WIFS '16*, Abu Dhabi, United Arab Emirates, 2016.

[194] Stuart J. Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. Pearson Education, 2 edition, 2003.

[195] Aksel Straume, Anders Johnsson, Gunnhild Oftedal, and Jonna Wilén. Frequency Spectra from Current vs. Magnetic Flux Density Measurements for Mobile Phones and Other Electrical Appliances. *Health Physics*, 93(4):279–287, 2007.

[196] Abdul Jabbar Jerri. The Shannon Sampling Theorem - Its Various Extensions and Applications: A Tutorial Review. *Proceedings of the IEEE*, 65(11):1565–1596, Nov 1977.

[197] Measuring Earths Magnetism. `https://earthobservatory.nasa.gov/images/84266/measuring-earths-magnetism`, 2018. Accessed: 2019-12-20.

[198] Jeffrey J. Love, Greg M. Lucas, Anna Kelbert, and Paul A. Bedrosian. Geoelectric Hazard Maps for the Mid-Atlantic United States: 100 Year Extreme Values and the 1989 Magnetic Storm. *Geophysical Research Letters*, 45(1):5–14.

[199] Electromagnetic fields and public health. `http://www.who.int/peh-emf/publications/facts/fs299/en/`, 2016. Accessed: 2019-12-20.

[200] Activity Manager: Android Developers. `https://developer.android.com/reference/android/app/ActivityManager.html`, 2018. Accessed: 2019-12-20.

[201] Best Practices for App Permissions: Android Developers. `https://developer.android.com/training/articles/user-data-permissions.html`, 2018. Accessed: 2019-12-20.

[202] Anupam Das, Nikita Borisov, and Matthew Caesar. Tracking Mobile Web Users Through Motion Sensors: Attacks and Defenses. In *NDSS '16*, San Diego, CA, USA, 2016.

[203] Jaewoo Chung, Matt Donahoe, Chris Schmandt, Ig-Jae Kim, Pedram Razavai, and Micaela Wiseman. Indoor Location Sensing using Geomagnetism. In *MobiSys '11*, Washington, DC, USA, 2011.

[204] Huan Liu and Lei Yu. Toward Integrating Feature Selection Algorithms for Classification and Clustering. *IEEE Transactions on Knowledge and Data Engineering*, 17(4):491–502, 2005.

[205] Andrew T. Campbell, Shane B. Eisenman, Nicholas D. Lane, Emiliano Miluzzo, Ronald Peterson, Hong Lu, Xiao Zheng, Mirco Musolesi, Kristof Fodor, and Gahng-Seop Ahn. The Rise of People-Centric Sensing. *IEEE Internet Computing Special Issue on Mesh Networks*, June/July 2008.

[206] Jiexin Zhang, Alastair R. Beresford, and Ian Sheret. SensorID: Sensor Calibration Fingerprinting for Smartphones. In *SP '19*, San Francisco, CA, USA, 2019.

[207] Mani Srivastava, Tarek Abdelzaher, and Boleslaw Szymanski. Human-centric sensing. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 370(1958):176–197, 2012.

[208] Xiaoyong Zhou, Soteris Demetriou, Dongjing He, Muhammad Naveed, Xiaorui Pan, XiaoFeng Wang, Carl A. Gunter, and Klara Nahrstedt. Identity, Location, Disease and More: Inferring Your Secrets from Android Public Resources. In *CCS '13*, 2013.

[209] AJ Brush, John Krumm, and James Scott. Exploring End-User Preferences for Location Obfuscation, Location-Based Services, and the Value of Location. In *UbiComp '10*, Copenhagen, Denmark, 2010.

[210] Dan Cvrcek, Marek Kumpost, Vashek Matyas, and George Danezis. A Study on the Value of Location Privacy. In *WPES '06*, Alexandria, VA, USA, 2006.

[211] Fuming Shih, Ilaria Liccardi, and Daniel Weitzner. Privacy Tipping Points in Smartphones Privacy Preferences. In *CHI '15*, Seoul, Republic of Korea, 2015.

[212] Chaoming Song, Zehui Qu, Nicholas Blumm, and Albert-László Barabási. Limits of Predictability in Human Mobility. *Science*, 327(5968):1018–1021, 2010.

[213] Apple Developer: Determining the Availability of Location Services. `https://developer.apple.com/documentation/corelocation/determining_the_availability_of_location_services`, 2018. Accessed: 2019-12-20.

[214] Location Permissions. `https://developer.android.com/reference/android/Manifest.permission`, 2018. Accessed: 2019-12-20.

[215] Windows Devices: Geolocation. `https://docs.microsoft.com/en-us/uwp/api/Windows.Devices.Geolocation`, 2018. Accessed: 2019-12-20.

[216] Yves-Alexandre De Montjoye, César A Hidalgo, Michel Verleysen, and Vincent D Blondel. Unique in the Crowd: The Privacy Bounds of Human Mobility. *Scientific reports*, 3:1376, 2013.

[217] Thiago H Silva, Pedro OS Vaz de Melo, Jussara M Almeida, Mirco Musolesi, and Antonio AF Loureiro. You Are What You Eat (and Drink): Identifying Cultural Boundaries by Analyzing Food and Drink Habits in Foursquare. In *ICWSM '14*, Ann Arbor, MI, USA, 2014.

[218] Lin Liao, Donald J. Patterson, Dieter Fox, and Henry Kautz. Learning and Inferring Transportation Routines. In *AAAI '04*, San Jose, CA, USA, 2004.

[219] Quannan Li, Yu Zheng, Xing Xie, Yukun Chen, Wenyu Liu, and Wei-Ying Ma. Mining User Similarity based on Location History. In *SIGSPATIAL '08*, Irvine, CA, USA, 2008.

[220] Jing Wang and Sajal K. Ghosh, R. K.and Das. A Survey on Sensor Localization. *Journal of Control Theory and Applications*, 8(1):2–11, feb 2010.

[221] Yin Chen, Dimitrios Lymberopoulos, Jie Liu, and Bodhi Priyantha. FM-based Indoor Localization. In *MobiSys '12*, Low Wood Bay, UK, 2012.

[222] Tak-Chung Fu. A Review on Time Series Data Mining. *Engineering Applications of Artificial Intelligence*, 24(1):164–181, 2011.

[223] Jun Han, Emmanuel Owusu, Le T Nguyen, Adrian Perrig, and Joy Zhang. ACcomplice: Location Inference using Accelerometers on Smartphones. In *COMSNETS '12*, Bangalore, India, 2012.

[224] Yan Michalevsky, Aaron Schulman, Gunaa Arumugam Veerapandian, Dan Boneh, and Gabi Nakibly. PowerSpy: Location Tracking Using Mobile Device Power Analysis. In *USENIX Security '15*, Washington, D.C., USA, 2015.

[225] Sarfraz Nawaz and Cecilia Mascolo. Mining Users' Significant Driving Routes with Low-Power Sensors. In *SenSys '14*, Memphis, TN, USA, 2014.

[226] Lei Zhang, Jiangchuan Liu, Hongbo Jiang, and Yong Guan. SensTrack: Energy-Efficient Location Tracking with Smartphone Sensors. *IEEE Sensors Journal*, 13(10):3775–3784, 2013.

[227] Saeed Aghabozorgi, Ali Seyed Shirkhorshidi, and Teh Ying Wah. Time-series Clustering – A Decade Review. *Information Systems*, 53:16–38, 2015.

[228] Anthony Bagnall, Jason Lines, Aaron Bostrom, James Large, and Eamonn Keogh. The Great Time Series Classification Bake Off: A Review and Experimental Evaluation of Recent Algorithmic Advances. *Data Mining and Knowledge Discovery*, 31(3):606–660, 2017.

[229] Thomas Kailath. The Divergence and Bhattacharyya Distance Measures in Signal Selection. *IEEE Transactions on Communication Technology*, 15(1):52–60, 1967.

[230] Tianqi Chen and Carlos Guestrin. XGBoost: A Scalable Tree Boosting System. In *KDD '16*, San Francisco, CA, USA, 2016.

[231] Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. *Deep Learning*, volume 1. MIT press Cambridge, 2016.

[232] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality Reduction by Learning an Invariant Mapping. In *CVPR '06*, New York, NY, USA, 2006.

[233] D. Brook and R.J. Wynne. *Signal processing: principles and applications*. Edward Arnold, London, England, UK, 1988.

[234] Lawrence R Rabiner. Digital Processing of Speech Signal. *Digital Processing of Speech Signal*, 1978.

[235] Sarah Myers West. Data Capitalism: Redefining the Logics of Surveillance and Privacy. *Business & Society*, 58(1):20–41, 2019.

[236] Shoshana Zuboff. Big Other: Surveillance Capitalism and the Prospects of an Information Civilization. *Journal of Information Technology*, 30(1):75–89, 2015.

[237] GDPR: Personal Data. `https://gdpr-info.eu/issues/personal-data/`, January 2020.

[238] Steven Euijong Whang and Hector Garcia-Molina. Joint entity resolution. In *ICDE '12*, Washington, DC, USA, 2012.

[239] Leif Sörnmo and Pablo Laguna. *Bioelectrical signal processing in cardiac and neurological applications*, volume 8. Academic Press, 2005.

[240] Robert Plonsey and Roger C Barr. *Bioelectricity: a quantitative approach*. Springer Science & Business Media, 2007.

[241] Plonsey Malmivuo, Jaakko Malmivuo, and Robert Plonsey. *Bioelectromagnetism: principles and applications of bioelectric and biomagnetic fields*. Oxford University Press, USA, 1995.