# Using Machine Learning to Improve Internet Privacy

Sebastian Zimmeck

Submitted in partial fulfillment of the
requirements for the degree
of Doctor of Philosophy
in the Graduate School of Arts and Sciences

COLUMBIA UNIVERSITY

2017

# ABSTRACT

## Using Machine Learning to Improve Internet Privacy

Sebastian Zimmeck

Internet privacy lacks transparency, choice, quantifiability, and accountability, especially, as the deployment of machine learning technologies becomes mainstream. However, these technologies can be both privacy-invasive as well as privacy-protective. This dissertation advances the thesis that machine learning can be used for purposes of improving Internet privacy. Starting with a case study that shows how the potential of a social network to learn ethnicity and gender of its users from geotags can be measured, various strands of machine learning technologies to further privacy are explored. While the quantification of privacy is the subject of well-known privacy metrics, such as $k$-anonymity or differential privacy, I discuss how some of those metrics can be leveraged in tandem with machine learning algorithms for purposes of quantifying the privacy-invasiveness of data collection practices. Further, I demonstrate how the current notice-and-choice paradigm can be realized by automated machine learning privacy policy analysis. The implemented system notifies users efficiently and accurately on applicable data practices. Further, by analyzing software data flows users are enabled to compare actual to purported data practices and regulators can enforce those at scale. The emerging cross-device tracking practices of ad networks, analytics companies, and others can be supplemented by machine learning technologies as well to notify users of privacy practices across devices and give them the choice they are entitled to by law. Ultimately, cross-device tracking is a harbinger of the emerging Internet of Things, in which I envision intelligent personal assistants that help users navigating through the increasing complexity of privacy notices and choices.

# Table of Contents

# List of Figures

# List of Tables

# Acknowledgments

Seven years at Columbia culminated in this thesis. It was not always easy, but it was always with heart and soul. Now I am embarking on a new journey. I am grateful for all the friends I made and the memorable moments that remain. In particular, I would like to thank my advisor Steve Bellovin, who gave me the academic freedom and trust to explore the space. I mean really. It was always a pleasure to work with you, Steve. Thank you.

I am fortunate to have further distinguished faculty on my thesis committee: Augustin Chaintreau, who never failed to grind fresh coffee in the face of fast approaching paper deadlines, Roxana Geambasu, whose passionate infusion of systems research with privacy was inspiring, Joel Reidenberg, who in his objectivity and detailedness was a true academic, and Sal Stolfo, whose intrusion detection expertise was only surpassed by his observations on life in general.

I immensely profited from the knowledge of many more teachers here at Columbia. First and foremost, I was always thrilled to collaborate with Tony Jebara. Al Aho provided programming language insights. Early on I learned from Adam Cannon and then from Xi Chen, Mihalis Yannakakis, Alexandros Biliris, Luca Carloni, Jonathan Gross, Paul Blaear, Dan Rubinstein, Alexander Pasik, Kaustubh Joshi, and Tal Malkin (and if it were not for Tal, I might not have been in Columbia's PhD program at all).

There were so many colleagues whose suggestions found their way into my work, one way or another. They are: Terry Tsai, Adrian Tang, Marios Pomonis, Theofilos Petsios, Marcin Szczodrak, Yannis Spiliopoulos, Riley Spahn, Suphannee Sivakorn, Kanad Sinha, Chris Riederer, Avner May, Amit Levy, Mathias Lecuyer, Georgios Kontaxis, Hyungtae Kim, Yuan Kang, Jill Jermyn, Stephen Fitz, Clément Canonne, Pablo Javier Barrio, Vaggelis Atlidakis, George Argyros, Suman Jana, Daisy Nguyen, Noura Farra, Andrea Lottarini, Cynthia Jihye Kwon, and, of course, Naser AlDuaji.

I would also like to thank Remi Moss and Jessica Rosa for their help in departmental affairs.

Remi picked up the phone a long time ago when I initially applied to Columbia and had missed the application date for the Master of Science program (glad you did, Remi).

In the course of my studies I had the chance to begin a terrific collaboration with Norman Sadeh, Florian Schaub, Shomir Wilson, and Bin Liu of the Usable Privacy Policy project at Carnegie Mellon, which I will, in fact, continue going forward.

What would have been my experience here in New York without a little help from my friends? Paul Schwartz, probably, the biggest pop and rock enthusiast I know, helped me out with many jobs and recommendations (I will pay it forward, Paul). Reelin' in the years with Nicole Ngo, Sandeep Sandhu, Matt Crosby, Peter Compitello, Julia DeWolf, Melanie Kambadur, Ryan Weng, Yiling Hu, Jenny Liu, and Andy Liu (with whom I spent many Thanksgivings and late nights hastily programming). Finally, I am grateful for the love and support of my parents without whom this work simply would not have been possible.

September 28, 2016<br>
Sebastian Zimmeck

To Jie S. Li and the life we built for ourselves
in New York and Seattle

# Chapter 1

# Introduction

The current state of Internet privacy is unsatisfactory: Internet users are often not aware of what happens with their data when they visit websites or use mobile apps. In many cases it is not obvious what types of data are collected or shared. Data practices are opaque. Further, opting out from targeted advertising is cumbersome, and self-regulatory efforts are only beginning to take shape. In addition, companies are frequently in doubt about their privacy obligations as well, and regulators have difficulty enforcing existing laws. Therefore, it is the broad theme of my research to advance privacy law on the Internet through technological solutions, more specifically, by leveraging machine learning (ML) technologies. In this dissertation I will sketch privacy technologies that advance transparency for Internet users, help companies in their efforts to develop compliant privacy standards on which they can compete in the marketplace, and assist governmental agencies and regulators with their privacy enforcement tasks.

## 1.1 ML Is the Problem. ML Is the Solution.

For the most part ML is perceived as a privacy-invading threat. The electronic traces that every Internet user leaves behind—whether Personally Identifiable Information (PII) or metadata—can be used to predict new information about that user (and oftentimes also about other users, such as friends on social networks). Many free Internet services are ad-financed and often frequently make use of ML technologies to learn more about their users and increase their revenue. However, in this work I take the opposite view and discuss the use of ML approaches for purposes of privacy enhancing technologies (PETs). Since many of the current privacy concerns are based on the exploitation of ML technologies it is only appropriate to mitigate those by leveraging the same technologies. In combination with other security and privacy technologies ML technologies provide a key element for protecting privacy in the modern Internet eco-system.

## 1.2 The Curious Relationship between Privacy and Technological Innovation

Privacy rights and concepts are often developed as a reaction to technological innovations. In the 19th century the right of privacy emerged against the background of the proliferating photo technology that enabled yellow press journalism. Today it is even more clear that the Internet and other new technologies, which lead to a dramatically increasing availability of user data to businesses and governments, pose new challenges for the protection of privacy [236]. Data business models fueled by the dispersion of data evolved and are commonplace. Internet users are tracked—often across devices—and their data is mined for purposes of contextual or targeted advertisements. However, more and more users are engaging in technological self-help, for example, by using ad blockers. Interestingly, Internet services are usually not enforcing their terms of services and privacy policies against users, which departs from the practice in many types of form contracts. In any case, law and regulations have yet to catch up to reality.

## 1.3 Privacy as a Right

Privacy is a fundamental right under the law in many jurisdictions. I see it as a natural right and adopt the definition of privacy as "[t]he right of individuals to control or influence what information related to them may be collected and stored and by whom and to whom that information may be disclosed [234]."[1] There are many more dimensions to privacy, for example, the philosophical [43] or economic [22] perspective of privacy. However, what I am considering in this dissertation is privacy as a right, which is, obviously, informed by the other aspects as well. To that end, the research presented here is used to advance law. Technology on its own would be

[1] With its focus on communication of information the definition used here stands in the tradition of Westin's [258].

an aimless endeavor to privacy. As it was argued for network research [156], many research efforts are stymied by a combination of economic and legal issues that were not considered in the research. Thus, researchers should pay attention to what is in the public interest, to the interests of the parties that may implement the idea, and to whether these interests coincide [156]. In this sense, I am addressing Internet privacy as it is rooted in the law.

**The Fourth Amendment.** At the outset, information privacy law in the United States is an amalgamate of various interrelated constitutional provisions, statutory laws, and regulations [236]. Privacy as a constitutional right—per the U.S. Constitution—is generally only applicable vis-à-vis the government and does not bind private actors. While the U.S. Constitution does not explicitly provide a privacy right, courts have used the Fourth Amendment's prohibition of unreasonable searches and seizures to construe a protective space for an individual's reasonable expectation of privacy [162]. In this regard, the Fourth Amendment protects an individual's privacy if he or she exhibited an actual expectation of privacy and if that expectation is recognized by society [162].[2] Given the existence of a reasonable expectation of privacy, police actions and other governmental conducts generally require a warrant.

Traditionally, each governmental action is treated as a discrete event that is evaluated individually for its Fourth Amendment relevance. For example, in United States v. Knotts [252] the Supreme Court evaluated the privacy implications of tracking a car during a single trip for less than a day as opposed to comprehensively analyzing the totality of multiple trips. More recently, however, in United States v. Jones [249] the Supreme Court made inroads to recognize that police surveillance and other governmental actions can become more privacy-invasive over time; even when fully occurring in the public sphere. This latter point stands in contrast to the Court's earlier opinions, e.g., in Knotts [252], holding that public observations can not be reasonably thought of as private. Under what became known as the mosaic theory intrusions can rise to the level of violating reasonable privacy expectations on the basis of extended observations, each of which by itself may not be sufficient to reach the threshold of a violation.

[2] There are other theories of how privacy is treated in the Constitution; see Griswold v. Connecticut, 381 U.S. 479, 485 (1965).

**Privacy as protection from identification and discrimination.** Privacy can be understood to protect from identification. If a person remains unrecognized or indistinguishable from one or more other people, his or her privacy will often be sufficiently protected. However, in addition to the identification risk privacy should also protect from discrimination. This understanding is a result of defining privacy as control over information collection, sharing, and storage. The holder of the privacy right can prevent the processing of potentially discriminating information. The control over this type of information is especially important as redlining—the practice of not providing services or maintaining an increased pricing level in ethnic or racially diverse neighborhoods—is re-appearing in a data-driven form. For example, the Federal Trade Commission (FTC) explored in a study on consumer auto insurance premiums [107] that credit-based insurance scores are distributed differently among racial and ethnic groups. The study finds that while the scores seem to derive only a relatively small amount of predictive power from their correlation with race and ethnicity, the observed difference is likely to have an effect on the insurance premiums that these groups pay. Different from redlining in its traditional form the discrimination appears inadvertently. However, as the researchers were not able to develop an alternative scoring model of the same efficacy without accounting for the differences in scores among racial and ethnic groups, the study highlights the difficulty of eliminating private facts from machine learning-based reasoning without incurring a performance penalty.

The FTC study deserves a closer look for another reason: auto insurance companies are making decisions based on predictive modeling. However, what is the meaning of predictions in terms of legal categories? They are not facts, but rather probabilities (different, for example, from the decisions to offer motorists insurance at a certain price point, which are indeed facts). At the outset there are various areas of law that arguably support probabilistic reasoning. For example, the Supreme Court's interpretation of evidentiary standards [248] are a seemingly good fit when Justice Harlan states that "[a]lthough the phrases 'preponderance of the evidence' and 'proof beyond a reasonable doubt' are quantitatively imprecise, they do communicate to the finder of fact different notions concerning the degree of confidence he is expected to have in the correctness of his factual conclusions." Thus, while the Court has eschewed to embrace bright lines there seem to be quantitative conceptions that inform the interpretation of evidentiary standards. However, it should also be noted that a purely mechanical quantification without considering the plausibility of evidence, for example, in the sense of a “plausible cause” [53], would shortcut the Fourth Amendment.

**Limits to privacy.** The privacy right is not without limits.[3] In particular, the Stored Communications Act, which is part of the Electronic Communications Privacy Act, provides the conditions under which governmental agencies can access electronic data held at private organizations. In 18 U.S.C. §2703(a), (b) it is stated that service providers have to disclose to the government content of electronic communications held in an account for more than 180 days under a subpoena or court order. Only if the communication was stored for fewer than 180 days does the law require a warrant from the government. The distinction is important because issuance of a warrant demands probable cause while the government can obtain a subpoena or court order under the lower standard of establishing reasonable grounds for the belief that the content of a searched item is relevant for an investigation [141]. However, as various governmental agencies in the past tried to obtain access to data stored at private Internet companies, the latter became increasingly reluctant to reveal the data of their users. Also, the increased use of encryption technologies in their products makes it harder for the government to gain access to the underlying unencrypted data even if the legal requirements are met and the companies are willing to help. Striking the right balance between individuals’ privacy rights and law enforcement is a challenging and unresolved task.

**Privacy protection vis-à-vis private organizations.** The Fourth Amendment is generally not applicable in the relationship between private entities. Rather, to a substantial extent privacy law is permeated by the goal of consumer protection and based on the notice and choice principle [70]. Internet users are notified about how their data are processed and they can use opt outs and other choice mechanisms to craft their relationship to the data processor. In this regard, Internet users’ privacy is often dependent on privacy policies (in addition there are a few narrow federal laws, state laws, and regulations, for example, covering childrens’ privacy rights). Typically, the provider of a web service posts a privacy policy on its website, which a user accepts by using the site. Thus, privacy policies are fundamental building blocks of web privacy, and the FTC as well as other regulators aim to enforce companies’ violations of the promises contained in the policies rigorously.

[3] There are many more limitations not discussed here. Particularly, European law in form of its General Data Protection Regulations [101], which includes the notorious right to be forgotten [265], is exacting.

However, many users do not read privacy policies and those who do find them oftentimes hard to understand [191]. The resulting information asymmetry leaves users uninformed about their privacy choices [191], can lead to market failure [188], and ultimately casts doubt on the notice and choice principle as a whole. Thus, while in theory the notice and choice principle is sound, there remain many practical challenges.

## 1.4 Transparency, Choice, Quantifiability, Accountability

Privacy as a right guarantees individuality and liberty. To effect these values my research is organized along four privacy principles: transparency, choice, quantifiability, and accountability. In the new data economy, as it is sometimes dubbed [17], users are assumed to be aware of their privacy rights through privacy policies. However, as McDonald et al. have shown [191], the current notice and choice approach is challenging. Various attempts to mitigate the lack of transparency, most notably P3P [75], remained unsuccessful. Consequently, this dissertations proposes the idea of using ML classifiers to automatically analyze privacy policy text and show users strongly condensed policy terms that can be more easily grasped (Chapter 5). However, giving users' choice is also a field for further improvement. Current mechanisms, for example, for opting out from receiving targeted advertisement, requires substantial investment of time and nearly expert privacy knowledge.

Beyond deficiencies in privacy transparency and choice recent findings suggest a lack of accountability as well. For example, a recent study detected more than 256 iOS apps in violation of Apple's App Store privacy policy due to the disclosure of device serial numbers and other data to third party library developers [132]. This finding suggests a deficiency in terms of accountability. However, the detection of those non-compliant apps appears to be a far cry from achieving systematic accountability of apps' data practices on the large scale. Furthermore, even in cases where privacy violations can be identified, it is unclear how their invasiveness can be measured. How can their harm be quantified? When is enough enough [46]? As I will show, this question can be addressed using ML methodology (Chapter 3).

The four principles—transparency, choice, quantifiability, and accountability—bear a strong resemblance to various sets of Fair Information Practice Principles (FIPPs) [127]. However, they are more abstract and leave more freedom for individual lawmaking and agreement between users and Internet services. For example, in my opinion, there is no need for limiting the use of data to the purpose they were originally collected for; a limitation that is included in some FIPPs. These purpose limitations should be made part of individual laws or agreements. However, declaring such a blanket statement generally applicable as part of a set of privacy principles appears to be overbearing.

## 1.5 Practical Considerations

There are various practical considerations that should be considered when designing PETs for Internet users. First, in the same way as security is a secondary task [126], privacy appears to be a secondary task as well. Generally, users are not interested in spending inordinate amounts of time fiddling with privacy settings or reading privacy policies. Thus, privacy technologies have to be as fast, automated, and comprehensible as possible. To some extent privacy is a question of usability and human-computer interaction; topics that this dissertation will, however, not discuss in detail.

Second, the classical security threat models [45] appear a poor fit for the types of Internet privacy questions examined here. After all, users can agree to trade privacy for services and enter into a contractual relationship with Internet services. Thus, as opposed to the assumption of an unlawful attack, privacy is a subject matter that is often based on lawful contractual relationships (and legal proceedings if the government is involved). Accordingly, assuming unlawful breaches of privacy by importing security threat models would oftentimes result in inept qualifications of privacy relationships.

Third, there is a substantial disconnect between the privacy ideal envisioned by the law and the actual privacy standards that users are experiencing. For example, what is written in privacy policies is often not an accurate description of the data practices occurring in reality. It was shown [44] that software developers are often unaware of their obligations and do not spend sufficient time to bring their software in conformity with the law (often unintentionally). Therefore, I intend to address the disconnect between written and actually occurring privacy practices and offer a solution for regulators as well as for software developers (Chapter 6).

## 1.6 Thesis: The Use of ML Technologies Is Essential for Improving Internet Privacy

The contributions of this dissertation provide support for the thesis that ML technologies are an essential element for advancing privacy on the Internet. First, in a case study I will demonstrate how ML can be used to detect a social network's potential to infer ethnicity and gender from its users' location data (Chapter 3). Under the privacy right, as understood here in the sense of control over information processing, an individual has control over whether others are able to determine his or her ethnicity or gender. Beyond the online inferences the demonstrated techniques can also be used to survey potential instances of discrimination and segregation in the real world. As such they illustrate that a person's online and offline privacy are often intertwined.

I will continue to show how ML can be leveraged for purposes of quantifying whether a privacy violation, which is understood to mean the non-compliance with a given privacy definition, exists (Chapter 4). Specifically, I will show how ML algorithms can be operationalized in the mosaic theory via existing privacy metrics, such as *k*–anonymity [241]. As the mosaic theory recognizes the occurrence of privacy violations on the basis of extended periods of observation, each of which by itself may not be sufficient to reach the threshold of such violation, ML provides the basis for the argument for why that is the case: the prolonged observation and consolidation of data can lead to insights that go beyond the sum of the individual observations.

Further, in order to improve privacy transparency I describe a system and its implementation to automatically analyze privacy policies (Chapter 5). Based on ML algorithms the system analyzes policy text and returns a label with the most important information allowing Internet users to gain a fast understanding of essential policy terms. In this regard, it should be noted that the automatic processing is not perfect, and mistakes can happen simply due to the nature of the approach being based on ML techniques. Making privacy policies more accessible by automatically analyzing the policy text (even it occasional mistakes) and extracting the most salient information gives users the opportunity to quickly grasp essential data practices and enhances the prevailing model of notice and choice.

The policy analysis results can be compared to actually occurring practices on websites, mobile apps or other software. I will illustrate a system implementation and its results for a large-scale study of Android apps and their corresponding policies (Chapter 6). This type of comparison enables regulators to hold software publishers accountable for their privacy practices. In fact, a custom-tailored version of the system is implemented for the California Department of Justice. The system is in the process of being evaluated as a privacy enforcement tool for the apps on the Google Play store. However, the demonstrated techniques can be also used by software developers to avoid potential privacy inconsistencies before deploying any software in the first place.

I will finally explore a rarely investigated but increasingly common practice: cross-device tracking, that is, the comprehensive tracking of Internet users on multiple devices (Chapter 7).[4] Cross-device tracking is a person-centric tracking approach as opposed to the traditional tracking of individual devices or browsers. Recognizing cross-device tracking and alerting users accordingly is becoming increasingly important since there is a surge of Internet services making use of this practice. As a recent FTC workshop revealed a fundamental lack of research on the privacy implications of cross-device tracking [115] the explorations presented here are aimed at understanding the phenomenon at a fundamental level. Thus, among others, I will explore the reach of tracking companies and the methodologies they use. These inquiries are a necessary first step for developing efficient privacy protections in the cross-device space.

[4] As data is accumulated over time the methodology introduced in Chapter 4 may be used to quantify whether someone's privacy right is violated.

# Chapter 2

# Related Work

Different strands of related work motivate my research, most notably previous studies on the ML analysis of privacy policies, program analysis of apps, human-computer interaction, crowdsourcing, and web tracking.

## 2.1 ML Privacy Policy Analysis

A core concept for notifying users of privacy practices on the Internet and obtaining their agreement to these is notice and choice. Through privacy policies and other notifications users are alerted of applicable practices. However, as McDonald et al. have shown [191], very few users read those notifications. Thus, helping users' understanding their privacy choices is a major motivation of my work. Initial work on automatic privacy policy analysis focused on making privacy policies machine-readable. That way a browser or other user agent could read the policies and alert the user of good and bad privacy practices. Reidenberg [223] suggested early on that web services should represent their policies in the Platform for Internet Content Selection (PICS) format [15]. This and similar suggestions lead to the development of P3P [71; 75], which provided a machine-readable language for specifying privacy policies and displaying their content to users [76]. To that end, the designers of P3P implemented various end users tools, such as Privacy Bird [72], a browser extension for Microsoft's Internet Explorer that notifies users of the privacy practices of a web service whose site they visit, and Privacy Bird Search [57], a P3P-enabled search engine that returns privacy policy information alongside search results.

The development of P3P was complemented by various other languages and tools. Of particular relevance was A P3P Preference Privacy Exchange Language (APPEL) [74], which enabled users to express their privacy preferences vis-à-vis web services. APPEL was further extended in the XPath project [26] and inspired the User Privacy Policy (UPP) language [27] for use in social networks. For industry use, the Platform for Enterprise Privacy Practices (E-P3P) [161] was developed allowing service providers to formulate, supervise, and enforce privacy policies. Similar languages and frameworks are the Enterprise Privacy Authorization Language (EPAL) [39], the SPARCLE Policy Workbench [54; 55], Jeeves [266], and XACML [18]. However, despite all efforts the adoption rate of P3P policies among web services remained low [16], and the P3P working group was closed in 2006 due to lack of industry participation [70].

I believe, instead of creating new machine-readable privacy policy formats it is more effective to use what is already there—privacy policies in natural language. As of now, Massey et al. [189] provided the most extensive evaluation of 2,061 of such policies, however, not focusing on their legal analysis but rather their readability and suitability for identifying privacy protections and vulnerabilities from a requirements engineering perspective. In addition, Hoke et al. [145] studied the compliance of 75 policies with self-regulatory requirements, and Cranor et al. [73] analyzed structured privacy notice forms of financial institutions identifying multiple instances of opt out practices that appear to be in violation of financial industry laws.

Different from previous studies I analyze policies automatically, on a large scale, from a legal perspective, and not limited to the financial industry. For analyzing policy content I rely on the flexibility of ML classifiers. My work is informed by the study of Costante et al., who presented a completeness classifier to determine which data practice categories are included in a privacy policy [69] and proposed rule-based techniques to extract data collection practices [68]. However, I am going beyond these works in terms of both breadth and depth. The analysis here covers a much larger policy corpus and focuses on legal questions that have not yet been automatically analyzed. Different from many existing works that focus on pre-processing of policies, e.g. by using topic modeling [65; 238] and sequence alignment [182; 219] to identify similar policy sections and paragraphs, I am interested in analyzing policy content.

## 2.2 Legal Information Extraction

Given the task of analyzing natural language policies, the question becomes how salient information can be extracted from unordered policy texts. While most works in legal information extraction relate to domains other than privacy, they still provide some guidance. For example, Westerhout et al. [256; 257] had success in combining a rule-based classifier with an ML classifier to identify legal definitions. In another line of work de Maat et al. [81; 82] aimed at distinguishing statutory provisions according to types (such as procedural rules or appendices) and patterns (such as definitions, rights, or penal provisions). They concluded that it was unnecessary to employ something more complex than a simple pattern recognizer [81; 82]. Other tasks focused on the extraction of information from statutory and regulatory laws [52; 51], the detection of legal arguments [194], or the identification of case law sections [173; 240].

There are some works in the privacy policy domain, most notably, as part of the Usable Privacy Policy Project [13; 230]. In particular, Ammar et al. presented a pilot study [32] with a focus on classifying provisions for the disclosure of information to law enforcement officials and users' rights to terminate their accounts. They concluded the feasibility of natural language analysis in the privacy policy domain in general. Wilson et al. discussed the creation and analysis of a privacy policy corpus [260]. In general, the discussed works confirm the suitability of rule and ML classifiers in the privacy policy domain. However, neither provides a comprehensive concept, nor addresses, for example, how to make use of crowdsourcing results. The latter point is especially important because, as shown in Section 5.3, automatic policy classification on its own is inherently limited. None of the previous works relieves the user from actually reading the analyzed policy. In contrast, it is the goal of the work in this dissertation to provide users with a privacy policy summary in lieu of the full policy. I want to extract from a policy essential provisions, make it more comprehensible, provide guidance on the analyzed practices, and give an overall evaluation of its privacy level.

## 2.3 Privacy Policy Crowdsourcing

There are various crowdsourcing repositories where crowd contributors evaluate the content of privacy policies and submit their results into a centralized collection for publication on the Web.

Sometimes policies are also graded. Among those repositories are ToS;DR [10], privacychoice [8], TOSBack [11], and TOSBack2 [12]. Crowdsourcing has the advantage that it combines the knowledge of a large number of contributors, which, in principle, can lead to a much more nuanced interpretation of ambiguous policy provisions than current classifiers could provide. However, all crowdsourcing approaches suffer from a lack of participation and, consequently, do not scale well. While the analysis results of the most popular websites may be available, those for many lesser known sites are not. In addition, some repositories only provide the possibility to look up the results on the web without offering convenient user access, for example, by means of a browser extension or other software.

The use of supervised ML techniques, as used suggested here, requires ground-truth. To support the development of these techniques crowdsourcing has been proposed as a viable approach for gathering rich annotations from unstructured privacy policies [230; 261]. While crowdsourcing poses challenges due to the policies' complexity [224], assigning annotation tasks to experts and setting stringent agreement thresholds and evaluation criteria [261] can in fact lead to reliable policy annotations. However, as it is a recurring problem that privacy policy annotations grapple with low inter-annotator agreement [224], I am introducing a measure for analyzing their reliability based on the notion that high annotator disagreement does not principally inhibit the use of the annotations for ML purposes as long as the disagreement is not systematic.

## 2.4 Privacy Requirement Inconsistencies

Given the inquiry into privacy policy content, I believe, it is a worthwhile task to check the extent to which policies align with actual data practices. In this regard, I find it particularly insightful to explore whether mobile apps' practices are consistent with the disclosures made in their policies and selected requirements from other laws. The legal dimension is an important one that gives meaning to the app analysis results. For example, for apps that do not provide location services the transfer of location data may appear egregious. Yet, a transfer might be permissible if adequately disclosed in a privacy policy. Only few efforts have attempted to combine code analysis of mobile apps with the analysis of privacy policies. I am seeking to fill this void by identifying privacy requirement inconsistencies connecting the analyses of apps, privacy policies, and privacy laws.

In terms of previous work, various studies, e.g., [268; 267], made inroads on creating privacy documentation or even privacy policies from program code. Other works focused on comparing program behavior with non-legal texts. For example, Huang et al. proposed AsDroid to identify contradictions between apps and user interface texts [150]. Kong et al. introduced a system to infer security and privacy related app behavior from user reviews [171]. Gorla et al. [139] used unsupervised anomaly detection techniques to analyze app store descriptions for outliers, and Watanabe et al. [255] used keyword-based binary classifiers to determine whether a resource that an app accesses (e.g., location) is mentioned in the app's description.

Different from most previous studies I analyze app behavior for compliance with privacy requirements derived from their privacy policies and selected laws. A step in this direction was provided by Bhoraskar et al., who found that 80% of ads displayed in apps targeted at children linked to pages that attempt to collect personal information in violation of the law [47]. The closest results to the effort here were presented by Enck et al. [96] and Slavin et al. [235]. In an analysis of 20 apps Enck et al. found a total of 24 potential privacy law violations caused by transmission of phone data, device identifiers, or location data. Slavin et al. proposed a system to help software developers detect potential privacy policy violations. Based on mappings of 76 policy phrases to Android API calls they discovered 341 such potential violations in 477 apps.

While my approach is inspired by TaintDroid [96] and Slavin et al.'s study [235], I move beyond their contributions. First, the privacy requirements here cover privacy questions previously not examined. Notably, different from Slavin et al., I address whether an app needs a policy and analyze the policy's own compliance (i.e., whether it describes how users are informed of policy changes and how they can access, edit, and delete data). I also analyze the collection and sharing of contact information. Second, TaintDroid, is not intended to have app store wide scale. Third, TaintDroid and Slavin et al.'s approaches do not neatly match to legal categories. They do not distinguish between first and third party practices [96; 235], do not account for negative policy statements (i.e., that an app does *not* collect certain data, as, for example, in the Snapchat policy, and base their analysis on a dichotomy of strong and weak violations [235] unknown to the law. Fourth, I introduce techniques that achieve a mean accuracy of 0.94 and a failure rate of 0.4%, which improve over the closest comparable results of 0.8 and 21% [235], respectively.

## 2.5 Mobile App Analysis

As far as the analysis on the mobile app side is concerned, different from the closest related works [96; 235], my analysis of Android apps reflects the fundamental distinction between first and third party data practices. Both have to be analyzed independently as one may be allowed while the other may not. First and third parties have separate legal relationships to a user of an app. Among the third parties, ad and analytics libraries are of particular importance. Gibler et al. found that ad libraries were responsible for 65% of the identified data sharing with the top four accounting for 43% [129]. Similarly, Demetriou et al. [85] explored their potential reach and Grace et al. [140] their security and privacy risks. They find that the most popular libraries have the biggest impact on sharing of user data, and, consequently, the analysis of sharing practices presented here focuses on those as well. In fact, 75% of apps' location requests serve the purpose of sharing it with ad networks [180].

One of my contributions lies in the extension of various app analysis techniques to achieve a meaningful analysis of apps' compliance with privacy requirements derived from their privacy policies and selected laws. The core functionality of the app analyzer in this dissertation is built on Androguard [34], a static analysis tool. In order to identify the recipients of data the system creates a call graph as described by Gibler et al. [129; 255] and uses PScout [41], which is comparable to Stowaway [121], to check whether an app has the required permissions for making a certain API call or allowing a library to make such. My work takes further ideas from FlowDroid [38], which targeted the sharing of sensitive data, its refinement in DroidSafe [138], and the ded decompiler for Android Application Packages (APKs) [97]. However, neither of the previous works is intended for large-scale privacy requirement analysis.

## 2.6 Cross-device Tracking

ML techniques are playing a central role in cross-device tracking. To explore the space Drawbridge [2], an ad network specializing in cross-device tracking, hosted the ICDM 2015: Drawbridge Cross-Device Connections competition asking researchers to leverage machine learning techniques to correlate devices to users [89]. Competition participants were given access to an anonymized proprietary dataset to train and test their features and algorithms. The competition resulted in eight short papers by some of the most successful participants [169; 176; 196; 33; 62; 163; 232; 254]. Different from the discussion in this study, these papers took the perspective of an ad network and focused exclusively on improving machine learning techniques and achieving a high F score. While these point are also part of my investigation, I am much more interested in the privacy of cross-device tracking.

The first place solution in the Drawbridge competition provided by Walthers [254], which reached an F-0.5 score of 0.9, is in many ways representative for the techniques used in the competition. As other participants' solutions [169; 176; 62], it identified IP addresses that devices of the same user were connected to as the most important feature. Intuitively, as conjectured by Cao et al. [62], devices with similar IP footprints are more likely to be used by the same individual. Thus, simply relying on IP history can already lead to an F-0.5 score of 0.86 [62]. However, various studies found that not all IP addresses are equally meaningful. In particular, because the same cellular IP addresses occur for many devices of different users they harbor less identifying potential [254; 163].

While the Drawbridge competition was about the correlation of different user devices, it did not address the purpose of the correlation: the identification of demographics, interests, and other monetizable information of the person behind the devices. Various studies exist on this topic, however, not in the context of cross-device tracking. For example, de Montjoye et al. [83] have shown that to some extent personality can be predicted from standard call detail records (CDRs), e.g., metadata about received and placed phone calls and text messages. A little closer to the effort here, Hu et al. [148] analyzed the problem of predicting Internet users' gender and age based on their browsing behaviors. They achieved an F-1 score of 0.8 for predicting gender and a score of 0.6 for categorizing users into five different age groups. In order to defend against these types of inference attacks while still allowing personalized advertisement Mor et al. [195] proposed Bloom cookies that encode a user's profile in a compact and privacy-preserving way. Recognizing the importance of IP addresses for identifying users they aim for unlinkability of all queries from the same IP. In this regard, I will explore the effect of linking devices through IP addresses on the accuracy of learning.

## 2.7 Web Tracking

Much research was published on web tracking of *individual devices*. However, to the best of my knowledge, none of the existing efforts discusses tracking *across devices*. Such tracking is notably different from traditional tracking that is focused on one device or browser. To track web users across devices companies' need first to distinguish different browser instances on the Internet. Commonly, HTTP cookies are used for this purpose. Since HTTP cookies and other traditional trackers maintain state they are widely used to track individual browsers. As Englehardt et al. [99] point out, if two websites are embedding the same tracker an adversary can link visits to those pages from the same browser instance even if the user's IP address varies. In their study they find that an adversary with the ability to passively observe web traffic on the Internet backbone can reconstruct up to 73% of a typical user's browsing history. They obtained their results using their web measurement platform OpenWPM [98], which they introduced in conjunction with a large-scale measurement of web tracking based on a crawl of a million websites.

If a browser does not accept HTTP or other cookies, it can still be tracked via browser fingerprinting, which was pointed out by Eckersley et al. [95] and extensively surveyed by Lerner et al. [178]. Web-based device fingerprinting is the process of collecting sufficient information through the browser to perform stateless device identification [21]. Such fingerprinting is also used to re-identify a browser in case cookies have been deleted. It can also be based on sensors as Das et al. [79] showed. With their FPDetective Acar et al. [21] conducted a large-scale study of web-based device fingerprinting. Panchenko et al. [208] and Hayes and Danezis [143] discussed fingerprinting attacks; Cai et al. [58; 59] explored defenses. Juarez et al [157] showed that user's browsing habits and other environment variables have a significant impact on the efficacy of the web fingerprinting attack. In this regard, three advanced web tracking mechanisms—canvas fingerprinting, evercookies, and use of cookie syncing—were explored by Acar et al. [20]. From a legal perspective it would be interesting to research the extent to which the government could use web tracking technologies—whether based on fingerprinting or traditional mechanisms—to track users without a warrant across government sites. Here I am now exploring the extent to which fingerprinting plays a role in cross-device tracking, particularly, examining the BlueCava library, which was a prominent part of Nikiforakis et al.'s work [201] on investigating the practices of three popular browser-fingerprinting companies.

In a very interesting contribution Olejnik et al. [202] reported a significant rate of stability in browser history footprints. They posit that it is not simple to change one's browsing habit. Their results show that for 69% of users the browsing history is unique and that users for whom they could detect at least four visited websites were uniquely identifiable by their histories in 97% of cases. They ponder: if web browsing patterns were unique for a given user, history analysis could potentially identify the same user across multiple browsers and devices. I want to address this question. There are also other differences among users that could single them out. Based on usage traces from 255 users of two different smartphone platforms with 7-28 weeks of data per user Falaki et al. [105] found, for example, that the mean number of interactions per day for a user varies from 10 to 200, that the mean interaction length varies from 10 to 250 seconds, and that the number of applications used varies from 10 to 90. This is especially noteworthy as Eubank et al. [100] found that the top third-party domains across different categories of devices are substantially similar. They found only few mobile-specific ad networks leading to very similar lists of top desktop and mobile third-party trackers.

## 2.8 Human-Computer Interaction

While I am not aware of any web tracking study investigating how users are tracked across devices, there are various studies on human-computer interaction that provide valuable clues how it might work. The goal of these studies is to improve website navigation, browser prediction of user destinations, and search result relevance for search engines [23]. To that end, some of these studies focus on website revisit patterns highlighting the identifying potential of such revisits. Tauscher and Greenberg [242] found that 58% of visited websites of a user constitute revisits. People tend to access only a few pages frequently and browse in small clusters of related pages. Adar et al's [23] analysis reveals various patterns of revisit, each with unique behavioral, content, and structural characteristics. They find that a five week period is sufficient to capture a wide variety of revisit patterns, although, it lacks seasonal or yearly patterns.

Some studies took a closer look at website revisits across devices. Tossell et al. [244] were able to detect that revisits occurred very infrequently with approximately 25% of URLs revisited by each user. They further find that, compared to desktops, mobile browsers are accessed less frequently, for shorter durations, and to visit fewer pages. Users' seem to rely on apps instead. Different from websites, apps have a revisit rate of 97.1% driven by a high number of visits to the five most frequently accessed apps. It appears that mobile web use is more concentrated and narrow than its desktop counterpart. Indeed, Kamvar et al.'s study [159] confirms this conjecture for the use of web search. However, interestingly enough search behavior on high-end phones resembles computer-based search behavior more than mobile search behavior.

In their quest for improving the sharing of bookmarks, URLs, and other web information between devices Kane et al. [160] found that users tend to visit many of the same domains on both their mobile device and desktop. Specifically, they found that a median of 75.4% of the domains viewed on the phone were also viewed on the desktop, and a median of 13.1% of the domains viewed on the desktop were also viewed on the phone. Despite the differing browsing habits across devices, particularly, the higher number of web sites visited on desktops, they conclude that users' web browsing activities are similar across devices. However, users do not use all of their devices in the same way but rather assign them different roles, as Dearman and Pierce [84] found. They also point out that associating a user's activities with a particular device is problematic because many activities span multiple devices.

Human-computer interaction also plays a role for the notifying users on privacy practices, notably, the automatic privacy policy analysis. However, whether the analysis of a privacy policy is based on crowdsourcing or automatic classifications, in order to notify users of the applicable privacy practices it is not enough to analyze policy content, but rather the results must also be presented in a comprehensible, preferably, standardized format [199]. In this sense, usable privacy is orthogonal to the other related areas: no matter how the policies are analyzed, a concise, user-friendly notification is always desirable. In particular, privacy labels may help to succinctly display privacy practices [164; 165; 167; 221; 222]. Also, privacy icons, such as those proposed by PrimeLife [123; 146], KnowPrivacy [16], and the Privacy Icons project [7], can provide visual clues to users. However, care must be taken that the meaning of the icons is clear to the users [146]. As of today there is no standard set of privacy labels, and, consequently, their recognizability remains problematic.

## 2.9 Quantifying Privacy

Various efforts were undertaken to measure privacy. While most of the introduced metrics were developed to quantify privacy in databases, they are also used for anonymizing users in web services and hiding demographic characteristics or other traits of a person. However, different from the approach I am taking here, all existing methods for quantifying privacy, most notably, differential privacy [93] and *k*–anonymity [241], assume an understanding of privacy that is void of ML. Their underlying assumption is that only the *direct* identification of a person or his or her characteristics is privacy-relevant. In other words, the privacy of an individual is considered violated only based on firsthand leakage of information. However, this view of privacy is incomplete since the ability to *learn* sensitive information from apparently innocuous information is a surreptitious and sensitive action that can be equally privacy-invasive.

Shortly after it was introduced by Sweeny [241] *k*-anonymity became the starting point for a whole family of privacy metrics that built upon and extended it. Similar to *k*-anonymity, *l*-diversity was originally proposed to protect the identity of individuals in databases [184]. It is founded on the observation that while *k*-anonymity prevents the disclosure of identities, it does not prevent the disclosure of sensitive attributes, such as height, eye color, ethnicity, or other quasi-identifiers of a person [184]. Beyond $k$-anonymity, $l$-diversity and its progeny, one of the most influential recent privacy metrics is differential privacy, which was introduced by Dwork [94]. Comparable to $k$-anonymity and $l$-diversity differential privacy does not take into account that undisclosed sensitive information can be learned from other information that is available. While shy from a complete solution to this problem, I will address the expansion of existing privacy metrics by incorporating the ML element.

# Chapter 3

# Case Study: Detecting Potential Ethnicity and Gender Inferences

In the following case study I will show how ML can be used to identify an Internet service's potential to infer ethnicity and gender from user-submitted data and how the collected datasets can be used to survey real-world segregation and potential discrimination.[1] As the inference of ethnicity and gender can have discriminatory impact it is important to provide users with transparency on a service's capabilities. It is crucial that the technique works from *outside* the service and allows an estimate of its capabilities *before* user sign-up. While the presented study illustrates the technique for Instagram, it is generalizable to other services as well.[2]

## 3.1 From Redlining to Big Data Discrimination

The disclosure or inference of someone's ethnicity or gender as discussed in this study can have substantial negative impact. Ethnic and gender discrimination has a long history in many coun-

[1] In the following ethnicity is meant to also encompasses race (both in the terminology of the United States Census 2010 [247]). It should be further noted that while the inference of sensitive data can impact privacy, the discriminatory use of such data does not fall under traditional notions of privacy but is rather protected by other rights.

[2] It should be noted that the results presented here are not meant to imply that Instagram is in fact engaging in any discriminatory practice. Also, it might be the case that Instagram is already aware of the ethnicity and gender for many of its users because of respective data they submitted directly. However, there are other services that might not be, and the technique introduced here is, in principle, equally applicable to those.

tries. The redlining of neighborhoods based on ethnicity in the U.S. in the 1930s for purposes of finding solvent mortgage debtors might be the earliest occurrence of data-driven discrimination [237]. It took the civil rights movement three decades later to clearly enunciate the problem and take on the struggle to end such practices. However, there are still improvements to be made. Instances of discrimination continue to happen, for example, in the gentrifying neighborhoods of New York City [130]. However, these types of redlining, whether gone-by or current, are not the only ones. Redlining can also occur online. The FTC recently explored this phenomenon in a public workshop posing the question whether big data is a tool for inclusion or exclusion [112].

The FTC reasoned: if ML technologies are used to predict that certain consumers would not be suitable candidates for prime credit offers, educational opportunities, or certain lucrative jobs, such educational opportunities, employment, and credit may never be offered to them [119]. The effect is equally bad, whether online or offline. Consequently, the FTC vowed to raise awareness about big data practices that harbor the potential for detrimental impact on underserved populations and wants to promote the use of technologies to make positive impact on those. The challenge is to enable the use of big data by companies in a way that benefits them and society, while minimizing legal and ethical risks [119]. Against this background, tools and systems for identifying and preventing online redlining and discrimination are of equal importance as its offline counterparts, and both are often intertwined.

## 3.2 Methodology and Data

This section will introduce the methodology and data used. The dataset is based on user profiles collected from the Instagram photo sharing network. As many Instagram photos are tagged with GPS latitude-longitude locations the accumulated location data can build up to comprehensive mobility profiles.[3] Based on this insight and given that many user profiles on social networks are publicly accessible it is possible to generalize the used technique and construct a dataset from readily available data as follows:

1. Public user profiles of a photo sharing service are crawled and photo metadata are extracted

[3] Our exploitation of GPS tags demonstrates an easy defense for the type of inferences presented here. If users do not tag their photos it would be much more difficult to track their locations.

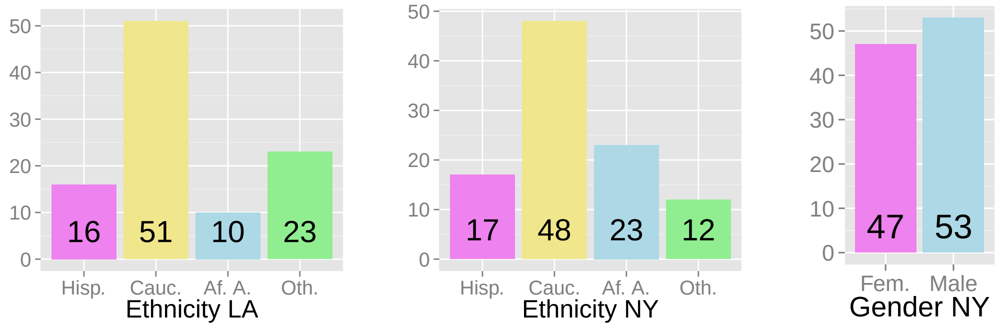
into a database.

2. Corresponding photos are labeled (with labels for ethnicity, gender, etc.) by crowd workers in an online labor marketplace.
3. The dataset is further enhanced with auxiliary data, e.g., with information that a certain location is close to a men's store.
4. Thereafter, the data can be used to analyze attributes on various demographic levels or train and test classifiers for individual inferences.

Based on the described methodology publicly available data from Instagram were collected and supplemented by Foursquare data (Instagram dataset). Specifically, the data was obtained by crawling Instagram from a root user and following further users subsequently through comments and likes. This approach biases results towards more active users. The crawl retrieved a total of 35,307,441 photo location points belonging to 118,374 unique users; users who did not have any geotags in their first 45 photos were skipped. Crowd workers then annotated users' ethnicities and genders based on the users' photos. Those photos often show the users as studies confirmed that 91% of teens post a photo of themselves on social networks [186] and that 46.6% of photos are either selfies or show the user posing with other friends [149]. However, given that an earlier study also identified 20% of Twitter profile photos as showing persons not associated with the accounts [212], annotators were instructed to disregard accounts for businesses, celebrities, and others where they had doubt about the identity of the account owner. They were also asked to make use of any tagged names to identify the account owners.

To match previous studies [155; 153; 154] annotations were obtained for the Los Angeles (LA) and New York City (NY) metropolitan areas. A user's home was specified by the ZIP code where the user had the most of his or her checkins, which are defined as Instagram latitude-longitude photo geotags. Each user was labeled by two annotators. In cases of disagreement a third annotator assigned an additional label to break the tie. In order to measure the quality of agreement Krippendorff's $\alpha$ [172] was used. Generally, values above 0.8 are considered as good agreement, values between 0.67 and 0.8 as fair agreement, and values below 0.67 as dubious [187].

The label categories are based on the categories of the United States Census 2010 (Census) [247]. More specifically, the ethnicity labels are based on the Census' Hispanic or Latino and Race categories, that is, each user is categorized either as Hispanic or Latino (Hispanic), White



| | *Ethnicity LA* | *Ethnicity NY* | *Gender NY* |
|---|---|---|---|
| Users (*n*) | 427 | 588 | 241 |
| Krippendorff's $\alpha$ Multi. | **0.74** | **0.68** | - |
| Krippendorff's $\alpha$ Binary | **0.78** | **0.74** | **0.85** |

***Figure 3.1:*** *Annotations for LA and NY. Top: percentages of user labels for the different categories. Bottom: total number of labeled users and annotation agreement results.*

alone (Caucasian), Black or African American alone (African American), or Other (combining all remaining Census categories). Just as the Census categories, the Hispanic category defined here includes Hispanics and Latinos of any race while the remaining categories do not include any Hispanics or Latinos. For the binary ethnicity categorization Caucasians are compared against all other categories taken together. Auxiliary information was added for each checkin, whenever available, in form of Foursquare's average venue popularity and venue category to estimate the types of places a user would visit. Figure 3.1 shows summary statistics for the labeled data. It also shows that agreement was at least fair and, thus, reliable ground truth for both ethnicity and gender classifications.

## 3.3 Mobility Patterns

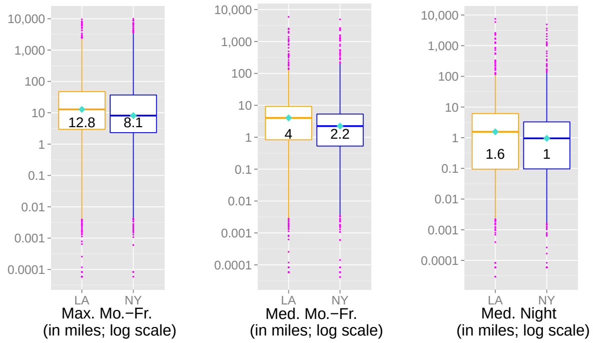
The introduced technique can be leveraged to create datasets for use in lieu of proprietary CDR datasets, for example, those analyzed by Isaacman et al. [154; 155]. As I will demonstrate, both contain similar mobility patterns. However, in order to make an adequate comparison of the mobility patterns of the Instagram dataset to those in the CDR dataset of Isaacman et al. I only

| | *Spring* | | *Winter* | |
|---|---|---|---|---|
| *Statistic* | LA | NY | LA | NY |
| Total Checkins<br>(Total CDRs) | 135,503<br>(74M) | 109,506<br>(62M) | 118,446<br>(247M) | 98,286<br>(161M) |
| Minimum Location/Day | 1 | 1 | 1 | 1 |
| 1st Quartile Location/Day | 1 | 1 | 1 | 1 |
| Median Location/Day<br>(Median Calls/Day)<br>(Median Texts/Day) | **1**<br>**(9)**<br>**-** | **1**<br>**(10)**<br>**-** | **1**<br>**(8)**<br>**(4)** | **1**<br>**(9)**<br>**(3)** |
| Mean Location/Day | 1.97 | 2.12 | 1.96 | 2.1 |
| 3rd Quartile Location/Day | 2 | 2 | 2 | 2 |
| Maximum Location/Day | 73 | 62 | 98 | 69 |

***Table 3.1:*** *Statistics of the LA and NY Spring and Winter subsets compared to the CDR dataset in [154] (where available, in parentheses). Calculations for the LA and NY subsets do not consider any day where a user had no checkins.*

consider checkins for the years 2011 through 2013 each for the Spring months from March 15 to May 15 and for the Winter months from November 15 to January 31 (the LA and NY Spring and Winter subsets, respectively). As it turns out, the mobility traces from the subsets are much more sparse. Most notably, while the CDR dataset has at least eight location points from call activity per day for the median user in LA and NY—and even 12 if text messages are added—the data in all of the Instagram subsets account for only one location point for the median user per day. Table 3.1 shows the distribution of the data in the subsets compared to those in the CDR dataset [154].

Another insightful metric for comparing mobility patterns is the *daily range*, defined as the maximum straight line distance a phone has traveled in a single day [155]. Daily ranges are characteristic for mobility because, for example, median daily ranges on weekdays represent a lower bound for a commute between home and work locations [155]. The maximum range (Max. Mo.–Fr.) is a user's longest distance and the median range (Med. Mo.–Fr.) a user's median distance, each taken on a single day for the entire Spring subset on a weekday [155]. The median range at night (Med. Night) represents the median distance a user has traveled on a day for the entire combined Spring and Fall subset from 7pm–7am [154]. Previous results [154; 155] are



| | *Max. Mo.–Fr.* | | *Med. Mo.–Fr.* | | *Med. Night* | |
|---|---|---|---|---|---|---|
| % | LA | NY | LA | NY | LA | NY |
| 98 | 2,471.7<br>(2,467) | 3,625.6<br>(2,455) | 133<br>(32) | 209.9<br>(29) | 117.4<br>(23.1) | 129.9<br>(19.4) |
| 75 | 47.5<br>(130) | 37<br>(111) | 9.3<br>(10) | 5.3<br>(8.2) | 6.1<br>(8) | 3.3<br>(5.6) |
| 50 | **12.8**<br>(36) | **8.1**<br>(27) | **4**<br>(5) | **2.2**<br>(3.8) | **1.6**<br>(4) | **1**<br>(2.6) |
| 25 | 3<br>(17) | 2.3<br>(12) | 0.8<br>(2) | 0.5<br>(1.3) | 0.1<br>(1.4) | 0.1<br>(0.7) |
| 02 | $\epsilon$<br>(1.6) | $\epsilon$<br>(1.3) | $\epsilon$<br>(0) | $\epsilon$<br>(0) | $\epsilon$<br>(0) | $\epsilon$<br>(0) |

***Figure 3.2:*** *Daily ranges in miles. Top: boxes show the 25th, 50th, and 75th percentiles; whiskers the 2nd and 98th percentiles. Bottom: table with the percentiles represented in the boxplots.*

shown in parentheses. Calculations do not consider any day where a user had a zero range, that is, multiple checkins at the same location or a single checkin only. It is defined $\epsilon < 0.005$ miles. The measured ranges are generally smaller than those reported by [154; 155]. However, the general trends in both datasets are similar. Most importantly, people in LA have generally greater ranges than people in NY. Also, in both areas people tend to travel longer during the day than at night. However, there are also differences: according to the Instagram data New Yorkers in the 98th percentiles travel farther than Angelinos. Figure 3.2 shows a subset of results for the Instagram dataset.

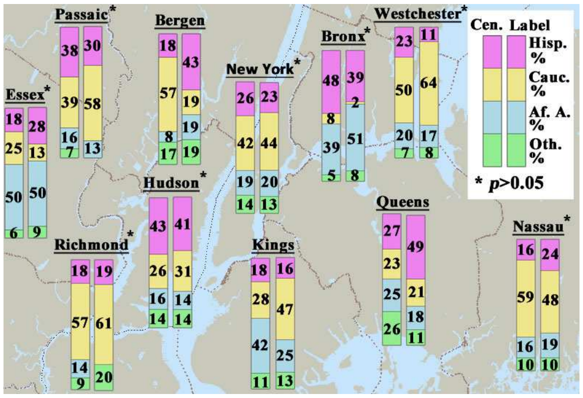
## 3.4 Demographic Patterns

The labeled Instagram data can be used to derive demographic patterns. In the following I discuss the adjustments that have to be made for the labels to be reliable.

**Adjusting Labels.** As the LA and NY subsets are annotated with ethnicity and gender labels it is possible to compare the resulting demographic distributions to the respective Census distributions. However, initial comparisons reveal substantial differences, which could be based on selection bias. For example, according to the Census there are more female than male residents (53% vs. 47%) living in Kings County [247] while the observed label frequencies suggest that there should be substantially fewer (43% vs. 57%). This result is even more surprising as the gender-specific usage rates of Internet (70% vs. 69%) [122] and Instagram (16% vs. 10%) [91] should further increase the percentage of women beyond the Census. However, while 86% of women social network account owners set their profile to private, only 74% of men do so [185]. Adjusting the Census distribution for this difference (as well as for gender-specific Internet and Instagram usage rates) leads to a distribution of females and males (49% vs. 51%) much closer to the distribution observed for the labeled data. Because the various differences are well known the adjustment to the Census distribution is more likely to represent the true population without having a skewed view through the idiosyncrasies of Instagram.

Similarly as for gender, I make adjustments for the varying percentages of Internet and Instagram usage rates among different ethnicities. However, even then there is still a substantial Hispanic underrepresentation, which was also observed for the southwest of the United States by [193]. This phenomenon is difficult to assess, specifically, as ethnicity is not significant for setting a social network profile private [179], activity levels (posting pictures, etc.) are not lower for Hispanics [239], and annotation disagreements for labeling in the Instagram dataset are not higher when the Hispanic label is involved. However, the reason for the underrepresentation seems to be the perception of Caucasian Hispanics as Caucasian alone. In a study, six of seven Caucasian Hispanics reported that others see them as Caucasian alone [192]. Therefore, it appears that most Caucasian Hispanics were actually labeled as Caucasian (i.e., annotators agreed on an incorrect classification). Consequently, the observed label frequencies were adjusted by adding to the Hispanic labels a number of labels corresponding to the Census percentage of Caucasian Hispanics and subtracting the same number from the Caucasian labels.

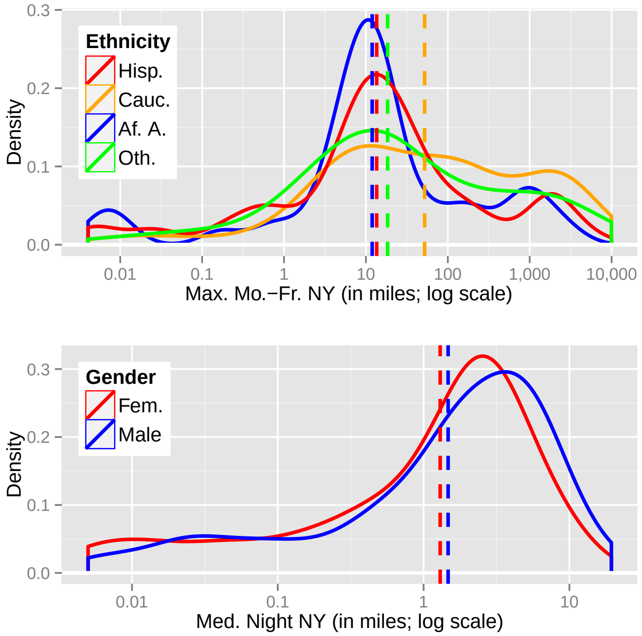
**Results.** When performing chi-square tests for goodness of fit comparing the gender and ethnicity distributions of labels to the corresponding Census distributions for different levels of gran-



| | *Ethnicity Multi-Cat.* | | *Ethnicity Binary* | | *Gender* |
|---|---|---|---|---|---|
| *Gran.* | LA | NY | LA | NY | NY |
| State | 0/1<br>(0%) | 0/1<br>(0%) | 1/1<br>(100%) | 0/1<br>(0%) | 1/1<br>(100%) |
| County | 1/2<br>(50%) | **8/11**<br>**(73%)** | 2/2<br>(100%) | 6/8<br>(75%) | 4/4<br>(100%) |
| PUMA | 12/16<br>(75%) | 11/17<br>(65%) | 2/2<br>(100%) | 5/6<br>(83%) | 1/1<br>(100%) |
| NTA | -<br>- | 9/16<br>(56%) | -<br>- | 7/7<br>(100%) | 2/2<br>(100%) |
| ZIP | 3/3<br>(100%) | 8/14<br>(57%) | 1/1<br>(100%) | 3/3<br>(100%) | -<br>- |

***Figure 3.3:*** *Chi-square goodness of fit test results for ethnicity and gender at various levels of Census-defined granularity. Top: detailed view of the multi-category ethnicity distributions for the NY county level. Left bars show the Census distributions (Cen.) and right bars the label distributions (Label) in the Instagram dataset. Bottom: complete results of the chi-square tests. NTAs are specific to NY and not available for LA.*

ularity most cases result in a value of $p > 0.05$, that is, do not present any evidence to reject the null hypothesis that the observed gender and ethnicity distributions follow the corresponding Census distributions. Figure 3.3 shows an example. For eight out of 11 counties in the NY area the tests resulted in $p > 0.05$ providing no evidence that the multi-category ethnicity distributions deviate significantly from the Census distributions. However, there are also differences. It is no surprise that this is true for the state level as the Instagram dataset only covers users from the LA and NY metropolitan areas. However, overall the results suggest that geotag data often replicates demographic trends. Below the ZIP code and NTA levels there was not enough data to perform chi-square tests. The recommendation by [227] is followed requiring the average expected fre-



| | *Max. Mo.-Fr. NY* | | | | *Med. Night NY* | |
|---|---|---|---|---|---|---|
| % | Hisp. | Cauc. | Af. A. | Oth. | Fem. | Male |
| 98 | 2,480.8 | 6,509.4 | 2,270.9 | 6,788.1 | 9.8 | 11.5 |
| 75 | 50.8 | 592.3 | 44 | 187 | 3.2 | 4.7 |
| 50 | **13.5** | **52.1** | **11.9** | **18.4** | **1.8** | **1.9** |
| 25 | 4.9 | 7 | 5.5 | 3.7 | 0.4 | 0.6 |
| 02 | ϵ | ϵ | ϵ | ϵ | ϵ | ϵ |

***Figure 3.4:*** *Daily ranges in miles. Top: density plot of the maximum daily ranges by ethnicity. Middle: density plot of the median daily ranges at night by gender. Bottom: table with the percentiles of the daily ranges represented in the plots.*

quency for a chi-square test with more than one degree of freedom to be at least two and for a test with one degree of freedom to be at least 7.5. To prevent skewing due to small sample sizes a Monte Carlo simulation with 2,000 replicates was used as well.

## 3.5 Differences in Moving Patterns by Ethnicity and Gender

Combining the previous methodologies of evaluating demographic and mobility patterns reveal that there are differences in how ethnic groups (and men and women) move. In particular, differences can be observed in daily ranges, home ranges, and temporal characteristics.

**Daily Ranges.** Figure 3.4 shows some of the daily range results for ethnic groups and genders based on sets of labeled users for LA and NY. These are the same types of daily ranges as described earlier in Figure 3.2, however, this time for all days of the year. I rounded small daily ranges up to 0.005 miles. Calculations do not consider any day where a user had a zero range, that is, had multiple checkins at the same location or a single checkin only. It is defined that $\epsilon < 0.005$ miles. Strikingly, Caucasians generally have a higher maximum daily range than the other ethnic groups. Indeed, a two sample Kolmogorov-Smirnov test reveals that the Caucasian range distribution differs significantly ($p < 0.05$) from the African American and Hispanic distribution. This result illustrates a more general finding: daily ranges of Caucasians often differ significantly from those of minorities. For 44% (8/18) of the comparisons of a Caucasian distribution to a minority distribution (three comparisons for maximum weekday, three for median weekday, three for median at night—each for LA and NY) the difference is significant at the 0.05 level. However, for the comparisons among minority distributions only 6% (1/18) are significantly different from each other.

The differences in ranges by ethnicity can be most prominently observed in the comparisons of Caucasians to African Americans and to Hispanics. However, it should be noted that at night all ethnicities exhibit very similar ranges. This finding stands in contrast to the difference in daily ranges between men and women. In fact, the only statistically significant difference ($p < 0.05$) that is observed between male and female ranges occur for the median daily ranges at night. As shown in Figure 3.4, women tend to travel smaller distances at night than men. There are many possible explanations for this phenomenon. One reason could be that women travel fewer times at night due to safety concerns [42] and, consequently, also avoid longer trips. In general, for both men and women—as well as for all ethnicities—observed daily ranges follow a (skewed) log normal distribution.

**Home Ranges.** In order to evaluate differences in mobility with respect to an individual's home location the analysis of daily ranges can be complemented with the evaluation of *home ranges*. A home range is a straight line distance between someone's home and another place to which the person travels. Figure 3.5 shows the resulting CCDFs for the home ranges of NY users. Different from daily ranges I calculate the home range not on a daily basis, but instead consider all home

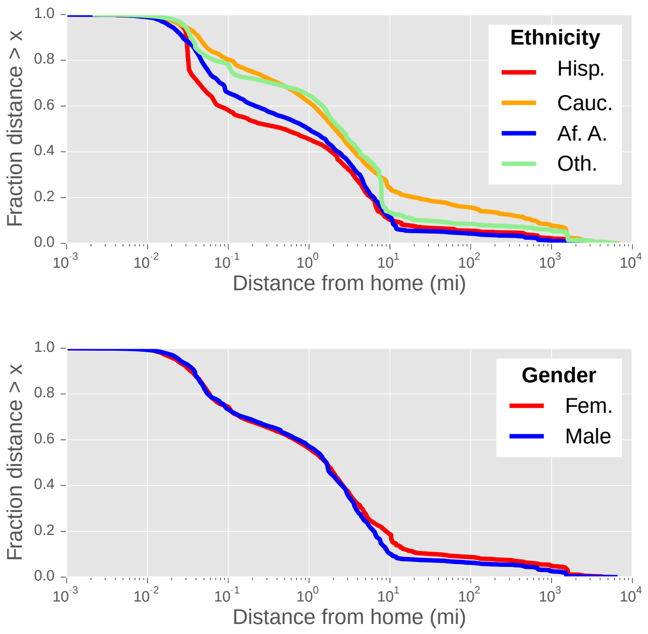


***Figure 3.5:*** *CCDFs of home ranges for NY. Top: CCDFs for different ethnic groups. Bottom: CCDFs for males and females.*

ranges—whether they were the maximum travel distance for a day or not. Based on a user's home location, that is, the ZIP code where the user had the most of his or her checkins, the distance between the home and each checkin for the different ethnic groups and genders can be calculated.

Both graphs show a noticeable decrease around the 2,500 mile mark, which is the distance from NY to major hubs on the West Coast of the United States (most notably LA (2,475 miles), San Francisco (2,563 mi), and Seattle (2,405 miles)). Men and women have very similar home ranges at the edges of the graph. However, women travel farther in the medium home ranges. This finding could be based on the fact that women were found to travel longer distances to work when they are employed full-time [175] and generally take more vacations than men [168]. It should be noted that the larger home ranges are not inconsistent with the previous observation of shorter ranges for women at night as that result does obviously not consider ranges during the day. The plot for ethnicity is in line with previous observations that Caucasians travel farther from home than minorities.

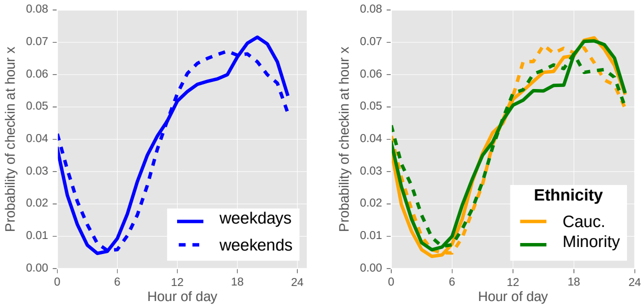


*Figure 3.6: Histograms of checkin times for NY. Left: Comparison of weekends and weekdays for all user groups. Right: Comparison of Caucasian and minority user groups for weekends and weekdays. Dashed lines correspond to weekends, solid lines to weekdays.*

**Temporal Checkin Characteristics.** Beyond spatial differences there are differences in temporal activity as well. Figure 3.6 shows histograms for checkins by hour of day. As might be expected, periodic behaviors with low checkin levels between 4–6am and peak levels from 3–8pm exist. On weekends the lows occur at later times than on weekdays suggesting that users may wake up later on weekends. There is also a dramatic increase in activity after 5pm on weekdays, which could correspond to the time at which many users get off of work. When broken up into Caucasians and minorities, the curves are very similar except with a more pronounced weekday after-work increase for minorities. It could be the case that Caucasians work more often in flexible environments. There is no substantial difference between genders or NY and LA.

## 3.6 Ethnic Segregation

Location data are the basis for measuring residential segregation, that is, the degree to which two or more groups live separately from one another in different parts of the urban environment [190]. Trends in residential segregation characterize a group's proximity to community resources (e.g., health clinics) and its exposure to environmental and social hazards (e.g., poor water quality and crimes) [220]. In the following I demonstrate how segregation can be analyzed based on the Instagram dataset. In this sense online data can also provide insight into redlining occurring offline. In addition to *residential* segregation I also introduce and evaluate *mobility* segregation, which is the degree to which two or more groups *move* to and from different parts of an area.

Mobility segregation allows for a dynamic view of segregation, for example, in order to determine a group's ease of access to community resources away from home.

**Methodology.** Various intersecting dimensions of segregation can be distinguished [190]. Two standard measures are explored here, each for a different dimension: the interaction index measures the dimension of exposure (the extent to which minority group members are exposed to majority group members in an area [190]) and the entropy index measures the dimension of evenness (the extent to which minority group members are over- or underrepresented in an area [190]). The interaction index, $B$, can be understood as the probability of a minority group member interacting with a majority group member and is defined [259] by

$$B_{kl} = \sum (\frac{n_{ik}}{N_k})(\frac{n_{il}}{n_i}), \tag{3.1}$$

where $n_{ik}$ is the population of ethnic minority group $k$ in area $i$ (e.g., in a ZIP code area), $N_k$ is the number of persons in group $k$ in the total population of all areas, $n_{il}$ is the population of ethnic majority group $l$ in area $i$, and $n_i$ is the area population.

The entropy index has the advantage over other indices that it can be used to measure segregation for more than two groups. It is defined [259], $H$, as

$$H = \frac{H^* - \bar{H}}{H^*}, \tag{3.2}$$

where $H^*$ is the population-wide entropy defined by

$$H^* = -\sum_{k=1}^{K} P_k ln(P_k), \tag{3.3}$$

and $\bar{H}$ is the weighted average of the individual areas' entropies defined by

$$\bar{H} = -\sum_{i=1}^{I} \frac{n_i}{N} \sum_{k=1}^{K} P_{ik} ln(P_{ik}), \tag{3.4}$$

where $K$ is the number of different ethnic groups, $P_k$ is the proportion of ethnicity $k$ in the total population, $I$ is the number of different areas, $n_i$ is the population in an area, $N$ is the sum of the population from all areas, and $P_{ik}$ is the proportion of the population of ethnicity $k$ in area $i$ (while it is defined that $P_{ik}ln(P_{ik}) = 0$ for $P_{ik} = 0$).

| | *Hisp./Cauc.* | | *Af. A./Cauc.* | | *Oth./Cauc.* | |
|---|---|---|---|---|---|---|
| *Gran.* | LA | NY | LA | NY | LA | NY |
| County | 0.29<br>(-2%) | 0.34<br>(+2%) | 0.27<br>(+1%) | 0.3<br>(-2%) | 0.3<br>(-3%) | 0.4<br>(0%) |
| PUMA | 0.32<br>(-6%) | **0.39**<br>**(+3%)** | 0.43<br>(+4%) | 0.42<br>(+7%) | 0.31<br>(-10%) | 0.49<br>(+5%) |
| NTA | -<br>- | 0.54<br>(+6%) | -<br>- | 0.43<br>(+3%) | -<br>- | 0.55<br>(+7%) |
| ZIP | 0.36<br>(-19%) | 0.56<br>(0%) | 0.33<br>(-23%) | 0.55<br>(+1%) | 0.58<br>(-1%) | 0.5<br>(-7%) |
| ∅ % Diff. | **5%** | | **6%** | | **5%** | |

***Table 3.2:*** *Interaction index ($B$) for different granularities based on labeled Instagram data. Differences to the interaction index calculated from Census data are shown in percentage points in parenthesis. For example, the probability of a Hispanic person to interact with a Caucasian person on the PUMA granularity level for NY is 39%. However, as shown in parenthesis, this result is an overestimation by three percentage points over the Census distribution probability of 36%. The last row of the table shows the mean difference between the labels and the Census for the three different ethnicities in absolute percentage points for both LA and NY together. Note that NTAs are not available for LA and that I also did not analyze the state level as the label and Census distributions differ significantly (Figure 3.3).*

For both interaction and entropy indices I make use of the sets of labeled users for LA and NY, however, exclude all areas for which the label distribution deviated significantly from the Census distribution as indicated by $p \leq 0.05$. Thus, for example, as shown in Figure 3.3, on the county level I do not include Queens, Kings, and Bergen. These exclusions are necessary as otherwise the accuracy of results decreases substantially. Recall that a user's home is defined as the ZIP code where he or she had the most checkins and that labels are adjusted per Census distributions (§3.4).

**Residential Segregation.** For the most part the interaction between Caucasian and minority group members can be considered fairly high [151]. All three minorities in LA and NY have similar probabilities of interacting with Caucasians. The measurement errors of 5% (Hisp./Cauc.

| | *Entropy* | | | | |
|---|---|---|---|---|---|
| *Metro* | County | PUMA | NTA | ZIP | ∅ % Diff. |
| LA | 0.01<br>(-2%) | 0.15<br>(+8%) | -<br>- | 0.15<br>(+9%) | **3%** |
| NY | 0.08<br>(0%) | 0.14<br>(+1%) | 0.08<br>(0%) | 0.09<br>(+4%) | |

***Table 3.3:*** *Entropy index ($H$) for different granularities based on labeled Instagram data. Differences to the entropy index calculated from Census data are shown in percentage points in parenthesis. As explained in Table 3.2, the last column shows the measurement error. As further explained in Table 3.2, I did not consider NTA (LA) and state granularities (LA and NY).*

and Oth./Cauc.) and 6% (Af. A./Cauc.) between labeled data and the Census suggest that the results are overall reliable. The inaccurate results for LA on the ZIP code level appear to have been caused by the smaller number of data points. While the level of interaction seems to increase when areas become more fine-grained, this phenomenon seems to be caused by the different area coverage for the various granularities. For example, it is not present when considering all NY city areas, where the Census distributions for the interaction of African Americans and Caucasians are: 0.41 (County), 0.25 (PUMA), 0.2 (NTA), and 0.22 (ZIP). Tables 3.2 and 3.3 show results for the interaction and entropy indices, respectively.

With entropy index scores ranging from 0.01 to 0.15, as shown in Table 3.3, I find another indicator for low segregation [151]. However, it should be noted that this low level of segregation is a characteristic of the particular areas investigated. For example, for all NY city areas at the NTA level I calculated an entropy of 0.31 indicating higher segregation. However, with mean differences of 5% (Hisp./Cauc.) and 6% (Af. A./Cauc. and Hisp./Oth.) between the results for the labeled data and the Census-based calculation the findings are generally reliable. As in the case of interaction, any existing inaccuracies could be due to small numbers of data points.

**Mobility Segregation.** I evaluate mobility segregation based on the same measures as residential segregation—interaction and entropy indices. However, instead of using home locations I leverage checkin data. More specifically, for each user I calculate the percentage that he or she spent at a

| | *Interaction* | | | *Entropy* |
|---|---|---|---|---|
| *Metro* | Hisp./Cauc. | Af. A./Cauc. | Oth./Cauc. | All Eth. |
| LA | 0.55<br>(+1%) | 0.57<br>(0%) | 0.58<br>(-1%) | 0.06<br>(+1%) |
| NY | 0.54<br>(-2%) | 0.53<br>(-1%) | 0.53<br>(-5%) | 0.06<br>(+2%) |
| ∅ % Diff. | **1%** | **1%** | **3%** | **1%** |

***Table 3.4:*** *Mobility interaction and entropy indices for ZIP code granularity based on labeled Instagram data. Differences to the residential interaction and entropy indices calculated from Census data are shown in percentage points in parenthesis. The last row of the table shows the mean difference between labeled and Census data in absolute percentage points for both LA and NY together.*

certain area and sum the resulting values for all users of a certain ethnicity. This method aims to avoid overcounting of active users. Results are shown in Table 3.4 and indicate that segregation levels in terms of where people go are similar to levels of where people live. Indeed, it would have been surprising to see higher segregation levels as members of minority groups may work in predominantly Caucasian areas. Furthermore, it would also have been a surprise to see lower levels of segregation as residential segregation is already relatively low.

## 3.7 Inferring Ethnicity and Gender

The distinctive mobility patterns that users of different ethnicity and gender often reveal enable prediction of those characteristics with reasonable accuracy using ML algorithms. Thus, they allow for an estimate to which extent a web service, in this case Instagram, is able to infer ethnicity and gender from its users. All following experiments were performed using scikit-learn's [211] implementations of logistic regression, decision trees, naive Bayes, and support vector machines (SVMs). The tasks are to distinguish between men and women and Caucasians and minorities. Both task are based on roughly equal class sizes.

Features falling into one of three groups were used: **general** location-based features, counts or percent of visits to each checkin; **Foursquare**-based features such as the average popularity of

| *Task* | *Parameters* | *Important Features* | *Base* | *Acc* | *AUC* | *F1* |
|---|---|---|---|---|---|---|
| Ethnicity NY | L1, $C = 0.01$ | Avg. ZIP ethnicities | 0.52 | **0.72** | **0.76** | **0.74** |
| Ethnicity LA | L1, $C = 1$ | Avg. ZIP ethnicities | 0.50 | **0.63** | **0.66** | **0.64** |
| Gender NY | L2, $C = 0.1$ | Men's Store | 0.53 | **0.58** | **0.59** | **0.55** |

***Table 3.5:*** *Results for the binary classifications of ethnicity and gender in NY and LA using logistic regression. The algorithm ran on all available features, such as counts of visits to different neighborhoods, the ethnicity of the most visited block, and the categories of nearby Foursquare venues. The baseline was obtained by predicting the class of a user based on the label distribution.*

visited venues or counts of visits to venues with certain categories; and **Census** derived features such as the average ethnic makeup of all visited locations and the ethnic makeup of a user's most-visited location. For each experiment five-fold cross validation was applied, that is, data was broken down into five groups, four of which were used for training and one for testing. After running all algorithms with all features, the best results are reported in Table 3.5.

The results suggest indeed that Instagram indeed can infer a user's ethnicity and gender from geotags. The accuracy for predicting ethnicity falls squarely within what has been reported for other types of data. On the lower bound, in their work of predicting individual Twitter users as African-American or not based on linguistic features of Tweets, [212] report as best performance an F-1 score of 0.655. On the upper bound, for predicting whether the ethnic origin of a phone user is inside or outside the United States based on a rich feature set containing Internet usage, call, text message, and location features [30] achieved an F-measure of 0.806 and for gender an F-measure of 0.611. Given that the data evaluated here contains far fewer features geotags appear surprisingly powerful in predicting ethnicity and gender.

Auxiliary information about a location derived from Foursquare or the Census may not always be available, such as in countries without publicly available census data or when locations are anonymized. Additionally, the granularity of location data can vary greatly depending on how it is created. For example, the GPS in a cell phone may have accuracy up to a few yards, while CDR data may cover several square miles. The granularity of location data is often lowered in order to increase the privacy of a dataset.

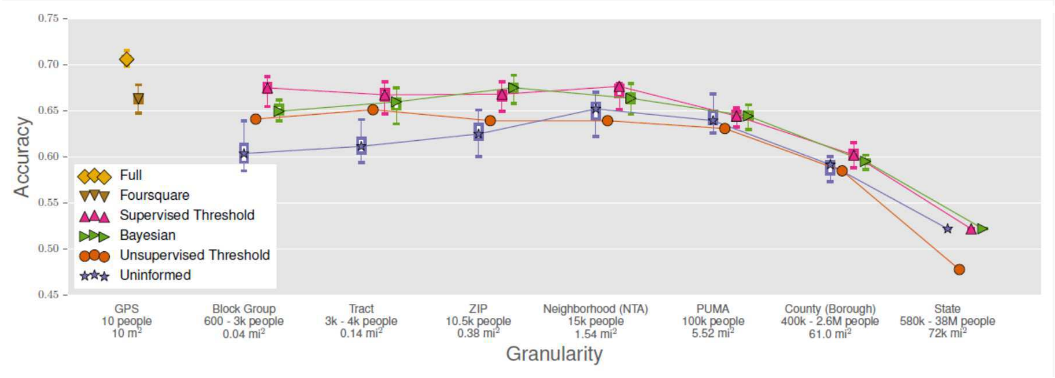


***Figure 3.7:*** *Accuracy of ethnicity prediction vs. granularity for the NY labeled data using several different inference techniques. Unsurprisingly, the Full algorithm, which uses features from Foursquare and the Census performs the best. Interestingly, however, much simpler algorithms with limited information achieve good results as well.*

In order to understand the impact of auxiliary information and granularity on the ability to make inferences, it is informative to compare the highest performing algorithm of §3.7 with algorithms that used only a subset of the Foursquare features or Census features. Additionally, to see if labeled profiles were necessary to infer ethnicity, simple decision rules that required no training were added.

Specifically, the following algorithms were tested:

- *Unsupervised Threshold*: To test if labeled data was necessary to guess ethnicity, a simple decision rule that used no labels was applied. Using Census data, I calculated the average percentage of Caucasian people living in all locations that a user visited. If this percentage was over the city's average, the algorithm predicts that the user was Caucasian. If it was under, it predicts that the user was part of a minority ethnicity.
- *Supervised Threshold*: As a point of comparison, the previously-mentioned decision rule was run again but this time it learned the threshold on a set of training data. The performance of this relative to the unsupervised threshold algorithm shows the impact of labeled data.
- *Uninformed*: The best performing algorithm (logistic regression) run on a reduced feature set of only the percentages of a user's checkins at each location serves as a lower bound on the performance of an algorithm on labeled data using only location information.

- *Bayesian*: A simple bayesian algorithm.
- *Foursquare*: Logistic regression using only the features derived from Foursquare.
- *Full*: The best performing algorithm from §3.7 which uses features derived from the U.S. Census and Foursquare. This serves as an upper bound on performance.

For all applicable algorithms, again five-fold cross validation was employed. To view the stability the process was repeated 30 times, using 30 different data partitionings into training and test sets. The results of this experiment are shown in Figure 3.7. All algorithms were ran on the dataset of NY users. To understand the impact of location granularity on prediction accuracy location data was represented at several different granularities defined by the Census ranging from block groups to states. Additionally GPS granularity was considered as well.

It can be observed that the Full algorithm achieves the best performance, as might be expected. Comparing it to the Uninformed result shows that auxiliary information provides a large performance boost. However, interestingly, many of the algorithms which only use counts of visits to areas within NY perform as well as the richer features derived from Foursquare. Another interesting result is that both the Bayesian algorithm and Uninformed algorithm perform well with the Uninformed algorithm outperforming the Unsupervised Threshold above the neighborhood granularity and the Bayesian algorithm outperforming the supervised threshold. This means that given enough labeled data of counts of visits to locations an algorithm with no auxiliary information can infer ethnicity with relative good accuracy.

The performance of all algorithms decreases at coarse granularities. This is most likely because the ethnicity distributions of larger regions are closer to the overall city distribution and provide less information. Several algorithms improve in performance at medium granularities such as ZIP and Neighborhood. This phenomenon is most likely caused by the sparsity of the dataset at the finest granularity, as many blocks are visited by only a few users. Overall, the results demonstrate the privacy implications of predicting from seemingly innocuous data demographic characteristics that might be considered sensitive.

## 3.8 Conclusion

As it is the claim of this thesis that ML is an essential technology to advance privacy on the Internet, the presented case study illustrates that ML algorithms can be used to identify a web service's potential to make ethnicity- and gender-specific inferences. The introduced technique is service-agnostic and can be leveraged for social networks beyond Instagram as well as other types of web services. In addition, the study also demonstrates that Internet privacy is often linked to offline privacy. The discussed methodology allows the study of discrimination and segregation both online and offline.

There are various extensions of the study. First, beyond ethnicity and gender, attributes such as age, occupation, and other lifestyle features may be analyzed, and naturally there are many other mobility properties to account for in addition to, for example, daily ranges. Second, better understanding the discriminative power of location data might inform the design of tools for raising user awareness about the information they reveal. This insight motivates revisiting mobility modeling and the inferences it renders possible to empower users to hide or make available their locations at will.

# Chapter 4

# Using Machine Learning to Quantify Potential Fourth Amendment Violations

As the discussed case study illustrates, social networks as well as other organizations with access to an individual's data can learn facts that the individual did not disclose directly. However, current privacy metrics are not suitable to quantify this type of privacy loss and do not translate into legal categories. To help mitigating this shortcoming I will demonstrate in the following how ML can be used to quantify privacy-invasive ML data practices, particularly, governmental practices based on extensive location surveillance. In this sense, ML enables the measurement of potential Fourth Amendment violations.

## 4.1 The Mosaic Theory

While traditionally each observation by law enforcement is treated as a discrete event that is evaluated separately for its Fourth Amendment relevance [252], ML provides a rationale to move beyond this limited view. The holistic perspective that evaluates collected data more comprehensively is known as mosaic theory.[1] ML provides a justification for the mosaic theory. At

[1] The term "mosaic theory" appears to have been first used by the Court of Appeals for the District of Columbia Circuit: "As with the 'mosaic theory' often invoked by the Government in cases involving national security information, 'What may seem trivial to the uninformed, may appear of great moment to one who has a broad view of the scene."' [253] It should be noted, though, that the mosaic theory is not (yet?) recognized by the Supreme Court.

its essence, the mosaic theory states that a set of observations about a person can create a more complete picture than the sum of individual observations. In other words, an observer can learn more than a simple tally of the collected data would suggest. This phenomenon is reflected in the increase of prediction accuracy with more data common to many ML tasks. Thus, troubling Fourth Amendment concerns emerge. As Justice Sotomayor expressed in her concurring opinion in Jones [250]:

> Disclosed in [GPS] data ... will be trips the indisputably private nature of which takes little imagination to conjure: trips to the psychiatrist, the plastic surgeon, the abortion clinic, the AIDS treatment center, the strip club, the criminal defense attorney, the by-the-hour motel, the union meeting, the mosque, synagogue or church, the gay bar and on and on."

In this sense, ML helps explain why there can be "privacy in public." The movements in public spaces can be meaningful information to learn information about an individual that can be protected by the privacy right. Furthermore, law enforcement is able to know more with considerably less effort. Thus, in addition to the increase in learning power the mosaic theory addresses the practical concern that the relative ease of data accumulation removes the economic check on abusive governmental activity that might otherwise exist. The fact that location tracking is cheap can be understood as eroding a vital bulwark of Fourth Amendment protection. While the increased efficiency in learning does not necessarily always create a Fourth Amendment violation, at some point an observer can learn disproportionately more relative to the expended effort. As Justice Alito stated in Jones, the economic aspect of automatic accumulation of data becomes increasingly troubling [251]:

> In the pre-computer age, the greatest protections of privacy were neither constitutional nor statutory, but practical. Traditional surveillance for any extended period of time was difficult and costly and therefore rarely undertaken. The surveillance at issue in this case—constant monitoring of the location of a vehicle for four weeks—would have required a large team of agents, multiple vehicles, and perhaps aerial assistance. Only an investigation of unusual importance could have justified such an expenditure of law enforcement resources.

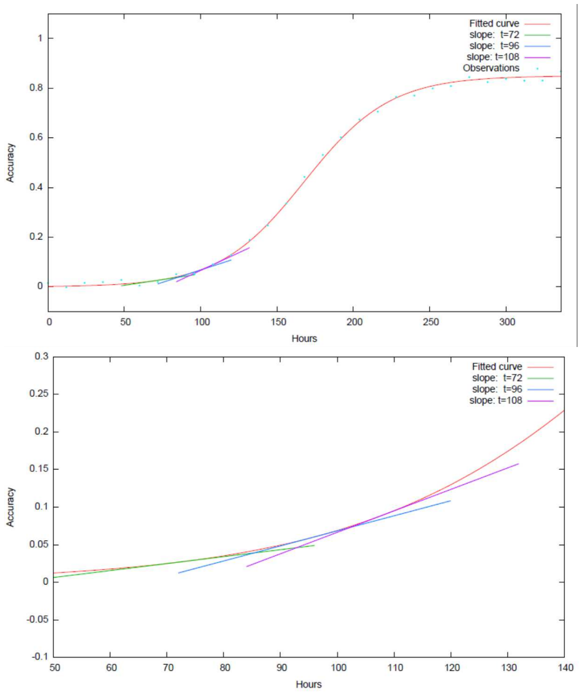


***Figure 4.1:*** *The change in slope of a graph can be used to identify at what point accuracy improves substantially given a certain quantity of input. Top: Graph with synthetic data. Bottom: Close-up of the graph with various slopes.*

The mosaic theory captures the fundamental idea that privacy can be compromised indirectly over time. Even if a particular sensitive trait of a person (such as sexual orientation) is not known to a government, the continued observation of that person disclose facts (such as visits to gay bars) that give away that trait. Individual facets of a person can turn into a much more complete mosaic of someone's character and life. Indeed, confirming Justice Sotomayor's intuition, as shown in the previous chapter, it is possible to infer an individual's ethnicity with reasonable accuracy solely based on location data. Few sparse location data points from Instagram were sufficient to identify the ethnicity for nearly three out of four people. Considering that, for example, Facebook has much more information about its users than just location, it is likely that their predictions of ethnicity [200] are even more accurate.

## 4.2 Determining the Formation of a Mosaic

The central question then is this: at what point does the tracking, aggregation, and processing of data by ML techniques arise to the quality of a search in violation of the Fourth Amendment? While it is difficult to provide a formal mathematical definition, it can be defined descriptively. Suppose we relate the amount of observations to the accuracy of a prediction, as shown in Figure 4.1, then the slope at a certain point visualizes that a given amount of data yields a certain accuracy. The change in slope, however, is what is significant here: if the slope is increasing as more data points are considered, and especially if it is increasing rapidly, the change in slope tells us that we have a better chance of learning proportionally more from later than from earlier observations. Thus, a certain threshold of a changing slope can be interpreted as the formation of a mosaic. In determining the threshold, which depends on the individual circumstances that are difficult to generalize, three aspects are particularly relevant: data granularity, quantity, and availability of auxiliary data. For example, as shown in the previous chapter for inferring ethnicity from the Instagram dataset, algorithmic performance decreased at the coarse granularities.

## 4.3 Applying Privacy Metrics

ML classifiers return probabilities for the existence of a class. Thus, their immediate results are not related to privacy. However, when used in combination with privacy metrics it becomes indeed possible to quantify privacy. To illustrate the point I focus on two well-known privacy metrics: $k$-anonymity and $l$-diversity. In $k$-anonymity the identity of a person is protected. By definition, $k$-anonymity is concerned with size $k$ of a group of people; when $k = 1$, a person is certainly identifiable. In contrast to $k$-anonymity, $l$-diversity deals with a larger set of protected attributes: quasi-identifiers. $l$-diversity generalizes $k$-anonymity in that any attribute can be specified as a quasi-identifier, and for each there must be at least $l$ possible values. However, how can $l$-diversity be mapped to the output of machine learning algorithms? In order to reconcile the two we either need to transform the ML outputs or formulate a different privacy metric in terms of probabilities. I propose the former and provide a simple rule for converting probabilities into an $l$-diverse answer: Given that a machine learning algorithm returns a probability, $p$, for the existence of an attribute, it holds that $l = \lfloor \frac{1}{p} \rfloor$.

Let us illustrate the rule by an example. If investigators believe that a suspected drug dealer driving in his car picked up a bag containing drugs in San Francisco, the machine learning algorithm may return a 40% probability for a pick-up stop in San Francisco. This result can be translated into 2-diversity. Now, why is that the case? In general, the probabilities for selecting the correct answer from two equally likely possibilities at random would be 50%, from three possibilities 33.1/3%, from four 25%, and so on. Thus, if the probability returned from the machine learning algorithm is greater than 50%, there is a higher chance of being correct when selecting this answer compared to any other answer. This can be interpreted as 1-diversity. However, if the probability returned is not greater than 50%, but greater than 33.1/3%, we have 2-diversity. If it is not greater than 33.1/3%, but greater than 25%, 3-diversity, and so on. Because in the example the probability that the suspect picked up something in San Francisco is 40%, it holds that $l = \lfloor \frac{1}{0.4} \rfloor = \lfloor 2.5 \rfloor = 2$, that is, the mapping creates 2-diversity.

The demonstrated transformation leads to another observation. Whatever question the investigators ask, it must be checked if the probability of the answer is greater than 50%. If that is the case, the corresponding answer is more likely to be correct than all others. Consequently, the prediction of an attribute (in case of $l$-diversity) or the identification of the suspect (in case of $k$-anonymity) is more likely to be successful than not and we have 1-anonymity and 1-diversity, respectively. Given such result and given that the type of information asked for is protected as well, a Fourth Amendment violation may exist. In other words, the mapping provides a rationale based on $k$-anonymity and $l$-diversity for quantifying a reasonable expectation of privacy violation at a 50% probability threshold. Thus, if either $k$-anonymity or $l$-diversity are used in the manner described, they import (and justify) a probabilistic understanding of privacy into the reasonable expectation of privacy analysis. However, in addition to the probability for the occurrence of a fact its plausibility [53] should also be considered.

## 4.4 Identifying a Privacy Violation

In order to establish a case under the mosaic theory, it is necessary to show that ML inferences can indeed violate the reasonable expectation of privacy. In other words, machine learning techniques must be used to deduce facts that are not otherwise ascertainable without violating clearly established principles, most fundamentally the privacy protections originating from the privacy of the home. The reasonable expectation of privacy of today's Fourth Amendment doctrine accommodates this notion and is explicitly couched in terms of societal expectations, i.e., what people as a whole believe is "reasonable." Consider Justice Harlan's concurrence in Katz [162]: "there is a twofold requirement, first that a person have exhibited an actual (subjective) expectation of privacy and, second, that the expectation be one that society is prepared to recognize as 'reasonable."' Societal expectations, though, are based on what is customary, and customary behavior by law enforcement is based in part on economic factors and is limited by what people will put up with. Thus, for example, visits to "the union meeting, the mosque, synagogue or church, the gay bar" [250] can be protected information under the Fourth Amendment if contemporary societal expectations consider them private. In this regard, it should be noted that sensitive information is not always protected by the Fourth Amendment. The inquiry has to be focused on the latter.

The ramifications of the quantification approach discussed here are diverse. As ML algorithms and features are increasingly used by government agencies and regulators the legal consequences of applying these technologies for purposes of investigating crimes and enforcing laws will become more prevalent. For example, if data analysis can lead to discovery of sensitive information that are protected under the Fourth Amendment, police would need to generally obtain a warrant before collecting or, at least, analyzing such data. Also, if there is a high likelihood that sensitive information can be inferred, governmental agencies cannot request from a company to turn over the user data that would enable such inferences (again, except if the agencies have a warrant or other exceptions are applicable).

## 4.5 Conclusion

As machine learning can have substantial privacy implications, it should be part of all efforts to quantify privacy. I have shown how ML can be operationalized in the mosaic theory—under which the prolonged observation of a person can lead to a violation of the reasonable expectation of privacy under the Fourth Amendment—via existing privacy metrics. In this regard, machine learning also provides a justification for the mosaic theory. However, the approach described here for measuring the degree of privacy loss is only a start. While I have shown a way to translate the output of machine learning algorithms into a legal definition of privacy via a commonly known privacy metric, it will be an important task for the future to develop a more general privacy metric that is mathematically sound, technically useful, and legally relevant. It should, for example, cover the distinction between PII and non-PII, which is a fundamental legal dichotomy.

# Chapter 5

# Automating Notice and Choice

Various technologies were proposed to mitigate the challenges for users to understand privacy notices and make their choices under the current privacy regime. However, none of them gained widespread acceptance—neither among users, nor in the industry. Most prominently, The Platform for Privacy Preferences (P3P) project [71; 75] was not widely adopted, mainly, because of a lack of incentive on part of the industry to express their policies in P3P format. In addition, P3P was also criticized for not having enough expressive power to describe privacy practices accurately and completely [70; 16]. Further, existing crowdsourcing solutions, such as Terms of Service; Didn't Read (ToS;DR) [10], do not scale well and are unlikely to gain more popularity at this point. Informed by these experiences I developed Privee—a novel software architecture for analyzing web privacy policies.

## 5.1 The Privee Concept

Figure 5.1 shows a conceptual overview of Privee, which makes use of automatic classifiers and complements them with privacy policy crowdsourcing. It integrates various components of the current web privacy ecosystem. Policy authors write their policies in natural language and do not need to adopt any special machine-readable policy format. When a user wants to analyze a privacy policy, Privee leverages the discriminative power of crowdsourcing. As we will see in Section 5.3 that classifiers and human interpretations are inherently limited by ambiguous language, it is especially important to resolve those ambiguities by providing a forum for discussion and developing

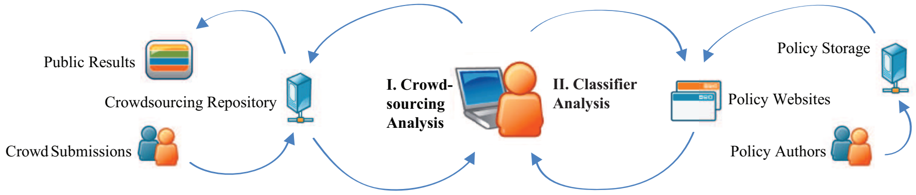


***Figure 5.1:*** *Privee system overview. When a user requests a privacy policy analysis, the system checks whether the analysis results are available at a crowdsourcing repository (to which crowd contributors can submit analysis results of policies). If results are available, they are returned and displayed to the user (I. Crowdsourcing Analysis). If no results are available, the policy text is fetched from the policy website, analyzed by automatic classifiers on the client machine, and then the analysis results are displayed to the user (II. Classifier Analysis).*

consensus among different crowd contributors. Further, Privee complements the crowdsourcing analysis with the ubiquitous applicability of rule and ML classifiers for policies that are not yet analyzed by the crowd. Because the computational requirements are low, as shown in Section 5.3.3, a real time analysis is possible.

As the P3P experience showed [70] that a large fraction of web services with P3P policies misrepresented their privacy practices, presumably in order to prevent user agents from blocking their cookies, any privacy policy analysis software must be guarded against manipulation. However, natural language approaches, such as Privee, have an advantage over P3P and other machine-readable languages. Because it is not clear whether P3P policies are legally binding [229] and the FTC never took action to enforce them [177], the misrepresentation of privacy practices in those policies is a minor risk that many web services are willing to take. This is true for other machine-readable policy solutions as well. In contrast, natural language policies can be valid contracts [1] and subject to the FTC's enforcement actions against unfair or deceptive acts or practices (15 U.S.C. §45(a)(1)). Thus, web services are more likely to ensure that their natural language policies represent their practices accurately.

When capturing privacy policy text it is crucial to do so completely and without additional text, in particular, free from advertisements on the policy website. Further, while it is true that an ill-intentioned privacy policy author might try to deliberately use ambiguous language to trick the classifier analysis, this strategy can only go so far as ambiguous contract terms are interpreted against the author (Restatement (Second) of Contracts, §206) and might also cause the FTC to challenge them as unfair or deceptive. Beyond safeguarding the classifier analysis, it is also important to prevent the manipulation of the crowdsourcing analysis. In this regard, the literature on identifying fake reviews should be brought to bear. For example, Wu et al. [262] showed that fake reviews can be identified by a suspicious grade distribution and their posting time following negative reviews. In order to ensure that the crowdsourcing analysis returns the latest results the crowdsourcing repository should also keep track of privacy policy updates.

## 5.2 The Privee Browser Extension

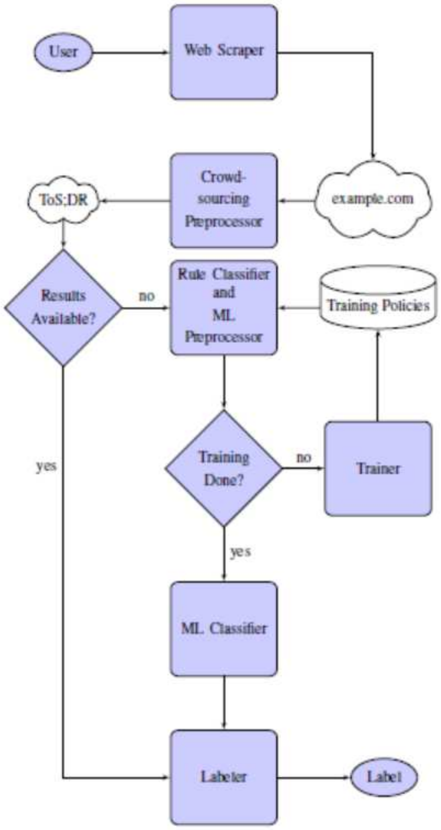


*Figure 5.2: Simplified Privee program flow.*

I implemented Privee as a proof of concept browser extension for Google Chrome. After the user has started the extension, the web scraper obtains the text of the privacy policy to be analyzed (example.com) as well as the current URL (http://example.com/). The crowdsourcing preprocessor then extracts from the URL the ToS;DR identifier and checks the ToS;DR repository for results. If results are available, they are retrieved and forwarded to the labeler, which converts them to a label for display to the user. However, if no results are available on ToS;DR the policy text is analyzed. First, the rule classifier attempts a rule-based classification. However, if that is not possible the ML preprocessor prepares the ML classification. It checks if the ML classifier is already trained. If that is the case, the policy is classified by the ML classifier, assigned a label according to the classifications, and the results are displayed to the user. Otherwise, a set of training policies is analyzed by the trainer first and the program proceeds to the ML classifier and labeler afterwards. The set of training policies is included in the extension package and only needs to be analyzed for the first run of the ML classifier. Thereafter, the training results are kept in persistent storage until deletion by the user. I wrote the Privee extension in JavaScript using the jQuery library and Ajax functions for client-server communication. While the extension is designed as an end user tool, it can also be used for research, for example, in order to easily compare different privacy policies. Figure 5.2 shows a simplified overview of the program flow. In this section I describe the various stages of program execution.

### 5.2.1 Web Scraper

The user starts the Privee extension by clicking on its icon in the Chrome toolbar. Then, the web scraper obtains the text of the privacy policy that the user wants to analyze and retrieves the URL of the user's current website. While the rule and ML classifier analysis only works from the site that contains the policy to be analyzed, the crowdsourcing analysis works on any website whose URL contains the policy's ToS;DR identifier.

### 5.2.2 Crowdsourcing Preprocessor

The crowdsourcing preprocessor is responsible for managing the interaction with the ToS;DR repository. It receives the current URL from the web scraper from which it extracts the ToS;Dr identifier. It then connects to the API of ToS;DR and checks for the availability of analysis results, that is, short descriptions of privacy practices and sometimes an overall letter grade. The results, if any, are forwarded to the labeler and displayed to the user. Then the extension terminates. Otherwise, the policy text, which the crowdsourcing preprocessor also received from the web scraper, is forwarded to the rule classifier and ML preprocessor.

### 5.2.3 Rule Classifier and ML Preprocessor

Generally, classifiers can be based on rule or ML algorithms. In preliminary experiments I found that for some classification categories a rule classifier worked better, in others an ML classifier, and in others again a combination of both [240; 257]. I will discuss classifier selection in Section 5.3.1 in more detail. This section will focus on the feature selection process for the rule classifier and ML preprocessor. Both rule classification and ML preprocessing are based on feature selection by means of regular expressions.

My preliminary experiments showed that classification performance depends strongly on feature selection. Ammar et al. [32] discuss a similar finding. Comparable to other domains [257], feature selection is particularly useful here for avoiding misclassifications due to the heavily imbalanced structure of privacy policies. For example, in many multi-page privacy policies there is often only one phrase that determines whether the web service is allowed to combine the collected information with information from third parties to create personal profiles of users. Especially, supervised ML classifiers do not work well in such cases, even with undersampling (removal of uninteresting examples) or oversampling (duplication of interesting examples) [170]. Possible solutions to the problem are the separation of policies into different content zones and applying a classifier only to relevant content zones [173] or—the approach adopted here—running a classifier only on carefully selected features.

The extension's feature selection process begins with the removal of all characters from the policy text that are not letters or whitespace and conversion of all remaining characters to lower case. However, the positions of removed punctuations are preserved because, as noted by Biagoli et al. [48], a correct analysis of the meaning of legal documents often depends on the position of punctuation. In order to identify the features that are most characteristic for a certain class I used the term frequency-inverse document frequency (tf-idf) statistic as a proxy. With tf-idf it is possible to measure how concentrated into relatively few documents the occurrences of a given word are in a document corpus [218]. Thus, words with high tf-idf values correlate strongly with the documents in which they appear and can be used to identify topics in that document that are not discussed in other documents. However, instead of using individual words as features the use of bigrams lead to better classification performance, which was also discussed in previous works [32; 194].

```
(ad|advertis.*)
    (compan.*|network.*|provider.*|servin.*|serve.*|vendor.*) |
    (behav.*|context.*|network.*|parti.*|serv.*) (ad|advertis.*)
```

***Listing 5.1:*** *Simplified pseudocode of the regular expression to identify whether a policy allows advertising tracking. For example, the regular expression would match "contextual advertising."*

The method by which the Privee extension selects characteristic bigrams, which usually consist of two words, but can also consist of a word and a punctuation mark, is based on regular expressions. It applies a three-step process that encompasses both rule classification and ML preprocessing. To give an example, for the question whether the policy allows advertising tracking (e.g., by ad cookies) the first step consists of trying to match the regular expression in Listing 5.1, which identifies bigrams that nearly always indicate that advertising tracking is allowed. If any bigram in the policy matches, no further analysis happens, and the policy is classified by the rule classifier as allowing advertising tracking. If the regular expression does not match, the second step attempts to extract further features that can be associated with advertising tracking (which are, however, more general than the previous ones). Listing 5.2 shows the regular expression used for the second step.

```
(ad|advertis|market) (.+)|(.+) (ad|advertis|market)
```

***Listing 5.2:*** *Simplified pseudocode of the regular expression to extract relevant phrases for advertising tracking. For example, the regular expression would match "no advertising."*

The second step—the ML preprocessing—is of particular importance for the analysis because it prepares classification of the most difficult cases. It extracts the features on which the ML classifier will run later. To that end, it first uses the Porter stemmer [215] to reduce words to their morphological root [48]. Such stemming has the effect that words with common semantics are clustered together [125]. For example, “collection,” “collected,” and “collect” are all stemmed into “collect.” As a side note, while stemming had some impact, there was no substantial performance increase for running the ML classifier on stemmed features compared to unstemmed features. In the third step, if no features were extracted in the two previous steps, the policy is classified as not allowing advertising tracking.

### 5.2.4 Trainer

In the training stage the Privee extension checks whether the ML classifier is already trained. If that is not the case, a corpus of training policies is preprocessed and analyzed. The analysis of a training policy is similar to the analysis of a user-selected policy, except that the extension does not check for crowdsourcing results and only applies the second and third step of the rule classifier and ML preprocessor phase. The trainer’s purpose is to gather statistical information about the features in the training corpus in order to prepare the classification of the user-selected policy. It stores the training results locally in the user’s browser memory using persistent web storage, which is, in principle, similar to cookie storage.

### 5.2.5 Training Data

The training policies are held in a database that is included in the extension package. The database holds a total of 100 training policies. In order to obtain a representative cross section of training policies, I selected the majority of policies randomly from the Alexa top 500 websites for the U.S. [28] across various domains (banking, car rental, social networking, etc.). However, a few random policies from lesser frequented U.S. sites and sites from other countries that published privacy policies in English were also included. The trainer accesses these training policies one by one and adds the training results successively to the client’s web storage. After all results are added the ML classifier is ready for classification.

### 5.2.6 ML Classifier

I now describe the ML classifier design and the classification categories.

**ML Classifier Design.** In order to test the suitability of different ML algorithms for analyzing privacy policies I performed preliminary experiments using the Weka library [142]. Performance for the different algorithms varied. I tested all algorithms available on Weka, among others the Sequential Minimal Optimization (SMO) algorithm with different kernels (linear, polynomial, radial basis function), random forest, J48 (C4.5), IBk nearest neighbor, and various Bayesian algorithms (Bernoulli naive Bayes, multinomial naive Bayes, Bayes Net). Surprisingly, the Bayesian algorithms were among the best performers. Therefore, I implemented naive Bayes in its Bernoulli and multinomial version. Because the multinomial version ultimately proved to have better performance, I settled on this algorithm.

As Manning et al. [187] observed, naive Bayes classifiers have good accuracy for many tasks and are very efficient, especially, for high-dimensional vectors, and they have the advantage that training and classification can be accomplished with one pass over the data. The naive Bayes implementation is based on their specification [187]. In general, naive Bayes classifiers make use of Bayes' theorem. The probability, $P$, of a document, $d$, being in a category, $c$, is

$$P(c|d) \propto P(c) \prod_{1 \leq k \leq n_d} P(t_k|c), \tag{5.1}$$

where $P(c)$ is the prior probability of a document occurring in category $c$, $n_d$ is the number of terms in $d$ that are used for the classification decision, and $P(t_k|c)$ is the conditional probability of term $t_k$ occurring in a document of category $c$ [187]. In other words, $P(t_k|c)$ is interpreted as a measure of how much evidence $t_k$ contributes for $c$ being the correct category [187]. The best category to select for a document in a naive Bayes classification is the category for which it holds that

$$\arg\max_{c \in \mathbb{C}} \hat{P}(c|d) = \arg\max_{c \in \mathbb{C}} \hat{P}(c) \prod_{1 \leq k \leq n_d} \hat{P}(t_k|c), \tag{5.2}$$

where $\mathbb{C}$ is a set of categories, which, in the case here, is always of size two (e.g., $\{$ad tracking, no ad tracking$\}$). The naive assumption is that the probabilities of individual terms within a document are independent of each other given the category [125]. However, the implementation here differs from the standard implementation and tries to alleviate the independence assumption. Instead of processing individual words of the policies the system tries to capture some context by processing bigrams.

Analyzing the content of a privacy policy requires multiple classification decisions. For example, the classifier has to decide whether personal information can be collected, disclosed to advertisers, retained indefinitely, and so on. This type of classification is known as multi-label classification because each analyzed document can receive more than one label. One commonly used approach for multi-label classification with $L$ labels consists of dividing the task into $|L|$ binary classification tasks [245]. However, other solutions handle multi-label data directly by extending specific learning algorithms [245]. It turned out to be simpler to implement the first approach. Specifically, at execution time multiple classifier instances are created—one for each classification category—by running the classifier on category-specific features extracted by the ML preprocessor.

**Classification Categories.** For which types of information should privacy policies actually be analyzed? In answering this question, one starting point are fair information practices [66]. Another one are the policies themselves. After all, while it is true that privacy law in the U.S. generally does not require policies to have a particular content, it can be observed that all policies conventionally touch upon four different themes: information collection, disclosure, use, and management (management refers to the handling of information, for example, whether information is encrypted). The four themes can be analyzed on different levels of abstraction. For example, for disclosure of information, it could simply be analyzed whether information is disclosed to outside parties in general, or it could be investigated more specifically whether information is disclosed to service providers, advertisers, governmental agencies, credit bureaus, and so on.

At this point it should be noted that not all information needs to be analyzed. In some instances privacy policies simply repeat mandatory law without creating any new rights or obligations. For example, a federal statute in the U.S.—18 U.S.C. §2703(c)(1)(A) and (B)—provides that the government can demand the disclosure of customer information from a web service provider after obtaining a warrant or suitable court order. As this law applies independently of a privacy policy containing an explicit statement to that end, the provision that the provider will disclose information to a governmental entity under the requirements of the law can be inferred from the law itself. In fact, even if a privacy policy states to the contrary, it should be assumed that such information disclosure will occur. Furthermore, if privacy policies stay silent on certain subject matters, default rules might apply and fill the gaps.

Another good indicator of what information should be classified is provided by user studies. According to one study [72], knowing about sharing, use, and purpose of information collection is very important to 79%, 75%, and 74% of users, respectively. Similarly, in another study [16] users showed concern for the types of personal information collected, how personal information is collected, behavioral profiling, and the purposes for which the information may be used. While it was only an issue of minor interest earlier [72], the question how long a company keeps personal information about its users is a topic of increasing importance [16]. Based on these findings, it appears advantageous to perform six different binary classifications, that is, whether or not a policy

- allows collection of personal information from users (Collection);
- provides encryption for information storage or transmission (Encryption);
- allows ad tracking by means of ad cookies or other trackers (Ad Tracking);
- restricts archiving of personal information to a limited time period (Limited Retention);
- allows the aggregation of information collected from users with information from third parties (Profiling);
- allows disclosure of personal information to advertisers (Ad Disclosure).

For purposes of the analysis, where applicable, it is assumed that the user has an account with the web service whose policy is analyzed and is participating in any offered sweepstakes or the like. Thus, for example, if a policy states that the service provider only collects personal information from registered users, the policy is analyzed from the perspective of a registered user. Also, if certain actions are dependent on the user’s consent, opt in, or opt out, it is assumed that the user consented, opted in, or did not opt out, respectively. As it was my goal to make the analysis results intuitively comprehensible to casual users, which needs to be confirmed by user studies, I tried to avoid technical terms. In particular, the term “personal information” is identical to what is known in the privacy community as personally identifiable information (while “information” on its own also encompasses non-PII, e.g., user agent information).

It is noteworthy that some of the analyzed criteria correspond to the semantics of the P3P

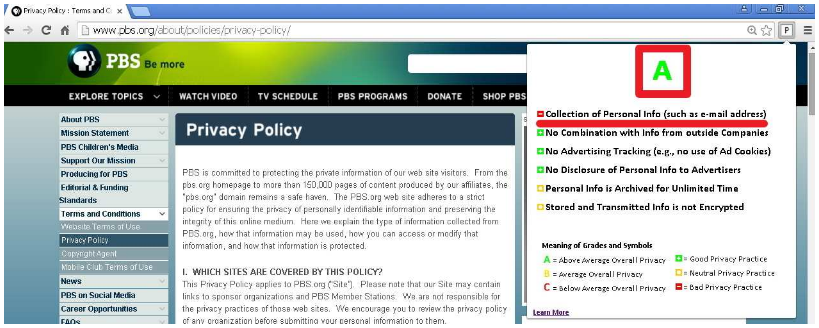
Compact Specification [5]. For example, the P3P token NOI indicates that a web service does not collect identified data while ALL means that it has access to all identified data. Thus, NOI and ALL correspond to the collection category. Also, in P3P the token IND means that information is retained for an indeterminate period of time, and, consequently, is equivalently expressed when the classifier comes to the conclusion that no limited retention exists. Further, PSA, PSD, IVA, and IVD are tokens similar to the profiling category. Generally, the correspondence between the semantics of the P3P tokens and the categories here suggests that it is possible to automatically classify natural language privacy policies to obtain the same information that web services would include in P3P policies without actually requiring them to have such.

### 5.2.7 Labeler

The extension's labeler is responsible for creating an output label. As it was shown that users casually familiar with privacy questions were able to understand privacy policies faster and more accurately when those policies were presented in a standardized format [165] and that most users had a preference for standardized labels over full policy texts [165; 166], I created a short standardized label format. Generally, a label can be structured in one or multiple dimensions. The multidimensional approach has the advantage that it can succinctly display different privacy practices for different types of information. However, one-dimensional formats, as used here, were shown to be substantially more comprehensible [167; 222].

In addition to the descriptions for the classifications, the labeler also labels each policy with an overall letter grade, which depends on the classifications. More specifically, the grade is determined by the number of points, $p$, a policy is assigned. For collection, profiling, ad tracking, and ad disclosure a policy receives one minus point, respectively. However, for not allowing one of these practices a policy receives one plus point. However, a policy receives a plus point for featuring limited retention or encryption, respectively. As most policies in the training set had zero points, zero points is the mean and grades are assigned as follows:

- A (above average overall privacy) if $p > 1$;
- B (average overall privacy) if $1 \leq p \geq -1$;
- C (below average overall privacy) if $p < -1$.



Privacy Policy

PBS is committed to protecting the private information of our web site visitors. From the pbs.org homepage to more than 150,000 pages of content produced by our affiliates, the "pbs.org" domain remains a safe haven. The PBS.org web site adheres to a strict policy for ensuring the privacy of personally identifiable information and preserving the integrity of this online medium. Here we explain the type of information collected from PBS.org, how that information may be used, how you can access or modify that information, and how that information is protected.

I. WHICH SITES ARE COVERED BY THIS POLICY?
This Privacy Policy applies to PBS.org ("Site"). Please note that our Site may contain links to sponsor organizations and PBS Member Stations. We are not responsible for the privacy practices of those web sites. We encourage you to review the privacy policy of any organization before submitting your personal information to them.

A

- 🟥 Collection of Personal Info (such as e-mail address)
- 🟩 No Combination with Info from outside Companies
- 🟩 No Advertising Tracking (e.g., no use of Ad Cookies)
- 🟩 No Disclosure of Personal Info to Advertisers
- 🟨 Personal Info is Archived for Unlimited Time
- 🟨 Stored and Transmitted Info is not Encrypted

Meaning of Grades and Symbols

| | |
|---|---|
| A = Above Average Overall Privacy | 🟩 = Good Privacy Practice |
| B = Average Overall Privacy | 🟨 = Neutral Privacy Practice |
| C = Below Average Overall Privacy | 🟥 = Bad Privacy Practice |

*Figure 5.3:* *Privee extension screenshot and detailed label view. The result of the privacy policy analysis is shown to the user in a pop-up.*

After the points are assigned to a policy, the corresponding label is displayed to the user as shown in Figure 5.3. In order to avoid confusion about the meaning of icons [146], short descriptions were used instead. The text in the pop-up is animated. If the user moves the mouse over it, further information is provided. The user can also find more detailed explanations about the categories and the grading by clicking on the blue ”Learn More” link at the bottom of the label. It should be noted that analysis results retrieved from ToS;DR usually differ in content from the classification results, and are, consequently, displayed in a different label format. The scheme introduced here should be understood as a proof of concept. There is no consensus on the selection of practices to display or the labels to use. Especially, it can be argued that a letter grading scheme incorrectly implies that the described practices are comparable, which in actuality might not be the case.

## 5.3 Experimental Results

Privee was run on a test set of 50 policies. Before this test phase the ML classifier was trained (with the 100 training policies that are included in the extension package) and tuned it (with a validation set of 50 policies). During the training, validation, and test phases the retrieval of crowdsourcing results was disabled. Consequently, the experimental results only refer to rule and ML classification. The policies of the test and validation sets were selected according to the same criteria as described for the training set in Section 5.2.5. In this section I first discuss the classification performance (Section 5.3.1), then the gold standard that I used to measure the performance (Section 5.3.2), and finally the computational performance (Section 5.3.3).

### 5.3.1 Classification Performance

In the validation phase I experimented with different classifier configurations for each of the six classification tasks. For the ad tracking and profiling categories the combination of the rule and ML classifier lead to the best results. However, for collection, limited retention, and ad disclosure the ML classifier on its own was preferable. Conversely, for the encryption category the rule classifier on its own was the best. It seems that the language used for describing encryption practices is often very specific making the rule classifier the first choice. Words such as "ssl" are very distinctive identifiers for encryption provisions. Other categories use more general language that could be used in many contexts. For example, phrases related to time periods must not necessarily refer to limited retention. For those instances the ML classifier seems to perform better. However, if categories exhibit both specific and general language the combination of the rule and ML classifier is preferable.

The results of the extension's privacy policy analysis are based on the processing of natural language. However, as natural language is often subject to different interpretations, the question becomes how the results can be verified in a meaningful way. Commonly applied metrics for verifying natural language classification tasks are accuracy (Acc.), precision (Prec.), recall (Rec.), and F-1 score (F-1). Accuracy is the fraction of classifications that are correct [187]. Precision is the fraction of retrieved documents that are relevant, and recall is the fraction of relevant documents that are retrieved [187]. Precision and recall are often combined in their harmonic mean, known as the F-1 score [147].

In order to analyze the extension's performance I calculated the accuracy, precision, recall, and F-1 score for the test policy set classifications. Table 5.1 shows the overall performance and the performance for each classification category. I also calculated the baseline accuracy (Base.) for comparison against the actual accuracy. The baseline accuracy for each category was determined by always selecting the classification corresponding to the annotation that occurred the most in the training set annotations, which I report in Figure 5.4. The baseline accuracy for the overall performance is the mean of the category baseline accuracies. Because the classification of privacy policies is a multi-label classification task I calculated the overall results based on the method for

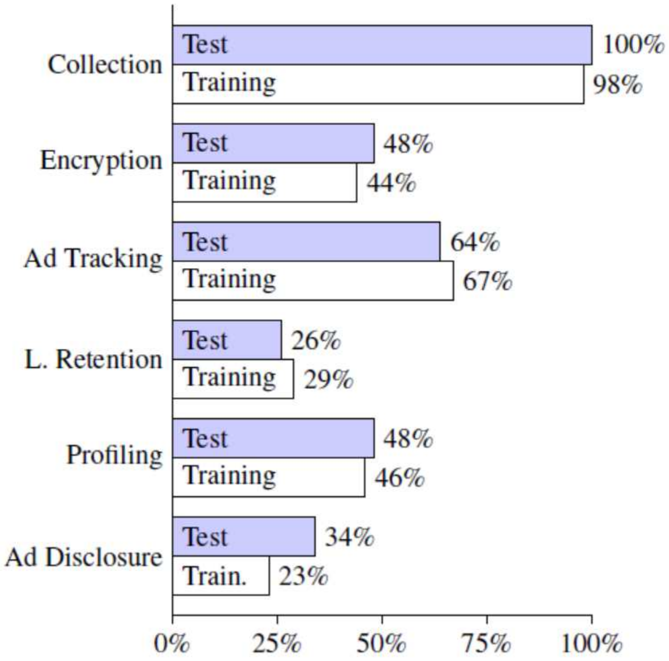


***Figure 5.4:*** *Annotation of positive cases in percent for the 50 test policies (blue) and the 100 training policies (white).*

measuring multi-label classifications given by Godbole and Sarawagi [131]. According to their method, for each document, $d_j$ in set $D$, let $t_j$ be the true set of labels and $s_j$ be the predicted set of labels. Then, the means is obtained by

$$Acc(D) = \frac{1}{|D|}\sum\nolimits_{i=1}^{|D|} \frac{|t_j \cap s_j|}{|t_j \cup s_j|}, \tag{5.3}$$

$$Prec(D) = \frac{1}{|D|}\sum\nolimits_{i=1}^{|D|} \frac{|t_j \cap s_j|}{|s_j|}, \tag{5.4}$$

$$Rec(D) = \frac{1}{|D|}\sum\nolimits_{i=1}^{|D|} \frac{|t_j \cap s_j|}{|t_j|}, \tag{5.5}$$

$$F-(D) = \frac{1}{|D|}\sum\nolimits_{i=1}^{|D|} \frac{2\,Prec(d_j)\,Rec(d_j)}{(Prec(d_j) + Rec(d_j))}. \tag{5.6}$$

From Table 5.1 it can be observed that the accuracies are at least as good as the corresponding baseline accuracies. For example, in the case of limited retention the baseline classifies all policies as not providing for limited retention because, as show in Figure 5.4, only 29% of the training policies were annotated as having a limited retention period, which would lead to a less accurate

| | Base. | Acc. | Prec. | Rec. | F-1 |
|---|---|---|---|---|---|
| Overall | **68%** | **84%** | **94%** | **89%** | **90%** |
| Collection | 100% | 100% | 100% | 100% | 100% |
| Encryption | 52% | 98% | 96% | 100% | 98% |
| Ad Tracking | 64% | 96% | 94% | 100% | 97% |
| L. Retention | 74% | 90% | 83% | 77% | 80% |
| Profiling | 52% | 86% | 100% | 71% | 83% |
| Ad Disclosure | 66% | 76% | 69% | 53% | 60% |

***Table 5.1:*** *Privee extension performance overall and per category. For the 300 test classifications (six classifications for each of the 50 test policies) I observed 27 misclassifications. 154 classifications were made by the rule classifier and 146 by the ML classifier. The rule classifier had 11 misclassifications (2 false positives and 9 false negatives) and the ML classifier had 16 misclassifications (7 false positives and 9 false negatives). It may be possible to decrease the number of false negatives by adding more rules and training examples. For the ad tracking category the rule classifier had an F-1 score of 98% and the ML classifier had an F-1 score of 94%. For the profiling category the rule classifier had an F-1 score of 100% and the ML classifier had an F-1 score of 53%. 28% of the policies received a grade of A, 50% a B, and 22% a C.*

classification of 74% in the test set compared to the actual accuracy of 90%. For the collection category it should be noted that there is a strong bias because nearly every policy allows the collection of personal information. However, in the validation set included two policies that did not allow this practice, but still were correctly classified by the extension. Generally, the F-1 performance results fall squarely within the range reported in the earlier works. For identifying law enforcement disclosures Ammar et Al. [32] achieved an F-1 score of 76% and Costante et al. reported a score of 83% for recognizing types of collected information [68] and 92% for identifying topics discussed in privacy policies [69].

In order to investigate the reasons behind the extension's performance I used two binary logistic regression models. Binary logistic regression is a statistical method for evaluating the dependence of a binary variable (the dependent variable) on one or more other variables (the independent variable(s)). In the first model each of the 50 test policies was represented by one data point with the dependent variable identifying whether it had any misclassification and the independent variables identifying (1) the policy's length in words, (2) its mean Semantic Diversity (SemD) value [144], and (3) whether there was any disagreement among the annotators in annotating the policy (Disag.). In the second model I represented each of 185 individual test classifications by one data point with the dependent variable identifying whether it was a misclassification and the independent variables identifying (1) the length (in words) of the text that the rule classifier or ML preprocessor extracted for the classification, (2) the text's mean SemD value, and (3) whether there was annotator disagreement on the annotation corresponding to the classification.

Hoffman et al.'s [144] SemD value is an ambiguity measure for words based on latent semantic analysis, that is, the similarity of contexts in which words are used. It can range from 0 (highly unambiguous) to 2.5 (highly ambiguous). I represented the semantic diversity of a document (i.e., a policy or extracted text) by the mean SemD value of its words. However, as Hoffman et al. only provide SemD values for words on which they had sufficient analytical data (31,739 different words in total), some words could not be taken into account for calculating a document's mean SemD value. Thus, in order to avoid skewing of mean SemD values only documents that had SemD values for at least 80% of their words were considered. In the first model all test policies were above this threshold. However, in the second model some of the 300 classifications were excluded. Particularly, all encryption classifications were excluded because words, such as "encryption" and "ssl" occurred often and had no SemD value. Also, in the second model the mean SemD value of an extracted text was calculated after stemming its words with the Porter stemmer and obtaining the SemD values for the resulting word stems (while the SemD value of each word stem was calculated from the mean SemD value of all words that have the respective word stem).

For the first model the analysis results are shown in Table 5.2 and for the second model in Table 5.3. Figure 5.5 shows the distribution of mean SemD values for the extracted texts in the second model. Using the Wald test, I evaluated the relationship between an independent variable and the dependent variable through the P value relating to the coefficient of that independent variable. If the P value is less than 0.05, the null hypothesis, i.e., that that coefficient is zero, is re-

| Per Policy | Length | SemD | Disag. |
|---|---|---|---|
| Mean | 2873.4 | 2.08 | 0.6 |
| Significance (P) | 0.64 | 0.74 | 0.34 |
| Odds Ratio (Z) | 1.15 | 1.11 | 0.54 |
| 95% Confidence Interval (Z) | 0.64-2.08 | 0.61-2.01 | 0.16-1.89 |

***Table 5.2:*** *Results of the first logistic regression model. The Nagelkerke pseudo $R^2$ is 0.03 and the Hosmer and Lemeshow value 0.13.*

| Per Extr. Text | Length | SemD | Disag. |
|---|---|---|---|
| Mean | 37.38 | 1.87 | 0.17 |
| Significance (P) | 0.22 | **0.02** | 0.81 |
| Odds Ratio (Z) | 0.58 | **2.07** | 0.86 |
| 95% Confidence Interval (Z) | 0.24-1.38 | 1.12-3.81 | 0.25-2.97 |

***Table 5.3:*** *Results of the second logistic regression model. The Nagelkerke pseudo $R^2$ is 0.11 and the Hosmer and Lemeshow value 0.051.*

jected. Looking at the results, it is noteworthy that both models do not reveal a statistically relevant correlation between the annotator disagreements and misclassifications. Thus, a document with a disagreement did not have a higher likelihood of being misclassified than one without. However, it is striking that the second model has a P value of 0.02 for the SemD variable. Standardizing the data points into Z scores and calculating the odds ratios it becomes clear that an increase of the mean SemD value in an extracted text by 0.17 (one standard deviation) increased the likelihood of a misclassification by 2.07 times (odds ratio). Consequently, the second model shows that the ambiguity of text in privacy policies, as measured by semantic diversity, has statistical significance for whether a classification decision is more likely to succeed or fail.

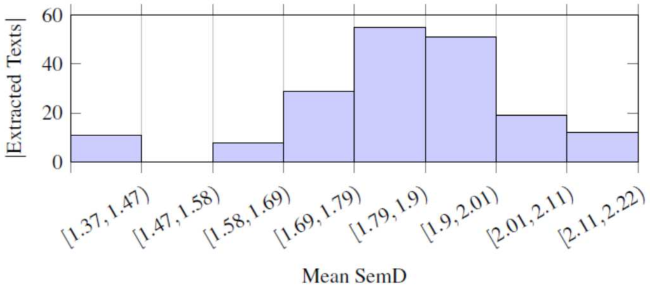


***Figure 5.5:*** *Mean SemD value distribution for the 185 extracted texts. The standard deviation is 0.17.*

Besides evaluating the statistical significance of individual variables, I also assessed the overall model fit. While the goodness of fit of linear regression models is usually evaluated based on the $R^2$ value, which measures the square of the sample correlation coefficient between the actual values of the dependent variable and the predicted values (in other words, the $R^2$ value can be understood as the proportion of the variance in a dependent variable attributable to the variance in the independent variable), there is no consensus for measuring the fit of binary logistic regression models. Various pseudo $R^2$ metrics are discussed. I used the Nagelkerke pseudo $R^2$ because it can range from 0 to 1 allowing an easy comparison to the regular $R^2$ (which, however, has to account for the fact that the Nagelkerke pseudo $R^2$ is often substantially lower than the regular $R^2$). While the Nagelkerke pseudo $R^2$ of 0.03 for the first model indicates a poor fit, the value of 0.11 for the second model can be interpreted as moderate. Further, the Hosmer and Lemeshow test, whose values were over 0.05 for both of the models, demonstrates the model fit as well.

In addition to the experiments just discussed, the models were also evaluated with further independent variables. Specifically, I evaluated the first model with the policy publication year, the second model with the extracted texts' mean tf-idf values, and both models with Flesch-Kincaid readability scores as independent variables. Also, using only ML classifications I evaluated the second model with the number of available training examples as independent variable. Only for the latter I found statistical significance at the 0.05 level. The number of training examples correlated to ML classification performance, which confirms Ammar et al.'s respective conjecture [32]. The more training examples the ML classifier had, the less likely a misclassification became.

### 5.3.2 Inter-annotator Agreement

Having discussed the classification performance, I now turn to the gold standard that was used to measure that performance. For the performance results to be reliable the gold standard must be reliable. One way of producing a gold standard for privacy policies is to ask the providers whose policies are analyzed to explain their meaning [16]. However, this approach should not be used, at least in the U.S., because the Restatement of Contracts provides that a contract term is generally given the meaning that *all* parties associate with it (Restatement (Second) of Contracts, §201). Consequently, policies should be interpreted from the perspective of both the provider and user. The interpretation would evaluate whether their perspectives lead to identical meanings or, if that is not the case, which one should prevail under applicable principles of legal interpretation. In addition, since technical terms are generally given technical meaning (Restatement (Second) of Contracts, §202(3)(b)), it would be advantageous if the interpretation is performed by annotators familiar with the terminology commonly used in privacy policies. The higher the number of annotations on which the annotators agree, that is, the higher the inter-annotator agreement, the more reliable the gold standard will be.

Because the annotation of a large number of documents can be very laborious, it is sufficient under current best practices for producing a gold standard to measure inter-annotator agreement only on a data sample [210], such that it can be inferred that the annotation of the remainder documents is reliable as well. Following this practice, I only measured the inter-annotator agreement for the test set, which would then provide an indicator for the reliability of the training and validation set annotation as well. To that end, I annotated all policies and additional annotations were obtained for the test policies from two other annotators. All annotators worked independently from each other. As the author who annotated the policies studied law and has expertise in privacy law and the two other annotators were law students with training in privacy law, all annotators were considered equally qualified, and the annotations for the gold standard were selected according to majority vote (i.e., at least two annotators agreed). After the annotations of the test policies were made, I ran the extension on these policies and compared its classifications to the annotations, which gave the results in Table 5.1.

The reliability of the gold standard depends on the degree to which the annotators agreed on the annotations. There are various measures for inter-annotator agreement. One basic measure

| | Disag. | % Ag. | K.'s $\alpha$/F.'s $\kappa$ |
|---|---|---|---|
| Overall | **8.12** | **84%** | **0.77** |
| Collection | 0 | 100% | 1 |
| Encryption | 6 | 88% | 0.84 |
| Ad Tracking | 7 | 86% | 0.8 |
| L. Retention | 9 | 82% | 0.68 |
| Profiling | 11 | 78% | 0.71 |
| Ad Disclosure | 16 | 68% | 0.56 |

***Table 5.4:*** *Inter-annotator agreement for the 50 test policies. The values for Krippendorff's $\alpha$ and Fleiss' $\kappa$ are identical.*

is the count of disagreements. Another one is the percentage of agreement (% Ag.), which is the fraction of documents on which the annotators agree [37]. However, disagreement count and percentage of agreement have the disadvantage that they do not account for chance agreement. In this regard, chance-corrected measures, such as Krippendorff's $\alpha$ (K.'s $\alpha$) [172] and Fleiss' $\kappa$ (F.'s $\kappa$) [124] are superior. For Krippendorff's $\alpha$ and Fleiss' $\kappa$ the possible values are constrained to the interval $[-1; 1]$, where 1 means perfect agreement, $-1$ means perfect disagreement, and 0 means that agreement is equal to chance [87]. Generally, values above 0.8 are considered as good agreement, values between 0.67 and 0.8 as fair agreement, and values below 0.67 as dubious [187]. However, those ranges are only guidelines [37]. Particularly, ML algorithms can tolerate data with lower reliability as long as the disagreement looks like random noise [225].

Based on the best practices and guidelines for interpreting inter-annotator agreement measurements, the results in Table 5.4 confirm the general reliability of the annotations and, consequently, of the gold standard. For every individual category, except for the ad disclosure category Krippendorff's $\alpha$ values indicated fair or good agreement. In addition, the overall mean agreement across categories is 0.77, and, therefore, provides evidence for fair overall agreement as well. For the overall agreement it should be noted that, corresponding to the multi-label classification task, the annotation of privacy policies is a multi-label annotation task as well. However, there are

| Per Policy | Length | SemD | Flesch-K. |
|---|---|---|---|
| Mean | 2873.4 | 2.08 | 14.53 |
| Significance (P) | 0.2 | 0.11 | 0.76 |
| Odds Ratio (Z) | 1.65 | 1.87 | 1.12 |
| 95% Confidence Interval (Z) | 0.78-3.52 | 0.87-4 | 0.55-2.29 |

***Table 5.5:*** *Results of the third logistic regression model. The Nagelkerke pseudo $R^2$ is 0.19 and the Hosmer and Lemeshow value 0.52.*

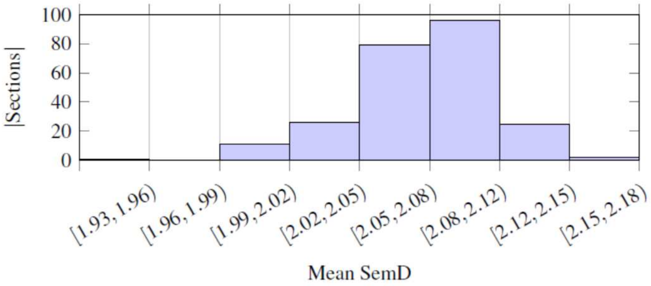


***Figure 5.6:*** *Mean SemD value distribution for the 240 policy sections. The standard deviation is 0.03.*

only very few multi-label annotation metrics, such as Passonneau's Measuring Agreement on Set-valued Items (MASI) [209]. As none of the metrics were suitable for the purposes here I selected as overall metric the mean over the results of the individual classification categories.

Inter-annotator agreement results were investigated by applying a third and fourth binary logistic regression model. In the third model each of the 50 test policies was represented by one data point with the dependent variable identifying whether the annotators had any disagreement in annotating the policy and the independent variables identifying (1) the policy's length in words, (2) its mean SemD value, and (3) its Flesch-Kincaid score. In the fourth model each of 240 individual annotations is represented by one data point with the dependent variable identifying whether the annotators disagreed for that annotation and the independent variables identifying (1) the length

| Per Section | Length | SemD | Flesch-K. |
|---|---|---|---|
| Mean | 306.76 | 2.08 | 15.59 |
| Significance (P) | 0.29 | **0.04** | 0.49 |
| Odds Ratio (Z) | 1.18 | **1.51** | 0.86 |
| 95% Confidence Interval (Z) | 0.87-1.6 | 1.02-2.22 | 0.56-1.32 |

***Table 5.6:*** *Results of the fourth logistic regression model. The Nagelkerke pseudo $R^2$ is 0.05 and the Hosmer and Lemeshow value 0.83.*

(in words) of the policy text section that the annotation is referring to, (2) the section's mean SemD value, and (3) its Flesch-Kincaid score. For the fourth model some of the 300 annotations were excluded because not every policy had a section for each category. For example, some policies did not discuss advertisement or disclosure of information. The Flesch-Kincaid readability score measures the number of school years an average reader would need to understand a text.

For the third and fourth model analysis results are shown in Table 5.5 and 5.6, respectively. Figure 5.6 shows the distribution of mean SemD values for the policy sections in the fourth model. Both models were significant, as indicated by their Nagelkerke and Hosmer and Lemeshow values. The results confirm that the readability of policies, as measured by the Flesch-Kincaid score, does not impact their comprehensibility [191]. In the third model I was unable to identify any statistically relevant variables (although, semantic diversity and length may be statistically significant in a larger data set). However, the fourth model proved to be more meaningful. Remarkably, corresponding to the finding in Section 5.3.1, according to which classifier performance correlates to semantic diversity, the statistically relevant P value of 0.04 for the mean SemD variable also indicates a correlation of inter-annotator agreement to semantic diversity. Standardizing the data points into Z scores and calculating the odds ratios it becomes clear that an increase of the mean SemD value of a section by 0.03 (one standard deviation) increased the likelihood of a disagreement by 1.51 times (odds ratio). It is astounding that even qualified annotators trained in privacy law had difficulties to avoid disagreements when semantic diversity increased to slightly above-mean levels.

While neither the first nor the second model in Section 5.3.1 showed a correlation between inter-annotator agreement and classifier performance, the results for the second and fourth model demonstrate that performance and agreement both correlate to one common variable—semantic diversity. More specifically, performance correlates to the semantic diversity of extracted text phrases and agreement correlates to the semantic diversity of policy sections. This result suggests, for example, that the relatively high number of misclassifications and disagreements in the ad disclosure category is inherent in the nature of the category. Indeed, in cases of fuzzy categories disagreements among annotators do not necessarily reflect a quality problem of the gold standard, but rather a structural property of the annotation task, which can serve as an important source of empirical information about the structural properties of the investigated category [24]. Thus, it is no surprise that for all six categories the values of Krippendorff's $\alpha$ correlate to the F-1 scores. The higher the value of Krippendorff's $\alpha$, the higher the F-1 score. Figure 5.7 shows the correlation.

As both classifier performance and inter-annotator agreement decrease with an increase in semantic diversity, the practicability of the notice and choice principle becomes questionable. After all, privacy policies can only provide adequate notice (and choice) if they are not too ambiguous. In order to further examine policy ambiguity I calculated the mean SemD value for the test policies over time. The test set analysis exhibited a statistically significant trend of decreasing semantic diversity with a P value of 0.049. Figure 5.8 illustrates the approach taken here. There are two possible explanations for the decrease over time. First, it could be a consequence of the FTC's enforcement actions and its call for policies to "be clearer, shorter, and more standardized" [110]. Second, we might be in the midst of a consolidation process leading to more standardized policy language. As de Maat et al [80] observed, drafters of legal documents tend to use language that adheres to writing conventions of earlier texts and similar statements. Independent of the reason, the result suggests that the notice and choice principle may overcome the problem of ambiguity over time.

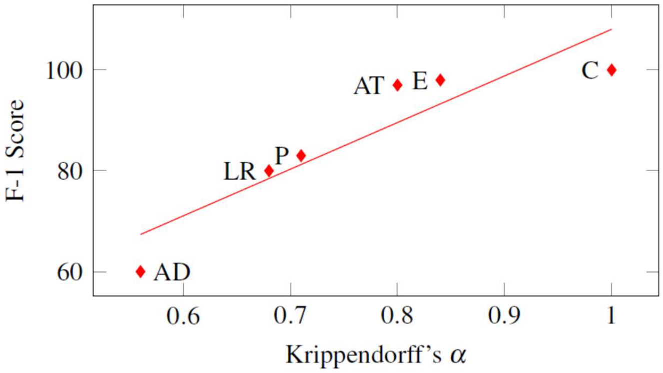


***Figure 5.7:*** *Linear regression plot with the F-1 score as dependent variable and Krippendorff's $\alpha$ as independent variable. The coordinate labels identify the categories: AD = Ad Disclosure, LR = Limited Retention, P = Profiling, AT = Ad Tracking, E = Encryption, and C = Collection. With an $R^2$ value of 0.83 the model has an excellent fit, which, however, should be interpreted in light of the small number of data points.*

### 5.3.3 Computational Performance

The extension's computational performance allows a real-time analysis. Table 5.7 shows the mean duration in seconds for obtaining analysis results for each of 50 randomly selected policies from ToS;DR (Crowdsourcing), processing each of the 50 test policies (Classifier), and processing each of the 50 test policies each with initial training (Training). Notably, retrieving policy results from ToS;DR is twice as fast as analyzing a policy with the classifiers.

## 5.4 Conclusion

In order to improve privacy transparency I developed Privee—a system to automatically analyze privacy policies. Based on ML algorithms Privee analyzes policy text and returns a label with the most important information allowing Internet users to gain a fast understanding of essential policy terms. Interestingly, experimental results reveal that the automatic classification of privacy policies encounters the same constraint as human policy interpretation—the ambiguity of natural language, as measured by semantic diversity. Such ambiguity seems to present an inherent limitation of what automatic privacy policy analysis can accomplish. Thus, on a more fundamental level, the viability of the notice and choice principle might be called into question altogether. However, based on the

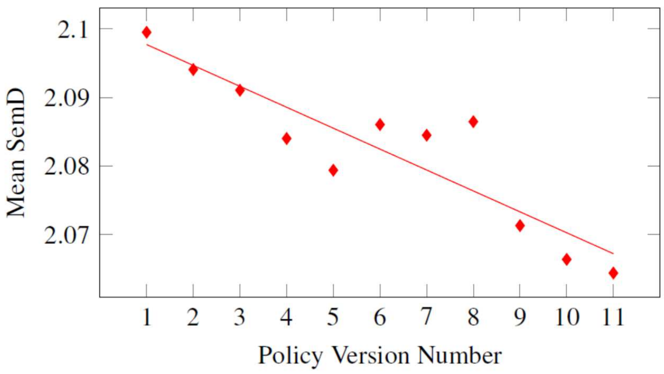


***Figure 5.8:*** *Linear regression plot for Symantec's privacy policy (which was part of the test set) with the mean SemD value of a policy version as dependent variable and the policy version number as independent variable. The first version of Symantec's policy dates back to August 5, 1999, and the eleventh version was adopted on August 12, 2013. The mean SemD value of Symantec's privacy policy decreased from 2.1 in the first version to 2.06 in the eleventh version as shown. A similar decrease occurred for 29 out of 44 test policies (6 of the test policies were only available in a single version and, therefore, could not be included in the analysis. However, for the 44 included policies there were on average 8 different versions over time.).*

presented indicators for a decrease of policy ambiguity over time I would caution to draw such conclusion, and I remain optimistic that the current notice and choice ecosystem is workable.

I believe that over time conventional language will develop that will make the meaning of many privacy policy provisions much clearer. As de Maat et al. [80] observed, drafters of legal documents tend to use language that adheres to writing conventions of earlier texts and similar information is usually expressed in syntactically similar statements. The FTC's call for privacy policies to "be clearer, shorter, and more standardized" [110] coupled with its enforcement power will likely also lead to a decrease in ambiguity. As anecdotal evidence serves Google's privacy policy whose mean SemD value decreased throughout the years from 2.1 in the year 1999 to 2.04 in 2013. Privacy policy crowdsourcing can supplement this development by providing a forum for identifying, discussing, and resolving ambiguities.

While Privee is the first architecture for automatically analyzing privacy policies, much more work remains to be done: What are the types of information that policies should be analyzed for? What is the most usable design for displaying the analysis results? What are the best features and

| Per Policy | Crowdsourcing | Classifier | Training |
|---|---|---|---|
| Mean | 0.39 sec | 0.78 sec | 20.29 sec |

***Table 5.7:*** *Computational performance of the Privee extension. The performance was evaluated on a Windows laptop with Intel Core2 Duo CPU at 2.13 GHz with 4 GB RAM. The space requirements for the installation on the hard disk are 2.11 MB (including 1.7 MB of training data and 286 KB for the jQuery library) and additional 230 KB during the program execution for storing training results.*

algorithms to train a privacy policy classifier? How can the interaction between the classifier and crowdsourcing analysis be improved? In particular, how can a program connect to many crowdsourcing repositories, and, possibly, decide which analysis is the best? Can crowdsourced policy results be fed into the classifier as training data? How can it be assured that the crowdsourcing results are always up to date? What are other ways to exploit the semantic diversity metric? And, finally, how can the whole architecture be made workable in the mobile world?

# Chapter 6

# Bridging the Gap between Notices and Actual Practices

Snapchat does "not ask for, track, or access any location-specific information." This is what Snapchat's privacy policy stated.[1] However, Snapchat's Android app transmitted Wi-Fi- and cell-based location data from users' devices to analytics service providers. These discrepancies remained undetected before they eventually surfaced when a researcher examined Snapchat's data deletion mechanism. His report was picked up by the Electronic Privacy Information Center and brought to the attention of the FTC, which launched a formal investigation requiring Snapchat to implement a comprehensive privacy program.[2]

The case of Snapchat illustrates that mobile apps are often deviating from their privacy policies. However, any inconsistencies can have dire consequences as the as they may lead to enforcement actions by the FTC and other regulators. This is especially true if discrepancies continue to exist for many years, which was the case for Yelp's collection of childrens' information.[3] These findings do not only demonstrate that regulators could benefit from a system that helps them identifying privacy requirement inconsistencies, but also that it would be a useful tool for companies to assess their privacy compliance as part of the software development process. This would be

[1] Complaint In the Matter of Snapchat, Inc. (December 31, 2014).

[2] Decision and Order In the Matter of Snapchat, Inc. (December 31, 2014).

[3] United States of America v. Yelp, Inc. (September 17, 2014).

valuable because researchers found that privacy violations often appear to be based on developers' difficulties in understanding privacy requirements [44] rather than on malicious intentions. Thus, for example, tools that automatically detect and describe third-party data collection practices may be helpful for developers [44]. Consequently, it is a major motivation of my work to help companies identifying red flags before they develop into serious and contentious privacy problems.

On various occasions, the FTC, which is responsible for regulating consumer privacy on the federal level, voiced dissatisfaction with the current state of apps' privacy compliance. Three times the FTC manually surveyed childrens' apps for privacy law compliance [108; 109; 118] and concluded that the "results of the survey are disappointing" [109]. Deviating from mandatory provisions, many publishers of childrens' apps did not disclose what types of data they collect, how they make use of the data, and with whom the data is shared [109]. A similar examination of 121 shopping apps revealed that many privacy policies are vague and fail to convey how apps actually handle consumers' data [114]. Given that the FTC limited its investigations to a small sample of apps, a presumably large number of discrepancies between apps and their privacy policies remained undetected. However, the FTC and other regulators have difficulty to achieve scale in their compliance checks. In this regard, I believe that the system can be leveraged by regulators to substantially increase the scope of their analyses.

In this chapter I present a privacy analysis system for Android that checks data practices of apps against privacy requirements derived from their privacy policies and selected laws. The work here enables app publishers to identify potentially privacy-invasive practices in their apps before they are published. Moreover, the work can also aid governmental agencies, such as the FTC, to achieve a systematic enforcement of privacy laws on a large scale. App store owners, researchers, and privacy advocates alike might also derive value from the approach presented here. My main contribution consists of the novel combination of machine learning and static analysis techniques to analyze apps' compliance with privacy requirements. However, I want to emphasize that this dissertation does not claim to resolve challenges in the individual techniques beyond what is necessary for the purposes here. This holds especially true for the static analysis of mobile apps and its many unresolved problems, for example, in the analysis of obfuscated code.

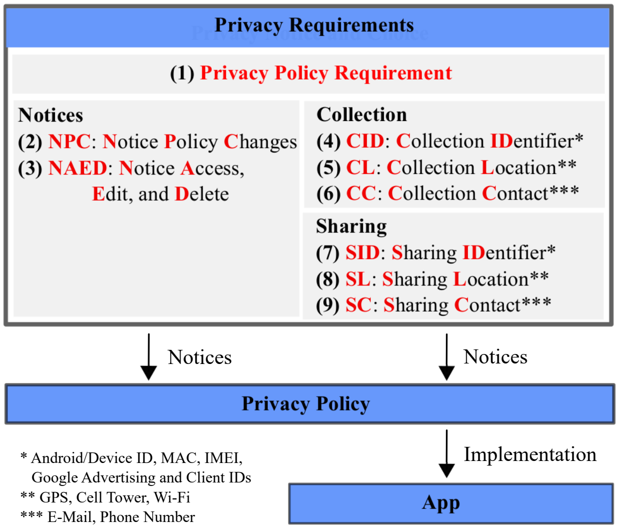

Privacy Requirements
(1) Privacy Policy Requirement
Notices
(2) NPC: Notice Policy Changes
(3) NAED: Notice Access, Edit, and Delete
Collection
(4) CID: Collection IDentifier*
(5) CL: Collection Location**
(6) CC: Collection Contact***
Sharing
(7) SID: Sharing IDentifier*
(8) SL: Sharing Location**
(9) SC: Sharing Contact***
Notices
Notices
Privacy Policy
Implementation
App
* Android/Device ID, MAC, IMEI, Google Advertising and Client IDs
** GPS, Cell Tower, Wi-Fi
*** E-Mail, Phone Number

***Figure 6.1:*** *Per the defined privacy requirements, apps that process Personally Identifiable Information (PII) need to (1) have a privacy policy, (2-3) include notices about policy changes and access, edit, and deletion rights in their policy, (4-6) notify users of data collection practices, and (7-9) disclose how data is shared with third parties. The notice requirements for policy changes and access, edit, and deletion are satisfied by including the notices in the policies while the collection and sharing practices must be also implemented in the apps.*

## 6.1 Privacy Policy Analysis

This section will discuss the automated large-scale ML analysis of privacy policies. It will first detail the law on privacy notice and choice (§ 6.1.1), then explain the check how many apps have a privacy policy (§ 6.1.2), and finally analyze the policy content (§ 6.1.3).

### 6.1.1 Notice and Choice

The privacy requirements are derived from apps' privacy policies and selected laws. Figure 6.1 provides an overview of the law on notice and choice and the nine privacy requirements that are analyzed (Privacy Policy Requirement, NPC, NAED, CID, CL, CC, SID, SL, SC). If an app does not adhere to a privacy requirement—by implementing a practice that is not covered in its policy—or if the app's policy does not notify users of policy changes and access, edit, and deletion rights, it is defined that a privacy requirement inconsistency occurs (which is also referred to to as non-compliance). In this regard, it should be cautioned that such inconsistency does not necessarily mean that a law is violated. First, not all privacy requirements might be applicable to all apps. Second, the system is based on a particular interpretation of the law. While I believe that the interpretation is sound and in line with the enforcement actions of the FTC and other regulatory agencies, reasonable minds may differ.[4] Third, the system makes is based on machine learning and static analysis and, thus, by its very nature errors can occur.

As to the individual privacy requirements, there is no generally applicable federal statute demanding privacy policies for apps. However, California and Delaware enacted comprehensive online privacy legislation that effectively serves as a national minimum privacy threshold given that app publishers usually do not provide state-specific app versions or exclude California or Delaware residents. In this regard, the California Online Privacy Protection Act of 2003 (CalOPPA) requires online services that collect PII to post a policy.[5] The same is true according to Delaware's Online Privacy and Protection Act (DOPPA).[6] In addition, the FTC's Fair Information Practice Principles (FTC FIPPs) call for consumers to be given notice of an entity's information practices before any PII is collected [106]. Further, the Children's Online Privacy Protection Act of 1998 (COPPA) makes policies mandatory for apps directed to or known to be used by children.[7] Thus, the existence of a privacy policy is treated as a privacy requirement.

CalOPPA and DOPPA further demand that privacy policies describe the process by which users are notified of policy changes.[8] COPPA also requires description of access, edit, and deletion rights.[9] Under the FTC FIPPs [106] as well as CalOPPA and DOPPA those rights are optional.[10] I concentrate the analysis on a subset of data types that are, depending on the context, legally protected: device IDs, location data, and contact information. App publishers are required to

[4] I am focusing on the U.S. legal system as I am most familiar with it. However, in principle, the techniques are applicable to any country with a privacy notice and choice regime.

[5] Cal. Bus. & Prof. Code §22575(a).

[6] Del. Code Tit. 6 §1205C(a).

[7] 16 CFR §312.4(d).

[8] Cal. Bus. & Prof. Code §22575(b)(3), Del. Code Tit. 6 §1205C(b)(3).

[9] 16 CFR §312.4(d)(3).

[10] Cal. Bus. & Prof. Code §22575(b)(2), Del. Code Tit. 6 §1205C(a).

disclose the collection of device IDs (even when hashed) and location data.[11] Device IDs and location data are also covered by CalOPPA[12] and for children apps according to COPPA[13] The sharing of these types of information with third parties requires consent as well.[14] Similarly, contact information, such as e-mail addresses, may be protected.[15]

It should be noted that ad identifiers are interpreted to be personal information since they can be used to track users over time and across devices. It is also assumed that a user did not opt out of ads (because otherwise no ad identifiers would be sent to opted out ad networks). I further interpret location data to refer to GPS, cell tower, or Wi-Fi location. I assume applicability of the discussed laws and perform the analysis based on the guidance provided by the FTC and the California Attorney General in enforcement actions and recommendations for best practices (e.g., [106] and [61]). Specifically, I interpret the FTC actions as disallowing the omission of data practices in policies and assume that silence on a practice means that it does not occur.[16] Finally, I assume that all apps in the U.S. Play store are subject to CalOPPA and DOPPA.[17] I believe this assumption is reasonable as I am not aware of any U.S. app publisher excluding California or Delaware residents from app use or providing state-specific app versions.

### 6.1.2 Privacy Policy Requirement

To assess whether apps fulfill the requirement of having a privacy policy I crawled the Google Play store (February 2016) and downloaded a sample ($n = 17,991$) of free apps (full app set).[18] The crawl was started with the most popular apps and followed random links on their Play store pages to other apps. I included all categories in the crawl, however, excluded Google's Designed for Families program (as Google already requires apps in this program to have a policy) and Android

[11] In the Matter of Nomi Technologies, Inc. (September 3, 2015).

[12] Cal. Bus. & Prof. Code §22577(a)(6) and (7) [61].

[13] 16 CFR §312.2(7) and (9).

[14] Complaint In the Matter of Goldenshores Technologies, LLC, and Erik M. Geidl (April 9, 2014).

[15] Complaint In the Matter of Snapchat, Inc. (December 31, 2014).

[16] Complaint In the Matter of Snapchat, Inc. (December 31, 2014).

[17] Cal. Bus. & Prof. Code §§22575–22579, Del. Code Tit. 6 §1205C.

[18] Whenever the Google Play store is referred to it is its U.S. site. Also, details on the various app and policy sets that are used are described in the appendix.

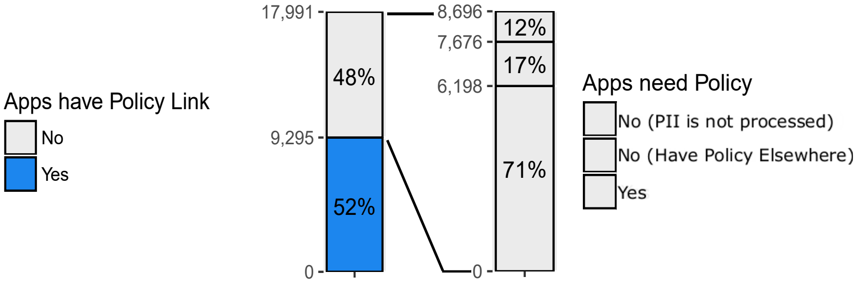


***Figure 6.2:*** *I analyze 17,991 free apps, of which 9,295 (52%) link to their privacy policy from the Play store (left). Out of the remaining apps, 6,198 (71%) appear to lack a policy while engaging in at least one data practice (i.e., PII is processed) that would require them to have one (right).*

Wear (as the focus is on mobile apps). It is assumed that the sample is representative in terms of app categories, which was confirmed with a two-sample Kolmogorov-Smirnov goodness of fit test (two-tailed) against a sample of a million apps [203]. It was not possible to reject the null hypothesis that both were drawn from the same distribution (i.e., p $> 0.05$). However, while the Play store hosts a long tail of apps that have fewer than 1K installs (56%) [203], the sample focuses on more popular apps as it only includes 3% of such fewer installed apps.

**Privacy Policy Requirement Inconsistencies.** Out of all policies in the full app set $n = 9,295$ apps provided a link to their policy from the Play store (full policy set) and $n = 8,696$ apps lacked such. As shown in Figure 6.2, the results suggest that 71% (6,198/8,696) apps without a policy link are indeed not adhering to the policy requirement. These app store privacy policy links can be used as proxies for actual policies, which is reasonable since regulators requested app publishers to post such links [111; 61] and app store owners obligated themselves to provide the necessary functionality [60]. The apps in the full app set were offered by a total of 10,989 publishers, and their app store pages linked to 6,479 unique privacy policies.

71% is achieved after making two adjustments. First, if an app does not have a policy it is not necessarily non-compliant with the policy requirement. After all, apps that are not processing PII are not obligated to have a policy. Indeed, since I found that 12% (1,020/8,696) of apps are not processing PII, I accounted for those apps. Second, despite the regulators' requests to post policy links in the Play store, some app publishers may still decide to post their policy elsewhere (e.g., inside their app). For that purpose I randomly examined 40 apps from the full app set that did not

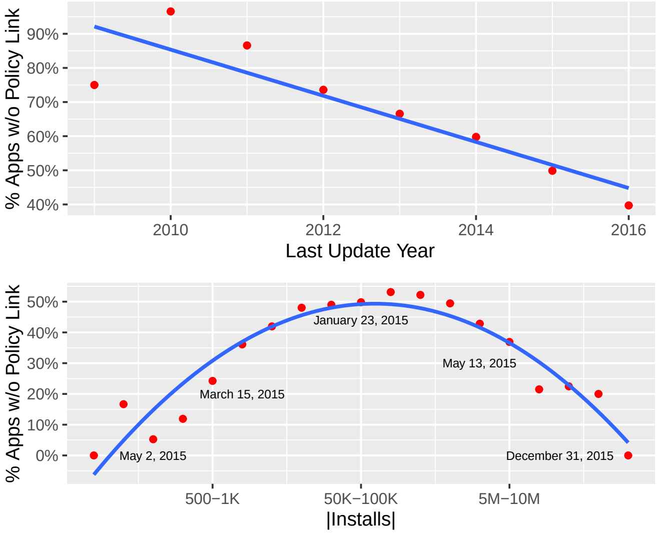


***Figure 6.3:*** *A linear regression model with the last app update year as independent variable and the percentage of apps without a policy link as dependent variable gives $r^2 = 0.79$ (top). In addition, a polynomial regression model using the number of installs as independent variable results in a multiple $r^2 = 0.9$ (bottom).*

have a policy link in the Play store but processed PII. I found that 83% (33/40) do not seem to have a policy posted anywhere (with a Clopper-Pearson confidence interval (CI) ranging from 67% to 93% at the 95% level based on a two-tailed binomial test).[19] Thus, accounting for an additional 17% (1,478/8,696) of apps having a policy elsewhere leaves us with $100\% - 12\% - 17\% = 71\%$ out of $n = 8,696$ apps appearing to be non-compliant with the policy requirement.

**Predicting Privacy Policy Requirement Inconsistencies.** As it appears that apps with frequent updates typically have a policy, this hypothesis was evaluated on the full app set using Pearson's chi-square test of independence. Specifically, it is the null hypothesis that whether an app has a policy is independent from the year when it was most recently updated. As the test returns p $\leq 0.05$, the null hypothesis can be rejected at the 95% confidence level. Indeed, as shown in the linear regression model of Figure 6.3, apps with recent update years have more often a policy than those that were updated longer ago. In addition to an app's update year there are other viable predictors as well. As shown in the polynomial regression model of Figure 6.3 the number of

[19] Except otherwise noted, all CIs in this paper are based on a two tailed binomial test and the Clopper-Pearson interval at the 95% level.

installs is insightful (p $\leq 0.05$). Apps with high install rates have more often a policy than apps with average install rates. Surprisingly, the same is also true for apps with low install rates. An explanation could be that those are more recent apps that did not yet gain popularity. Indeed, apps with low install rates are on average more recently updated than apps with medium rates. For example, apps with 500 to 1K installs were on average updated on March 15, 2015 while apps with 50K to 100K installs have an average update date as of January 23, 2015.

Further, apps with an Editors' Choice or Top Developer badge usually have a policy, which is also true for apps that offer in-app purchases. It is further encouraging that apps with a content rating for younger audiences often have a policy. Most apps for Everyone 10+ (75%), Teen (65%), and Mature 17+ (66%) audiences have a policy while apps that have an Everyone rating (52%) or are unrated (30%) often lack one.[20] Further, various app categories are particularly susceptible for not having a policy. Apps in the Comics (20%), Libraries & Demo (10%), Media & Video (28%), and Personalization (28%) categories have particularly low policy penetration, as compared to an average of 52% of apps having a policy across categories. Combining these predictors enables us to zoom in to areas of apps that are unlikely to have a policy. For instance, in the Media & Video category the percentage of apps with a policy decreases from 28% for rated apps to 12% for unrated apps. A similar decrease occurs in the Libraries & Demo category from 10% to 8%.

### 6.1.3 Privacy Policy Content

Let us now move from examining whether an app has a policy to the analysis of policy content. As a basis for the evaluation manually created policy annotations are used.

#### 6.1.3.1 Inter-annotator Agreement

For training and testing the classifiers the OPP-115 corpus [260] is leveraged—a corpus of 115 privacy policies annotated by ten law students that includes 2,831 annotations for the practices discussed here. The annotations, which are described in detail in [260], serve as the ground-truth for the ML classifiers (§ 6.1.3.3). Each annotator annotated a mean of 34.5 policies (median 35). The annotations are selected according to majority agreement (i.e., two out of three annotators agreed on it). As it is irrelevant from a legal perspective how often a practice is described in a

[20] The ratings are based on the categories of the Entertainment Software Rating Board (ESRB).

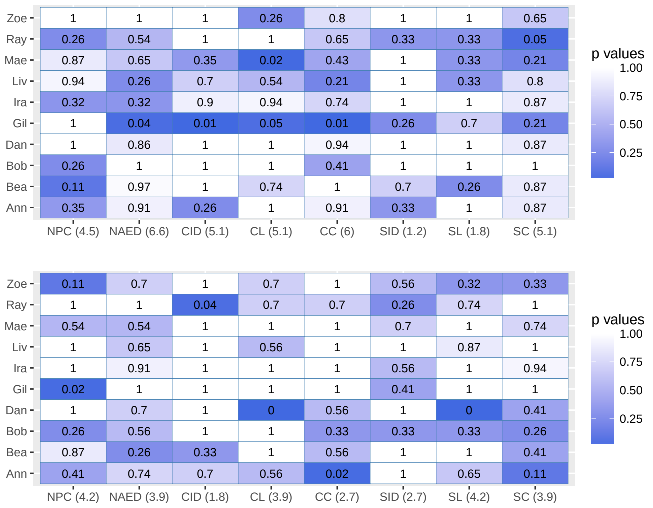
| ***Practice*** | ***\|Ann\|*** | ***Ag**$_{pol}$* | ***% Ag**$_{pol}$* | ***Fleiss**$_{pol}$**/Krip**$_{pol}$* |
|---|---|---|---|---|
| NPC | 395 | 86/115 | 75% | 0.64 |
| NAED | 414 | 80/115 | 70% | 0.59 |
| CID | 449 | **92/115** | 80% | **0.72** |
| CL | 326 | 85/115 | 74% | 0.64 |
| CC | 830 | 86/115 | 75% | 0.5 |
| SID | 90 | 101/115 | 88% | **0.76** |
| SL | 51 | 95/115 | 83% | 0.48 |
| SC | 276 | 85/115 | 74% | 0.58 |

***Table 6.1:*** *The table shows absolute numbers of annotations (|Ann|) as well as various agreement measures, specifically, absolute agreements (Ag$_{pol}$), percentage agreements (% Ag$_{pol}$), Fleiss' $\kappa$ (Fleiss$_{pol}$), and Krippendorff's $\alpha$ (Krip$_{pol}$). All agreement measures are computed on the full corpus of 115 policies and on a per-policy basis (e.g., for 92 out of 115 policies the annotators agreed on whether the policy allows collection of identifiers).*

policy, it is measured whether annotators agree that a policy describes a given practice at least once.

High inter-annotator agreement signals the reliability of the ground-truth on which classifiers can be trained and tested. As agreement measures I use Fleiss' $\kappa$ and Krippendorff's $\alpha$, which indicate that agreement is good above 0.8, fair between 0.67 and 0.8, and doubtful below 0.67 [187]. From the results in Table 6.1 it follows that the inter-annotator agreement for collection and sharing of device IDs with respective values of 0.72 and 0.76 is fair. However, it is below 0.67 for the remaining classes. While results showing stronger agreement would have been clearer, the annotations with the observed agreement levels can still provide reliable ground-truth as long as the classifiers are not misled by patterns of systematic disagreement, which can be explored by analyzing the disagreeing annotations [225].

To analyze whether disagreements contain systematic patterns I evaluate the number of each annotator's disagreements with the other two annotators. If he or she is in a minority position for a statistically significant number of times, there might be a misunderstanding of the annotation task or other systematic reason for disagreement. However, if there is no systematic disagreement,



| | NPC (4.5) | NAED (6.6) | CID (5.1) | CL (5.1) | CC (6) | SID (1.2) | SL (1.8) | SC (5.1) |
|---|---|---|---|---|---|---|---|---|
| Zoe | 1 | 1 | 1 | 0.26 | 0.8 | 1 | 1 | 0.65 |
| Ray | 0.26 | 0.54 | 1 | 1 | 0.65 | 0.33 | 0.33 | 0.05 |
| Mae | 0.87 | 0.65 | 0.35 | 0.02 | 0.43 | 1 | 0.33 | 0.21 |
| Liv | 0.94 | 0.26 | 0.7 | 0.54 | 0.21 | 1 | 0.33 | 0.8 |
| Ira | 0.32 | 0.32 | 0.9 | 0.94 | 0.74 | 1 | 1 | 0.87 |
| Gil | 1 | 0.04 | 0.01 | 0.05 | 0.01 | 0.26 | 0.7 | 0.21 |
| Dan | 1 | 0.86 | 1 | 1 | 0.94 | 1 | 1 | 0.87 |
| Bob | 0.26 | 1 | 1 | 1 | 0.41 | 1 | 1 | 1 |
| Bea | 0.11 | 0.97 | 1 | 0.74 | 1 | 0.7 | 0.26 | 0.87 |
| Ann | 0.35 | 0.91 | 0.26 | 1 | 0.91 | 0.33 | 1 | 0.87 |

| | NPC (4.2) | NAED (3.9) | CID (1.8) | CL (3.9) | CC (2.7) | SID (2.7) | SL (4.2) | SC (3.9) |
|---|---|---|---|---|---|---|---|---|
| Zoe | 0.11 | 0.7 | 1 | 0.7 | 1 | 0.56 | 0.32 | 0.33 |
| Ray | 1 | 1 | 0.04 | 0.7 | 0.7 | 0.26 | 0.74 | 1 |
| Mae | 0.54 | 0.54 | 1 | 1 | 1 | 0.7 | 1 | 0.74 |
| Liv | 1 | 0.65 | 1 | 0.56 | 1 | 1 | 0.87 | 1 |
| Ira | 1 | 0.91 | 1 | 1 | 1 | 0.56 | 1 | 0.94 |
| Gil | 0.02 | 1 | 1 | 1 | 1 | 0.41 | 1 | 1 |
| Dan | 1 | 0.7 | 1 | 0 | 0.56 | 1 | 0 | 0.41 |
| Bob | 0.26 | 0.56 | 1 | 1 | 0.33 | 0.33 | 0.33 | 0.26 |
| Bea | 0.87 | 0.26 | 0.33 | 1 | 0.56 | 1 | 1 | 0.41 |
| Ann | 0.41 | 0.74 | 0.7 | 0.56 | 0.02 | 1 | 0.65 | 0.11 |

***Figure 6.4:*** *Analysis of disagreement among annotators for the different data practices with binomial tests. Smaller p values mean fewer disagreements. If there are no disagreements, it is defined $p = 1$. An annotator can be in the minority when omitting an annotation that the two other annotators made (top) or adding an extra annotation (bottom). The results show few instances of systematic disagreement. The numbers in parentheses show the average absolute disagreements for the respective practices.*

annotations are reliable despite low agreement levels [225].[21] Assuming a uniform distribution each annotator should be in the minority in 1/3 of all disagreements. I test this assumption with the binomial test for goodness of fit. Specifically, I use the binomial distribution to calculate the probability of an annotator being $x$ or more times in the minority by adding up the probability of being exactly $x$ times in the minority, being $x + 1$ times in the minority, up to $x + n$ (that is, being always in the minority), and comparing the result to the expected probability of 1/3. I use a one-tailed test as it is not of interest to find whether an annotator is fewer times in the minority than in 1/3 of the disagreements.

As shown in Figure 6.4, there are only few cases with systematic disagreement. More specifically, for 7% (11/160) of disagreements there was statistical significance (p $\leq 0.05$) for rejecting the null hypothesis that the disagreements are equally distributed. We see that nearly half of the

[21] Arguably, low agreement levels present a problem from a legal perspective as there is no common interpretation of a respective policy fragment.

| ***Practice*** | ***Classifier*** | ***Parameters*** | ***Base*** | ***Acc$_{pol}$*** | ***95% CI*** | ***Prec$_{neg}$*** | ***Rec$_{neg}$*** | ***F-1$_{neg}$*** | ***F-1$_{pos}$*** | ***Pos*** |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | *(n=40)* | *(n=40)* | *(n=40)* | *(n=40)* | *(n=40)* | *(n=40)* | *(n=40)* | *(n=9,050)* |
| NPC | SVM | RBF, weight | 0.7 | 0.9 | 0.76–0.97 | 0.79 | 0.92 | **0.85** | 0.93 | **46%** |
| NAED | SVM | linear | 0.58 | 0.75 | 0.59–0.87 | 0.71 | 0.71 | **0.71** | 0.78 | **36%** |
| CID | Log. Reg. | LIBL | 0.65 | 0.83 | 0.67–0.93 | 0.77 | 0.71 | **0.74** | 0.87 | 46% |
| CL | SVM | linear | 0.53 | 0.88 | 0.73–0.96 | 0.83 | 0.95 | **0.89** | 0.86 | 34% |
| CC | Log. Reg. | LIBL, L2, weight | 0.8 | 0.88 | 0.73–0.96 | 0.71 | 0.63 | **0.67** | 0.92 | 56% |
| SID | Log. Reg. | LBFGS solver, L2 | 0.88 | 0.88 | 0.73–0.96 | 0.94 | 0.91 | **0.93** | **0.55** | **10%** |
| SL | SVM | linear, weight | 0.95 | 0.93 | 0.8–0.98 | 0.97 | 0.95 | **0.96** | - | **12%** |
| SC | SVM | poly (4 degrees) | 0.73 | 0.78 | 0.62–0.89 | 0.79 | 0.93 | **0.86** | **0.47** | **6%** |

***Table 6.2:*** *Classifiers, parameters, and classification results for the policy test set (n=40) and the occurrence of positive classifications (Pos) in a set of n=9,050 policies (full app/policy set). The best results were obtained by always setting the regularization constant to $C = 1$ and for NPC, CC, and SL adjusting weights inversely proportional to class frequencies with scikit-learn's* `class_weight` *(weight). Except for the SL practice, all classifiers' accuracies (Acc$_{pol}$) reached the baseline (Base) of always selecting the most often occurring class in the training set. $Prec_{neg}$, $Rec_{neg}$, and F-1$_{neg}$ are the scores for the negative classes (e.g., data is not collected or shared) while F-1$_{pos}$ is the F-1 score for positive classes.*

systematic disagreements occur for Gil. However, excluding Gil's and other affected annotations from the training set for the classifiers had only little noticeable effect. For some practices the classification accuracy slightly increased, for others it slightly decreased. Thus, I believe that the annotations are sufficiently reliable to serve as ground-truth for the classifiers. As other works have already explored, low levels of agreement in policy annotations are common and do not necessarily reflect their unreliability [224; 269]. In fact, different from the approach of analyzing systematic annotation differences, it could be argued that an annotator's addition or omission of an annotation is not a disagreement with the others' annotations to begin with.

```
def location_feature_extraction(policy):

 location_keywords = [’geo’, ’gps’]
 sharing_keywords = [’share’, ’partner’]
 rel_sentences = ’’
 features = ’’

```

```
for sentence in policy:
 for keyword in location_keywords:
  if (keyword in sentence):
   rel_sentences += sentence

tokens = word_tokenize(rel_sentences)
bigrams = ngrams(tokens,2)

for bigram in bigrams:
 for keyword in sharing_keywords:
  if (keyword in bigram):
   features += bigram, bigram[0], bigram[1]

return features
```

***Listing 6.1:*** *Pseudocode for the sharing of location (SL).*

#### 6.1.3.2 Feature Selection

One of the most important tasks for correctly classifying data practices described in privacy policies is appropriate feature selection. Listing 6.1 shows a simplified example of the algorithm for the location sharing practice. Using information gain and tf-idf I identified the most meaningful keywords for each practice and created sets of keywords. One set of keywords refers to the data type of the practices (e.g., for the location sharing practice `geo` and `gps`) and is used to extract all sentences from a policy that contain at least one of the keywords. On these extracted sentences the algorithm is using a second set of keywords that refers to the actions of a data practice (e.g., for the location sharing practice `share` and `partner`) to create unigram and bigram feature vectors [269]. Those feature vectors are then used to classify the practices. If no keywords are extracted, the classifier will select the negative class. The Porter stemmer is applied to all processed text.

For finding the most meaningful features as well as for the subsequent classifier tuning nested cross-validation with 75 policies separated into ten folds in the inner loop and 40 randomly selected policies as held out test set (policy test set) was performed. The inner cross-validation was used to select the optimal parameters during the classifier tuning phase and the held out policy test set for the final measure of classification performance. I stratified the inner cross-validation to avoid misclassifications due to skewed classes. After evaluating the performance of the classifiers with the policy test set I added the test data to the training data for the final classifiers to be used in the large-scale analysis.

#### 6.1.3.3 Classification

During the tuning phase I prototyped various classifiers with scikit-learn [211], a Python library. Support vector machines and logistic regression had the best performance. I selected classification parameters individually for each data practice.

**Classifier Performance for Policy Test Set.** The classification results for the policy test set, shown in Table 6.2, suggest that the ML analysis of privacy policies is generally feasible. For the negative classifications the classifiers achieve *F-1*$_{neg}$ scores between 0.67 and 0.96. These scores are the most important measures for the task here because the identification of a privacy requirement inconsistency demands that a practice occurring in an app is *not* covered by its policy (§ 6.3.1). Consequently, it is less problematic that the sharing practices, which are skewed towards the negative classes, have relatively low *F-1*$_{pos}$ scores of 0.55 (SID) and 0.47 (SC) or could not be calculated (SL) due to a lack of true positives in the policy test set.

**Classification Results for Full App/Policy Set.** I applied the classifiers to the policies in the full app/policy set with $n = 9,050$ policies. I obtained this set by adjusting the full policy set ($n = 9,295$) to account for the fact that not every policy link might actually lead to a policy: for 40 randomly selected apps from the full policy set I checked whether the policy link in fact lead to a policy, which was the case for 97.5% (39/40) of links (with a CI of 0.87 to 1 at the 95% level). As the other 2.5% of links lead to some other page and would not contain any data practice descriptions, 2.5% of policies without any data practice descriptions were excluded leaving $n = 9,295 - 245 = 9,050$ policies in the full app/policy set. This adjustment increases the occurrence of positive data practice instances in the full app/policy set and keeps discrepancies between apps and policies at a conservative level as some apps with lacking data practice descriptions are now excluded.[22]

[22] I also checked the random sample of 40 apps for policies dynamically loaded via JavaScript because for such policies the feature

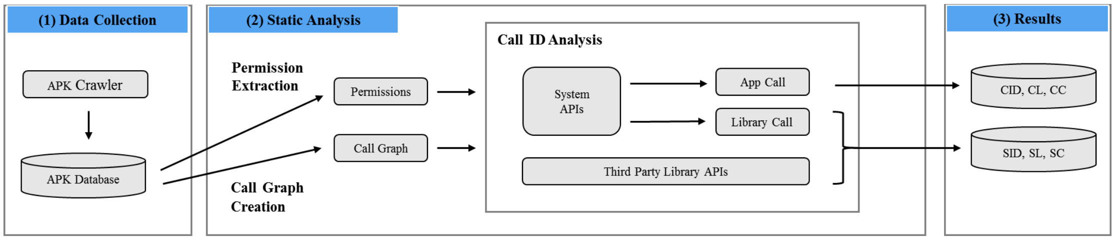


***Figure 6.5:*** *(1) The system first crawls the U.S. Google Play store for free apps. (2) Then, it performs for each app a static analysis. Specifically, it applies permission extraction, call graph creation, and call ID analysis, the latter of which is based on Android system and third party APIs. (3) Finally, results for the collection and sharing practices are generated and stored.*

It appears that many privacy policies fail to satisfy privacy requirements. Most notably, per Table 6.2, only 46% describe the notification process for policy changes, a mandatory requirement for apps that do not exclude California and Delaware residents. Similarly, only 36% of policies contain a statement on user access, edit, and deletion rights, which COPPA requires for childrens' apps, that is, apps intended for children or known to be used by children. For the sharing practices I expected more policies to engage in the SID, SL, and SC practices. The respective 10%, 12%, and 6% are rather small percentages for a presumably widely occurring practice, especially, given that the focus is on policies of free apps that often rely on targeted advertising.

**Runtime Performance and Failure Rate.** The analysis of all practices for the policies in the full app/policy set required about half an hour in total running ten threads in parallel on an Amazon Web Services (AWS) EC2 instance m4.4xlarge with 2.4 GHz Intel Xeon E5-2676 v3 (Haswell), 16 vCPU, and 64 GiB memory [31]. The feature extraction took up the majority of time and the training and classification finished in about one minute. There was no failure in extracting policy features or analyzing policies.

extraction would fail. However, as neither of the policies in the sample was loaded dynamically, I do not make an adjustment in this regard. Note, though, in the system built for the California Department of Justice (§ 6.4) functionality for analyzing dynamically loaded policies was implemented as well.

## 6.2 Mobile App Analysis

In order to compare the policy analysis results to what apps actually do according to their code let us now turn to the app analysis approach. Let us first discuss the system design (§ 6.2.1) and follow up with the analysis results (§ 6.2.2).

### 6.2.1 System Design

The app analysis system is based on Androguard [34], an open source static analysis tool written in Python that provides extensible analytical functionality. Apart from the manual intervention in the construction and testing phase the system's analysis is fully automated. Figure 6.5 shows a sketch of the system architecture. A brief example for sharing of device IDs will convey the basic program flow of the data-driven static analysis.

For each APK the system builds an API invocation map, which is utilized as a partial call graph (call graph creation). To illustrate the functionality with an example, for the practice of sharing device IDs (SID) all calls to the `android.telephony.Telephony Manager.getDeviceId` API are included in the call graph because the caller can use it to request a device ID. All calls to this and other APIs that can be used to request a device ID are included in the call graph and passed to the identification routine (call ID analysis), which checks the package names of the callers against the package names of selected third party libraries that are analyzed. In order to make use of the `getDeviceId` API a library needs the `READ_PHONE_STATE` permission. Only if the analysis detects that the library has the required permission (permission extraction), the app is classified as sharing device IDs with third parties.[23] I identified relevant Android API calls for the types of information and the permission each call requires by using PScout [41].

The static analysis is informed by a manual evaluation of Android and third party APIs. Because sharing of data most often occurs through third party libraries [97], it is appropriate to leverage the insight that the observation of data sharing for a given library allows extension of that result to all apps using the same library [129]. As the top libraries have the farthest reach [129] I focus on those. I used AppBrain [36] to identify the ten most popular libraries by app count that

[23] Android's permission model as of Android 6.0 does not distinguish between permissions for an app and permissions for a library, which, thus, can request all permissions of the app.

process device ID, location, or contact data. To the extent they were accessible I also analyzed previous library versions dating back to 2011. After all, apps sometimes continue to use older library versions even after the library API has been updated. For each library I opened a developer account, created a sample app, and observed the data flows from the developer perspective. For these apps as well as for a sample of Google Play store apps that implement the selected libraries I additionally observed their behavior from the outside by capturing and decrypting packets via a man-in-the-middle attack and a fake certificate [216]. I also analyzed library documentations. These exercises enable to see which data types were sent out to which third parties.

### 6.2.2 Analysis Results

**Performance Results for App Test Set.** Before getting into the analysis results for the full app set I discuss the performance of the app analysis on a set of 40 apps (app test set), which were selected randomly from the publishers in the policy test set to obtain corresponding app/policy test pairs for the later analysis of privacy requirement inconsistencies (§ 6.3.1). To check whether the data practices in the test apps were correctly analyzed by the system I dynamically observed and decrypted the data flows from the test apps to first and third parties, performed a manual static analysis for each test app with Androguard [34], and studied the documentations of third party libraries. Thus, for example, it is possible to infer from the proper implementation of a given library that data is shared as explained in the library's documentation. I did not measure performance based on micro-benchmarks, such as DroidBench [38], as those do not fully cover the data practices investigated here.

In the context of privacy requirement inconsistencies (§ 6.3.1) correctly identifying positive instances of apps' collection and sharing practices is more relevant than identifying negative instances because only practices that are occurring in an app need to be covered in a policy. Thus, the results for the data practices with rarely occurring positive test cases are especially noteworthy: CC, SL, and SC all reached $F\text{-}1_{pos} = 1$ indicating that the static analysis is able to identify positive practices even if they rarely occur. Further, the F-$1_{pos}$ scores, averaging to a mean of 0.96, show the overall reliability of the approach. For all practices the accuracy is also above the baseline of always selecting the test set class that occurs the most for a given practice. Overall, as shown in Table 6.4, the results demonstrate the general reliability of the analysis.

| ***3rd Party Library*** |
|---|
| Crashlytics/Fabric |
| Crittercism/Aptel. |
| Flurry Analytics |
| Google Analytics |
| Umeng |
| AdMob* |
| InMobi* |
| MoPub* |
| MillennialMedia* |
| StartApp* |

***Table 6.3:*** *Analytics and ad* libraries.*

| ***Pract*** | ***Base*** | ***95% CI*** | ***Prec***$_{pos}$ | ***Rec***$_{pos}$ | ***F-1***$_{pos}$ | ***F-1***$_{neg}$ | ***Pos***$_{w/\ pol}$ | ***Pos***$_{w/o\ pol}$ |
|---|---|---|---|---|---|---|---|---|
| | *(n=40)* | *(n=40)* | *(n=40)* | *(n=40)* | *(n=40)* | *(n=40)* | *(n=9,295)* | *(n=8,696)* |
| CID | 0.8 | 0.76–0.97 | 0.89 | 1 | **0.94** | 0.67 | 95% | 87% |
| CL | 0.55 | 0.64–0.91 | 0.73 | 1 | **0.85** | 0.71 | 66% | 49% |
| CC | 0.78 | 0.91–1 | 1 | 1 | **1** | 1 | **25%** | 12% |
| SID | 0.68 | 0.83–0.99 | 1 | 0.93 | **0.96** | 0.93 | **71%** | **62%** |
| SL | 0.93 | 0.91–1 | 1 | 1 | **1** | 1 | 20% | 16% |
| SC | 0.98 | 0.91–1 | 1 | 1 | **1** | 1 | **2%** | 0% |

***Table 6.4:*** *App analysis results for the app test set (n=40) and the percentages of practices' positive classifications for the full app set (n=17,991). More specifically,* $Pos\_w/\ pol$ *and* $Pos\_w/o\ pol$ *are showing what percentage of apps engage in a given practice for the subset of apps in the full app set with a policy (n=9,295) and without a policy (n=8,696), respectively. Precision, recall, and F-1 score with the* $_{pos}$ *and* $_{neg}$ *subscripts refer to the scores for the positive and negative classes.*

**Data Practice Results for Full App Set.** For all six data practices there is a mean of 2.79 positive practices per app for apps with policies and 2.27 cases for apps without policies. As all practices generally need to be described in a policy (§ 6.1.1), it is already clear that there are substantial amounts of inconsistencies between apps and policies simply due to missing policies. For example, sharing of device IDs was detected in 62% of apps that did not have a policy, which, consequently, appear to be in non-compliance of privacy requirements. Furthermore, for apps that had a policy only 10% disclosed the SID practice (§ 6.1.3.2) while it occurred in 71% of apps. Thus, 61% of those apps appear to be in non-compliance as well. The only practices for which it is not possible to immediately infer the existence of inconsistencies are the CC and SC practices with policy disclosures of 56% and 6% and occurrences in apps of 25% and 2%, respectively. There could be two reasons for this finding.

First, there could be a higher sensitivity among app publishers to notify users of practices related to contact data compared to practices that involve device identifiers and location data. Second, different from device ID and location data, contact information is often provided by the user through the app interface bypassing the APIs considered for the static analysis (most notably, the `android.accounts.AccountManager.getAccounts` API). Thus, the result demonstrates that the analysis approach has to be custom-tailored to each data type and that the user interface should receive heightened attention in future works [235]. It also illustrates that the results only represent a lower bound, particularly, for the sharing practices (SID, SL, SC), which are limited to data sent to the ten publishers of the libraries in Table 6.3.

**Limitations.** I want to point out various limitations of the approach introduced here. At the outset the approach is generally subject to the same limitations that all static analysis techniques for Android are facing, most notably, the difficulties of analyzing native code [25], obfuscated code [181], and indirect techniques (e.g., reflection). However, there are various considerations that ameliorate exposure of the approach to these challenges. First, if an app or a library uses native code, it cannot hide its access to Android System APIs [129]. In addition, the use of native code in ad libraries is minimal [181]. Indeed, there was rarely native code observed in the analysis. Similarly, the need to interact with a variety of app developers effectively prohibits the use of indirect techniques [50]. However, code obfuscation presented in fact an obstacle. The static analysis failed in 0.4% (64/18,055) due to obfuscation (i.e., an app's Dex file completely in bytecode). However, the failure rate improves over the closest comparable rate of 21% [235].

It is a further limitation of the approach suggested here that the identification of data practices is limited to observations from the outside (e.g., server-side code is not considered). While this limitation is not a problem for companies' analysis of their own apps, which I see as a major application of the system, it can become prevalent for regulators, for instance. In many cases decrypting HTTPS traffic via a man-in-the-middle attack and a fake certificate will shed some light. However, it appears that some publishers are applying encryption inside their app or library. In those cases, the analysis will need to rely on inferring the data practice in question indirectly. For example, it remains possible to check whether a library is properly implemented in an app according to the library's documentation, which lends evidence to the inference that the app indeed makes use of the documented data practices.

Also, the results for the sharing practices only refer to the ten third parties listed in Table 6.3. The percentages for sharing of contacts, device IDs, or locations would almost certainly be higher if additional libraries are considered. In addition, the definition of sharing data with a third party only encompasses sharing data with ad networks and analytics libraries. However, as it was shown that ad libraries are the recipients of data in 65% of all cases [129], I believe that this definition covers a substantial portion of sharing practices. It should be finally noted that the investigation does not include collection or sharing of data that occurs offline or at the backend. However, as the analysis already identifies a substantial percentage of non-compliant apps, I think that there is value in the introduced techniques even with these limitations.

**Runtime Performance.** In terms of runtime performance, using ten threads in parallel on an AWS EC2 instance m4.10xlarge with 2.4 GHz Intel Xeon E5-2676 v3 (Haswell), 40 vCPU, and 160 GiB memory [31] the analysis of all 17,991 APKs took about 31 hours. The mean runtime is 6.2 seconds per APK analysis.

## 6.3 Privacy Requirement Inconsistencies

In this section I marry the policy (§ 6.1) and app (§ 6.2) analyses. I explore to which extent apps are non-compliant with privacy requirements (§ 6.3.1) and show how app metadata can be used to zoom in on sets of apps that have a higher likelihood of non-compliance (§ 6.3.2).

### 6.3.1 Identifying Individual Privacy Requirement Inconsistencies

Non-compliance of apps with privacy requirements is not necessarily based on malicious behavior of software developers.

**Privacy Requirement Inconsistencies** App developers were found to often lack an understanding of privacy-best practices [44], and it could be that many of the privacy requirement inconsistencies are a result of this lack of understanding. Many developers struggle to understand what type of data third parties receive, and with limited time and resources even self-described privacy advocates and security experts grapple with implementing privacy and security protection [44]. In this regard, the analysis approach can provide developers with a valuable indicator for instances of non-compliance. For identifying privacy requirement inconsistencies positive app classes and negative policy classes are relevant. In other words, if a data practice does not occur in an app, it does not need policy coverage because there can be no privacy requirement inconsistency to begin with. Similarly, if a user is notified about a data practice in a policy, it is irrelevant whether the practice is implemented in the app or not. Either way, the app is covered by the policy. Based on these insights the performance of the approach is analyzed.

**Performance Results for App/Policy Test Set.** To check the performance of the system for correctly identifying privacy requirement inconsistencies a test set with corresponding app/policy pairs (app/policy test set) is used. The set contains the 40 apps from the app test set (§ 6.2.2) and their associated policies from the policy test set (§ 6.1.3.3). An app and a policy are associated if the app or its Play store page links to the policy or if the policy explicitly declares itself applicable to mobile apps. As only 23 policies satisfy this requirement some policies are associated with multiple apps. Making 240 classifications in the app/policy test set—that is, classifying six practices for each of the 40 app/policy pairs—the system correctly identified 32 privacy requirement inconsistencies (TP). It also returned five false negatives (FN), 10 false positives (FP), and 193 true negatives (TN). As shown in Table 6.5, accuracy results range between 0.86 and 1 with a mean of 0.94. Although not fully comparable, AsDroid achieved an accuracy of 0.79 for detecting stealthy behavior [150] and Slavin et al. [235] report an accuracy of 0.8 for detecting discrepancies between app behavior and policy descriptions.

The F-1$_{pos}$ scores for the analysis, ranging from 0.7 to 1, indicate the overall reliable identification of privacy requirement inconsistencies. While I think that these results are generally promising, precision value of $Prec_{pos} = 0.54$ for the CL practice is relatively low. This result illustrates a broader point that is applicable beyond location collection. False positives seem to occur because the analysis takes into account too many Android system APIs that are only occasionally used for purposes of the data practice in question. Despite the believe that it is better to err on the side of false positives, which is especially true for an auditing system [129], in hindsight I probably would have left out some APIs. The opposite problem seems to occur in the SID practice. I included too few relevant APIs. In this regard, it is a challenge to identify a set of APIs that at the same time captures the bulk of cases for a given practice without being over-inclusive.

**Privacy Requirement Inconsistencies for Full App/Policy Set.** As indicated by the high inconsistency percentages shown in Table 6.5, privacy requirement inconsistencies seem to be a widespread phenomenon. Specifically, collection of device IDs and locations as well as sharing of device IDs are practices that appear to be inconsistent for 50%, 41%, and 63% of apps, respectively. It is further noteworthy that for SL and SC nearly every detection of the practice goes hand in hand with a privacy requirement inconsistency. For the apps that share location information (20%, per Table 6.4) nearly all (17%, per Table 6.5) do not properly disclose such sharing. Similarly, for the 2% of apps that share contact data only a handful provide sufficient disclosure. For

| ***Practice*** | ***Acc***<br>*(n=40)* | ***Acc**$_{pol}$ · **Acc**$_{app}$*<br>*(n=40)* | ***95% CI***<br>*(n=40)* | ***Prec**$_{pos}$*<br>*(n=40)* | ***Rec**$_{pos}$*<br>*(n=40)* | ***F-1**$_{pos}$*<br>*(n=40)* | ***F-1**$_{neg}$*<br>*(n=40)* | ***MCC***<br>*(n=40)* | ***TP, FP, TN, FN***<br>*(n=40)* | ***Inconsist***<br>*(n=9,050)* |
|---|---|---|---|---|---|---|---|---|---|---|
| CID | **0.95** | 0.74 | 0.83–0.99 | 0.75 | 1 | **0.86** | 0.97 | 0.84 | 6, 2, 32, 0 | **50%** |
| CL | **0.83** | 0.7 | 0.67–0.93 | **0.54** | 1 | **0.7** | 0.88 | 0.65 | 8, 7, 25, 0 | **41%** |
| CC | **1** | 0.88 | 0.91–1 | - | - | - | 1 | - | 0, 0, 40, 0 | **9%** |
| SID | **0.85** | 0.84 | 0.7–0.94 | 0.93 | 0.74 | **0.82** | 0.87 | 0.71 | 14, 1, 20, 5 | **63%** |
| SL | **1** | 0.93 | 0.91–1 | 1 | 1 | **1** | 1 | 1 | 3, 0, 37, 0 | **17%** |
| SC | **1** | 0.78 | 0.91–1 | 1 | 1 | **1** | 1 | 1 | 1, 0, 39, 0 | **2%** |

***Table 6.5:*** *Results for identifying privacy requirement inconsistencies in the app/policy test set (n=40) and the percentage of privacy requirements inconsistencies for all 9,050 app/policy pairs (Inconsistency). Assuming independence of policy and app accuracies, Acc$_{pol}$· Acc$_{app}$, that is, the product of policy analysis accuracy () and app analysis accuracy (), indicates worse results than the directly measured accuracy. However, the Matthews correlation coefficient (MCC), a measure that is particularly insightful for evaluating classifier performance in skewed classes, indicates a positive correlation between the observed and predicted classes.*

the majority of those cases it is not even necessary to perform a policy analysis to detect privacy requirement inconsistencies.

From a big picture view, the average number of 1.83 inconsistencies per app is high compared to the closest previous averages with 0.62 (113/182) cases of stealthy behavior [150] and potential privacy violations of 1.2 (24/20) [96] and 0.71 (341/477) [235]. Figure 6.6 shows the details. It should also be noted that for apps without a policy essentially every data collection or sharing practice causes an inconsistency. For example, all 62% apps without a policy that share device IDs (Table 6.4) are non-compliant. Thus, overall the results suggest a broad level of inconsistency between apps and policies. As the system is currently evaluated for its use in privacy enforcement with the California Department of Justice (§ 6.4) I did not yet contact any affected app publishers of the findings.

### 6.3.2 Predicting Inconsistencies from App Metadata for Groups of Apps

Analyzing individual apps for privacy requirement compliance at scale is a time- and resource-intensive task. Thus, it is worthwhile to first estimate an app population's non-compliance as a whole before digging deep into individual analyses. My suggestion is to systematically explore app metadata for correlations with privacy requirement inconsistencies based on statistical models.

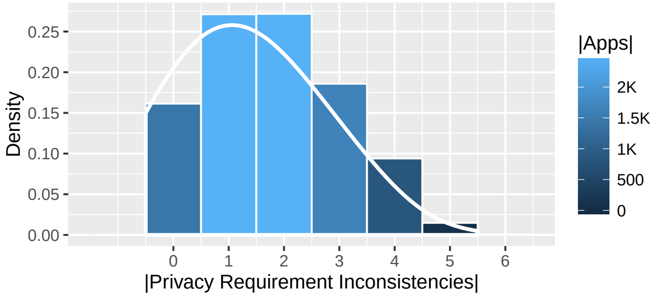


***Figure 6.6:*** *For the full app/policy set (n = 9,050) 2,455 apps have one inconsistency, 2,460 have two, and only 1,461 adhere completely to their policy. Each app exhibits a mean of 1.83 (16,536/9,050) inconsistencies (with the following means per data practice: CID: 0.5, CL: 0.41, CC: 0.09, SID: 0.63, SL: 0.17, SC: 0.02).*

This broad macro analysis supplements the individual app analysis and reveals areas of concern on which, for example, privacy activists can focus on. To illustrate this idea I evaluate a binary logistic regression model that determines the dependence of whether an app has a privacy requirement inconsistency (the dependent variable) from six Play store app metadata variables (the independent variables). The results, shown in Table 6.6, demonstrate correlations at various statistical significance levels with p values ranging from 0.0001 to 0.08. Particularly, with an increase in the number of user ratings the probability of privacy requirement inconsistencies decreases. There is also a decreasing effect for apps with a badge and for apps whose content has not yet been rated.

Interestingly, apps with higher overall Google Play store scores do not have lower odds for privacy requirement inconsistencies. In fact, the opposite is true. With an increase in the overall score the odds of an inconsistency become higher. An increase of the overall score by one unit, e.g., from 3.1 to 4.1 (on a scale of 1 through 5), increases the odds of an inconsistency by a factor of 1.4. A reason could be that highly rated apps provide functionality and personalization based on user data, whose processing is insufficiently described in their privacy policies. At least, users do not seem to rate apps based on privacy considerations. I found the word “privacy” in only 1% (220/17,991) of all app reviews. Beyond an app’s score the odds for a privacy requirement inconsistency also increase for apps that feature in-app purchases or interactive elements. Also, supplementing the model with category information reveals that the odds for an inconsistency significantly (p $\leq 0.05$) surge for apps in the Finance, Health & Fitness, Photography, and Travel

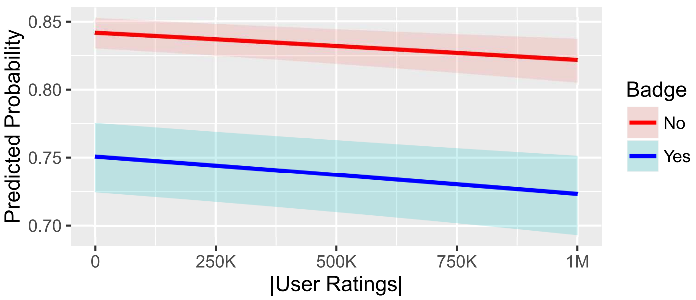


***Figure 6.7:*** *The graph shows the predicted probability of an app having a privacy requirement inconsistency dependent on the number of user ratings and the assignment of a badge. The overall score is held at the mean and in-app purchases, interactive elements, and unrated content are held to be not present. The shaded areas identify the profile likelihood CIs at the 95% level.*

& Local categories while they decrease for apps in the Libraries & Demo category.

In order to evaluate the overall model fit based on statistical significance I checked whether the model with independent variables (omitting the category variables) had significantly better fit than a null model (that is, a model with the intercept only). The result of a chi-square value of 151.03 with six degrees of freedom and value of p $\leq 0.001$ indicates that the model has indeed significantly better fit than the null model. To see the impact of selected aspects of the model it is useful to illustrate the predicted probabilities. An example is contained in Figure 6.7. Apps with a Top Developer or Editor's Choice badge have a nearly 10% lower probability of a privacy requirement inconsistency. That probability further decreases with more user ratings for both apps with and without badge.

## 6.4 Case Study: Assisting the California Department of Justice in enforcing CalOPPA

Currently the system's use in enforcement actions is evaluated with the California Department of Justice, specifically, the Office of the Attorney General, on evaluating the system's suitability for supplementing the enforcement of CalOPPA. To that end, a custom-made version of the system is implemented for the Attorney General (§ 6.4.1) and various new analysis functionality is being added (§ 6.4.3). The preliminary feedback received up to this point on the performance of the

| ***Variable*** | ***Pos*** | ***p value*** | ***OR*** | ***95% CI*** |
|---|---|---|---|---|
| $\lvert$User Ratings$\rvert$ | 100% | **0.0001** | $\mathbf{0.\overline{9}}$ | 0.9999998–$0.\overline{9}$ |
| Overall Score | **100%** | <**0.0001** | **1.4** | 1.24–1.57 |
| Badge | 21% | <**0.0001** | 0.57 | 0.49–0.65 |
| In-app Purchases | 27% | **0.08** | 1.15 | 0.99–1.34 |
| Interactive Elm | 45% | <**0.0001** | 1.33 | 1.17–1.53 |
| Content Unrated | 5% | **0.002** | 0.68 | 0.53–0.87 |

***Table 6.6:*** *Significant variables for predicting apps' non-compliance with at least one privacy requirement as evaluated on the full app-policy set (n=9,050). Top Developer and Editor's Choice badges are assigned by Google. Interactive elements and unrated content refer to the respective ESRB classifications. Pos% are the percentages of positive cases (e.g., 100% apps have an overall score), OR is the odds ratio, and the 95% CI is the profile likelihood CI.*

system is encouraging and, as I believe, an indicator for making further strides towards the current direction (§ 6.4.3).

### 6.4.1 System Implementation

The system implementation for the Office of the Attorney General, shown in Figure 6.8, allows users to input either the package name or Play Store page URL of an app that they would like to analyze. The system then automatically runs the analysis and displays the results. Analyses can be requested for individual apps, however, the system also supports batch processing. The frontend of the system consists of a web application, which has the advantage that it does not require users to install any special software on their local computers. As it is easier to use a graphical user interface instead of a command line interface is used.

The system has to be available at all times, so that people working in the Office of the Attorney General would be able to analyze an app whenever it becomes necessary. As such a system is mostly resource-intensive when apps are being analyzed, however, otherwise stays idle, an AWS EC2 t2.large instance with up to 3.0 GHz Intel Xeon, 2 vCPU, and 8 GiB memory [31] is leveraged. Each of these instances has enough resources to analyze three apps in about ten minutes. Should it become necessary it is possible to immediately scale the number of instances

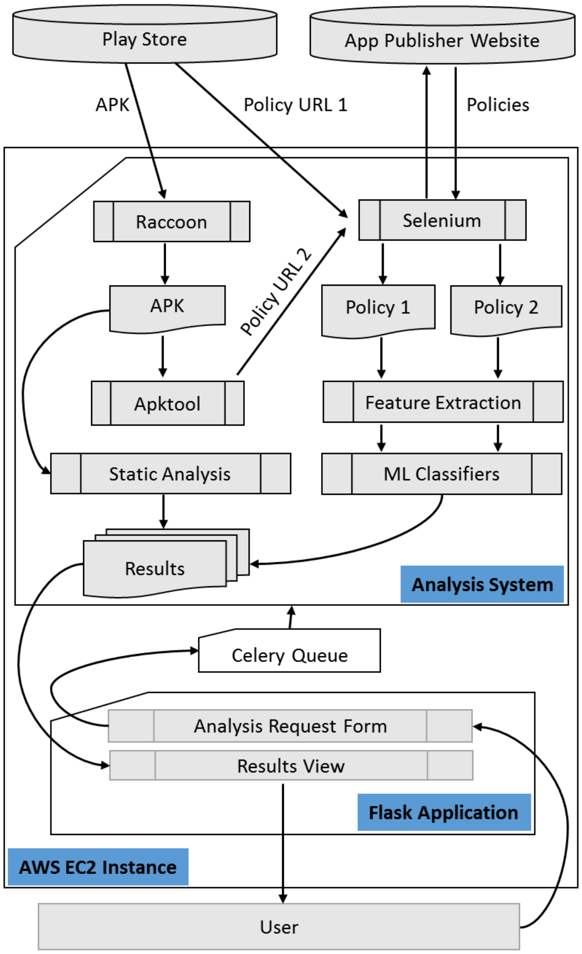
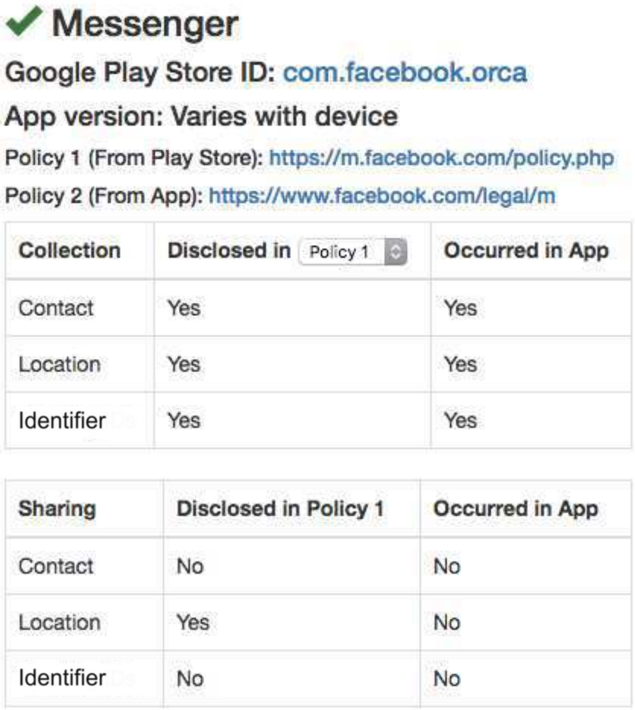


***Figure 6.8:*** *The system allows users to analyze apps for privacy policy compliance. An app can be subject to multiple privacy policies—for example, one policy linked to from inside the app and one linked to from the app's Play Store page. In these cases the app is checked against both policies.*

and increase the throughput quickly.

The interface applies the Flask Python web framework [226] running on the Apache web server [243] with a Web Server Gateway Interface module [92]. All analysis requests are added to a Celery task queue [40] that communicates with the Flask application using the RabbitMQ message broker [213]. When users are submitting analysis requests from the web interface, which is served by Flask, the requests are put into the task queue and executed one at a time. Once an analysis is finished the results are written to a JSON file, which is loaded by the Flask application, and displayed in the users' browsers.

In order to download APK files for requested apps from the Play store the system makes use of Raccoon [204], which is also used in the original system. The system obtains the privacy policy links for the requested apps from their Play store pages. To download the websites that the links lead to a Firefox browser with Selenium [231] and PyVirtualDisplay [214] is automated, which allows to run a real browser without having a graphical user interface. Using a real browser instead of just crawling the HTML of the policy pages is advantageous as it is possible to obtain policies that are loaded dynamically via JavaScript.



✓ Messenger

Google Play Store ID: com.facebook.orca

App version: Varies with device

Policy 1 (From Play Store): https://m.facebook.com/policy.php

Policy 2 (From App): https://www.facebook.com/legal/m

| Collection | Disclosed in Policy 1 | Occurred in App |
|---|---|---|
| Contact | Yes | Yes |
| Location | Yes | Yes |
| Identifier | Yes | Yes |

| Sharing | Disclosed in Policy 1 | Occurred in App |
|---|---|---|
| Contact | No | No |
| Location | Yes | No |
| Identifier | No | No |

***Figure 6.9:*** *A screenshot from the web application's results view for the analysis of the Facebook Messenger app, which was not flagged for any inconsistencies.*

After the website with the privacy policy is downloaded any elements that are not part of the policy, such as advertisements or page navigation elements, are removed. The system then runs the feature extraction routines (§ 6.1.3.2) as well as ML classifiers (§ 6.1.3.3) on the policy and the static analysis (§ 6.2) on the downloaded APK. Finally, the results are displayed to the user with flags raised for all privacy requirement inconsistencies. Figure 6.9 shows an example of the results view.

### 6.4.2 Adding Additional Functionality

Tailoring the system for use by the Office of the California Attorney General requires a strong focus on usability. Various users come from non-technical backgrounds and were easily thrown off by some of the terminology used in the presentation of the analysis results. For example, instead of using the terms “true” and “false” for the occurrence and absence of a practice, they instead found the terms “yes” and “no” clearer. For these types of usability refinements as well as for the other changes to the system an iterative development cycle is used where the future development of the system is based on weekly user feedback.

Users were also interested in receiving additional information, which lead us to expand the analysis. For example, one additional piece of information is the breakdown of third parties in the sharing practices. The initial version of the report simply showed what information was being shared without mentioning the third parties. For example, a report would show that the user’s contact and device ID were being shared without disclosing that, say, contact information is shared with InMobi and the device ID with Crashlytics. However, this distinction is important under an interpretation of CalOPPA according to which the sharing of contact information makes a stronger case.[24]

Given the importance of contact information, the implementation of additional functionality to detect further instances of contact sharing is finalized. As I believe that the relatively low detection rate for the collection and sharing of contact information is due to the fact that such information is often supplied by the user, which the original system does not check (§ 6.2.2), the system will be enhanced in this regard. In particular, leveraging the Facebook Login library [103] that is included in many apps and that, by default, gives the app access to a user’s name and Facebook ID, which can be used to identify and contact a user, is instructive. The usage of Facebook Login functionality can be detected in an app by extracting the app’s manifest and resource files with Apktool [246] and then searching for signatures that would be required for Facebook login. These include an activity or extent filter dedicated to the login interface, a login button on the layout, and the invocation of an initialization, destruction, or configuration routine from the Facebook Login library.

[24] Compare Cal. Bus. & Prof. Code §22577(a)(3) and (7).

Another added feature is the retrieval of privacy policy links from inside apps. The initial policy crawler had just downloaded policies that were linked from an app's Play store page. As the Attorney General provided guidance to app publishers for linking the policy from both the Play store as well as from inside the app [61], the new approach is intended to cover both possibilities. The links in an app can be found by extracting strings from the APK file using Apktool and then extracting URLs from within these strings which contain keywords, such as "privacy." If a link inside an app differs from the app's Play store policy link or if there are multiple links in the app, the system analyzes the documents those links are leading to as well. The interface allows the user to pick which policy to show results for.

### 6.4.3 Preliminary Feedback

The users of the system at the Office of the California Attorney General reported that the system has the potential to increase their productivity. Particularly, as they have limited resources it can give them guidance on the areas of mobile apps to focus on. Since they have limited time, they can put less effort into analyzing practices in apps for which the system does not find inconsistencies. Instead, they can spend most of their time examining the specific inconsistencies in apps that are flagged. In addition, the users expressed that the system was useful for showing them the current overall state of CalOPPA compliance. For example, the analysis results alerted them to the many policies which use vague language in the descriptions of their collection and sharing practices.

## 6.5 Conclusion

The law of notice and choice is intended to enable enforcement of data practices in mobile apps and other online services. However, verifying whether an app actually behaves according to the law and its privacy policy is decisively hard. To alleviate this problem I propose the use of an automated analysis system based on machine learning and static analysis. The system advances app privacy in three main thrusts: it increases transparency for otherwise largely opaque data practices, offers the scalability necessary for potentially making an impact on the app eco-system as a whole, and provides a first step towards the automation of privacy requirement checks.

The results suggest the occurrence of privacy requirement inconsistencies on a large scale.

This finding raises the question of extending the approach to other areas. While I focused on the Android platform, the approach is, in principle, adaptable to other mobile platforms, for example, for iOS using [86; 174]. The approach can also be made workable for the analysis of websites' data practices, e.g., leveraging [233], for which first and third party cookies and other tracking mechanisms can be observed to evaluate collection and sharing of data. The Internet of Things and sensor data represent other rich use cases. Fitness trackers with APIs for monitoring the heart rate and other body sensor data could be a first step towards exploring these areas.

I believe that it is necessary to develop public policy and law alongside the privacy requirement analysis system I propose. In my opinion, regulators are moving in the right direction by pushing for app store standardization [60] and early enforcement of potentially invasive privacy practices [113]. Approaches like the one proposed here can relieve regulators through automation and allow them to focus their limited resources to move from a purely reactionary approach towards systematic oversight. As I also think that many software publishers do not intend non-compliance with privacy requirements, but rather lose track of their obligations or are unaware of them, I also advocate for implementation of a privacy law check in software development tools and as part of the app vetting process in app stores.

# Chapter 7

# Cross-device Tracking

As online users are increasingly accessing the Internet from multiple devices a new form of tracking is emerging: cross-device tracking. This practice—in most cases for purposes of advertising—is aimed at crossing the boundary between a user's individual devices and browsers. It establishes a person-centric approach to recognize users across devices and seeks to combine the input from the various data sources into a single comprehensive user profile. By its very nature such tracking across devices can reveal a complete picture of a person and, thus, become more privacy-invasive than the siloed tracking via HTTP cookies or other traditional tracking mechanisms. Cross-device tracking is also a form of tracking in which ML techniques play a major role for detecting which devices belong to the same user.

To my knowledge no rigorous privacy analysis of cross-device tracking has been conducted. Thus, the work presented here should be understood as a foundational privacy analysis from which mechanisms for privacy protection can be developed. Possible privacy mechanisms could involve notifying users of the cross-device trackers' occurrences on apps or websites and developing and opt-out model across devices that is convenient to use without hampering legitimate industry interests. However, in order to develop meaningful privacy protection mechanisms a variety of basic questions have to be further explored: How can cross-device tracking be detected? What are the methods used by cross-device tracking companies? Where and to which extent does cross-device tracking occur? Is the current self-regulatory approach promising or should regulators and lawmakers step in? In the following I aim to provide some basic insight into these fundamental questions to ultimately develop sound privacy protection mechanisms.

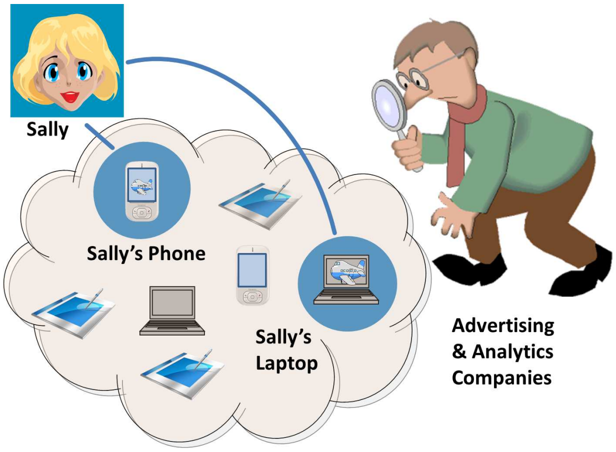


***Figure 7.1:*** *Identifying Sally's phone and desktop among the other devices on the Internet based on device and software metadata.*

In particular, I demonstrate a method to detect the occurrence of cross-device tracking, which can be implemented in an ML classifier. Also, based on cross-device tracking data that I collected from 126 Internet users I explore the frequency of trackers capable of crossing device boundaries. I show that the similarity of IP addresses and Internet history of a user across devices gives rise to a matching rate of F-1 = 0.91 for connecting a mobile to a desktop device. This finding is particularly noteworthy in light of the increase in learning power that ad networks and analytics services can achieve by leveraging Internet history from more than one device. Given these privacy implications of cross-device tracking I also examine compliance with applicable self-regulation for 40 cross-device companies and find that some are not transparent about their practices. The work presented here provides a foundation for use in ML technologies, particularly, personal privacy assistants [6].

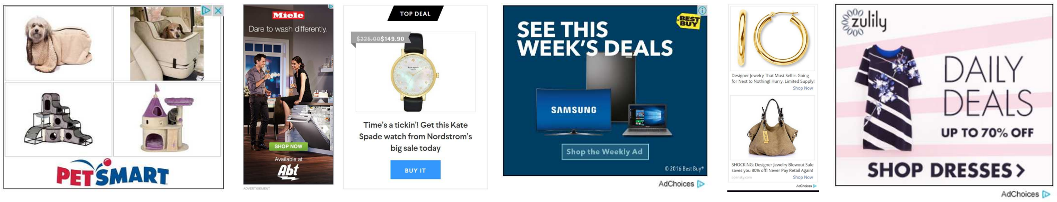
In a study commissioned by Facebook the Gesellschaft für Konsumforschung revealed that in the U.S. and the U.K. 60% of online adults use at least two devices every day [128]. Also, more than 40% of Internet users start an activity on one device and finish it on another [128]. A similar study by Google showed that 98% of surveyed users in the U.S. move between devices on the same day and 90% use multiple screens sequentially to accomplish a task over time [136]. The increased Internet-connectivity of devices, particularly, of smartphones, enables ad networks, analytics services, and other Internet companies to learn much more about their users than they previously could.[1] I want to shed some light on the privacy implications of cross-device tracking practices.

Cross-device tracking (sometimes also referred to as cross-device retargeting [206], cross-platform optimization [35], or multi-platform tracking [67]) is the tracing of an individual's usage of the Internet on multiple devices and combining all resulting information into one comprehensive user profile. Ad networks and analytics services are at the forefront of cross-device tracking because it enables more efficient user targeting and attributing conversion. As illustrated in Figure 7.1, ad networks could deliver ads to Sally on her desktop for a flight whose booking she abandoned earlier on her phone. Cross-device tracking goes beyond the tracking of standalone devices but rather aims to identify all devices of a person. The correlation of multiple devices equates ultimately to the tracking of a *person* and, as such, is potentially much more privacy-invasive than the tracking of unconnected devices.

Currently, many companies in the ad space add cross-device functionality to their systems. At the outset two basic types of cross-device tracking can be distinguished: deterministic and probabilistic cross-device tracking. Deterministic cross-device tracking occurs in a first-party relationship in which a user's device can usually be identified with near certainty. For example, if a user logged into his or her social network account from one device and later logs into the same account from another device, the social network can assume that the two devices belong to the same user (save for any device sharing or account hacking). At that point the user can be traced through all websites and apps that make use of the social network's plugins, software development kits (sdks), or other tracking software—even when the user is not logged in[63].

For the most part I focus on probabilistic cross-device tracking. Different from its deterministic implementations, probabilistic techniques are used by services that only have a third-party relationship with Internet users. To that end ad networks and analytics providers are making use of cookies and other tracking mechanisms that are deployed on the websites and apps of the publishers they cooperate with and that have a first-party user relationship. Applying machine learning they then correlate the various data streams to identify which ones belong to the same users. However, as I will discuss (§ refbreadth) probabilistic and deterministic cross-device tracking are not

[1] I am using the term ad network loosely encompassing ad exchanges, demand side platforms, supply side platforms, and ad tech companies.



**A. PetSmart**
nytimes.com
adsense.com
Google AdSense
Google Display Network

**B. Miele/Abt**
latimes.com
as.chango.com
Rubicon Project
Tapad

**C. Kate Spade**
aol.com
redirectingat.com
Skimlinks
Lotame

**D. Best Buy/Samsung**
aol.com
adsense.com
Google AdSense
Google Display Network

**E. Jewelry**
aol.com
opensky.com
The OpenSky Project
N/A

**F. zulily**
aol.com
r1.ace.advertising.com
Advertising.com/AOL
Advertising.com/AOL

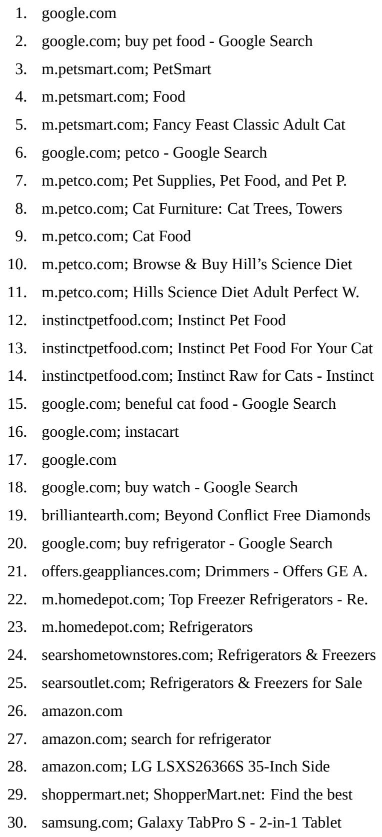
***Figure 7.2:*** *Screenshots of selected ads served to the desktop browser after visiting the websites shown below on the mobile browser. I had not seen any of these ads in the initial desktop browsing session two months earlier.*

mutually exclusive but are rather complementary as companies of different provenance cooperate with each other and exchange data.

Some of the ad networks that apply probabilistic cross-device tracking claim to match billions of devices [2]. Social networks and webmail providers have cross-device functionality naturally built into their services [3]. Given this depth and scope of cross-device tracking the FTC recently hosted a cross-device workshop [115]. The event facilitated an initial public discussion about the privacy implications of this new form of Internet tracking. Regulators, industry representatives, academics, and various other stakeholders discussed privacy risks, consumer transparency, and the extent to which industry self-regulation can provide appropriate privacy standards. As evidenced by a recent case the FTC is determined to enforce cross-device tracking violations [120], however, is hampered by insufficient insight into the used technologies [116].

## 7.1 Case Study: Detecting Cross-device Tracking

In order to discover how cross-device tracking actually occurs in the wild I conducted an exploratory case study. While I would not want to claim the experiment as a comprehensive survey of the cross-device phenomenon in the real world, I think that it provides sufficient evidence for its occurrence and underlines the basic workings of the ad industry in this realm. It provides a first glimpse into the emerging cross-device landscape highlighting some of its players and their partnerships. The underlying method of the experiment can be used to generically test for cross-device



1. google.com
2. google.com; buy pet food - Google Search
3. m.petsmart.com; PetSmart
4. m.petsmart.com; Food
5. m.petsmart.com; Fancy Feast Classic Adult Cat
6. google.com; petco - Google Search
7. m.petco.com; Pet Supplies, Pet Food, and Pet P.
8. m.petco.com; Cat Furniture: Cat Trees, Towers
9. m.petco.com; Cat Food
10. m.petco.com; Browse & Buy Hill's Science Diet
11. m.petco.com; Hills Science Diet Adult Perfect W.
12. instinctpetfood.com; Instinct Pet Food
13. instinctpetfood.com; Instinct Pet Food For Your Cat
14. instinctpetfood.com; Instinct Raw for Cats - Instinct
15. google.com; beneful cat food - Google Search
16. google.com; instacart
17. google.com
18. google.com; buy watch - Google Search
19. brilliantearth.com; Beyond Conflict Free Diamonds
20. google.com; buy refrigerator - Google Search
21. offers.geappliances.com; Drimmers - Offers GE A.
22. m.homedepot.com; Top Freezer Refrigerators - Re.
23. m.homedepot.com; Refrigerators
24. searshometownstores.com; Refrigerators & Freezers
25. searsoutlet.com; Refrigerators & Freezers for Sale
26. amazon.com
27. amazon.com; search for refrigerator
28. amazon.com; LG LSXS26366S 35-Inch Side
29. shoppermart.net; ShopperMart.net: Find the best
30. samsung.com; Galaxy TabPro S - 2-in-1 Tablet

***Figure 7.3:*** *The complete mobile browser history (without the visits to the Alexa-ranked homepages in the first two months of the experiment). The list shows the domains as well as the titles of the webpages, and the order reflects the order of visits.*

tracking without resorting to formally requesting information from cross-device companies or using the limited ad preference tools that a few of them provide (e.g., the BlueKai registry [205]). Particularly, the method can be implemented in an ML classifier.

**Establishing an IP Link.** I began the experiment by establishing an IP address connection between two devices—a desktop and a mobile device—that cross-device companies could pick up. During the time of the experiment I kept the IP address of the router to which both phone and desktop were connected unchanged. Using a fresh desktop browser without any cookies or other user data I visited the homepages of five random ad-financed news websites, that is, aol.com, latimes.com. nytimes.com, wsj.com, and washingtonpost.com (the test homepages), and observed the ads that were served. I refreshed each test homepage about ten times. The next two months I occasionally and randomly visited highly ranked homepages from the Alexa [28] rankings on the mobile browser; in total about 100 pages.

**Observing Cross-device Ads.** After the two months had passed I used the same mobile browser for visiting the websites shown in Figure 7.3. Specifically, after performing the shown Google searches I clicked on some ads of the Google results page. I then waited a few hours and switched to the desktop browser. Then, I accessed the test homepages from the start of the experiment, refreshed them about ten times, and took again note of the ads that were served. Some of the ads, neither of which were seen before on the test homepages, strikingly resembled the browsing history on the mobile. Figure 7.2 shows these ads and associated information, that is, the name of the ad (e.g., PetSmart), the domain on which it was served (e.g., nytimes.com), the domain of the tracker (e.g., adsense.com), the ad network serving the ad (e.g., Google AdSense), and the presumably involved cross-device tracking provider (e.g., Google Display Network).

It is noteworthy that the Kate Spade watch ad in Figure 7.2C. appeared nearly every time I refreshed the AOL homepage. I believe this ad was shown due to the earlier Google search for “buy watch” shown in line 18 of Figure 7.3. The PetSmart ad (occurring twice) and the Miele/Abt kitchen appliances ad (occurring once) also have a connection to the mobile browsing history. These results are indicative for the occurrence of cross-device tracking. Especially, given that Google’s AdSense network serves ads for 261 general ad categories, of which only three relate to pets, [134] the probability that I randomly received the PetSmart ads seems small. However, it should also be noted that the majority of ads served still seemed generic or served based on the website context.

**Observing Cross-browser Ads.** To examine the effect of switching browsers I opened a different unused browser on the desktop. As before I reloaded the five test homepages and observed the ads shown to us. Again the Kate Spade ad was shown nearly every time I refreshed AOL. However, I also received two ads related to the mobile browsing history that I had not seen earlier. The BestBuy/Samsung ad seems to be due to accessing the Samsung website and the jewelry ad may be served based on my click on an ad for diamonds. These results seem to imply that the desktop and mobile were matched independently of the browser used on the desktop. The same appears to hold for the browsers on the mobile. When I switched browsers on the phone I realized that the jewelry ad was served many times, which still was the case when deleting cookies, history, and cached files on the original phone browser. Curiously, the Zulily ad, which was served on both of the desktop browsers (and which does not appear to be related to the mobile history) kept being served despite clicking on a dismiss button.

**Identifying Ad Networks.** Based on the domains of the trackers that were observed from the ads it is possible to connect the ads to the ad networks that served them. One of the largest networks that serves ads across devices is the Google Display Network, which indeed receives ad inventory from one of two sources: the DoubleClick Ad Exchange or—as observed—AdSense [133]. Similarly, AOL has its own cross-device capabilities with its Advertising.com platform [35]. The ads served by Rubicon Project and Skimlinks demonstrate another common theme of the cross-device tracking environment. Smaller ad networks often have partnerships with other networks that have specialized cross-device capabilities; in case of Rubicon Project, Tapad [228], in case of Skimlinks, Lotame [183]. It should be cautioned, though, that the lack of insight into the ad serving backends presents an obstacle for making reliable claims on any cooperations beyond what is publicly known.

**Direction of Ad Serving.** Having checked ad serving from mobile to desktop I was also interested in the reverse direction. However, searching Google on the desktop for buying flowers, boats, and chocolate did not seem to lead to ads for these products on the mobile browser. I continued to see ads for refrigerators, jewelry, and pet food. An explanation for this result could be that ad networks attach more weight to history on the device to which an ad is served and less to other connected devices. However, this explanation does not seem likely to us. The reason is that after deleting all user information from the mobile browser I received generic ads and still no ads for flowers, boats, or chocolate. It seems that the ad serving was intentionally limited to one direction; from mobile to desktop. After all, while reasons for switching devices vary, in general, people tend to move from a smaller to a larger screen. [136; 128]. Also, since cross-device tracking is strongly campaign-driven it might simply a miss of campaigns at the time. Similarly, as for cross-device tracking from desktop to mobile I was not able to notice any correlation in ad serving when conducting the experiment with mobile apps.

## 7.2 The CDT Dataset

One of the major reasons for the scarcity of academic research in cross-device tracking—besides the field being in its infancy—is the lack of publicly available data.[2] Generally, only proprietary industry data exists. Therefore, I decided to collect my own cross-device tracking dataset (the CDT dataset), which will be provided in anonymized form to interested researchers for further exploration. I will also make available all data collection software. Here is how the data was collected.

**Data Collection Procedure.** Before starting the data collection Columbia University's Institutional Review Board permitted it. The collection system was built such that interested users could sign up on the project website, at which point a device fingerprint for each signed up device was taken. Users were asked to supply basic information on their demographics (e.g., age, gender, native language), interests (e.g., finance, games, shopping) [135], and personas (e.g., avid runners, bookworms, pet owners) [264]. In order to capture users' mobile and desktop history they were asked to install browser extensions and an app for automatically collecting such information. Details on the types of information are contained in the appendices.

A limitation of the data collection is that only Android phones are supported and users could only sign up if they were regularly making use of Android's native browser, Google Chrome, or the Samsung S-Browser. I did not support iOS or other operating systems. However, the app only requires Android 4.0.3 and runs without root access. Every minute it checks whether there is a new foreground app running on the device. If it detects a new app, it transmits a new app history data point to the server. It also checks every minute for new entries in the browsing history database of the phone's browsers, which will be transmitted accordingly.[3] On the desktop side I provided users of all operating systems with data collection browser extensions for Google Chrome, Mozilla Firefox, and Opera. At the conclusion of the study each user received an Amazon gift card for $15 to $50 depending on the length of their study participation.

**Dataset Characteristics.** The data collection covers a total of 126 users—125 desktop and 108

[2] The Drawbridge dataset [89] was only accessible to participants of the Drawbridge competition and limited in its use for purposes of the competition.

[3] For a few Google Chrome users on Android 6.0 or higher the system did not receive the full browsing history due to browser restrictions. I asked affected users to send us their history manually.

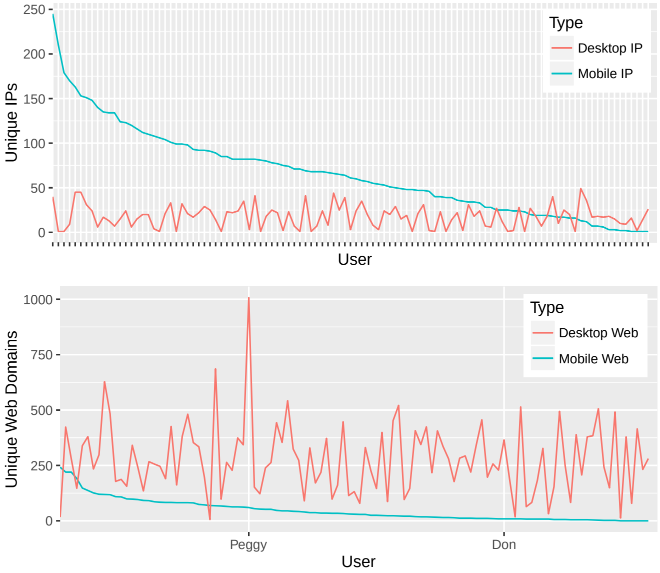


***Figure 7.4:*** *IPs (top) and domains (bottom) for each user in the dataset. For example, to the right of Don, 28 users had fewer than ten unique mobile domains; to the right of Peggy, 72 users visited 55 unique mobile domains or fewer.*

mobile users with an intersection of 107 users for which both were obtained.[4] While the data reflects reality accurately in the sense that not every Internet user has multiple devices, it fails to represent users in the real world with more than two devices. However, despite this limitation I believe that it faithfully reflects real multi-device usage on the Internet to a large extent because, according to an analysis of the Drawbridge data, the vast majority of mobile devices are associated with only one desktop browser [33]. Therefore, it seems plausible that probabilistic cross-device tracking companies are currently focusing on correlating two devices. Consequently, this understanding of the problem is adopted here as well.

118 users in the study were affiliates of Columbia University; mostly students and a few employees. Based on this population I believe that the dataset is more homogeneous than a similar dataset from, say, the general population of New York City. For the median user about three weeks of data were collected of which IP addresses and domains are of particular importance for probabilistic cross-device tracking because they can be used to measure the similarity between devices (§ 7.3.2). As illustrated in Figure 7.4, for mobile devices IP addresses harbor strong identifying

[4] Desktop users also include users of laptops.

| | Desktop Web | Mobile Web | Mobile Apps |
|---|---|---|---|
| Users | 125 | 102 | 104 |
| IPs | **1,994** | **5,784** | |
| Domains | **23,517** | **3,876** | **845** |

***Table 7.1:*** *Summary statistics for unique IPs, users, and domains in the CDT dataset in total.*

| | Desktop Web | Mobile Web | Mobile Apps |
|---|---|---|---|
| Days | 19, **22**, 26 | 9, **17**, 23 | 19, **22**, 24 |
| IPs | 6, **17**, 24 | 25, **63**, 92 | |
| Domains | 149, **251**, 374 | 9, **31**, 70 | 19, **30**, 44 |

***Table 7.2:*** *Summary statistics for the CDT dataset per user showing the 25th, 50th, 75th percentiles.*

potential while desktops are often characterized by their domains. However, there does not seem to be a correlation between desktop and mobile devices to the effect that lower usage of one would imply more usage of the other or that both are used to an equal degree.

Tables 7.1 and 7.2 show selected summary statistics for the CDT dataset. It is noteworthy that the total unique mobile IP count (5,784) nearly triples the total unique desktop IP count (1,994), which reflects mobile usage on the go.[5] However, the high number of unique desktop domains (23,517), compared to the homogeneous usage of apps (845), underscores the diversity of desktop browsing. While it is much more diverse in terms of domains (3,876), mobile web usage pales compared to app usage. As shown by the 25th, 50th, and 75th percentiles, the median user accessed the mobile web only for 17 days visiting only 31 unique domains.[6] While app usage is more popular with a median of 22 days, the median usage of 30 unique apps is comparable to that of the mobile web. However, the median number of unique mobile IPs (63) more than triples desktop IPs (17) likely due to usage on the go.

[5] As there was not a mobile IP for every transmitted data point the unique mobile IP count is likely even higher.

[6] A day counts if it had at least one desktop web, mobile web, or app access, respectively. Also, uniqueness of a domain depends on its top and second level, e.g., linkedin.com and blog.linkedin.com are the same domain.

| | ***Desktop Devices*** | ***Mobile Devices*** |
|---|---|---|
| User Agent | **4.46**, 0.64, 4.96 | **6.42**, 0.95, 8.43 |
| Display Size/Colors | **5.34**, 0.77, 6.08 | **1.7**, 0.25, 2.07 |
| Fonts | **6.11**, 0.88, 7.33 | **1.2**, 0.18, 1.32 |
| Accept Headers | 2.86, 0.41, 3.29 | 2.33, 0.34, 2.99 |
| System Language | 0.41, 0.06, 0.51 | 0.87, 0.13, 1.1 |
| Time Zone | 0.25, 0.04, 0.35 | 0.45, 0.07, 0.73 |
| Mobile Carrier | - | **2.27**, 0.48, 2.4 |
| Overall | **6.93**, 0.99, **11.34** | **6.61**, 0.98, **9.44** |

***Table 7.3:*** *Entropy, normalized entropy, and estimated entropy for various browser features on desktop and mobile devices. For the overall result all features are concatenated.*

## 7.3 Methods for Cross-device Tracking

How is it possible to track Internet users across devices? First, such tracking requires that all devices of interest can be identified (§ 7.3.1). Second, they also have to be correlated (§ 7.3.2). If both requirements are met, device tracking transcends into person tracking.

### 7.3.1 Identifying Devices

HTTP cookies are the traditional mechanism to identify desktop devices. Indeed, many cross-device tracking companies are employing cookies for their tracking purposes as well. Thus, if users are allowing cookies, their desktop devices can be easily identified. In order to track mobile devices the use of advertising identifiers, such as Apple's Identifier for Advertising (IDFA), is common and often combined with cookie tracking. However, as users are increasingly installing tracking protection and adblocking software, which some consider a mainstream technology on mobile by now [207], unconventional identification technologies are becoming more prevalent. While it does not seem that they will generally replace cookies and advertising identifiers any time soon, they are important supplements. Most notably, various cross-device ad networks—for example, BlueCava [49] and AdTruth [102]—are making use of device fingerprinting.

**Entropy Calculations and Estimations.** To get a better understanding of the effectiveness of fingerprinting techniques used in the context of cross-device tracking I calculated the Shannon entropy for various browser features. It is of particular interest to evaluate mobile and desktop devices separately to reveal any differences that might exist between the two device types. In total, the CDT dataset contains 108 mobile device fingerprints and 126 desktop fingerprints.[7] For the mobile fingerprints there were 8 duplicates and for the desktop fingerprints 3. As every mobile device in the set reveals 6.61 bits of identifying information it can be concluded that the 98th device ($2^{6.61} = 97.68$) must be a duplicate. For the desktop that threshold is reached at the 122nd device.

Table 7.3 shows details of the results. $H_n(p) = -\sum_{i=1}^{n} p_i log_b p_i / log_b n$ is the normalized entropy, where $p_i = 1/n$ and $b = 2$, which will result in a value between 0 (all feature values are the same) and 1 (all feature values are different). The estimated entropy is calculated according to Chao and Shen [64], which is intended to give a prediction beyond the sample of fingerprints. Based on this estimation the mobile devices in the CDT set have 9.44 identifying bits while desktop devices have 11.34. Both the actual entropy as well as its estimate suggest that mobile devices are overall less identifiable than desktop devices. However, the results also indicate that some features substantially differ in their impact depending on whether they are used for identifying a mobile or desktop device.

**Entropy Differences between Mobile and Desktop.** Particularly, mobile user agents appear to be far more diverse than user agents on desktops (6.42 vs. 4.46 bits), and, thus, are much more revealing. One reason is that the phone manufacturer and type of phone is part of the mobile's user agent. However, mobiles usually do not contain extensive amounts of system fonts, which are a major contributor to the identifiability of desktops (1.2 vs. 6.11 bits); in addition to displays (1.7 vs. 5.34). There are also idiosyncratic features that are only available on one device type. Most notably, the mobile carrier (2.27 bits) of a phone that can be obtained via reverse IP lookups of cellular IPs is not present on desktops. For the 27 users in the dataset who provided their fingerprint on a cellular connection there were six different mobile carriers. It should be noted, though, that the mobile carrier feature can only be used within the subset of devices that reveal such. There are also features that are generally only meaningful for desktops, for example, plugins.

Overall, some features show more diversity in mobile devices while others have more on the

[7] One user did not submit a mobile fingerprint and one submitted two for different devices. The latter also submitted an additional desktop fingerprint.

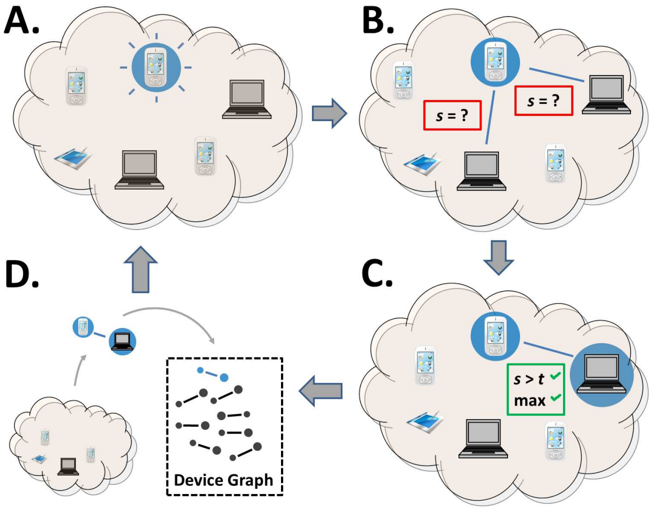


***Figure 7.5:*** *A. The routine begins with identifying a mobile device. B. The similarity, $s$, between the mobile device and each desktop device is calculated. 3. The mobile-desktop pair with the maximum similarity, $max$, that is above the similarity threshold, $t$, is determined. 4. If such pair exists, it is added to the device graph and the next iteration starts with a new mobile device. This routine is performed for three similarities: (1) IPs, (2) Web, (3) Apps/Web. If a device can not be matched in one stage, a match is attempted in the next.*

desktop side. Device fingerprinting seems to work sufficiently on mobile devices to be useful for cross-device fingerprinting, although, likely, more as a supplement to cookie- and advertising identifier-based techniques. However, in this regard the substantial limitations that we imposed on users for participating in the study should be considered (i.e., requiring them to allow first party cookies and JavaScript, run Android 4.0.3 or higher, and use the native browser, Chrome, or the S-Browser) as well as the conservative approach (i.e., the order in which fonts and plugins were detected were not used, which might not be necessary [95]), and only a limited set of fingerprint features was investigated. Thus, entropy in a real-world measurement would likely be higher.

### 7.3.2 Correlating Devices

After identifying the observed devices, cross-device companies try to match those that appear similar, which is the core problem to solve. The goal is to represent all observed devices in a graph known as Device Graph [9], Connected Consumer Graph [2], Intent & Identity Graph [78], or similar proprietary moniker. From a graph-theoretical perspective a device graph is built by creating connected components (each of which represents a user) with a maximum number of vertices (devices) and edges (connections between devices) [77]. The graph must result in a maximum weight matching with the weights being similarity scores between devices. In the case here, since the task is to only connect mobile devices to desktop devices, find the maximum weight matching for a bipartite graph is the goal.

**The CDT Algorithm.** The cross-device tracking algorithm (the CDT algorithm) is outlined in Figure 7.5. In order to determine the similarity between devices I explored various distance measures [62; 195]—specifically, the Jaccard index, cosine similarity, and the Bhattacharyya coefficient. The Jaccard index is defined for the sets $A$ and $B$ as $J(A, B) = (A \cap B)/(A \cup B)$, cosine similarity is defined for the feature vectors $\hat{A}$ and $\hat{B}$ as $\cos(\theta) = \hat{A}\hat{B}/\|\hat{A}\|\|\hat{B}\|$, and the Bhattacharyya coefficient is defined for the distributions $p$ and $q$ as $BC(p, q) = \sum_{x \in X} \sqrt{p(x)q(x)}$. Jaccard index and cosine similarity range between 0 (no similarity) and 1 (maximum similarity). To get a comparable similarity score for the Bhattacharyya coefficient it was normalized for the range between 0 and 1 as well.

The CDT algorithm works in a staged fashion, that is, it first tries to match devices based on the similarity of mobile and desktop IPs, then it attempts a matching using the similarity between mobile and desktop web domains, and finally it tries the similarity between mobile app and desktop web domains. Thus, if at one stage a mobile device does not resemble any desktop device, that is, none of the similarity scores for the mobile device reached the pre-defined threshold, it remains to be matched at a later stage. Using a random subset of data from 64 users as training set I experimented with different settings for matching thresholds and similarity features (e.g., I also tried system language and time zone). I also tried to exclude certain domains or IPs. When a satisfying performance was achieved the best setting—based on the Bhattacharyya coefficient—was evaluated on a test set consisting of data from 44 users (the test set). Table 7.4 shows the results for the test set.

**Test Set Results.** Running the CDT algorithm on the test set results in an accuracy of 84% with 37 true positives, 0 true negatives, 5 false positives, and 2 false negatives. Precision, recall, and F-1 score are 0.88, 0.95, and 0.91, respectively. The F-0.5 score [169], which emphasizes precision over recall, reaches 0.91 as well. The results confirm that IP addresses are of critical importance

| | ***IPs*** | ***Web*** | ***Apps & Web*** |
|---|---|---|---|
| Stage | 1 | 2 | 3 |
| Measure | Bhatta | Bhatta’ | Bhatta* |
| Threshold $t$ | 0.07 | 0.13 | 0.02 |
| Set Size | 44 | 17 | 8 |
| Match% (Sim) | **61%** (0.33) | **53%** (0.16) | **13%** (0.03) |

***Table 7.4:*** *Bhatta’ excludes the Alexa Top 50 domains and Columbia University’s domain. Similarly, Bhatta* excludes the most used 100 apps observed in the training set. Set size is the number of mobile devices to match at the given stage. Match% is the percentage of mobile devices successfully matched. Sim is the mean similarity.*

for matching devices and are in line with Cao et al.’s results [62], who reached an average F-0.5 score of 0.86 in the Drawbridge competition using only IP features. However, different from the participants in the Drawbridge competition the results here suggest that visited web domains are a good indicator for device similarity as well. In fact, there are situations in which they can be more revealing than IPs, for example, when users share a household and have the same IP. Domains lead to the match of another 9 users in the test set. However, as it appears from the description of the Drawbridge competition [158] mobile web history was absent from the competition dataset and, thus, not tested.

**Interpreting Cross-device Tracking Results.** The performance results for probabilistic cross-device tracking must be interpreted against the background of an ad network. A false positive can occur if a mobile device is similar to an unrelated desktop device. Those mismatches might happen for people living in the same household (in case of IP similarity) or individuals having the same interests (in case of web or web/app similarity). However, in these situations a mismatched device might still be a meaningful ad target [90]. Further, a false negative can be caused by setting the similarity thresholds too high. Those cases present a tradeoff between scale and accuracy. Setting the thresholds lower will improve scale and setting them higher accuracy. Therefore, it is not surprising that Dawbridge claims to have a matching accuracy of 97.3% [89]. In fact, changing the similarity threshold in the third stage of the CDT algorithm from $t = 0.02$ to $t = 0.2$ would lead to an accuracy of 100%. However, at the same time the number of device pairs for which a

| | ***IPs*** | ***Web*** | ***Apps & Web*** |
|---|---|---|---|
| Jaccard | 60% (0.03) | 29% (0.06) | 13% (0.01) |
| Cosine | **66%** (0.43) | 10% (0.46) | 5% (0.02) |
| Bhatta | **66%** (0.32) | 29% (0.42) | 6% (0.18) |
| Bhatta” | 64% (0.3) | **58%** (0.18) | **16%** (0.12) |

***Table 7.5:*** *The experiments on the full CDT data with an unstaged complete run for the different feature types (IPs, Web, Apps & Web) confirm the test results. The Bhatta” analysis excludes the 5 most visited IPs (IPs), the top 100 Alexa U.S. domains and Columbia University’s domain (Web), and the 100 most used apps (Apps & Web).*

match would have been attempted would decrease from 44 to 36 since all device pair similarities in the third stage were below the threshold of $t = 0.2$ (they ranged from 0 to 0.16).

**Experiments on the Full CDT Data.** After performing the evaluation on the test set I proceeded to experiment with the full dataset. Table 7.5 shows some of the match percentages and mean similarity scores. Generally, the similarity of IP addresses across devices leads to the most matches with 66%. However, the similarity of web domains is also a strong signal with 58% correct matches. This performance was achieved by applying the Bhattacharyya coefficient and excluding popular domains. The result demonstrates that carefully-crafted features are of utmost importance for the match accuracy [254]. Different from excluding domains and apps the exclusion of the most frequently occurring IP addresses actually caused the performance to deteriorate. The combination of features was also not successful leading us to believe that the staged evaluation is a good choice. App usage also did not correlate to desktop web usage as much as expected. App usage seems to be less diverse than mobile web usage, which provides stronger features.

**Generalizability of Results.** To make the experiments as realistic as possible they always included one user who only had mobile data and no desktop data. In the test set evaluation (Table 7.4) that user’s mobile device was incorrectly matched in the third stage to another user’s desktop. Further, in all experiments, particularly, in each stage of the test set evaluation, desktop data from 18 users for whom we did not have any mobile data were included. However, it is obvious that the dataset is orders smaller than the real data that cross-device tracking companies are ordinarily working with. This difference in size begs the question to which extent the findings are applicable to larger datasets. For the similarity of IPs this question was already reliably answered. The Drawbridge competition results, for instance, by Landry et al. [176], are based on a set of 61,156 mobile devices and confirm the meaningfulness of IP features. For web history features the situation is different as the Drawbridge data did not contain those for mobiles. However, another argument can be made.

Whether web data can be correlated across devices depends on two premises: first, users visiting a subset of domains both on their mobile and desktop devices and, second, domains being sufficiently diverse to allow meaningful distinction between users. To examine the first premise I randomly selected 50 U.S. domains out of the top 5,000 sites that were quantified by Quantcast [217] and observed a mean of 17.1% users visiting a website both on a mobile and desktop device during a 30-day period. At the 95% confidence level using the bootstrap technique this finding translates to lower and upper bounds of 14.4% and 19.5%, respectively, meaning that in 95% of the cases the true estimate of a user visiting a site on both mobile and desktop devices is between 14.4% and 19.5%. Thus, it appears that visiting websites is a broadly occurring phenomenon. As to the second premise, all 102 users in the dataset who visited at least one mobile website had a unique web history. The resulting entropy is 6.67 bits and the estimated entropy according to Chao and Shen [64] comes out at 13.41 bits. Given this information gain there is also a reasonable claim to be made that web data is extent distinctive enough to distinguish thousands of devices, especially, as not even full URLs were considered.

**Practical Considerations and Limitations.** Finally, there are various considerations of identifying and correlating devices in practice. Extreme sparse and large-scale data make user cross-device matching a challenging problem. [232]. In this regard, the CDT algorithm has a runtime of $\mathcal{O}(n(n-1)/2)$. Also, as discussed in more detail below (§ 7.5), despite the broad coverage that some cross-device trackers have, by no means do they have access to all IP, web, and app data of users. In this sense, the task here was easier. However, I did not have a full IP history either as the IP address was not collected with every data point that was submitted. Also, through the confined space and users being mostly students from one University the data is probably more homogenous than real data would be. Another consideration concerns the time periods that data covers. In this regard, it remains unknown for which duration cross-device companies can track users.

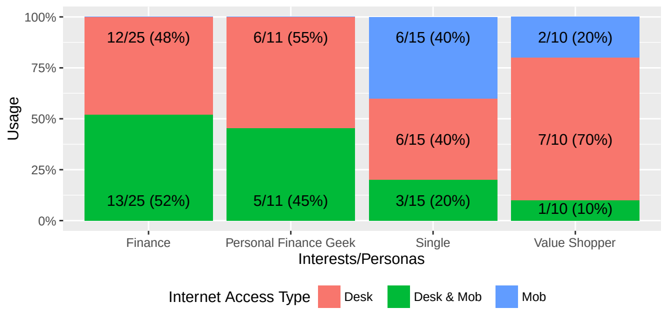
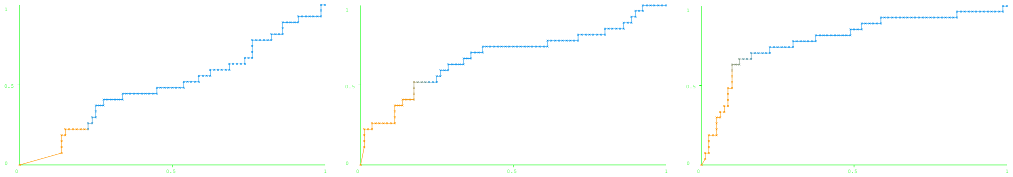


***Figure 7.6:*** *About half of all users with an Interest in finance access respective domains only on their desktop while the other half is using both mobile (app or web) and desktop devices. However, no user is accessing those domains exclusively from a mobile device, which also holds for personal finance geeks. For singles and value shoppers the picture looks different, though. Most are using either a mobile or desktop device, however, not both.*

## 7.4 Learning from Cross-device Data

Cross-device tracking can be more privacy-invasive than traditional tracking of individual devices. After all, cross-device companies are potentially able to obtain a fuller picture of a person and learn much more than they could by only observing unconnected devices. As discussed in the previous section, the average user accesses about a sixth of all websites on both mobile and desktop devices. However, it is also true that people are using different devices for different purposes. This phenomenon is illustrated in Figure 7.6 for users in the dataset that expressed an interest in finance, value shopping, and dating. In terms of methodology, I used Alexa category rankings [29] and Google Play store categories [137] to identify 25 domains for each of these interests that have both a website and an app. Then, I checked for the users in the dataset how often, if at all, they access these domains from their different devices. The result suggests that having data available from both mobile and desktop devices could indeed increase the predictive power in machine learning experiments (assuming that the respective domains are used as features).

**Increase in Predictive Accuracy.** Indeed, the results in Figure 7.7, which are based on 10-fold cross validation, indicate that predicting an interest in finance for users in the dataset is more accurate if both desktop and mobile data is available. The results are based on using the 25 financial domains as starting point for feature creation on the Weka machine learning toolkit [142].

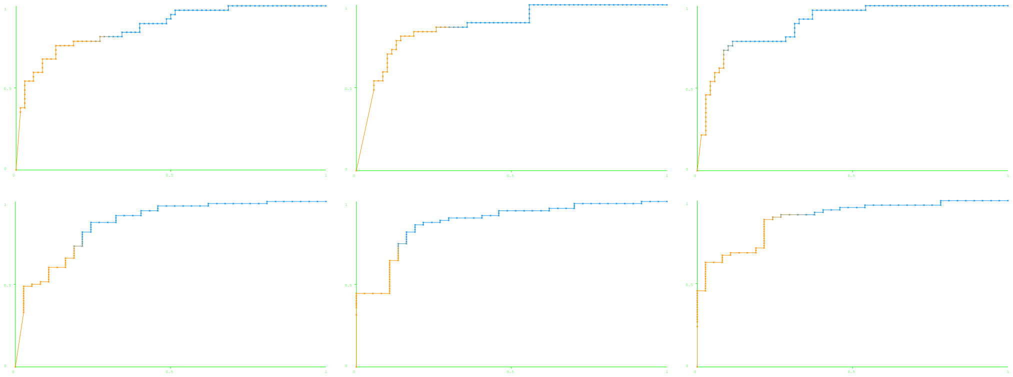


| | ***Features*** | ***Acc*** | ***Prec*** | ***Rec*** | ***F-1*** | ***ROC*** |
|---|---|---|---|---|---|---|
| Mob | 90 | 64.5% | 0.26 | 0.22 | **0.24** | **0.5** |
| Desk | 106 | 74.8% | 0.5 | 0.52 | **0.51** | **0.68** |
| Mob&Desk | 107 | 83.2% | 0.68 | 0.63 | **0.65** | **0.79** |

***Figure 7.7:*** *Logistic regression for predicting an interest in finance from app and web domains. As before, mobile data (Mob) includes both mobile web and apps and desktop data (Desk) covers desktop web domains. The F-1 score for predictions based on both types of data is substantially higher than predictions from each source individually. The three ROC curves visualize this finding (left: Mob, middle: Desk, right: Mob&Desk). True positives are displayed on the y-axis and false positives on the x-axis. The results for the positive cases, that is, predicting that users have an interest in finance, are shown in orange while the negative predictions for not having an interest in finance are displayed in blue. As can be observed from the ROC curves, especially, the former benefits from having both mobile and desktop data available.*

I tried various feature engineering techniques and all standard algorithms, among which were logistic regression, stochastic gradient descent, support vector machines, various versions of naive Bayes, and various tree-based algorithms, such as random forest. Logistic regression turned out to be the classifier with the best performance. Due to the class imbalance of only 23% users expressing an interest in finance I ran logistic regression as a cost-sensitive classifier increasing the cost for a false positive of 1.5 times over the cost for a false negative. Certainly, the results can be improved. However, what I want to show here is that it is an advantage to have data from various sources. The advantage is quantitative as there are simply more data points available. However, it is also qualitative because it allows the creation of more characteristic features as evidenced by the nearly equal number of desk and mob&desk features (106 vs. 107).

**Compensating for the Lack of Cross-device Data.** While the results indicate that predictive performance increases with the availability of both desktop and mobile web data, it appears that



| | ***Features*** | ***Acc*** | ***Prec*** | ***Rec*** | ***F-1*** | ***ROC*** |
|---|---|---|---|---|---|---|
| Desk | 3,395 | 75.7% | 0.79 | 0.76 | **0.76** | **0.87** |
| Mob&Desk | 3,006 | 76.6% | 0.81 | 0.77 | **0.77** | **0.87** |
| Desk | 5,929 | 84.1% | 0.84 | 0.84 | **0.84** | **0.89** |

***Figure 7.8:*** *Accuracy, precision, recall, and F-1 score are based on the average for the men and women classes as weighted by the number of instances in those classes. The ROC curves are visualizing the ROC areas for women (top) and men (bottom). From left to right the curves are for desktop web domains with 3,395 features, mobile and desktop domains with 3,006 features, and desktop domains with 5,929 features.*

the use of higher-dimensional feature vectors on desktop data can sometimes compensate and even outperform these results. Using logistic regression it is possible to predict the gender of users. Since there are about one third women and two thirds men the algorithm is adjusted for the gender skew by penalizing the misclassification of a woman as a man 1.5 times of the misclassification of a man as a woman. The results in Figure 7.8, which are based on 10-fold cross validation, show that doubling the number of features in the desktop web domains relative to the number of domains used in the mobile and desktop domain combination data increases the F-1 score from 0.77 to 0.84. However, holding the number of features constant at about the same level (3,395 vs. 3,006) demonstrates the higher value in the combined desktop and mobile features. From the perspective of an ad network or analytics provider it is certainly of interest to work with low-dimensional data to avoid performance bottlenecks.

**Learning Sensitive Information.** From the results it appears that sensitive traits of a person, such as ethnicity or religion, can be much better inferred with web domain data from two types

| | ***Accuracy*** | ***Precision*** | ***Recall*** | ***F-1*** | ***ROC*** |
|---|---|---|---|---|---|
| Chinese | 87.5% | 1 | 0.88 | **0.93** | **0.96** |
| English | 91.3% | 0.91 | 0.91 | **0.91** | **0.78** |
| Indian | 44.4% | 0.4 | 0.44 | **0.42** | **0.65** |
| Weight Avg | 86% | 0.87 | 0.86 | **0.86** | **0.79** |

***Table 7.6:*** *Logistic regression results for predicting a user's native language from visited domains based on data from both mobile devices (web and app) and desktop devices (web). I used 10-fold cross validation.*

of devices than from one. Table 7.6 shows results for predicting the native languages spoken by users in the dataset, which can be used to infer ethnicity. Using 25 popular domains for each of the U.S., China, and India that have an app and a website to create meaningful features (116), the results for using mobile and desktop features were better than for mobile alone (weighted average F-1 0.78) and desktop alone (weighted average F-1 0.83). However, the results are based on a small sample of 86 native speakers (8 Chinese language origin, 9 Indian language origin, and 69 English). Interestingly, the prediction of Indian users did not perform as well as the identification of Chinese users. I believe that the reason is that Chinese users have a common core of domains they use (e.g., Baidu and Tencent), which is not the case for Indian users making it harder to identify the latter.

Accessing religious web domains and apps can be an obvious predictor for adherence to a particular faith. However, such predictions are also possible based on subtler user behaviors. Most notably, as the data collection for the study covered the last two days of the Jewish Passover holiday a few users in the study did not use both of their signed up devices as the Jewish faith prescribes abstinence from using electronics. Among all users in the study who were signed up at the time the pattern of holiday observation became very clear. This signal is more clear given the insight into multiple devices because during the two days of Passover some users did not use one of their devices, however, used the other. Only those users observant of Passover did not use both devices. In this sense, cross-device tracking can be more privacy-invasive than the tracking of unconnected individual devices and can also lead to a privacy violation, which is also true for cross-device tracking companies' observation of users ethnicities.

## 7.5 A Small Glimpse into the Scope of Cross-device Tracking on the Internet

The degree to which cross-device tracking is permeating the Internet is unknown. While I leave a comprehensive inquiry for another day, some initial inroads will be provided. I crawled the websites and apps in the dataset for their inclusion of third party trackers (§ 7.5.1) and analyzed potential cross-device usage, particularly, accounting for industry collaborations and consolidation (§ 7.5.2).

### 7.5.1 Obtaining Third Party Tracking Data

In order to examine the extent to which cross-device tracking is happening I examined the trackers on the domains and apps that the users in the study visited. Automating a Firefox browser with Selenium [231] as well as a Lightbeam [197] and user agent switcher [198] browser extension I recorded the trackers on each domain. Third party connections found in a subdomain were added to the domain, however, not vice versa. Thus, for example, the domain linkedin.com contains all trackers on blog.linkedin.com but not the other way around. Both desktop and mobile sessions were started with a fresh browser that did not contain any user data. For the desktop crawl a Windows 10 user agent simulated and for the mobile crawl an Android Nexus 5 user agent.

**Limitations.** One limitation of the approach is that some websites were not accessible (e.g., sites that required a user login). In some cases the crawl was also redirected or the requested page was not found. However, these limitations only affected few URLs. Also, it should be noted that the crawl of the sites was conducted about a month after finishing collecting data from the study participants. Thus, in the meantime, some websites might have different trackers than at the time they were actually visited. Ideally, it would have been possible to capture the trackers live from the devices of the users. However, such recording is an expensive proposition in terms of mobile device performance, and, especially, the constraints of the Android environment (e.g., the sandboxing of browser apps) make it difficult to capture trackers directly on the device.

**Data Collection Procedure.** For detecting trackers inside of apps I selected a total of 153 third party sdks listed on AppBrain [36] encompassing sdks of ad networks (e.g., LiveRail), social networks (e.g., Twitter), analytics services (e.g., comScore), crash reporters (e.g., Crashlytics), and

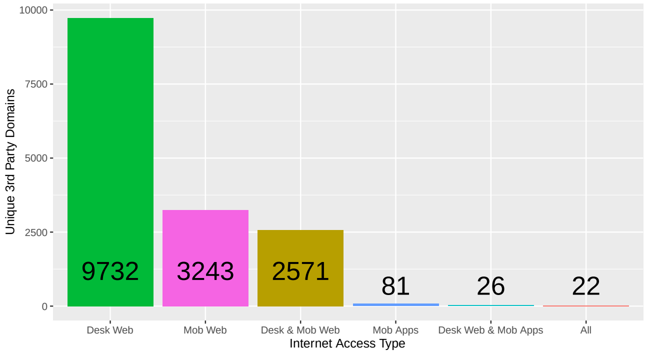


***Figure 7.9:*** *Total unique third party trackers in the dataset. 2,571 trackers occurred on both desktop and mobile websites. Out of the 153 sdks from AppBrain 81 acre contained in the dataset. 26 intersected with the desktop websites, and 22 were present on desktop and mobile websites as well as in apps. For the most part, system apps, banking apps, and apps by the phone manufacturer did not contain trackers.*

payment processors (e.g., Amazon In-App Purchasing). Then, I crawled the AppBrain statistics to determine which of the libraries are included in the apps of the users in the study. The approach for detecting trackers should be understood as a lower bound for various reasons. First, trackers not identified in Lightbeam and sdks not included in the set of 153 will remain undetected. Second, apps are limited to tracking via sdks and does not account for WebViews and app-internal browsers that could also contain tracking cookies [77]. Third, the reach of companies' tracking activity is not always clear due to unknown industry collaborations or backend data exchanges. Finally, I rely on companies' representation that they track users across devices and do not make any own independent determination beyond detecting the presence of their trackers.

### 7.5.2 The Converging Cross-Device Ecosystem

As shown in Figure 7.9, the mobile websites in the dataset (3,876 per Table 7.1) contained 3,243 unique third party trackers. 2,571 of those were also present on desktop websites. Thus, there appears to be a large number of cross-device trackers across mobile and desktop websites. The number of trackers inside the apps in the dataset is substantially smaller, clearly, as a consequence

| | ***Desk Web*** | ***Mob Web*** | ***Mob Apps*** |
|---|---|---|---|
| BlueCava | **0.2%** | **0.5%** | - |
| comScore | 11.3% | 15.1% | 1.7% |
| Flurry Analytics | - | 0.3% | 4.3% |
| comScore & Flurry | **11.3%** | **15.1%** | **6.1%** |
| Google Analytics | 58% | 43.6% | 5.1% |
| Facebook | 21.4% | 17.1% | 20.3% |
| LiveRail | 1% | 1.6% | 0.3% |
| Facebook & LiveRail | **21.6%** | **17.7%** | **20.3%** |
| PayPal | 1.1% | 0.6% | 0.9% |
| Tapad | 1.1% | 1.9% | - |
| Apsalar | - | - | 0.3% |
| Tapad & Apsalar | **1.1%** | **1.9%** | **0.3%** |
| Twitter | 11.5% | 6% | 0.7% |

***Table 7.7:*** *Companies' percentage for covering websites and apps for the average user in the dataset (out of the 107 users for which both mobile and desktop domains were collected.)*

of the limited set started out with.[8] While many unique trackers across device boundaries were detected this finding does not allow a claim on how broadly cross-device companies disseminated their trackers.

**Tracking of the Average User in the Dataset.** For an illustrative cross-section of cross-device tracking companies—some smaller, some bigger, some deterministic, some probabilistic—I calculated the percentage that each user in the dataset is tracked across his or her different devices. Table 7.7 shows the results. We see the phenomenon of a few general ad companies having a broad scope of trackers while specialized cross-device tracking companies have a smaller market share. The former is clearly represented by Google and Facebook on the deterministic spectrum of cross-device tracking and comScore on the probabilistic end. An example for the latter is Tapad. Also, the detection of BlueCava fingerprinting scripts on both mobile and desktop websites confirms Acar et. al's claim [21] that one of the use cases for fingerprinting consists of reaching customers

[8] The app tracker count includes affiliated company's sdks. Thus, for example, the Facebook sdk inside the Instagram app is counted as a tracker.

across devices.

**Industry Collaborations and Consolidation.** Evaluating the results in light of known industry collaborations demonstrates something else: partnerships are often complementary. For example, while comScore's cookies are a mainstay on both the desktop and mobile web, their reach into mobile apps is much more limited. As the opposite is true for Flurry Analytics it makes a lot of sense that both are collaborating in their cross-device efforts as part of the Flurry Pulse platform [263]. A similar observation, albeit on a smaller scale, can be made for the cooperation between Tapad and Appsalar [56]. Sometimes, such strategic partnerships also come into existence through one company acquiring another, for example, in the case of Facebook's acquisition of LiveRail.

This acquisition also shows that the line between probabilistic and deterministic cross-device tracking is not as clear-cut as the dichotomic usage of the terms suggest. LiveRail receives some user data from Facebook to track users probabilistically [104]. However, Facebook's deterministic tracking might profit from LiveRail as well. After all, some Internet users do not have a Facebook account, in which case they still can be tracked probabilistically. In general, whether through collaboration or acquisition the ad industry is experiencing a consolidation and concentration that broadens companies' access to cross-device data. This development makes privacy protection more challenging.

## 7.6 Does Self-Regulation Work?

In the U.S. there are no statutes or regulations for cross-device tracking, but rather the field is subject to self-regulation, most notably by the Digital Advertising Alliance (DAA) and the Network Advertising Initiative (NAI). A hallmark of the U.S. privacy regime is the notion of data transparency vis-à-vis web consumers. In fact, according to a recent guidance, the DAA requires cross-device companies to disclose "the fact that data collected from a particular browser or device may be used with another computer or device that is linked to the browser or device on which such data was collected." [88] I examined compliance with this transparency requirement for 40 randomly selected ad networks with DAA membership that advertised their cross-device capabilities.

Specifically, I manually checked if they disclose their cross-device tracking activity in their privacy policies or opt-out statements. While 23 did so, 17 omitted to mention cross-device tracking at all. After contacting these 17 companies, I received feedback from four. One ad network simply claimed that they are “not violating anything.” Another amended its policy as required per the DAA without explicitly getting back to us. A third company explained to us that their cross-device functionality is not yet rolled out to clients. Finally, a fourth ad network notified us that they will update their policy once the NAI’s cross-device code of conduct would become available.

Based on the interpretation of the DAA guidance, I find indications that there is some lack of transparency when it comes to the disclosure of cross-device tracking. It does not seem to be the case that the DAA guidance is rigorously enforced. To be clear, the vast majority of consumers will likely not take the time to understand the tracking practices on a per-company level either way. Using tracking and ad blockers is a much more efficient approach from a consumer perspective.[9] However, for audit and enforcement purposes as well as to gain trust in the marketplace I think that it is certainly a worthwhile endeavor for companies to properly disclose their practices.

## 7.7 Conclusion

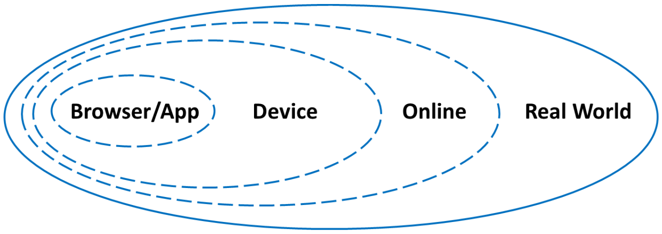


***Figure 7.10:*** *This thesis discusses how ad networks are crossing device boundaries within the online space. However, there are also early attempts to cross the online-offline boundary.*

This chapter can be considered as groundwork for developing privacy protections for cross-device tracking. Among others, I have demonstrated how to identify cross-device tracking. This identification can be implemented in an ML classifier. Overall, cross-device tracking challenges

[9] Thus, if one wants to think of cross-device tracking in terms of a threat model, the most effective defense would be to block tracking. In this sense, the defenses against cross-device tracking are the same as the defenses against the tracking of individual devices.

current notions of Internet privacy. The various angles discussed here are all deserving of a more detailed and comprehensive examination than could be done here. Especially, as shown—whether it is the correlation of devices or the learning from data—the machine learning techniques applied for purposes of cross-device tracking have notable privacy implications. Thus, I understand the work here as a tour d'horizon of the cross-device tracking landscape. I highlighted some aspects that I believe to be particularly important; others are left open. For example, there are various ad preference managers that allow consumers insights into how they are tracked by individual companies (e.g. the BlueKai Registry [205]). It would be interesting to see whether these preference managers can be leveraged to understand data flows between companies.

Proprietary research in cross-device tracking is way ahead of academia. While a few big points are known (for example, that IP addresses are the most crucial feature for correlating devices), many details on how ad networks operate in this space remain opaque. To shed more light on the subject I will publicize the dataset (in anonymized fashion) together with the developed software for further exploration. As cross-device tracking matures and becomes an integral part of tracking on the Internet I advocate for a comprehensive view of the phenomenon that also includes the legal environment. Establishing an enforceable self-regulatory framework for companies to be transparent about their practices will help to protect consumer privacy and allow ad networks to earn their advertising dollars responsibly. Thus, I believe the FTC's current approach is the right one.

# Chapter 8

# Conclusions

It is the thesis of this dissertation that Internet privacy can be improved based on the use of ML technologies (in many cases in tandem with other technologies, such as static code analysis, as illustrated in Chapter 6). First, in a case study I have demonstrated how ML classifiers can be used to identify ethnicity- and gender-specific location patterns (Chapter 3). I also showed how ML can be leveraged for purposes of quantifying privacy-invasiveness (Chapter 4), particularly, as part of the mosaic theory and in combination with privacy metrics, such as *k*–anonymity [241]. Further, in order to improve privacy transparency I described a system to automatically analyze privacy policies using ML classifiers (Chapter 5). The policy analysis results can be compared to actually occurring practices on websites, mobile apps or other software (Chapter 6). This type of comparison enables regulators to hold software publishers accountable for their privacy practices. Finally, I explored the foundations for developing PETs for a rarely investigated but increasingly common practice: cross-device tracking (Chapter 7).

Internet privacy is a multi-dimensional concept. It transcends the boundaries of various academic disciplines and is characterized by sociological, legal, and engineering aspects—to name a few. In addition, there are also many different technologies affecting it: web tracking, cryptographic protocols, and social networks are some examples. In this dissertation I am interpreting privacy as a legal right. It is my thesis that privacy can be advanced based on ML technologies. It is the central theme of this work to explore the uses of ML to advance privacy. Thus, while it is true that privacy is threatened by machine learning technologies, those same technologies can also be used to improve privacy.

Ultimately, the inter-connectedness of devices and the penetration of many wakes of life by data collection and sharing mechanisms raises privacy challenges of a new quality. In this regard, cross-device tracking can be seen as an early harbinger of the Internet of Things (IoT). Ensuring transparency and practicable control mechanisms for information that is traversing device boundaries and permeates in and out of the offline world, as depicted in Figure 7.10, is in its infancy. The massive volume of granular data allows those with access to it to perform ML analyses that would not have been possible before [117]. However, in this environment the FTC will continue to place emphasis on the notice and choice principle [117]. Lacking interfaces of many IoT devices would require companies to give notice in different way [117]. Given the understanding of the IoT as "a world-wide network of interconnected objects uniquely addressable, based on standard communication protocols [152]," privacy notices and choice should evolve into comprehensive personal privacy assistants: they will perform ML privacy policy analysis, check whether devices adhere to what is claimed in the policies, or establish or deny connection between devices.

What is needed is an intelligent and scalable mechanism that empowers users to efficiently and accurately obtain data processing information and control. Such mechanism could warn users if it detects, for example, as discussed in Chapter 3, that sensitive information could be inferred from certain collected data. A hallmark of this new paradigm is the application of ML to relieve the user from being constantly involved in privacy decisions [19]. A device could learn a user's preferences on one device (e.g., data should not be shared with third party advertisers) and use those as default preferences on all devices [117]. Another example could be a central appliance hub that stores data locally and learns preferences based on prior behavior and predict future privacy preferences as new appliances are added to the hub [117]. Along these lines I envision an intelligent personal privacy assistant that is deployable in the current and future IoT environment.

The privacy assistant could be comprised of a central control unit (e.g., an app on a phone) connected to all other devices (e.g., cars, smartwatches, Wi-Fi routers, household appliances, cars) that resolves privacy settings and new privacy queries based on user input and learned privacy preferences. For difficult questions the user will be alerted and the control unit continues to learn based on user input. It obtains the data, analyzes them, and acts on them according to the user preferences. Such privacy assistant must understand natural language privacy policies as well as interface with APIs of other domains (e.g., cars, appliances). Thus, developing such assistant is partly a question of operating system research and standard setting. Opening APIs and their standardization is necessary, which is a policy problem and very likely not completely solvable. However, various standards are currently in development [4; 14] and open for privacy considerations. The capability of devices interacting with each other—in many cases without human input—should be developed in tandem with intelligent privacy assistants that can act on the same premises.

# Bibliography

[1] Claridge v. RockYou, Inc., 785 F. Supp. 2d 855 (N.D. Cal. 2011).

[2] Drawbridge, Inc. `http://www.drawbrid.ge/`. Last accessed: March 6, 2017.

[3] Google Display Network. `https://www.google.com/ads/displaynetwork/`. Last accessed: March 6, 2017.

[4] IEEE Standards Association. `http://standards.ieee.org/innovate/iot/`. Last accessed: March 6, 2017.

[5] P3P compact policy cross-reference. `http://compactprivacypolicy.org/compact_token_reference.htm`. Last accessed: March 6, 2017.

[6] Personalized privacy assistant project. `http://www.privacyassistant.org/`. Last accessed: March 6, 2017.

[7] Privacy Icons. `http://www.azarask.in/blog/post/privacy-icons/`. Last accessed: March 6, 2017.

[8] privacychoice. `http://www.privacychoice.org`. Last accessed: March 6, 2017.

[9] Tapad, Inc. `http://www.tapad.com/`. Last accessed: March 6, 2017.

[10] Terms of Service; Didn't Read (ToS;DR). `http://tosdr.org/index.html`. Last accessed: March 6, 2017.

[11] TOSBack. `http://tosback.org/`. Last accessed: March 6, 2017.

[12] TOSBack2. `https://github.com/pde/tosback2`. Last accessed: March 6, 2017.

[13] Usable Privacy Policy Project. `http://www.usableprivacy.org/home`. Last accessed: March 6, 2017.

[14] W3C web of things interest group. `https://www.w3.org/WoT/IG/`. Last accessed: March 6, 2017.

[15] Platform for Internet Content Selection (PICS). `http://www.w3.org/PICS/`, 1997. Last accessed: March 6, 2017.

[16] KnowPrivacy. `http://knowprivacy.org/`, June 2009. Last accessed: March 6, 2017.

[17] Data economy – a CNBC special report. `http://www.cnbc.com/data-economy/`, Nov 2013. Last accessed: March 6, 2017.

[18] eXtensible Access Control Markup Language (XACML) version 3.0. `http://docs.oasis-open.org/xacml/3.0/xacml-3.0-core-spec-os-en.html`, January 2013. Last accessed: March 6, 2017.

[19] *Twelfth Symposium on Usable Privacy and Security, SOUPS 2016, Denver, CO, USA, June 22-24, 2016*. USENIX Association, 2016.

[20] Gunes Acar, Christian Eubank, Steven Englehardt, Marc Juarez, Arvind Narayanan, and Claudia Diaz. The web never forgets: Persistent tracking mechanisms in the wild. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, CCS '14, pages 674–689, New York, NY, USA, 2014. ACM.

[21] Gunes Acar, Marc Juarez, Nick Nikiforakis, Claudia Diaz, Seda Gürses, Frank Piessens, and Bart Preneel. Fpdetective: Dusting the web for fingerprinters. In *Proceedings of the 2013 ACM SIGSAC Conference on Computer &#38; Communications Security*, CCS '13, pages 1129–1140, New York, NY, USA, 2013. ACM.

[22] Alessandro Acquisti. The economics of personal data and the economics of privacy, 2010.

[23] Eytan Adar, Jaime Teevan, and Susan T. Dumais. Large scale analysis of web revisitation patterns. In *CHI '08: Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, pages 1197–1206, New York, NY, USA, 2008. ACM.

[24] Alina Adreevskaia and Sabine Bergler. Mining WordNet for fuzzy sentiment: Sentiment tag extraction from WordNet glosses. In *11th conference of the European chapter of the Association for Computational Linguistics*, EACL ’06, pages 209–216, Stroudsburg, PA, USA, 2006. ACL.

[25] Vitor Afonso, Antonio Bianchi, Yanick Fratantonio, Adam Doupe, Mario Polino, Paulo de Geus, Christopher Kruegel, and Giovanni Vigna. Going native: Using a large-scale analysis of android apps to create a practical native-code sandboxing policy. In *Proceedings of the ISOC Network and Distributed System Security Symposium (NDSS)*, San Diego, CA, February 2016.

[26] Rakesh Agrawal, Jerry Kiernan, Ramakrishnan Srikant, and Yirong Xu. An XPath-based preference language for P3P. In *Proceedings of the 12th international conference on World Wide Web*, WWW ’03, pages 629–639, New York, NY, USA, 2003. ACM.

[27] Esma Aïmeur, S’ebastien Gambs, and Ai Ho. UPP: User privacy policy for social networking sites. In *Fourth International Conference on Internet and Web applications and services*, ICIW ’09, pages 267–272, Washington, DC, USA, 2009. IEEE Computer Society.

[28] Alexa. The top 500 sites on the web. `http://www.alexa.com/topsites/countries/US`. Last accessed: March 6, 2017.

[29] Alexa. The top 500 sites on the web. `http://www.alexa.com/topsites/category`. Last accessed: March 6, 2017.

[30] Yaniv Altshuler, Nadav Aharony, Michael Fire, Yuval Elovici, and Alex Pentland. Incremental learning with accuracy prediction of social and individual properties from mobile-phone data. In *SocialCom/PASSAT*, pages 969–974. IEEE, 2012.

[31] Amazon. Amazon ec2 instance types. `https://aws.amazon.com/ec2/instance-types/`. Last accessed: March 6, 2017.

[32] Waleed Ammar, Shomir Wilson, Norman Sadeh, and Noah Smith. Automatic categorization of privacy policies: A pilot study. Technical Report CMU-ISR-12-114, CMU-LTI-12-019, Carnegie Mellon University, December 2012.

[33] Thakur Raj Anand and Oleksii Renov. Machine learning approach to identify users across their digital devices. In *IEEE International Conference on Data Mining Workshop, ICDMW 2015, Atlantic City, NJ, USA, November 14-17, 2015*, pages 1676–1680, 2015.

[34] Androguard. `http://doc.androguard.re/html/index.html`. Last accessed: March 6, 2017.

[35] AOL, Inc. Cross-platform optimization. `https://www.advertising.com/advertiser/our-platform`. Last accessed: March 6, 2017.

[36] AppBrain. Android library statistics. `http://www.appbrain.com/stats/libraries/`. Last accessed: March 6, 2017.

[37] Ron Artstein and Massimo Poesio. Inter-coder agreement for computational linguistics. *Comput. Linguist.*, 34(4):555–596, December 2008.

[38] Steven Arzt, Siegfried Rasthofer, Christian Fritz, Eric Bodden, Alexandre Bartel, Jacques Klein, Yves Le Traon, Damien Octeau, and Patrick McDaniel. Flowdroid: Precise context, flow, field, object-sensitive and lifecycle-aware taint analysis for android apps. In *Proceedings of the 35th ACM SIGPLAN Conference on Programming Language Design and Implementation*, PLDI ’14, pages 259–269, New York, NY, USA, 2014. ACM.

[39] Paul Ashley, Satoshi Hada, Günter Karjoth, Calvin Powers, and Matthias Schunter. Enterprise Privacy Authorization Language (EPAL 1.2). Technical report, IBM, November 2003.

[40] Ask Solem & Contributors. Celery - distributed task queue. `http://docs.celeryproject.org/en/latest/`. Last accessed: March 6, 2017.

[41] Kathy Wain Yee Au, Yi Fan Zhou, Zhen Huang, and David Lie. Pscout: Analyzing the android permission specification. In *Proceedings of the 2012 ACM Conference on Computer and Communications Security*, CCS ’12, pages 217–228, New York, NY, USA, 2012. ACM.

[42] Emily Badger. This is how women feel about walking alone at night in their own neighborhoods. http://www.washingtonpost.com/blogs/wonkblog/wp/2014/05/28/this-is-how-women-feel-about-walking-alone-at-night-in-their-own-neighborhoods/, May 2014.

[43] Katayoun Baghai. Privacy as a human right: A sociological theory. *Sociology*, 46(5):951–965, October 2012.

[44] Rebecca Balebako, Abigail Marsh, Jialiu Lin, Jason Hong, and Lorrie F. Cranor. The privacy and security behaviors of smartphone app developers. In *Workshop on Usable Security (USEC)*, 2014.

[45] Steven M. Bellovin. *Thinking Security: Stopping Next Year's Hackers*. Addison-Wesley, 2015.

[46] Steven M. Bellovin, Renée M. Hutchins, Tony Jebara, and Sebastian Zimmeck. When enough is enough: Location tracking, mosaic theory, and machine learning. *N.Y.U. J.L. Liberty*, 8(2):556–629, 2014.

[47] Ravi Bhoraskar, Seungyeop Han, Jinseong Jeon, Tanzirul Azim, Shuo Chen, Jaeyeon Jung, Suman Nath, Rui Wang, and David Wetherall. Brahmastra: Driving apps to test the security of third-party components. In *Proceedings of the 23rd USENIX Conference on Security Symposium*, SEC'14, pages 1021–1036, Berkeley, CA, USA, 2014. USENIX Association.

[48] C. Biagioli, E. Francesconi, A. Passerini, S. Montemagni, and C. Soria. Automatic semantics extraction in law documents. In *Proceedings of the 10th International Conference on Artificial Intelligence and Law*, ICAIL '05, pages 133–140, New York, NY, USA, 2005. ACM.

[49] BlueCava, Inc. `http://bluecava.com/`. Last accessed: March 6, 2017.

[50] Theodore Book and Dan S. Wallach. A case of collusion: A study of the interface between ad libraries and their apps. In *Proceedings of the Third ACM Workshop on Security and Privacy in Smartphones & Mobile Devices*, SPSM '13, pages 79–86, New York, NY, USA, 2013. ACM.

[51] Travis D. Breaux and Annie I. Antón. Mining rule semantics to understand legislative compliance. In *Proceedings of the 2005 ACM Workshop on Privacy in the Electronic Society*, WPES '05, pages 51–54, New York, NY, USA, 2005. ACM.

[52] Travis D. Breaux and Annie I. Antón. Analyzing regulatory rules for privacy and security requirements. *IEEE Trans. Software Eng.*, 34(1):5–20, January 2008.

[53] Kiel Robert Brennan-Marquez. Plausible cause. *Vanderbilt Law Review*, 70, 2017. forthcoming.

[54] Carolyn Brodie, Clare-Marie Karat, John Karat, and Jinjuan Feng. Usable security and privacy: a case study of developing privacy management tools. In *Proceedings of the 2005 Symposium On Usable Privacy and Security*, SOUPS '05, pages 35–43, New York, NY, USA, 2005. ACM.

[55] Carolyn A. Brodie, Clare-Marie Karat, and John Karat. An empirical study of natural language parsing of privacy policy rules using the SPARCLE policy workbench. In *Proceedings of the second Symposium On Usable Privacy and Security*, SOUPS '06, pages 8–19, New York, NY, USA, 2006. ACM.

[56] Business Wire. Apsalar and tapad partner to help brands understand and action cross-device customer behavior. `http://www.businesswire.com/news/home/20150826005126/en/Apsalar-Tapad-Partner-Brands-Understand-Action-Cross-Device`. Last accessed: March 6, 2017.

[57] Simon Byers, Lorrie Faith Cranor, Dave Kormann, and Patrick McDaniel. Searching for privacy: design and implementation of a P3P-enabled search engine. In *Proceedings of the 4th international conference on Privacy Enhancing Technologies*, PET '04, pages 314–328, Berlin, Heidelberg, Germany, 2005. Springer.

[58] Xiang Cai, Rishab Nithyanand, and Rob Johnson. Cs-buflo: A congestion sensitive website fingerprinting defense. In *Proceedings of the 13th Workshop on Privacy in the Electronic Society*, WPES '14, pages 121–130, New York, NY, USA, 2014. ACM.

[59] Xiang Cai, Rishab Nithyanand, Tao Wang, Rob Johnson, and Ian Goldberg. A systematic approach to developing and evaluating website fingerprinting defenses. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, CCS '14, pages 227–238, New York, NY, USA, 2014. ACM.

[60] California Department of Justice. Attorney general kamala d. harris secures global agreement to strengthen privacy protections for users of mobile applications. `http://www.oag.ca.gov/news/press-releases/attorney-general-kamala-d-harris-secures-global-agreement-strengthen-privacy`, February 2012. Last accessed: March 6, 2017.

[61] California Department of Justice. Making your privacy practices public. `https://oag.ca.gov/sites/all/files/agweb/pdfs/cybersecurity/making_your_privacy_practices_public.pdf`, May 2014. Last accessed: March 6, 2017.

[62] Xuezhi Cao, Weiyue Huang, and Yong Yu. Recovering cross-device connections via mining IP footprints with ensemble learning. In *IEEE International Conference on Data Mining Workshop, ICDMW 2015, Atlantic City, NJ, USA, November 14-17, 2015*, pages 1681–1686, 2015.

[63] Abdelberi Chaabane, Mohamed Ali Kaafar, and Roksana Boreli. Big friend is watching you: Analyzing online social networks tracking capabilities. In *Proceedings of the 2012 ACM Workshop on Workshop on Online Social Networks*, WOSN '12, pages 7–12, New York, NY, USA, 2012. ACM.

[64] Anne Chao and Tsung-Jen Shen. Nonparametric estimation of shannon's index of diversity when there are unseen species in sample. *Environmental and Ecological Statistics*, 10(4):429–443, 2003.

[65] Parvathi Chundi and Pranav M. Subramaniam. An Approach to Analyze Web Privacy Policy Documents. In *KDD Workshop on Data Mining for Social Good*, 2014. 00000.

[66] Corey A. Ciocchetti. The future of privacy policies: A privacy nutrition label filled with fair information practices. *J. Marshall J. Computer & Info. L.*, 26:1–46, 2008.

[67] comScore, Inc. Media metrix multi-platform. `http://www.comscore.com/Products/Audience-Analytics/Media-Metrix-Multi-Platform`. Last accessed: March 6, 2017.

[68] Elisa Costante, Jerry den Hartog, and Milan Petkovic. What websites know about you: Privacy policy analysis using information extraction. In Roberto Di Pietro, Javier Herranz, Ernesto Damiani, and Radu State, editors, *Data Privacy Management and Autonomous Spontaneous Security*, volume 7731 of *Lecture Notes in Computer Science*, pages 146–159, Berlin, Heidelberg, Germany, 2013. Springer.

[69] Elisa Costante, Yuanhao Sun, Milan Petković, and Jerry den Hartog. A machine learning solution to assess privacy policy completeness: (short paper). In *Proceedings of the 2012 ACM Workshop on Privacy in the Electronic Society*, WPES ’12, pages 91–96, New York, NY, USA, 2012. ACM.

[70] Lorrie Faith Cranor. Necessary but not sufficient: Standardized mechanisms for privacy notice and choice. *J. on Telecomm. and High Tech. L.*, 10(2):273–307, 2012.

[71] Lorrie Faith Cranor, Brooks Dobbs, Serge Egelman, Giles Hogben, Jack Humphrey, Marc Langheinrich, Massimo Marchiori, Martin Presler-Marshall, Joseph M. Reagle, Matthias Schunter, David A. Stampley, and Rigo Wenning. The Platform for Privacy Preferences 1.1 (P3P1.1) specification. World Wide Web Consortium, Note NOTE-P3P11-20061113, November 2006.

[72] Lorrie Faith Cranor, Praveen Guduru, and Manjula Arjula. User interfaces for privacy agents. *ACM Trans. Comput.-Hum. Interact.*, 13(2):135–178, June 2006.

[73] Lorrie Faith Cranor, Kelly Idouchi, Pedro Giovanni Leon, Manya Sleeper, and Blase Ur. Are they actually any different? comparing thousands of financial institutions privacy practices. In *Workshop on the Economics of Information Security*, June 2013.

[74] Lorrie Faith Cranor, Marc Langheinrich, and Massimo Marchiori. A P3P Preference Exchange Language 1.0 (APPEL 1.0). World Wide Web Consortium, Working Draft WD-P3P-preferences-20020415, April 2002.

[75] Lorrie Faith Cranor, Marc Langheinrich, Massimo Marchiori, Martin Presler-Marshall, and Joseph M. Reagle. The Platform for Privacy Preferences 1.0 (P3P1.0) specification. World Wide Web Consortium, Recommendation REC-P3P-20020416, April 2002.

[76] Lorrie Faith Cranor and Joel R. Reidenberg. Can user agents accurately represent privacy notices? *TPRC*, Sept. 2002.

[77] Criteo SA. Building the cross device graph at criteo. `http://labs.criteo.com/2016/06/building-cross-device-graph-criteo/`. Last accessed: March 6, 2017.

[78] Cross Pixel Media, Inc. Media metrix multi-platform. `http://www.crosspixel.net/ourdata/`. Last accessed: March 6, 2017.

[79] Anupam Das, Nikita Borisov, and Matthew Caesar. Tracking mobile web users through motion sensors: Attacks and defenses. In *23nd Annual Network and Distributed System Security Symposium, NDSS 2016, San Diego, California, USA, February 21-24, 2016*, 2016.

[80] Emile de Maat, Kai Krabben, and Radboud Winkels. Machine learning versus knowledge based classification of legal texts. In *Proceedings of the 2010 conference on Legal Knowledge and Information Systems: JURIX 2010: The Twenty-Third Annual Conference*, pages 87–96, Amsterdam, The Netherlands, The Netherlands, 2010. IOS Press.

[81] Emile de Maat and Radboud Winkels. Automatic classification of sentences in dutch laws. In *Proceedings of the 2008 conference on Legal Knowledge and Information Systems: JURIX 2008: The Twenty-First Annual Conference*, pages 207–216, Amsterdam, The Netherlands, The Netherlands, 2008. IOS Press.

[82] Emile de Maat and Radboud Winkels. A next step towards automated modelling of sources of law. In *Proceedings of the 12th International Conference on Artificial Intelligence and Law*, ICAIL ’09, pages 31–39, New York, NY, USA, 2009. ACM.

[83] Yves-Alexandre de Montjoye, Jordi Quoidbach, Florent Robic, and Alex (Sandy) Pentland. Predicting personality using novel mobile phone-based metrics. In *Proceedings of*

*the 6th International Conference on Social Computing, Behavioral-Cultural Modeling and Prediction*, SBP'13, pages 48–55, Berlin, Heidelberg, 2013. Springer-Verlag.

[84] David Dearman and Jeffery S. Pierce. It's on my other computer!: Computing with multiple devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, pages 767–776, New York, NY, USA, 2008. ACM.

[85] Soteris Demetriou, Whitney Merrill, Wei Yang, Aston Zhang, and Carl Gunter. Free for all! assessing user data exposure to advertising libraries on android. In *23rd Annual Network and Distributed System Security Symposium, NDSS 2016, San Diego, California, USA, February 22-24, 2016*. The Internet Society, 2016.

[86] Zhui Deng, Brendan Saltaformaggio, Xiangyu Zhang, and Dongyan Xu. iris: Vetting private api abuse in ios applications. In *Proceedings of the 22Nd ACM SIGSAC Conference on Computer and Communications Security*, CCS '15, pages 44–56, New York, NY, USA, 2015. ACM.

[87] Barbara Di Eugenio and Michael Glass. The kappa statistic: a second look. *Comput. Linguist.*, 30(1):95–101, March 2004.

[88] Digital Advertising Alliance. Application of the self-regulatory principles of transparency and control to data used across devices. `http://www.aboutads.info/sites/default/files/DAA_Cross-Device_Guidance-Final.pdf`. Last accessed: March 6, 2017.

[89] Drawbridge, Inc. Drawbridge challenges scientific community to better the accuracy of its cross-device consumer graph.

[90] Dstillery, Inc. A tale of two crosswalks. `http://dstillery.com/a-tale-of-two-crosswalks/`. Last accessed: March 6, 2017.

[91] Maeve Duggan and Joanna Brenner. The demographics of social media users - 2012. *Pew Research Center*, 2013.

[92] Graham Dumpleton. Modwsgi. `https://modwsgi.readthedocs.io/en/develop/`. Last accessed: March 6, 2017.

[93] Cynthia Dwork. Differential privacy. In *33rd International Colloquium on Automata, Languages and Programming, part II (ICALP 2006)*, volume 4052 of *Lecture Notes in Computer Science*, pages 1–12. Springer Verlag, July 2006.

[94] Cynthia Dwork. Differential privacy. In *Automata, Languages and Programming, 33rd International Colloquium, ICALP 2006, Venice, Italy, July 10-14, 2006, Proceedings, Part II*, pages 1–12, 2006.

[95] Peter Eckersley. How unique is your web browser? In *Proceedings of the 10th International Conference on Privacy Enhancing Technologies*, PETS'10, pages 1–18, Berlin, Heidelberg, 2010. Springer-Verlag.

[96] William Enck, Peter Gilbert, Byung-Gon Chun, Landon P. Cox, Jaeyeon Jung, Patrick McDaniel, and Anmol N. Sheth. Taintdroid: An information-flow tracking system for realtime privacy monitoring on smartphones. In *Proceedings of the 9th USENIX Conference on Operating Systems Design and Implementation*, OSDI'10, pages 1–6, Berkeley, CA, USA, 2010. USENIX Association.

[97] William Enck, Damien Octeau, Patrick McDaniel, and Swarat Chaudhuri. A study of android application security. In *Proceedings of the 20th USENIX Conference on Security*, SEC'11, pages 21–21, Berkeley, CA, USA, 2011. USENIX Association.

[98] Steven Englehardt and Arvind Narayanan. Online tracking: A 1-million-site measurement and analysis. [Technical Report], May 2016.

[99] Steven Englehardt, Dillon Reisman, Christian Eubank, Peter Zimmerman, Jonathan Mayer, Arvind Narayanan, and Edward W. Felten. Cookies that give you away: The surveillance implications of web tracking. In *Proceedings of the 24th International Conference on World Wide Web*, WWW '15, pages 289–299, New York, NY, USA, 2015. ACM.

[100] Christian Eubank, Marcela Melara, Diego Perez-botero, and Arvind Narayanan. Shining the floodlights on mobile web tracking  a privacy survey.

[101] European Parliament and the Council of the European Union. General data protection regulation. http://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679&from=EN, April 2016. Last accessed: March 6, 2017.

[102] Experian Ltd. Device recognition by adtruth. http://www.experian.co.uk/marketing-services/products/adtruth-device-recognition.html. Last accessed: March 6, 2017.

[103] Facebook. Add facebook login to your app or website. https://developers.facebook.com/docs/facebook-login. Last accessed: March 6, 2017.

[104] Facebook Inc. Explaining facebook's recent advertising technology updates.

[105] Hossein Falaki, Ratul Mahajan, Srikanth Kandula, Dimitrios Lymberopoulos, Ramesh Govindan, and Deborah Estrin. Diversity in smartphone usage. In *Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services*, MobiSys '10, pages 179–194, New York, NY, USA, 2010. ACM.

[106] Federal Trade Commission. Privacy online: A report to congress. https://www.ftc.gov/reports/privacy-online-report-congress, 1998. Last accessed: March 6, 2017.

[107] Federal Trade Commission. Credit-based insurance scores: Impacts on consumers of automobile insurance: A report to congress by the federal trade commission. https://www.ftc.gov/reports/credit-based-insurance-scores-impacts-consumers-automobile-insurance-report-congress-federal, July 2007. Last accessed: March 6, 2017.

[108] Federal Trade Commission. Mobile apps for kids: Current privacy disclosures are disappointing. http://www.ftc.gov/os/2012/02/120216mobile_apps_kids.pdf, February 2012. Last accessed: March 6, 2017.

[109] Federal Trade Commission. Mobile apps for kids: Disclosures still not making the grade. https://www.ftc.gov/reports/mobile-apps-kids-

disclosures-still-not-making-grade, December 2012. Last accessed: March 6, 2017.

[110] Federal Trade Commission. Protecting consumer privacy in an era of rapid change. http://www.ftc.gov/reports/protecting-consumer-privacy-era-rapid-change-recommendations-businesses-policymakers, March 2012. Last accessed: March 6, 2017.

[111] Federal Trade Commission. Mobile privacy disclosures. www.ftc.gov/os/2013/02/130201mobileprivacyreport.pdf, February 2013. Last accessed: March 6, 2017.

[112] Federal Trade Commission. Big data: A tool for inclusion or exclusion? https://www.ftc.gov/news-events/events-calendar/2014/09/big-data-tool-inclusion-or-exclusion, September 2014. Last accessed: March 6, 2017.

[113] Federal Trade Commission. FTC warns childrens app maker BabyBus about potential COPPA violations. https://www.ftc.gov/news-events/press-releases/2014/12/ftc-warns-childrens-app-maker-babybus-about-potential-coppa, 2014. Last accessed: March 6, 2017.

[114] Federal Trade Commission. What's the deal? a federal trade commission study on mobile shopping apps. https://www.ftc.gov/reports/whats-deal-federal-trade-commission-study-mobile-shopping-apps-august-2014, August 2014. Last accessed: March 6, 2017.

[115] Federal Trade Commission. FTC cross-device tracking workshop. https://www.ftc.gov/news-events/events-calendar/2015/11/cross-device-tracking, November 2015. Last accessed: March 6, 2017.

[116] Federal Trade Commission. FTC cross-device tracking workshop, segment 1, transcript, November 2015.

[117] Federal Trade Commission. Internet of things, January 2015.

[118] Federal Trade Commission. Kids' apps disclosures revisited. `https://www.ftc.gov/news-events/blogs/business-blog/2015/09/kids-apps-disclosures-revisited`, 9 2015.

[119] Federal Trade Commission. Big data: A tool for inclusion or exclusion? `https://www.ftc.gov/system/files/documents/reports/big-data-tool-inclusion-or-exclusion-understanding-issues/160106big-data-rpt.pdf`, January 2016. Last accessed: March 6, 2017.

[120] Federal Trade Commission. FTC issues warning letters to app developers using 'silverpush' code. `https://www.ftc.gov/news-events/press-releases/2016/03/ftc-issues-warning-letters-app-developers-using-silverpush-code`, March 2016. Last accessed: March 6, 2017.

[121] Adrienne Porter Felt, Erika Chin, Steve Hanna, Dawn Song, and David Wagner. Android permissions demystified. In *Proceedings of the 18th ACM Conference on Computer and Communications Security*, CCS '11, pages 627–638, New York, NY, USA, 2011. ACM.

[122] Thom File. Computer and internet use in the united states. http://www.census.gov/prod/2013pubs/p20-569.pdf, May 2013.

[123] Simone Fischer-Hübner and Harald Zwingelberg. UI prototypes: Policy administration and presentation - version 2. Technical Report D4.3.2, Karlstad University, 2010.

[124] J.L. Fleiss. Measuring nominal scale agreement among many raters. *Psychological Bulletin*, 76(5):378–382, 1971.

[125] E. Francesconi and A. Passerini. Automatic classification of provisions in legislative texts. *Artif. Intell. Law*, 15(1):1–17, March 2007.

[126] Simson Garfinkel and Heather Richter Lipford. *Usable Security*. Morgan & Claypool, 2014.

[127] Robert Gellman. Fair information practices: A basic history. `http://bobgellman.com/rg-docs/rg-FIPShistory.pdf`, June 2016. Last accessed: March 6, 2017.

[128] Gesellschaft für Konsumforschung. Finding simplicity in a multi-device world. https://blog.gfk.com/2014/03/finding-simplicity-in-a-multi-device-world/, March 2014. Last accessed: March 6, 2017.

[129] Clint Gibler, Jonathan Crussell, Jeremy Erickson, and Hao Chen. Androidleaks: Automatically detecting potential privacy leaks in android applications on a large scale. In *Proceedings of the 5th International Conference on Trust and Trustworthy Computing*, TRUST'12, pages 291–307, Berlin, Heidelberg, 2012. Springer-Verlag.

[130] DW Gibson. 'i put in white tenants': The grim, racist (and likely illegal) methods of one brooklyn landlord. http://nymag.com/daily/intelligencer/2015/05/grim-racist-methods-of-one-brooklyn-landlord.html, May 2015.

[131] Shantanu Godbole and Sunita Sarawagi. Discriminative methods for multi-labeled classification. In *Proceedings of the 8th Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 22–30, Berlin, Heidelberg, Germany, 2004. Springer.

[132] Dan Goodin. Researchers find 256 ios apps that collect users personal info. http://arstechnica.com/security/2015/10/researchers-find-256-ios-apps-that-collect-users-personal-info/, Oct 2015. Last accessed: March 6, 2017.

[133] Google, Inc. Display inventory and ad formats on the google display network. https://support.google.com/partners/answer/172610?hl=en. Last accessed: March 6, 2017.

[134] Google, Inc. General ad categories. https://support.google.com/adsense/answer/3016459?hl=en. Last accessed: March 6, 2017.

[135] Google, Inc. Topics used for personalized ads. https://support.google.com/ads/answer/2842480?hl=en. Last accessed: March 6, 2017.

[136] Google, Inc. The new multi-screen world study. https://www.thinkwithgoogle.com/research-studies/the-new-multi-screen-world-study.html, August 2012. Last accessed: March 6, 2017.

[137] Google Play store. `https://play.google.com/store/apps?hl=en`. Last accessed: March 6, 2017.

[138] Michael I. Gordon, Deokhwan Kim, Jeff Perkins, Limei Gilham, Nguyen Nguyen, and Martin Rinard. Information-flow analysis of Android applications in DroidSafe. In *Proceedings of the 22nd Annual Network and Distributed System Security Symposium (NDSS)*, 2015.

[139] Alessandra Gorla, Ilaria Tavecchia, Florian Gross, and Andreas Zeller. Checking app behavior against app descriptions. In *Proceedings of the 36th International Conference on Software Engineering*, ICSE 2014, pages 1025–1035, New York, NY, USA, 2014. ACM.

[140] Michael C. Grace, Wu Zhou, Xuxian Jiang, and Ahmad-Reza Sadeghi. Unsafe exposure analysis of mobile in-app advertisements. In *Proceedings of the Fifth ACM Conference on Security and Privacy in Wireless and Mobile Networks*, WISEC '12, pages 101–112, New York, NY, USA, 2012. ACM.

[141] Griffin v. Wisconsin. 1987. 483 U.S. 868.

[142] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. The WEKA data mining software: An update. *SIGKDD Explor. Newsl.*, 11(1):10–18, November 2009.

[143] Jamie Hayes and George Danezis. k-fingerprinting: A robust scalable website fingerprinting technique. In *25th USENIX Security Symposium (USENIX Security 16)*, pages 1187–1203, Austin, TX, August 2016. USENIX Association.

[144] P. Hoffman, M.A. Lambon Ralph, and T.T. Rogers. Semantic diversity: A measure of semantic ambiguity based on variability in the contextual usage of words. *BRM*, 45(3):718–730, 2013.

[145] Candice Hoke, Lorrie Cranor, Pedro Leon, and Alyssa Au. Are They Worth Reading? An In-Depth Analysis of Online Trackers Privacy Policies. *I/S : a journal of law and policy for the information society*, April 2015.

[146] Leif-Erik Holtz, Katharina Nocun, and Marit Hansen. Towards displaying privacy information with icons. In Simone Fischer Hübner, Penny Duquenoy, Marit Hansen, Ronald Leenes, and Ge Zhang, editors, *Privacy and Identity Management for Life*, volume 352 of *IFIP Advances in Information and Communication Technology*, pages 338–348, Berlin, Heidelberg, Germany, 2011. Springer.

[147] George Hripcsak and Adam S. Rothschild. Technical brief: Agreement, the F-measure, and reliability in information retrieval. *JAMIA*, 12(3):296–298, 2005.

[148] Jian Hu, Hua-Jun Zeng, Hua Li, Cheng Niu, and Zheng Chen. Demographic prediction based on user's browsing behavior. In *Proceedings of the 16th International Conference on World Wide Web*, WWW '07, pages 151–160, New York, NY, USA, 2007. ACM.

[149] Yuheng Hu, Lydia Manikonda, and Subbarao Kambhampati. What we instagram: A first analysis of instagram photo content and user types, 2014.

[150] Jianjun Huang, Xiangyu Zhang, Lin Tan, Peng Wang, and Bin Liang. Asdroid: Detecting stealthy behaviors in android applications by user interface and program behavior contradiction. In *Proceedings of the 36th International Conference on Software Engineering*, ICSE 2014, pages 1036–1046, New York, NY, USA, 2014. ACM.

[151] John Iceland, Daniel Weinberg, and Lauren Hughes. The residential segregation of detailed Hispanic and Asian groups in the United States: 1980-2010. *Demographic Research*, 3:593–624, 2014.

[152] INFSO D.4 Networked Enterprise & RFID IMFSO G.2 Micro & Nanosystems, RFID Working Group of the European Technology Platform on Smary Systems Integration (EPOSS). Internet of things in 2020, September 2008.

[153] Sibren Isaacman, R Becker, R Cáceres, S Kobourov, Margaret Martonosi, J. Rowland, and A. Varshavsky. Identifying important places in people's lives from cellular network data. *Pervasive Computing*, pages 133–151, 2011.

[154] Sibren Isaacman, R Becker, R Cáceres, S Kobourov, Margaret Martonosi, J. Rowland, and A. Varshavsky. Ranges of human mobility in Los Angeles and New York. In *Pervasive*

*Computing and Communications Workshops (PERCOM Workshops), 2011 IEEE International Conference on*, pages 88–93, 2011.

[155] Sibren Isaacman, Richard Becker, Ramón Cáceres, Stephen Kobourov, James Rowland, and Alexander Varshavsky. A tale of two cities. In *HotMobile '10: Proceedings of the Eleventh Workshop on Mobile Computing Systems & Applications*. ACM Request Permissions, February 2010.

[156] Scott Jordan. Intersections between networking research and public policy, Oct 2015. Electrical Engineering Distinguished Lecture, Columbia University.

[157] Marc Juarez, Sadia Afroz, Gunes Acar, Claudia Diaz, and Rachel Greenstadt. A critical evaluation of website fingerprinting attacks. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, CCS '14, pages 263–274, New York, NY, USA, 2014. ACM.

[158] Kaggle, Inc. ICDM 2015: Drawbridge cross-device connections. `https://www.kaggle.com/c/icdm-2015-drawbridge-cross-device-connections/data`. Last accessed: March 6, 2017.

[159] Maryam Kamvar, Melanie Kellar, Rajan Patel, and Ya Xu. Computers and iphones and mobile phones, oh my!: A logs-based comparison of search users on different devices. In *Proceedings of the 18th International Conference on World Wide Web*, WWW '09, pages 801–810, New York, NY, USA, 2009. ACM.

[160] Shaun K. Kane, Amy K. Karlson, Brian R. Meyers, Paul Johns, Andy Jacobs, and Greg Smith. *Exploring Cross-Device Web Use on PCs and Mobile Devices*, pages 722–735. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.

[161] Günter Karjoth, Matthias Schunter, and Michael Waidner. Platform for Enterprise Privacy Practices: privacy-enabled management of customer data. In *Proceedings of the 2nd international conference on Privacy Enhancing Technologies*, PET '02, pages 69–84, Berlin, Heidelberg, Germany, 2003. Springer.

[162] Katz v. United States. 1967. 389 U.S. 361 (Harlan, J., concurring).

[163] Girma Kejela and Chunming Rong. Cross-device consumer identification. In *IEEE International Conference on Data Mining Workshop, ICDMW 2015, Atlantic City, NJ, USA, November 14-17, 2015*, pages 1687–1689, 2015.

[164] Patrick Gage Kelley. Designing a privacy label: assisting consumer understanding of online privacy practices. In *CHI '09 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '09, pages 3347–3352, New York, NY, USA, 2009. ACM.

[165] Patrick Gage Kelley, Joanna Bresee, Lorrie Faith Cranor, and Robert W. Reeder. A "nutrition label" for privacy. In *Proceedings of the 5th Symposium On Usable Privacy and Security*, SOUPS '09, pages 4:1–4:12, New York, NY, USA, 2009. ACM.

[166] Patrick Gage Kelley, Lucian Cesca, Joanna Bresee, and Lorrie Faith Cranor. Standardizing privacy notices: an online study of the nutrition label approach. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, pages 1573–1582, New York, NY, USA, 2010. ACM.

[167] Patrick Gage Kelley, Lorrie Faith Cranor, and Norman Sadeh. Privacy as part of the app decision-making process. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, pages 3393–3402, New York, NY, USA, 2013. ACM.

[168] Kelton. 4th annual springhill suites annual travel survey. http://news.marriott.com/springhill-suites-annual-travel-survey.html, April 2013.

[169] Michael Sungjun Kim, Jiwei Liu, Xiaozhou Wang, and Wei Yang. Connecting devices to cookies via filtering, feature engineering, and boosting. In *IEEE International Conference on Data Mining Workshop, ICDMW 2015, Atlantic City, NJ, USA, November 14-17, 2015*, pages 1690–1694, 2015.

[170] Lukasz Kobylinski and Adam Przepiorkowski. Definition extraction with balanced random forests. In *Proceedings of the 6th international conference on Advances in Natural Language Processing*, GoTAL '08, pages 237–247, Berlin, Heidelberg, Germany, 2008. Springer.

[171] Deguang Kong, Lei Cen, and Hongxia Jin. Autoreb: Automatically understanding the review-to-behavior fidelity in android applications. In *CCS '15*. ACM, 2015.

[172] K. Krippendorff. *Content analysis: An introduction to its methodology.* SAGE, Beverly Hills, CA, USA, 1980.

[173] Florian Kuhn. A description language for content zones of German court decisions. In *Proceedings of the LREC 2010 Workshop on the Semantic Processing of Legal Texts*, SPLeT '10, pages 1–7, 2010.

[174] Andreas Kurtz, Andreas Weinlein, Christoph Settgast, and Felix Freiling. Dios: Dynamic privacy analysis of ios applications. Technical Report CS-2014-03, Friedrich-Alexander-Universität Erlangen-Nürnberg, Dept. of Computer Science, June 2014.

[175] Mei-Po Kwan. Gender, the home-work link, and space-time patterns of nonemployment activities. *Economic Geography*, 75(4):pp–370, 1999.

[176] Mark Landry, Sudalai Rajkumar S, and Robert Chong. Multi-layer classification: ICDM 2015 drawbridge cross-device connections competition. In *IEEE International Conference on Data Mining Workshop, ICDMW 2015, Atlantic City, NJ, USA, November 14-17, 2015*, pages 1695–1698, 2015.

[177] Pedro Giovanni Leon, Lorrie Faith Cranor, Aleecia M. McDonald, and Robert McGuire. Token attempt: The misrepresentation of website privacy policies through the misuse of P3P compact policy tokens. In *Proceedings of the 9th Annual ACM Workshop on Privacy in the Electronic Society*, WPES '10, pages 93–104, New York, NY, USA, 2010. ACM.

[178] Adam Lerner, Anna Kornfeld Simpson, Tadayoshi Kohno, and Franziska Roesner. Internet jones and the raiders of the lost trackers: An archaeological study of web tracking from 1996 to 2016. In *25th USENIX Security Symposium (USENIX Security 16)*, Austin, TX, August 2016. USENIX Association.

[179] Kevin Lewis, Jason Kaufman, and Nicholas Christakis. The taste for privacy: An analysis of college student privacy settings in an online social network. *J. Computer-Mediated Communication*, 14(1):79–100, 2008.

[180] Jialiu Lin, Bin Liu, Norman Sadeh, and Jason I. Hong. Modeling users' mobile app privacy preferences: Restoring usability in a sea of permission settings. In *SOUPS '14*. USENIX Assoc., July 2014.

[181] Bin Liu, Bin Liu, Hongxia Jin, and Ramesh Govindan. Efficient privilege de-escalation for ad libraries in mobile apps. In *Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services*, MobiSys '15, pages 89–103, New York, NY, USA, 2015. ACM.

[182] Fei Liu, Rohan Ramanath, Norman Sadeh, and Noah A. Smith. A step towards usable privacy policy: Automatic alignment of privacy statements. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 884–894, August 2014.

[183] Lotame Solutions, Inc. Skimlinks and lotame unleash enhanced retail intent data. `https://www.lotame.com/resource/skimlinks-lotame-dmp/`. Last accessed: March 6, 2017.

[184] Ashwin Machanavajjhala, Daniel Kifer, Johannes Gehrke, and Muthuramakrishnan Venkitasubramaniam. L-diversity: Privacy beyond k-anonymity. *ACM Trans. Knowl. Discov. Data*, 1(1), March 2007.

[185] Mary Madden. Privacy management on social media sites. *Pew Research Center*, 2012.

[186] Mary Madden, Amanda Lenhart, Sandra Cortesi, Urs Grasser, Maeve Duggan, Aaron Smith, and Meredith Beaton. Teens, social media, and privacy. *Pew Research Center*, 2013.

[187] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, New York, NY, USA, 2008.

[188] Florencia Marotta-Wurgler. Does contract disclosure matter? *JITE*, 168(1):94–119, 2012.

[189] Aaron K. Massey, Jacob Eisenstein, Annie I. Antón, and Peter P. Swire. Automated text mining for requirements analysis of policy documents. In *21st IEEE International Re-*

*quirements Engineering Conference, RE 2013, Rio de Janeiro-RJ, Brazil, July 15-19, 2013*, pages 4–13, 2013.

[190] Douglas S. Massey and Nancy A. Denton. The dimensions of residential segregation. *Social Forces*, 67(2):281–315, 1988.

[191] Aleecia M. Mcdonald, Robert W. Reeder, Patrick Gage Kelley, and Lorrie Faith Cranor. A comparative study of online privacy policies and formats. In *Proceedings of the 9th international symposium on Privacy Enhancing Technologies*, PETS '09, pages 37–55, Berlin, Heidelberg, Germany, 2009. Springer.

[192] Sara McDonough and David L. Brunsma. Navigating the color complex: How multiracial individuals narrate the elements of appearance and dynamics of color in twenty-first-century america. In Ronald E. Hall, editor, *The Melanin Millennium*. Springer, Dordrecht, 2013.

[193] Alan Mislove, Sune Lehmann, Yong-Yeol Ahn, Jukka-Pekka Onnela, and J. Niels Rosenquist. Understanding the Demographics of Twitter Users. In *Proceedings of the 5th International AAAI Conference on Weblogs and Social Media (ICWSM'11)*, Barcelona, Spain, July 2011.

[194] Marie-Francine Moens, Erik Boiy, Raquel Mochales Palau, and Chris Reed. Automatic detection of arguments in legal texts. In *Proceedings of the 11th International Conference on Artificial Intelligence and Law*, ICAIL '07, pages 225–230, New York, NY, USA, 2007. ACM.

[195] Nitesh Mor, Oriana Riva, Suman Nath, and John Kubiatowicz. Bloom cookies: Web search personalization without user tracking. In *NDSS*. The Internet Society, 2015.

[196] Roberto Diáz Morales. Cross-device tracking: Matching devices and cookies. In *IEEE International Conference on Data Mining Workshop, ICDMW 2015, Atlantic City, NJ, USA, November 14-17, 2015*, 2015.

[197] Mozilla. Lightbeam for Firefox. `https://www.mozilla.org/en-US/lightbeam/`. Last accessed: March 6, 2017.

[198] MyBrowserAddon. User-agent switcher. `http://mybrowseraddon.com/useragent-switcher.html`. Last accessed: March 6, 2017.

[199] National Telecommunications and Information Administration. Short form notice code of conduct to promote transparency in mobile app practices. `http://www.ntia.doc.gov/files/ntia/publications/july_25_code_draft.pdf`, July 2013. Last accessed: March 6, 2017.

[200] Annalee Newitz. Facebooks ad platform now guesses at your race based on your behavior, March 2016.

[201] Nick Nikiforakis, Alexandros Kapravelos, Wouter Joosen, Christopher Kruegel, Frank Piessens, and Giovanni Vigna. Cookieless monster: Exploring the ecosystem of web-based device fingerprinting. In *Proceedings of the 2013 IEEE Symposium on Security and Privacy*, SP '13, pages 541–555, Washington, DC, USA, 2013. IEEE Computer Society.

[202] Lukasz Olejnik, Claude Castelluccia, and Artur Janc. On the uniqueness of web browsing history patterns. *Annales des Télécommunications*, 69(1-2):63–74, 2014.

[203] Kenneth Olmstead and Michelle Atkinson. Apps permissions in the Google Play store. `http://www.pewinternet.org/2015/11/10/apps-permissions-in-the-google-play-store/`, November 2015. Last accessed: March 6, 2017.

[204] Onyxbits. `http://www.onyxbits.de/raccoon`. Last accessed: March 6, 2017.

[205] Oracle. The BlueKai registry - putting consumers in control of their digital footprint. `http://bluekai.com/registry/`. Last accessed: March 6, 2017.

[206] Oracle. Cross-device retargeting. `https://www.crosswise.com/cross-device-learning-center/cross-device-retargeting/`. Last accessed: March 6, 2017.

[207] PageFair. Adblocking goes mobile. `https://pagefair.com/downloads/2016/05/Adblocking-Goes-Mobile.pdf`. Last accessed: March 6, 2017.

[208] Andriy Panchenko, Fabian Lanze, Andreas Zinnen, Martin Henze, Jan Pennekamp, Klaus Wehrle, and Thomas Engel. Website fingerprinting at internet scale. In *23rd Annual Network and Distributed System Security Symposium (NDSS 2016), San Diego, CA, USA*. Internet Society, 2 2016.

[209] Rebecca Passonneau. Measuring Agreement on Set-valued Items (MASI) for semantic and pragmatic annotation. In *Proceedings of the international Conference on Language Resources and Evaluation*, LREC '06, 2006.

[210] Rebecca J. Passonneau and Bob Carpenter. The benefits of a model of annotation. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 187–195, Stroudsburg, PA, USA, 2013. ACL.

[211] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[212] Marco Pennacchiotti and Ana-Maria Popescu. A machine learning approach to twitter user classification, 2011.

[213] Pivotal Software, Inc. Rabbitmq. `https://www.rabbitmq.com/`. Last accessed: March 6, 2017.

[214] ponty. Pyvirtualdisplay. `https://pypi.python.org/pypi/PyVirtualDisplay`. Last accessed: March 6, 2017.

[215] Martin Porter. An algorithm for suffix stripping. *Program: electronic library and information systems*, 14(3):130–137, 1980.

[216] Progress Software Corporation. Fiddler. `http://www.telerik.com/fiddler`. Last accessed: March 6, 2017.

[217] Quantcast. Top sites. `https://www.quantcast.com/top-sites`. Last accessed: March 6, 2017.

[218] Anand Rajaraman and Jeffrey David Ullman. *Mining of massive datasets*. Cambridge University Press, New York, NY, USA, 2012.

[219] Rohan Ramanath, Fei Liu, Norman Sadeh, and Noah A. Smith. Unsupervised alignment of privacy policies using hidden markov models. In *Proceedings of the Annual Meeting of the Association of Computational Linguistics (Short Papers)*, pages 605–610, 2014.

[220] S. F. Reardon. *A Conceptual Framework for Measuring Segregation and its Association with Population Outcomes*, chapter 7, pages 169–192. John Wiley Sons, San Francisco, CA, USA, 2006.

[221] Robert W. Reeder. *Expandable Grids: a user interface visualization technique and a policy semantics to support fast, accurate security and privacy policy authoring*. PhD thesis, Pittsburgh, PA, USA, 2008. AAI3321049.

[222] Robert W. Reeder, Patrick Gage Kelley, Aleecia M. McDonald, and Lorrie Faith Cranor. A user study of the Expandable Grid applied to P3P privacy policy visualization. In *Proceedings of the 7th ACM Workshop on Privacy in the Electronic Society*, WPES '08, pages 45–54, New York, NY, USA, 2008. ACM.

[223] Joel R. Reidenberg. The use of technology to assure internet privacy: Adapting labels and filters for data protection. *Lex Electronica*, 3(2), 1997.

[224] Joel R. Reidenberg, Travis Breaux, Lorrie Faith Cranor, Brian French, Amanda Grannis, James T. Graves, Fei Liu, Aleecia McDonald, Thomas B. Norton, Rohan Ramanath, N. Cameron Russell, Norman Sadeh, and Florian Schaub. Disagreeable privacy policies: Mismatches between meaning and users' understanding. *Berkeley Technology Law Journal*, 30(1):39–88, 2015.

[225] Dennis Reidsma and Jean Carletta. Reliability measurement without limits. *Comput. Linguist.*, 34(3):319–326, September 2008.

[226] Armin Ronacher. Flask. `http://flask.pocoo.org/`. Last accessed: March 6, 2017.

[227] John T. Roscoe and Jackson A. Byars. An Investigation of the Restraints with Respect to Sample Size Commonly Imposed on the Use of the Chi-Square Statistic. *Journal of the American Statistical Association*, 66(336):755–759, December 1971.

[228] Rubicon Project. Rubicon project advertising technology privacy policy, april 11, 2016. `http://rubiconproject.com/rubicon-project-yield-optimization-privacy-policy/`. Last accessed: March 6, 2017.

[229] Ira S. Rubinstein. Privacy and regulatory innovation: Moving beyond voluntary codes. *ISJLP*, 6(3):355–423, 2011.

[230] Norman Sadeh, Alessandro Acquisti, Travis D. Breaux, Lorrie Faith Cranor, Aleecia M. McDonald, Joel R. Reidenberg, Noah A. Smith, Fei Liu, N. Cameron Russell, Florian Schaub, and Shomir Wilson. The usable privacy policy project: Combining crowdsourcing, machine learning and natural language processing to semi-automatically answer those privacy questions users care about. Tech. report CMU-ISR-13-119, Carnegie Mellon University, 2013.

[231] SeleniumHQ. Seleniumhq browser automation. `http://www.seleniumhq.org/`. Last accessed: March 6, 2017.

[232] Lars Ropeid Selsaas, Bikash Agrawal, Chunming Rong, and Tomasz Wiktorski. AFFM: auto feature engineering in field-aware factorization machines for predictive analytics. In *IEEE International Conference on Data Mining Workshop, ICDMW 2015, Atlantic City, NJ, USA, November 14-17, 2015*, pages 1705–1709, 2015.

[233] Shayak Sen, Saikat Guha, Anupam Datta, Sriram K. Rajamani, Janice Tsai, and Jeannette M. Wing. Bootstrapping privacy compliance in big data systems. In *SP '14*. IEEE Comp. Soc., 2014.

[234] Robert W. Shirey. Internet Security Glossary, Version 2. ISE RFC 4949, March 2013.

[235] R. Slavin, X. Wang, M.B Hosseini, W. Hester, R. Krishnan, J. Bhatia, T.D. Breaux, and J. Niu. Toward a framework for detecting privacy policy violation in android application

code. In *ACM/IEEE 38th International Software Engineering Conference*, ICSE '16, May 2016. to appear.

[236] Daniel Solove and Paul Schwartz. *Information Privacy Law*. Aspen Publishers, 4th edition, 2011.

[237] Gregory D. Squires. *From Redlining to Reinvestment: Community Responses to Urban Disinvestment*. Temple University Press, 1992.

[238] John W. Stamey and Ryan A. Rossi. Automatically identifying relations in privacy policies. In *27th ACM International Conference on Design of Communication*, SIGDOC '09, pages 233–238, New York, NY, USA, 2009. ACM.

[239] Statista. Social networking time per user in the united states in july 2012, by ethnicity (in hours and minutes). http://www.statista.com/statistics/248158/social-networking-time-per-us-user-by-ethnicity/, 2012.

[240] Manfred Stede and Florian Kuhn. Identifying the content zones of German court decisions. In *Lecture Notes in Business Information Processing*, volume 37 of *BIS '09*, pages 310–315, Berlin, Heidelberg, Germany, 2009. Springer.

[241] Latanya Sweeney. K-anonymity: A model for protecting privacy. *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.*, 10(5):557–570, October 2002.

[242] LINDA TAUSCHER and SAUL GREENBERG. How people revisit web pages. *Int. J. Hum.-Comput. Stud.*, 47(1):97–137, July 1997.

[243] The Apache Software Foundation. Apache. `http://httpd.apache.org/`. Last accessed: March 6, 2017.

[244] Chad Tossell, Philip Kortum, Ahmad Rahmati, Clayton Shepard, and Lin Zhong. Characterizing web use on smartphones. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pages 2769–2778, New York, NY, USA, 2012. ACM.

[245] G. Tsoumakas and I. Katakis. Multi label classification: An overview. *IJDWM*, 3(3):1–13, 2007.

[246] Connor Tumbleson and Ryszard Wiśniewski. Apktool. `https://ibotpeaches.github.io/Apktool/`. Last accessed: March 6, 2017.

[247] United States Census Bureau. 2010 census. http://factfinder2.census.gov/faces/nav/jsf/pages/index.xhtml, 2010.

[248] United States v. Jones. 1970. 397 U.S. 358, 370 (Harlan, J., concurring).

[249] United States v. Jones. 2012. 132 S. Ct. 945.

[250] United States v. Jones. 2012. 132 S. Ct. 945, 955 (Sotomayor, J., concurring) (quoting People v. Weaver, 12 N.Y.3d 433, 441-42 (2009)).

[251] United States v. Jones. 2012. 132 S. Ct. 945 (Alito, J., concurring).

[252] United States v. Knotts. 1983. 460 U.S. 276.

[253] United States v. Maynard. 2010. 615 F.3d 544, 562.

[254] Jeremy Walthers. Learning to rank for cross-device identification. In *IEEE International Conference on Data Mining Workshop, ICDMW 2015, Atlantic City, NJ, USA, November 14-17, 2015*, pages 1710–1712, 2015.

[255] Takuya Watanabe, Mitsuaki Akiyama, Tetsuya Sakai, and Tatsuya Mori. Understanding the inconsistencies between text descriptions and the use of privacy-sensitive resources of mobile apps. In *Eleventh Symposium On Usable Privacy and Security (SOUPS 2015)*, pages 241–255, Ottawa, July 2015. USENIX Association.

[256] Eline Westerhout. Definition extraction using linguistic and structural features. In *Proceedings of the 1st Workshop on Definition Extraction*, WDE '09, pages 61–67, Stroudsburg, PA, USA, 2009. ACL.

[257] Eline Westerhout. Extraction of definitions using grammar-enhanced machine learning. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics: Student Research Workshop*, EACL '09, pages 88–96, Stroudsburg, PA, USA, 2009. ACL.

[258] Alan F. Westin. *Privacy and freedom*. Atheneum, New York, 1970.

[259] Michael J. White. Segregation and diversity measures in population distribution. *Population Index*, 52(2):198–221, 1986.

[260] Shomir Wilson, Florian Schaub, Aswarth Abhilash Dara, Frederick Liu, Sushain Cherivirala, Pedro Giovanni Leon, Mads Schaarup Andersen, Sebastian Zimmeck, Kanthashree Mysore Sathyendra, N. Cameron Russell, Thomas B. Norton, Eduard Hovy, Joel Reidenberg, and Norman Sadeh. The creation and analysis of a website privacy policy corpus. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, ACL '16, Berlin, Germany, August 2016. ACL.

[261] Shomir Wilson, Florian Schaub, Rohan Ramanath, Norman Sadeh, Fei Liu, Noah A. Smith, and Fredrick Liu. Crowdsourcing annotations for websites' privacy policies: Can it really work? In *WWW '16: 25th International World Wide Web Conference*, 2016.

[262] Guangyu Wu, Derek Greene, and Pádraig Cunningham. Merging multiple criteria to identify suspicious reviews. In *Proceedings of the Fourth ACM Conference on Recommender Systems*, RecSys '10, pages 241–244, New York, NY, USA, 2010. ACM.

[263] Yahoo! Inc. Flurry pulse. `https://developer.yahoo.com/flurry-pulse/`. Last accessed: March 6, 2017.

[264] Yahoo! Inc. Personas. `https://developer.yahoo.com/flurry/docs/analytics/lexicon/personas/`. Last accessed: March 6, 2017.

[265] Jane Yakowitz. More crap from the E.U. `https://blogs.harvard.edu/infolaw/2012/01/25/more-crap-from-the-e-u/`, January 2012. Last accessed: March 6, 2017.

[266] Jean Yang, Kuat Yessenov, and Armando Solar-Lezama. A language for automatically enforcing privacy policies. In *Proceedings of the 39th annual ACM SIGPLAN-SIGACT symposium on Principles Of Programming Languages*, POPL '12, pages 85–96, New York, NY, USA, 2012. ACM.

[267] Le Yu, Tao Zhang, Xiapu Luo, and Lei Xue. Autoppg: Towards automatic generation of privacy policy for android applications. In *Proceedings of the 5th Annual ACM CCS Workshop on Security and Privacy in Smartphones and Mobile Devices*, SPSM ’15, pages 39–50, New York, NY, USA, 2015. ACM.

[268] Mu Zhang, Yue Duan, Qian Feng, and Heng Yin. Towards automatic generation of security-centric descriptions for android apps. In *CCS ’15*. ACM, 2015.

[269] Sebastian Zimmeck and Steven M. Bellovin. Privee: An architecture for automatically analyzing web privacy policies. In *23rd USENIX Security Symposium (USENIX Security 14)*, pages 1–16, San Diego, CA, Aug 2014. USENIX Association.

# Appendices

# Appendix A

# Datasets used in § 6

## A.1 Policy and App Datasets

1. Full App Set - Total Apps Collected (n=17,991)
2. Full Policy Set - Policies Obtained via the Play Store Policy Link for the Apps in the Full App Set (n=9,295)
3. Full App/Policy Set - App/Policy Pairs from the Full App and Policy Sets adjusted for Links not leading to a Policy (n=9,050)
4. App Test Set - Random Apps from the Publishers in the Policy Test Set (n=40)
5. Policy Test Set - Random Policies from the OPP-115 Corpus (n=40)
6. App/Policy Test Set - Apps from the App Test Set and Associated Policies from the Policy Test Set (n=40)

# Appendix B

# Cross-device Tracking Dataset Details (§ 7)

## B.1 Device Fingerprint

1. User Agent
2. Browser Engine
3. Installed Browser Plugins
4. Installed Adobe Flash Plugin/Version
5. Installed Microsoft Silverlight Plugin/Version
6. Flash Cookies Enabled
7. Time Zone
8. Screen (Color Depth, Screen Dimensions)
9. System Language
10. First Party HTTP Cookies Enabled
11. Third Party HTTP Cookies Enabled
12. JavaScript Enabled
13. Java Enabled
14. Do Not Track Enabled
15. Touch Enabled
16. Latency (Request Duration, Roundtrip Duration)

17. Installed Fonts
18. IP Address
19. HTML5 Web Storage Enabled (Local, Session)
20. HTML5 Geolocation Enabled (Latitude, Longitude)
21. HTTP Accept Headers
22. Internet Connection Type (Wi-Fi, Cellular)
23. Internet Service Provider

## B.2 App and Browsing History

1. IP Address
2. Browser Vendor
3. Date
4. Time
5. Time Zone
6. Browser Tab ID
7. Full HTTP Referrer URL
8. Full URL/App Package Name
9. URL Title
10. Third Party Trackers/SDKs
11. App/URL Mapping

## B.3 Google Interest Categories ($n = 126$ users)

1. Arts and Entertainment (68%)
2. Food and Drink (64%)
3. Computers and Electronics (63%)
4. Science (62%)
5. News (60%)
6. Books and Literature (55%)
7. Jobs and Education (52%)

8. Games (43%)
9. Travel (40%)
10. Law and Government (37%)
11. Shopping (36%)
12. Hobbies and Leisure (34%)
13. People and Society (34%)
14. Beauty and Fitness (33%)
15. Internet and Telecom (33%)
16. Sports (29%)
17. Online Communities (24%)
18. Finance (23%)
19. Pets and Animals (23%)
20. Business and Industrial (21%)
21. World Localities (15%)
22. Reference (13%)
23. Autos and Vehicles (11%)
24. Home and Garden (11%)
25. Real Estate (4%)

## B.4 Flurry Analytics Personas ($n = 126$ users)

1. Music Lovers (47%)
2. Movie Lovers (46%)
3. Food and Dining Lovers (40%)
4. Singles (39%)
5. Bookworms (33%)
6. Entertainment Enthusiasts (31%)
7. Tech and Gadget Enthusiasts (31%)
8. Casual and Social Gamers (30%)
9. News and Magazine Readers (23%)

10. Leisure Travelers (21%)
11. Sports Fans (21%)
12. Health and Fitness Enthusiasts (20%)
13. Mobile Payment Makers (19%)
14. Value Shoppers (18%)
15. Parenting and Education (15%)
16. Pet Owners (14%)
17. Business Professionals (13%)
18. American Football Fans (11%)
19. Hardcore Gamers (11%)
20. Photo and Video Enthusiasts (11%)
21. Fashionistas (10%)
22. Personal Finance Geeks (10%)
23. Avid Runners (7%)
24. Flight Intenders (6%)
25. Social Influencers (6%)
26. Catalog Shoppers (5%)
27. Auto Enthusiasts (3%)
28. Business Travelers (3%)
29. Small Business Owners (3%)
30. Home Design Enthusiasts (2%)
31. Real Estate Followers (2%)
32. High Net Individuals (1%)
33. Mothers (1%)
34. Home and Garden Pros (0%)
35. New Mothers (0%)
36. Slots Players (0%)

## B.5 Native Language ($n = 126$ users)

1. English (64%)
2. Chinese Origin (8%)
3. Indian Origin (8%)
4. Greek (3%)
5. Spanish (3%)
6. French (2%)
7. Korean (2%)
8. Portuguese (2%)
9. Turkish (2%)
10. Vietnamese (2%)
11. Others (5%)

## B.6 Age Groups ($n = 126$ users)

1. 18–20 (18%)
2. 21–25 (51%)
3. 26–30 (21%)
4. 31–35 (6%)
5. Over 35 (3%)

## B.7 Gender ($n = 126$ users)

1. Female (34%)
2. Male (66%)