Fairness and Privacy Violations in Black-Box Personalization Systems: Detection and Defenses

Submitted in partial fulfillment of the requirements for

the degree of

Doctor of Philosophy

in

Electrical and Computer Engineering

Amit Datta

B.Tech., Computer Science and Engineering, IIT Kharagpur

Carnegie Mellon University
Pittsburgh, PA

March 2018

Acknowledgements

I am thankful to many individuals for making my doctoral program a very rich experience. Throughout my program, I have come across many individuals who have helped create the researcher that I am today.

First and foremost, I would like to thank my adviser, Prof. Anupam Datta. From him, I have learnt everything that I know about research. He has encouraged me to work on difficult research problems and has closely guided my search for a solution. He has been very supportive of me and my ideas. With his feedback, I have been able to exponentially improve my reading, writing and presentation skills, which will continue to serve me for the rest of my life. Anupam played a key role in making my doctoral program an extremely rich and rewarding learning experience.

I am thankful to all of my dissertation committee members for providing thoughtful feedback on my work. The productive discussions with them considerably improved the content and presentation of my dissertation. I am and will always remain grateful to Dr. Michael C. Tschantz, who I had the pleasure of closely working with throughout my doctoral program. He has helped shape my thinking all the way from the bird's eye view of a problem to the low-level details. Dr. Saikat Guha's study of real-world advertising systems inspired the applications of my research. I am also grateful to Saikat for giving me the opportunity to gain research exposure at Microsoft Research as an intern. Prof. Lorrie Cranor encouraged me to explore the policy and legal implications of my research findings. Detailed feedback from Prof. Lujo Bauer and Prof. Nicolas Christin helped enhance the quality of the dissertation significantly.

I am grateful to my collaborators Lay Kuan Loh, Amber Lu, Jael Makagon, Prof. Deirdre Mulligan and Prof. Jeannette Wing with whom I have published various parts of this thesis. I would like to thank Dr. Divya Sharma, Dr. Arunesh Sinha, Shayak Sen, Dr. Piotr Madziel, Gihyuk Ko, Dr. Saurabh Shintre, Rajat Kateja, Soo-Jin Moon, and other members of the CyLab community for many thought-provoking discussions.

This journey would have been bleak without the many friendships that I developed during the program. I am thankful to Pratiti, Arka, Utsav, Megha, Sudipto, Deepoo, and many others, for being there when I needed them the most.

I am thankful to my family for their constant support and encouragement. None of this would have been possible without my parents, Asesh and Mousumi Datta. I am grateful to my sister, Arishma Datta, for motivating me to stay on track. Finally, I am indebted to Arthita Ghosh, who has been a pillar of support as my girlfriend, fiancée and wife as I navigated through the program.

My research has been partly funded by the CyLab Security and Privacy Institute at Carnegie Mellon University under grants CCF0424422, CNS1064688, and 1704845 from the National Science Foundation.

Abstract

Black box personalization systems have become ubiquitous in our daily lives. They utilize collected data about us to make critical decisions such as those related to credit approval and insurance premiums. This leads to concerns about whether these systems respect expectations of fairness and privacy. Given the black box nature of these systems, it is challenging to test whether they satisfy certain fundamental fairness and privacy properties. For the same reason, while many black box privacy enhancing technologies offer consumers the ability to defend themselves from data collection, it is unclear how effective they are. In this doctoral thesis, we demonstrate that carefully designed methods and tools that soundly and scalably discover causal effects in black box software systems are useful in evaluating personalization systems and privacy enhancing technologies to understand how well they protect fairness and privacy. As an additional defense against discrimination, this thesis also explores legal liability for ad platforms in serving discriminatory ads.

To formally study fairness and privacy properties in black box personalization systems, we translate these properties into information flow instances and develop methods to detect information flow. First, we establish a formal connection between information flow and causal effects. As a consequence, we can use randomized controlled experiments, traditionally used to detect causal effects, to detect information flow through black box systems. We develop AdFisher as a general framework to perform information flow experiments scalably on web systems and use it to evaluate discrimination, transparency, and choice on Google's advertising ecosystem. We find evidence of gender-based discrimination in employment-related ads and a lack of transparency in Google's transparency tool when serving ads for rehabilitation centers after visits to websites about substance abuse.

Given the presence of discrimination and the use of sensitive attributes in personalization systems, we explore possible defenses for consumers. First, we evaluate the effectiveness of publicly available privacy enhancing technologies in protecting consumers from data collection by online trackers. Specifically, we use a combination of experimental and observational approaches to examine how well the technologies protect consumers against fingerprinting, an advanced form of tracking. Next, we explore legal liability for an advertising platform like Google for delivering employment and housing ads in a discriminatory manner under Title VII and the Fair Housing Act respectively. We find that an ad platform is unlikely to incur liability under Title VII due to its limited coverage. However, we argue that housing ads violating the Fair Housing Act could create liability if the ad platform targets ads toward or away from protected classes without explicit instructions from the advertiser.

Contents

Contents v

List of Tables vii

List of Figures x

1 Introduction 1

2 A Methodology for Information Flow Experiments 6

2.1 Introduction 6

2.2 Information Flow Analysis 10

2.3 Causality 15

2.4 Experimental Design 17

2.5 Statistical Analysis 20

2.6 Prior Studies of WDUD 26

2.7 Comparison of Test Statistics 28

2.8 Conclusions and Suggestions 30

3 Discrimination, Opacity, and Choice in Google's Ad Ecosystem 32

3.1 Introduction 32

3.2 Prior Work 37

3.3 Privacy Properties 38

3.4 Methodology 41

3.5 AdFisher 45

3.6 Experiments 49

3.7 Discussion and Conclusion 55

4 An Evaluation of Privacy Enhancing Technologies against Browser Fingerprinting 57

4.1 Introduction . . . 57

4.2 Prior Work . . . 62

4.3 Trackers and PETs . . . 62

4.4 Observational Evaluation of PETs . . . 65

4.5 Experimental Evaluation of PETs . . . 68

4.6 Hybrid Evaluation of PETs . . . 75

4.7 Application: Informing PET design . . . 78

4.8 Conclusion and Discussion . . . 80

5 A Legal Analysis of Discrimination in Online Advertising 81

5.1 Introduction . . . 81

5.2 Related Work . . . 83

5.3 A primer on Google's web advertising . . . 84

5.4 Possible Causes of Discrimination . . . 87

5.5 Title VII and Prohibitions on Discriminatory Advertising . . . 94

5.6 Conclusions . . . 104

6 Conclusions and Future Work 106

6.1 Conclusions . . . 106

6.2 Future Work . . . 106

A Appendices for A Methodology for Information Flow Experiments 109

A.1 System Formalism . . . 109

A.2 Universal Unsoundness and Incompleteness Proofs . . . 111

A.3 Background: Causality . . . 112

A.4 Interference and Causation . . . 114

A.5 Details of Experiments . . . 119

B Appendices for Experiments on Google's Ad Ecosystem 131

B.1 Tables . . . 131

B.2 Details of Methodology . . . 131

B.3 Holm-Bonferroni Correction . . . 137

C Appendices for An Evaluation of PETs against Fingerprinting 138

C.1 Additional Tables . . . 138

C.2 Additional Experiment 145

Bibliography **148**

List of Tables

2.1 General Terminology and Two Instances of Experimental Science. In the chemical study, a scientist studies whether a chemical is carcinogenic when added to the diet of mice. In a behavioral marketing study, the scientist studies whether changes in visitor behavior causes changes in ads. 17

2.2 Prior Works Experimenting on Online Marketing Systems with Other Methods. 26

2.3 The number of runs out of 20 experiments that various tests consider to show information usage. 30

3.1 Privacy Properties Tested on Google's Ad Settings 39

4.1 List of PETs we study, their abbreviation, and approach to protection. Most PETs are browser extensions, * indicates full browsers, and ** indicates browser configurations. 64

4.2 PET masks as purported and observed by PETInspector. □ indicates the PET's documentation claiming the attribute is masked. The remaining symbols represent the possible outputs of PETInspector: ✓ indicates observed masking, × indicates no masking found even when our experiment is likely to detect it, and · indicates inconclusive results (not shown for □s). 73

4.3 Trackability metrics for PETs. TBB* denotes an evaluation done with a handcrafted, more accurate model of TBB's masking. 77

4.4 Entropy-based effectiveness of PETs on fingerprint samples scaled according to the PET's popularity, sorted according to the effectiveness 77

4.5 Comparison of effectiveness and loss of TBB's original spoofing strategies with alternate strategies. Trackability metrics and losses are rounded to three decimal places and to three significant digits respectively. 78

5.1 Possible causes of the 'Gender and Jobs' experiment finding organized around four actors . . . 88

A.1 For Experiments 2 and 3, the list of websites returned by Google upon searching with corresponding term. These websites were used for creating the profile of an auto enthusiast. 120

A.2 For Experiment 2, the number of unique ads collected. I denotes the set of all ads collected from the primary browser instance running in isolation, while P denotes the same collected from the primary browser instance running in parallel. This table shows the number of ads collected in each round as well as the total number of ads and the number of unique ads in I and P. The stars represent numbers from the instances running in isolation. 124

A.3 For Experiment 3, how the ads were distributed over the 10 different instances. T denotes the set of all ads collected from the trained instances, while U denotes the same collected from the untrained instances. The number of ads collected by each instance in $\{i_1 \ldots i_{10}\}$ is shown in the left half of the table. The right half of the table shows the total number of ads and the number of unique ads in T and U. 125

A.4 p-values for the permutation tests in the experiment-control experiment 126

A.5 For Experiment 3, how the ads were distributed over the 10 different instances in the *control-control* experiment. 5 out these 10 were randomly assigned to T, while the remaining to U. Observe that data-set 8 is an outlier because the instances in that round returned much fewer ads. 127

A.6 For Experiment 3, p-values for the for control-control experiment. Note that the significant p-values are from data-set 8, which we showed in Table A.5 to an outlier. 128

A.7 For Experiment 3, how the ads were distributed over the 10 different instances in the *treatment-treatment* experiment. 5 out these 10 were randomly assigned to T, while the remaining to U. 129

A.8 For Experiment 3, p-values for the for treatment-treatment experiment 130

B.1 Summary of our experimental results. Ads are collected from the Times of India (TOI) or the Guardian (G). We report how long each experiment took, how many ads were collected for it, and what result we concluded. 132

B.2 Results from the discrimination experiments sorted by unadjusted p-value. TOI stands for Times of India. G stands for the Guardian. * denotes statistically significant results under the Holm-Bonferroni method. All these experiments had 100 blocks 132

B.3 Top URL+titles for the gender and jobs experiment on the Times of India in May. 133

B.4 Results from transparency experiments. TOI stands for Times of India, G stands for the Guardian. Every experiment for this property ran with 100 blocks. * denotes statistically significant results under the Holm-Bonferroni method. 133

B.5 Top URL+titles for substance abuse experiment on the Times of India in May. 134

B.6 Top URL+titles for disability experiment on the Times of India in May. 134

B.7 Results from effectful choice experiments using the Times of India sorted by unadjusted p-value. * denotes statistically significant results under the Holm-Bonferroni method. 134

B.8 Top URL+titles for the dating experiment on Times of India in May. 135

B.9 Setup for and ads from ad choice experiments. All experiments used 10 blocks. The same keywords are used to remove ad interests, as well as create the test statistic for permutation test. 135

B.10 P-values from ad choice experiments sorted by the (unflipped) p-value. The Bonferroni adjusted p-value is only adjusted for the two hypotheses tested within a single experiment (row). The Holm-Bonferroni adjusts for all 6 hypotheses. * denotes statistically significant results under the Holm-Bonferroni method. 135

C.1 Comparison of the samples. The sample of users we use is compared GlobalStat's sample and publically available information about the AmIUnique data set, of which our sample is a subset. Percentages of users with various attributes shown. 139

C.2 Configurations of simulated browsing platforms in our main experiment. Font types Mordred, OldLondon, and OldLondonAlternate have been shortened as M, OL, and OLA respectively. ($\star$) Browsing platform # 7 is simulated on a regularly used MacBook and has over 150 different fonts installed. 139

C.3 Observed effects of PETs on attribute values in our main experiment. For brevity, we remove attributes not modified by any PET. 140

C.4 Mask models of PETs generated by our experimental method and used as input to the hybrid method 140

C.5 List of attribute values revealed by baseline Chrome and Firefox on the seven simulated browsing platforms in our main experiment. For clarity of presentation, we replace lists with long values by the length of the list. 141

C.6 Popularity of PETs in our list as of Dec 2017. The popularity of extensions were obtained from the Firefox add-on library and the Chrome extensions webstore. Tor's popularity was obtained from the Tor Metrics webpage. 142

C.7 Attributes in the `amiunique` dataset 143

C.8 Uniqueness metrics for different PETs on samples scaled according to their popularity 144

C.9 Configurations of simulated browsing platforms from our session experiment in Appendix C.2. FF represents Firefox, while C represents Chrome. Font types Mordred, OldLondon, and OldLondonAlternate have been shortened as M, OL, and OLA respectively. (⋆) Browsing platform # 7 is simulated on a regularly used MacBook and has over 150 different fonts installed. 145

C.10 Comparison of purported and observed PET behaviors on attributes from our session experiment in Appendix C.2. □ indicates documented spoofing, ✓ indicates observed modification. 146

C.11 Observed effects of PETs on attribute values from our session experiment in Appendix C.2. . . 147

List of Figures

2.1 Taxonomy of analyses 12

2.2 Ads collected from the first browser instance to visit the Chicago Tribune. The time interval for collection was one minute. The x-axis is time measured in hh:mm. The y-axes ranges over unique ads ordered by the time at which the instance first observed it in the experiment. 21

3.1 Screenshot of Google's Ad Settings webpage in 2015 33

3.2 Experimental setup to carry out significance testing on eight browser agents comparing the effects of two treatments. Each agent is randomly assigned a treatment which specifies what actions to perform on the web. After these actions are complete, they collect measurements which are used for significance testing. 42

3.3 Our experimental setup with training and testing blocks. Measurements from the training blocks are used to build a classifier. The trained classifier is used to compute the test statistic on the measurements from the testing blocks for significance testing. 43

3.4 Screenshot of an ad with the top URL+title for identifying agents that visited webpages associated with substance abuse 52

4.1 Hybrid method for PET evaluation 60

4.2 Analysis Engine in PETInspector . . . 71

4.3 Ranking of PETs by the experimental method. Arrows show the preorder, with PETs at an equivalent order being grouped together. The y-axis shows the number of attributes masked. . 74

5.1 Creating responsive ads on Google AdWords . . . 86

5.2 Ads approved by Google in 2015. Ads in the left (right) column were targeted to women (men). 89

5.3 Ad disapproved by Google in 2017. . . . 90

5.4 Ads approved and served by Google in 2017: truck driver jobs only to men, secretary jobs only to women, and housing disparately. . . . 91

A.1 The second browser instance visiting the Chicago Tribune for Experiment 1 The time interval for collection was one minute. The x-axis is time measured in hh:mm. . . . 121

A.2 For Experiment 1, plots of ads from Instances 1 and 2 in of the six experiments with varying time intervals between reloads. Observe that the pooling behavior appears for the first time in A.2(d), where the pool seems to switch somewhere around the 80th reload. After that the number of these switches keep increasing in successive plots with the reload interval. . . . 122

A.3 For Experiment 1, plots of ads from the Times of India, BBC, and Fox News. The time interval for collection was one minute for each plot. The x-axis is time measured in hh:mm. . . . 123

C.1 The distribution of anonymity set sizes, plotted in log-scale, of fingerprints revealed by Chrome (left) and Firefox (right) PETs against aware-exact-match. Fingerprints are arranged in the decreasing order of the corresponding anonymity set size. The fingerprint with index 0 has the largest anonymity set size for the corresponding PET. Longer tails indicate higher numbers of browsing platforms residing in small anonymity sets. . . . 138

Chapter 1

Introduction

Motivation. Black box software systems increasingly use personal data to make predictions and decisions affecting our daily lives. Such personalization systems make decisions related to credit, employment, insurance premiums, search results, news, and advertisements based on personal data. This leads to concerns about whether these systems respect expectations of fairness and privacy.

There are several examples of discrimination by software systems. Sweeney found race associated with first names affected search ads suggestive of arrest records [135]. Angwin et al. found race was associated with incorrect predictions of recidivism [9]. Other examples include webcam software being unable to detect faces of African-Americans [129], the 'Street Bump' app inadvertently directing Boston's pothole repair crews to affluent neighborhoods [118], and image labeling software incorrectly labeling African-Americans [157]. There are also examples of privacy violations through the inference and use of sensitive personal information. Retail store Target was publicly criticized for inferring the pregnancy status of one of its customers from her shopping history to target advertisements of baby products to her [33]. Google ads, targeted on the basis of visits to websites about sleep apnea, were found be in violation of Canada's privacy laws [110]. Other studies have found evidence of personalization on the basis of visits to websites about depression [148] and indicated sexual preference [59]. These and other examples drove the White House to publish a report urging actors behind such systems to preserve social values [39]. The report highlighted the need for software systems to prevent discrimination and satisfy privacy expectations like *transparency* into how decisions are made and provide *choice* for individuals to rectify inaccurate data and inferences.

Properties. The technical core of a class of fairness and privacy properties is naturally captured by information flow notions. We start with noninterference [48], a standard formalization of information flow. As personalization systems may be probabilistic in nature, we employ a probabilistic version of noninterference. Intuitively, probabilistic noninterference requires the system to behave identically regardless of any sensitive inputs to the system. If a sensitive input causes the system to produce different outputs, then the system is said to have probabilistic interference.

Concrete instances of violations of privacy and fairness properties are captured by probabilistic interference. For example, discrimination can result from probabilistic interference from protected attributes (like race or gender) to unrelated decisions (like recidivism predictions or employment decisions). Similarly, a privacy violation may occur when the system has probabilistic interference from a sensitive attribute (like pregnancy status) to non-private decisions (like targeted advertisements). Justifying why these instances are privacy or fairness violations is outside the scope of this thesis. We assume that instances of fairness and privacy violations are provided by other sources.

We study discrimination, transparency and choice on Google's advertising ecosystem. To study *discrimination*, we search for a flow of information from gender to employment-related ads. The selection of this example was motivated by Title VII of the Civil Rights Act which prohibits publication of employment related ads indicating a preference based on gender. The decision to study transparency and choice was prompted by the 'Notice/Awareness' and 'Choice/Consent' principles in the Fair Information Practice Principles recommended by the United States Federal Trade Commission [27]. To demonstrate a lack of *transparency*, we seek a flow from browsing activities to ads served, in the absence of a flow to Ad Settings.[1] To examine *choice*, we look for a flow of information from user choices on Ad Settings to ads served.

Detection. Traditional methods of information flow detection rely on white box analyses of software programs. However, these methods cannot be applied to black box software systems. We show that probabilistic interference from an input to an output is equivalent to a causal effect of the input on the output. This raises the question *whether it is possible to soundly and scalably detect causal effects in black box software systems*.

We adapt randomized controlled experiments, originally designed to detect causal effects in natural or biological systems, to study software systems. By running carefully designed experiments, we detect causal effects of inputs on outputs from a software system. These experiments form the basis of our methodology to soundly detect information flow.

To detect information flow to the large numbers of outputs a personalization system may produce (e.g., ads produced by the Google ad ecosystem), we extend the methodology to be scalable. To avoid missing

[1] Google's transparency and choice tool: `www.google.com/settings/ads`

subtle effects due to small samples, we modify the experimental setup and the nature of statistical analyses to increase the sample size. Additionally, to avoid having to guess which effect to test for, we use machine learning to automate the selection of the effect. Prior studies of black box personalization systems are unable to detect information flow in a scalable manner since they either approach the problem with non-statistical analyses [33,59,148], make strong assumptions [85], or require considerable insight into what effect to test for [135].

We apply information flow experiments to study fairness and privacy properties in personalization systems. We develop AdFisher as a general framework to detect information flow in web systems and use it to evaluate discrimination, transparency, and choice in Google's advertising ecosystem. We find evidence of gender-based discrimination in employment-related ads and a lack of transparency in Google's transparency tool when serving ads for rehabilitation centers after visits to websites about substance abuse.

Defenses. Given the presence of discrimination and the use of sensitive attributes in personalization systems, we next explore possible defenses for consumers. Many defenses, which require cooperation from personalization systems (e.g., Do Not Track [28], Adnostic [139], Privad [60], etc.), have not seen much adoption. We focus our attention on defenses which function independent of support from personalization systems. One such defense is to prevent these systems from obtaining data about consumers' online activities. This blocking of data collection is made difficult by the presence of a vast network of data aggregators which track online activities using sophisticated techniques (like browser fingerprinting [34]) and sell the data to personalization systems. Privacy Enhancing Technologies (PETs) aim to provide users with the ability to prevent such data collection. Given the numerous PETs available, it is unclear which PET consumers should adopt. Since many PETs are black boxes it is difficult to evaluate how effectively they hinder data collection. We find that the problem of evaluating PETs also reduces to the question of *detecting causal effects in black box software systems*. In this case, we are interested in measuring the causal effect of a PET on the ability of data aggregators to collect data about users' browsing activities.

Prior studies on PET evaluations adopt purely experimental methods (e.g., [81,91,108,120]). Experimental methods have the advantage of precision and control in discovering causal effects and can be applied to new PETs that currently lack a user base. However, this approach is not able to evaluate PETs in the real-world as the experimental units may be unlike real-world browsers. Observational methods, such as those used to study browser fingerprinting [34]), have the advantage of scale and sampling from real-world browsers, but they are unable to draw causal conclusions unless the observational data satisfy strong assumptions.

We propose a novel hybrid method combining experimental and observational approaches, which offers the advantages of both. We use experiments to find causal effects of a PET on various fingerprintable attributes to create a model of the PET. We then apply this model on a pre-existing data set of real-world browser fingerprints to produce a counterfactual PET-modified data set. By comparing the abilities of a data aggregator in tracking the original fingerprints with that of the PET-modified fingerprints, we evaluate the effectiveness of the PET. We apply our evaluations to 26 different PETs, 15 of which explicit claim to protect against fingerprinting. We find that 13 of the 15 anti-fingerprinting PETs do not provide much additional protection than using no PET at all. We find the Tor Browser Bundle to be the most effective among the ones we evaluate.

The thesis of this work is that *carefully designed methods and tools that soundly and scalably discover causal effects in black box software systems are useful in evaluating personalization systems and anti-tracking tools to understand how well they protect fairness and privacy.*

As an alternative defense against discrimination, this thesis also explores legal liability for ad platforms like Google for serving discriminatory ads. In particular, we consider the application of Title VII of the Civil Rights Act and the Fair Housing Act, which prohibit expressing a preference based on sex for employment and housing related advertisements, and their interaction with the Communications Decency Act, which provides broad immunity to online intermediaries like advertising platforms.

We find that the language of Title VII makes it applicable only to specific categories of entities (like employer or employment agency) and may not apply to an advertising platform like Google. The Fair Housing Act, however, does not have the same restrictions and may consider Google as a covered entity. This is where another law, Section 230 of the Communications Decency Act, comes into play. It provides immunity to online intermediaries for illegal content created or developed by a third party. Section 230 provides Google with immunity if discrimination came about solely from the advertiser's inputs. However, the law does not provide immunity if the online intermediary "materially contributes" to the illegality of the content. We argue that the targeting of ads by an ad platform towards or away from protected classes without explicit instructions from the advertiser constitutes material contribution. As a result, the immunity should not extend to the ad platform in such scenarios. Policy makers may want to address gaps in the language of Title VII so it also applies to online intermediaries in this age of online advertising. The threat of legal liability may incentivize companies behind personalization systems to take precautionary measures to avoid discrimination.

Structure of Dissertation. The remainder of the dissertation is structured as follows:

- In Chapter 2, we develop a methodology for detecting information flow in black box systems and carry out proof-of-concept experiments to demonstrate their applicability.

- In Chapter 3, we scale up the information flow experiments methodology and apply it to study discrimination, transparency, and choice on the Google advertising ecosystem.
- In Chapter 4, we study how to evaluate publicly available privacy enhancing technologies which aim to protect users from tracking via fingerprinting.
- In Chapter 5, we explore potential legal liability for an advertising platform like Google for delivering ads in a discriminatory manner.
- In Chapter 6, we discuss some future directions and conclusions.

Remarks. Parts of this dissertation appear in the proceedings of the Computer Security Foundations Symposium (2015) as *'A Methodology for Information Flow Experiments'* (Chapter 2), the Proceedings of Privacy Enhancing Technologies (2015) as *'Automated Experiments on Ad Privacy Settings: A Tale of Opacity, Choice, and Discrimination'* (Chapter 3), and the proceedings of the Conference on Fairness, Accountability, and Transparency (2018) as *'Discrimination in Online Advertising: A Multidisciplinary Inquiry'* (Chapter 5).

The legal analyses of Chapter 5 were carried out in collaboration with law scholars Jael Makagon and Deirdre K. Mulligan.

Chapter 2

A Methodology for Information Flow Experiments

2.1 Introduction

In this chapter, we develop a formal methodology for detecting information flow in black box systems, We first prove a connection between information flow and causality. With this connection, the problem of demonstrating information flow reduces to a problem of demonstrating causal effect. We adapt Fisher's randomized controlled experiments to detect causal effects and apply a rigorous statistical analysis for such experiments. We demonstrate the utility of our methods by running some information flow experiments to detect personalization of Google ads based on browsing activity.

2.1.1 Web Data Usage Detection

Suppose you are shown a car ad by Google while reading an article on a news webpage. You might wonder whether the ad appears because you visited a car dealer website earlier in the day. That is, you would like to know whether information flows from the car dealer's website (to the ad network) to the news webpage.

More generally, concerns about privacy (e.g., [141]) have led to much interest in determining whether ad networks, such as Google's DoubleClick and the Yahoo Ad Exchange, use certain types of information [11, 12, 37, 59, 85, 88, 135, 148]. We call this problem *web data usage detection* (WDUD). In this Chapter, we show how to conduct experiments in a systematic way that can help answer this and other kinds of privacy-related questions.

While WDUD studies are, in essence, attempting to track the flow of information from inputs to some system to outputs of it, they differ from traditional information flow analyses (IFAs). The traditional motivation for IFA, designing secure programs, leads to viewing the analyst as verifying that a system under his control protects information sensitive to the operator of the system. Thus, the problems studied and analyses proposed tend to presume that the analyst has access to the program running the system in question or total control over its inputs and environment. (See [123] for a survey.)

In the setting of WDUD, the analyzed system can be adversarial toward the person studying the system. The analyst may be aligned with (or even equal to) a *data subject*, an entity whose information is collected by the system. In this setting, the analyst has no access to the program running the system in question, little control over its inputs, and a limited view of its behavior. Thus, the analyst lacks the abilities presupposed by traditional IFAs. To understand the WDUD problem as an instance of IFA requires a fresh perspective on IFA.

The original motivation underlying much of IFA research also obscures its connection to other areas of research. For example, copyright protection [134,146], traitor tracing [23], data leak detection [113,130,149], and the detection of plagiarism [100] are all in essence information flow analyses in which the analyst has limited access to the system in question (often a person). However, to keep the presentation clear, we focus on WDUD, leaving a discussion of related areas to future work.

2.1.2 Chapter Contributions

We develop a formal methodology for conducting IFA for black-box information flow problems, and for WDUD in particular. The overarching contribution of this work is relating IFA in these nontraditional settings to experiments designed to determine causation. We show that the ability of the analyst to control some inputs enables *information flow experiments* that manipulate the system in question to discover the use of information without a white-box model of the system. We present an easy-to-apply, statistically rigorous methodology for information flow experiments that future studies on WDUD and other IFAs for black boxes may use to draw statistically sound conclusions.

Our methodology is supported by a chain of contributions that follows the chapter's outline:

§ 2.2 a systematization of black-box IFA problems
§ 2.3 a proof of a connection between IFA and causality
§ 2.4 an experimental design leveraging this connection
§ 2.5 a rigorous statistical analysis for such experiments

In particular, while the link has been subject to prior comment [93, 101], we believe we are the first to formally prove a connection between standard notions of information flow and causation. We are also the first to provide a method for conducting WDUD studies that comes with a methodology showing that a positive result entails a flow of information with a quantified certainty.

These contributions are each necessary for creating a chain of sound reasoning from intuition about vague problems to rigorous quantified results. This chain of reasoning provides a systematic, unifying view of these problems, which leads to a concrete methodology based on well studied scientific methods. While the notion of experimental science is hardly new, our careful justification provides guidance on the choices involved in actually conducting an information flow experiment.

The systematization of experimental approaches to security, privacy, and accountability is becoming increasingly important as technological trends (e.g., Cloud and Web services) result in analysts and auditors having limited access to and control over systems whose properties they are expected to study. This chapter provides a useful starting point towards such a systematization by providing a common model and a shared vocabulary of concepts that places problems of security and privacy into the context of causality, experimentation, and statistical analysis.

2.1.3 Chapter Overview

We explore, both empirically and theoretically, how to conduct IFA over black-box systems while avoiding unjustified assumptions. First, in Section 2.2, we build upon traditional IFA by starting with noninterference [48], a standard formalization of information flow. We identify the limited abilities of the analyst in these problems and cast WDUD as a form of analysis between the extremes of white-box program analysis and black-box monitoring. In doing so, we shift IFA from its traditional context of program analysis using white-box models of software to the new context of investigating black-box systems that hide much of their behavior and operate in uncontrolled environments. We thereby highlight an interesting flavor of black-box IFA that lacks prior formal study.

In particular, we focus on WDUD as it is the least understood nontraditional IFA problem. We formalize it in terms of noninterference showing its relationship to IFA. We prove that sound information flow detection is impossible in this setting (Theorem 1).

Motivated by this impossibility result, we look for an alternative statistical approach. To do so, we build upon research on causality [114], strengthening the connection between the two research areas. In Section 2.3, we prove that a system has interference from a high-level user H to a low-level user L in the sense of IFA if and only if inputs of H can have a causal effect on the outputs of L while the other inputs to the system remain fixed (Theorem 3). This connection allows us to appeal to inductive methods employed in experimental science to study IFA. Such methods provide precisely what we need to make high-assurance statistical claims about flows despite our impossibility result. We leverage this observation to approach WDUD with information flow experiments.

Section 2.4 discusses the design of information flow experiments. We show a correspondence between the features of WDUD and the features of a scientific study (Table 2.1). We discuss the limitations and abilities of experiments to find interference. In particular, we explain the difficulty of finding that a single system has interference. We also identify the ability to find that a system and its environment acting together has interference. For example, we find that while we cannot claim Google itself has interference, we can determine that Google and it's ad ecosystem does.

In Section 2.5, we review significance testing as a systematic method of quantifying the degree of certainty that an information flow experiment has observed interference. We conduct pilot studies to explore what assumptions, and therefore statistical analyses, are appropriate for WDUD. We identify permutation testing [49], a method of significance testing, as particularly well suited. In essence, it uses randomization, similarly to security algorithms, to defeat adversaries, making it appropriate for a security setting.

Section 2.6 compares our method to those found in prior WDUD studies. In Section 2.7, we empirically benchmark our interpretations of some of their approaches with our own WDUD study. This WDUD study is the first to come with a methodology showing its relationship to IFA.

We then provide practical suggestions, summarized in Section 2.8, for systematically conducting future WDUD studies. We end by discussing directions for new research that further strengthens the connection between information flow and causality and applies it to other security problems.

Throughout this work, we present our own experiments to illustrate the abstract concepts we present. These results may also be of independent interest to the reader.

2.1.4 Prior Work

Ruthruff, Elbaum, and Rothermel note the usefulness of experiments for program analysis [122]. Whereas our work focuses on problems where traditional white-box analyses are impossible, their work examines experiments in the more traditional setting where the analyst has control over the system in question. Furthermore, we develop a formalism relating informal flow and causality, provide proofs, and present a statistical analysis.

While we could not find any prior articulation of the formal correspondence between informal flow and causality (our Theorem 3), we are not the first to note such a connection. McLean [93] and Mowbray [101] each proposed a definition of information flow that uses the lack of a causal connection to rule out security violations even if there is a flow of information from the point of view of information theory. Sewell and Vitek provide a "causal type system" for reasoning about information flows in a process calculus [128]. We differ from these works by showing an equivalence between a standard notion of information flow, noninterference [48], and a standard notation of causality, Pearl's [114], rather than using a notion of causality to adjust an information theoretic notion of information flow. Furthermore, Mowbray's formalism requires white-box access to the system while McLean's only considers temporal ordering as a source of causal knowledge. More importantly, they use causality to handle problematic edge cases in their formalisms whereas we reduce interference to causality so that we may apply standard methods from experimental science to IFA.

Our identification of WDUD as an interesting problem for IFA was inspired by prior WDUD studies. Sweeney uses a method similar to ours to study how search ads are affected by search terms [135]. Our work provides a formal justification of her method in terms of information flows. Other prior studies either approach the problem with non-statistical analyses [11, 59, 88, 148] or make assumptions that our experiments show unlikely to hold in the setting we study [12, 37, 85]. Section 2.6 details these works.

We draw on works from experimental design and statistics, whose discussion we defer until the point of use.

2.2 Information Flow Analysis

In this section, we discuss prior work on information flow analysis starting with noninterference, a formalization of information flows. We next discuss the analyses used in prior work to determine whether a flow of information exists. We present them systematically by the capabilities they require of the analyst. We end by discussing the capabilities of the analyst in our motivating applications, and WDUD in particular, how prior analyses are inappropriate given these capabilities, and the inherent limitations of these capabilities.

2.2.1 Noninterference

Goguen and Meseguer introduced *noninterference* to formalize when a sensitive input to a system with multiple users is protected from untrusted users of that system [48]. Intuitively, noninterference requires the system to behave identically from the perspective of untrusted users regardless of any sensitive inputs to the system.

As did they, we will define noninterference in terms of a synchronous finite-state Moore machine. The inputs that the system accepts are tuples where each component represents the input received on a different input channel. Similarly, our outputs are tuples representing the output sent on each output channel. For simplicity, we will assume that the machine has only two input channels and two output channels, but all results generalize to any finite number of channels.

We partition the four channels into H and L with each containing one input and one output channel. Typically, H corresponds to all channels to and from high-level users, and L to all channels to or from low-level users. The high-level information might be private or sensitive information that should not be mixed with public information, denoted by L. In the area of taint analysis, the roles are reversed in that the tainted information is untrusted and should not be mixed with trusted information on the trusted channel. However, either way, the goal is the same: keep information on the input channel of H from reaching the output channel of L.

We will often have a single user using channels of both sets since we are concerned with not only to whom information flows but also under what contexts. To this end, we interpret *channel* rather broadly to include virtual channels created by multiplexing, such as a field of an HTML form or the ad container of a web page. We also allow each channel's input/output to be a null message indicating no new input/output.

A system q consumes a sequence $\vec{\imath}$ of input pairs where each pair contains an input for the high and the low input channels. We write $q(\vec{\imath})$ for the output sequence $\vec{o}$ that q would produce upon receiving $\vec{\imath}$ as input where output sequences are defined as a sequence of pairs of high and low outputs.

For an input sequence $\vec{\imath}$, let $\lfloor \vec{\imath} {\downarrow} L \rfloor$ denote the sequence of low-level inputs that results from removing the high-level inputs from each pair of $\vec{\imath}$. That is, it "purges" all high-level inputs. We define $\lfloor \vec{o} {\downarrow} L \rfloor$ similarly for output sequences.

Definition 1 (Noninterference)**.** *A system q has* noninterference *from L to H iff for all input sequences $\vec{\imath}_1$ and $\vec{\imath}_2$,*

$$\lfloor \vec{\imath}_1 {\downarrow} L \rfloor = \lfloor \vec{\imath}_2 {\downarrow} L \rfloor \textit{ implies } \lfloor q(\vec{\imath}_1) {\downarrow} L \rfloor = \lfloor q(\vec{\imath}_2) {\downarrow} L \rfloor$$

Intuitively, if inputs only differ in high-level inputs, then the system will provide the same low-level outputs.

To handle systems with probabilistic transitions, we will employ a probabilistic version of noninterference similar to the previously defined *P-restrictiveness* [55] and *probabilistic nondeduciblity on strategies* [56]. To define it, we let $Q(\vec{\imath})$ denote a probability distribution over output sequences given the input $\vec{\imath}$, a concept that can be made formal given the probabilistic transitions of the machine [56]. We define $\lfloor Q(\vec{\imath}) {\downarrow} L \rfloor$ to be the distribution μ over sequences $\vec{\ell}$ of low-level outputs such that $\mu(\vec{\ell}) = \sum_{\vec{o} \text{ s.t. } \lfloor \vec{o} \downarrow L \rfloor = \vec{\ell}} Q(\vec{\imath})(\vec{o})$. Probabilistic Noninterference compares such distributions for equality.

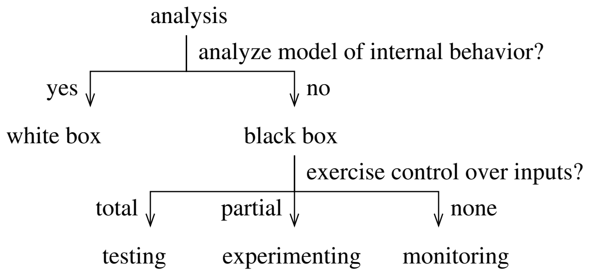


Figure 2.1: Taxonomy of analyses

Definition 2 (Probabilistic Noninterference)**.** *A system Q has* probabilistic noninterference *from L to H iff for all input sequences $\vec{\imath}_1$ and $\vec{\imath}_2$,*

$$\lfloor \vec{\imath}_1 {\downarrow} L \rfloor = \lfloor \vec{\imath}_2 {\downarrow} L \rfloor \textit{ implies } \lfloor Q(\vec{\imath}_1) {\downarrow} L \rfloor = \lfloor Q(\vec{\imath}_2) {\downarrow} L \rfloor$$

2.2.2 Analysis

Information flow analysis (IFA) is a set of techniques to determine whether a system has noninterference (or similar properties) for interesting sets H and L. Examples include analyses employing type systems [123, 144], model checking of code [14], or dynamic approaches that instrument the code running the system to track values carrying sensitive information (e.g., [92, 107, 142, 143]).

The above methods are inappropriate for WDUD since they require *white-box* access to the program. That is, the analyst must be able to study and/or modify the code. In our applications, the analyst must treat the program as a *black box*. That is, the analyst can only study the I/O behavior of the program and not its internal structure. Black-box analyses vary based on how much access they require to the system in question. Figure 2.1 shows a taxonomy of analyses.

Numerous black-box analyses for detecting information flows operate by running the program rather than analyzing its code [22, 31, 70, 84, 152]. They run the program multiple times with varying inputs to detect changes in output that imply interference. However, these black-box analyses continue to require access to the internal structure of the program even if they do not analyze that structure. For example, the analysis of Yumerefendi et al. requires the binary of a program to copy it into a virtual machine for producing I/O traces [152]. In theory, such black-box analyses could be modified to not require any access to code by completely controlling the environment in which the program executes. To do so, the analyst would run a single copy of the program and reset its environment to simulate having multiple copies of the system. We call this form of black-box analysis, with total control over the system, *testing* as it is the setting typical to software testing.

Testing will not work for our applications. For example, in the setting of WDUD, the analyst cannot reset and run the program multiple times since the analyst has only limited interactions with the program over a network. Thus, it cannot force the program into the same initial environment to reset it. Furthermore, unlike a program, Google's ad *system* is stateful and, thus, modifying its environment alone would be insufficient to reset it. In this setting, the analyst must analyze the *system* as it runs, not a program whose environment the analyst can change at will.

At the opposite extreme of black-box analysis is *monitoring*, which passively observes the execution of a system. While some monitors are too powerful by being able to observe the internal state of the running system (e.g. [127]), others match our needs in that the analyst only has access to a subset of the program's outputs (e.g., [46]). However, all monitors are too weak since they cannot provide inputs to the system as our application analysts can. We need a form of black-box analysis between the extremes of testing and monitoring.

Thus, we find no prior work on IFA that corresponds to the capabilities of the analyst in WDUD or our other motivating applications.

2.2.3 Information Flow Experiments

Unlike the primary motivation of traditional IFA, developing programs with Mandatory Access Controls (MAC), our motivating examples involve situations in which the analyst and the system in question are not aligned. Thus, the information available to the analyst is much more limited than in the traditional security setting. In particular, the analyst

1. has no model of or access to the program running the system,
2. cannot observe or directly control the internal states of the system,
3. has limited control over and knowledge of the environment of the system,
4. can observe a subset of the system's outputs, and
5. has control over a subset of the inputs to the system.

We will call performing IFA in this setting *experimenting*. Experiments are an interactive extension of a limited form of execution monitoring that allows analyst inputs but limits the analyst to only observing a subset of system I/O.

Prior work shows that no monitor can detect information flows [94, 127, 145]. We argue that experiments, with their additional ability to control some inputs to the system, do not improve upon this situation. In particular, we prove that no non-degenerate experiment can be sound for interference or for noninterference, even on deterministic systems. (Although, we will later show that experiments do enable statistical analyses with probabilistic soundness properties.)

Experiment Model. We model an experiment as a pair $\langle \vec{\imath}, d\rangle$ where $\vec{\imath}$ is an input sequence and d is a *decision function* from the set of output sequences to $\{\mathsf{ni}, ?, \mathsf{in}\}$. $\vec{\imath}$ represents the sequence of inputs that the analyst supplies to the system in question and d represents how the analyst goes from the sequence of resulting outputs to either the conclusion that the system q has interference (in), has noninterference (ni), or that she does not know (?), all for a fixed H and L. The result of the experiment $\langle \vec{i}, d\rangle$ on the deterministic system q is $d(q(\vec{\imath}))$. We allow the analyst just one I/O sequence since the analyst cannot restart the system, which would include resetting its hard drives, clocks, etc. to their initial states. The analyst can embed into its single sequence multiple subsequences each corresponding to a run of a program on the system.

Adversary Model. The system in question q might be under the control of an adversary trying to trick the analyst as to whether the system has interference. We model the adversary as being able to select any automaton for the system q. In essence, the following theorems show that for any experiment, the adversary can select a system that fools the analyst.

To prove the unsoundness of black-box experiments for interference, we consider an arbitrary system q for which an experiment returns a positive result indicating interference. In our setting, the experiment must base its decision solely upon its interactions with the system. Thus, it will return the same positive result for a system q_{N} that always produces the same outputs as q did irrespective of its inputs. Since q_{N} always produces these outputs, it has noninterference making the positive result false.

Theorem 1 (Unsoundness for Interference)**.** *For all experiments $\langle \vec{\imath}, d\rangle$, if there exists a system q such that $d(q(\vec{\imath})) = \mathsf{in}$, then there exists a system q_{N} with noninterference from H to L such that $d(q_{\mathsf{N}}(\vec{\imath})) = \mathsf{in}$.*

The argument for noninterference is symmetric, but requires that interference is possible given the system's input and output space. That is, the system must have at least two high inputs and two low outputs.

Theorem 2 (Unsoundness for Noninterference)**.** *If H has two inputs and L has two outputs, then for all experiments $\langle \vec{\imath}, d\rangle$, if there exists a system q such that $d(q(\vec{\imath})) = \mathsf{ni}$, then there exists a system q_{I} with interference from H to L such that $d(q_{\mathrm{I}}(\vec{\imath})) = \mathsf{ni}$.*

Appendix A.2 contains proofs for these theorems. Note that these theorems hold even if the analyst can observe every input in H and L making the above shift of focus to the composite system of Google operating in its environment unsuccessful. However, as we will later see, we can probabilistically handle the lack of total internal control of the composite system using statistical techniques. Since we can never be sure whether we have started a particular sequence of inputs from the same initial state as another sequence, we use many instances of each sequence instead of one for each. Intuitively, if the outputs for one group of inputs are consistently different from outputs for the other group of inputs, then it is likely that the difference is introduced by the difference between the groups instead of from the initial states differing. We formalize this idea to present a probabilistically sound method of detecting interference. We leave detecting noninterference to future work.

2.3 Causality

In this section, we discuss a formal notion of causality motivated by the studies of the natural sciences. We then prove that noninterference corresponds to a lack of a causal effect. This result allows us to repose WDUD as a problem of statistical inference from experimental data using causal reasoning.

2.3.1 Background

Let us start with a simple example. A scientist might like to determine whether a chemical causes cancer in mice. More formally, she is interested in whether the value of the *experimental factor* X, recording whether the mouse ingests the chemical, causes an effect to a *response variable* Y, an indicator of mouse cancer, holding all other factors (possible causes) constant.

Pearl [114] provides a formalization of *effect* using *structural equation models* (SEMs), a formalism widely used in the sciences (e.g., [64]). A probabilistic SEM $M = \langle \mathcal{V}_{\mathsf{en}}, \mathcal{V}_{\mathsf{ex}}, \mathcal{E}, \mathcal{P} \rangle$ includes a set of *variables* partitioned into *endogenous* (or dependent) variables $\mathcal{V}_{\mathsf{en}}$ and *exogenous* (or independent) variables $\mathcal{V}_{\mathsf{ex}}$. M also includes in $\mathcal{E}$, for each endogenous variable V, a *structural equation* $V := F_V(\vec{V})$ where $\vec{V}$ is a list of other variables other than V and F_V is a possibly randomized function. A structural equation is directional like variable assignments in programming languages. Each exogenous variable is defined by a probability distribution given by $\mathcal{P}$. Thus, every variable is a random variable defined in terms of a probability distribution or a function of them.

Let M be an SEM, X be an endogenous variable of M, and x be a value that X can take on. Pearl defines the *sub-model* $M[X{:=}x]$ to be the SEM that results from replacing the equation $X := F_X(\vec{V})$ in $\mathcal{E}$ with the equation $X := x$. The sub-model $M[X{:=}x]$ shows the *effect* of setting X to x. Let Y be an endogenous variable called the *response variable*. We define *effect* in a manner similar to Pearl [114].

Definition 3 (Effect)**.** *The experimental factor X has an* effect *on Y given $Z := z$ iff there exists x_1 and x_2 such that the probability distribution of Y in $M[X{:=}x_1][Z{:=}z]$ is not equal to its distribution in $M[X{:=}x_2][Z{:=}z]$.*

Intuitively, there is an effect if $F_Y(x_1, \vec{V}) \neq F_Y(x_2, \vec{V})$ where $\vec{V}$ are the random variables other than X and Y.

2.3.2 The Relationship of Interference and Causality

Intuitively, interference is an effect from a high-level input to a low-level output. Noninterference corresponds to lack of an effect, which Pearl calls *causal irrelevance* [114]. We can make this connection formal by providing a conversion from a probabilistic system to an SEM.

Given a probabilistic Moore Machine Q, we define a SEM M_Q. Intuitively, it contains endogenous variables for each input and output and exogenous variables for each user. In more detail, for each time t, M_Q contains the endogenous variables HI_t. It also contains HO_t for high outputs, LI_t for low inputs, and LO_t for low outputs, all at the time t. M_Q also has exogenous variables HU_t and LU_t that represent the behavior of high and low users of the system at time t.

The behavior of Q provides functions $F_{\mathsf{lo},t}$ defining the low output at time t in terms of the previous and current inputs. The output may depend upon previous inputs via a variable S_t representing the state of the system. Similar functions exist for the other variables. For example, the function $F_{\mathsf{s},t}$ for updating the state is determined by the transition function of Q. (Details may be found in Appendix A.4.1.)

The following lemma shows that Q and M_Q are equivalent. To state it, we use $\vec{V}^t$ to denote the vector holding those variables V_t with an index of t or less (in order). We let I^t represent a similar vector of input variables combining HI^t and LI^t. We use $\vec{V}^t = \vec{v}$ as shorthand for $\bigwedge_{j=1}^{t} \vec{V}[j] = \vec{v}[j]$.

Lemma 1. *For all Q, t, $\vec{\imath}$, and $\vec{\mathsf{lo}}$ of lengths t and $t+1$, respectively, $\mathcal{P}(\vec{\mathsf{LO}}^{t+1}{=}\vec{\mathsf{lo}} \mid \mathsf{do}(\vec{I}^t{:=}\vec{\imath})) = \lfloor Q(\vec{\imath}){\downarrow}L \rfloor(\vec{\mathsf{lo}})$.*

The key theorem follows from Lemma 1 and the properties of SEMs and interference:

Theorem 3. *Q has probabilistic interference iff there exists low inputs $\vec{\mathsf{li}}$ of length t such that $\vec{\mathsf{HI}}^t$ has an effect on $\vec{\mathsf{LO}}^t$ given $\vec{\mathsf{LI}}^t := \vec{\mathsf{li}}$ in M_Q.*

Notice that Theorem 3 requires that the low-level inputs to the system in question be fixed to a set value $\vec{\mathsf{li}}$. This requirement is a reflection of how noninterference only requires that low-level outputs be equal when low-level inputs are equal (Definition 1). The proofs are in Appendices A.4.3 and A.4.4.

General Terms	Chemical Study	Behavioral Marketing
natural process	cancer	marketing
population of units	mice	browser instances
experimental factor	diet	visitor behavior
treatments	chemical or placebo	behavioral profiles
constant factors	water allowance	IP address etc.
noise factors	age, weight, etc.	other users, advertisers
response variables	tumor count	sequences of ads
effect	carcinogenic	use of data

Table 2.1: General Terminology and Two Instances of Experimental Science. In the chemical study, a scientist studies whether a chemical is carcinogenic when added to the diet of mice. In a behavioral marketing study, the scientist studies whether changes in visitor behavior causes changes in ads.

2.4 Experimental Design

Having reduced the problem of information flow experiments to that of checking for effects, we can employ the checking method often used in empirical sciences, randomized controlled experiments. However, doing so requires mapping the features of WDUD and other black-box IFA problems into the standard terms of experimental design. Furthermore, it requires scoping the experiment to be within the limited abilities of the analyst. In particular, we must respect the requirement of Theorem 3 that the analyst be able to fix all low-level inputs. We discuss each of these issues before turning to the actual running of the experiment.

2.4.1 The Setup of Experiments

A randomized controlled experiment randomly assigns each *experimental unit*, such as a mouse, to either a *control* or an *experimental* treatment. The treatment determines the value of the unit's *experimental factor*, which maps to the changed variable X in Definition 3. The experimenter holds other factors under her control constant to isolate the effect of the treatment. These factors map to Z in Definition 3. The experimenter measures a *response*, some feature, of each unit. The experimenter attempts to determine whether the treatments have an effect on the measured responses.

For example, consider a WDUD study to determine whether a pattern of behavior, or profile, affects the ads that Google shows to a user. Table 2.1 summarizes how to view it and an archetypal cancer study as experiments.

In the case of WDUD, the natural experimental unit might appear to be Google. However, since a randomized controlled experiment requires multiple experimental units and there is just one Google, we must select some subsets of interactions with Google as the experimental units. Since one of the major goals of WDUD is to determine the nature of Google's behavioral tracking of people, interactions with Google at the granularity of people could be an appropriate experimental unit. However, since we desire automated studies, we substitute separate automated browser instances for actual people. In particular, we can use multiple browser instances with separate caches and cookies to simulate multiple users interacting with the web tracker. We can apply treatments to browsers by having them controlled by different scripts that automate different behaviors.

The treatments are various behavioral profiles that the analyst is interested in comparing. The constant factors can include anything the analyst can control: the IP address, the browser used, the time of day, etc. The response may be the ads shown to the simulated browser.

2.4.2 Scoping the System

Properly scoping an the experiment for WDUD is particularly important. Suppose in the above example, the system in question is Google. Since the profile of the user is of interest, it dictates the high-level inputs. Since every input must be either low-level or high-level, all inputs not determined by the profile are low-level. These low-level inputs include some that the analyst cannot observe or control, such as inputs from advertisers to Google. However, Theorem 3 requires that the all the low-level variables remain fixed. That is, to use Theorem 3, the analyst must select the system and its inputs so that she can ensure that the low-level inputs are fixed.

The analyst must shrink the set of low-level inputs to just those that she can fix. One means of achieving this goal is to consider more inputs high-level, but if the inputs converted to be high-level are already known to determine the ads shown (such as inputs from advertisers), then the analysis would be of little interest. Another means would be to alter Google so that it no longer accepts such inputs, but the analyst does not have such control over Google. However, the analyst does have control over which system she studies. Rather than study Google in isolation, she could study the composite system of Google and the advertisers operating in parallel. By doing so, she converts the uncontrolled low-level inputs to Google from the advertisers into internal messages of the composite system, which are irrelevant to whether interference occurs.

The practical consequences of these limitations for WDUD is that we cannot determine that Google has interference on its own. Rather, we can only determine that Google operating in its environment has interference. That is, we can determine that the composite system consisting of Google and the other systems making up the ad ecosystem has interference.

This limitation means that we cannot explain how observed interference occurs. Upon seeing interference, one explanation is that Google directly used the information in question to select ads. However, it is also possible that Google shared the information with an advertiser that used the information to change its bids, which, in turn, caused Google to change its ads. If the output to the advertiser is low-level, then Google itself does not have interference in the second case.

Nevertheless, we know that some part of the ad ecosystem used the information. This finding can be useful in its own right if one is interested in the complete process of how ads are selected. It could also justify a white-box investigation by either internal auditors or external regulators who may compel internal access.

Lastly, note this scoping does not enable sound nor complete analyses of the composite system: Theorems 1 and 2 continue to apply since they do not require non-empty sets of low-level inputs. The analyst's continued inability to observe the internal state of the system means that the analyst must still employ statistical analyses.

2.4.3 Running the Experiment

With the system properly scoped, we run such a randomized controlled experiment as follows:

1. Assign each browser either an experimental or control profile at random.

2. Each browser instance simulates those profiles by interacting with webpages.

3. Each browser instance collects ads from (possibly other) webpages.

4. Compare the collected ads from browsers with one profile to browsers with the other profile.

In more detail, the analyst prepares a vector $\vec{x}$ with a length equal to the number of units that hold treatment values. Typically, half will be control treatments and half experimental treatments. She randomly assigns each experimental unit k to an index i_k of $\vec{x}$ so that no unit is assigned the same index. For each k, she then applies the treatment in the i_kth slot of $\vec{x}$ to the unit k, which in our setting implies providing inputs corresponding to a profile. Units assigned the same treatment form a *group*.

Groups may vary due to *noise factors*, variations among the experimental units other than those from the application of treatments. Proper randomization over larger sample sizes makes negligible the probability that the groups vary in a systematic manner. If the analyst also ensures that no other systematic differences are introduced to the groups after the application of the treatment, the units will not systematically differ between the groups under the *null hypothesis* that the treatment has no effect. Thus, any difference in responses that consistently shows up in one group but not the other can only be explained by chance under the null hypothesis. If given the sample size, this chance is small, then the analyst can reject the null hypothesis as unlikely, providing probabilistic evidence of a causal relationship, which we quantify in the next section.

2.5 Statistical Analysis

To quantify the probability that the collected responses could appear to show a flow of information due to a chance occurrence, we use *significance testing* [44].

A *statistical test* of the data provides a *p-value*, the probability of seeing results at least as extreme as the observed data under the assumption that the null hypothesis is true. A small p-value implies that the data is unlikely under the null hypothesis. Typically, scientists are comfortable rejecting the null hypothesis if the p-value is below a threshold of 0.05 or 0.01 depending on the field. Rejecting the null hypothesis makes the alternative hypothesis that there is an effect more plausible. In our case, using significance testing requires selecting a test of independence. We discuss the process of selecting one and detail the one we have selected, permutation testing.

2.5.1 Selecting a Test of Independence and Pilot Studies

Some tests of independence require assumptions about the system in question. These assumptions enable powerful statistical techniques, which in some cases allow smaller sample sizes or more detailed characterizations of a research finding.

Parametric tests assume that the behavior of the system in question is drawn from some known family of distributions with a small number of unknown parameters.

Another common assumption is that, under the null hypothesis, the responses of the experimental units are independent of one another and identically distributed (i.i.d.).

Some statistical analyses require that giving or withholding a treatment from one unit will not have an effect upon the other units (e.g., [29, p. 19]), that is, the absence of *cross-unit effects*.

Given our understanding of how ad networks operate, these assumptions appear suspicious. The complex behavior of ad networks makes selecting a family of distributions to model one difficult. The fact that budgets control the number of ad impressions creates the possibility that one browser instance receiving an ad might decrease the probability that another receives it, invalidating the i.i.d. assumption. Any choice of experimental unit other than all of Google, which leads to a sample size of one, will possibly exhibit cross-unit effects by virtue of units being multiplexed onto a single system.

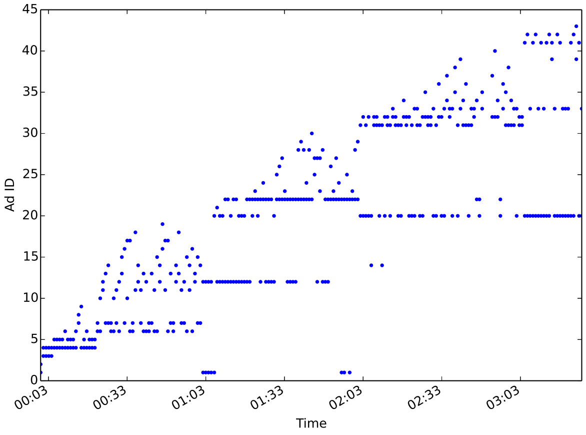


Figure 2.2: Ads collected from the first browser instance to visit the Chicago Tribune. The time interval for collection was one minute. The x-axis is time measured in hh:mm. The y-axes ranges over unique ads ordered by the time at which the instance first observed it in the experiment.

To empirically explore how reasonable these assumptions are in our setting, we conducted two pilot studies, which show them difficult to defend in our setting. Appendix A.5 contains additional details about these and our other experiments.

Experiment 1. *We collected ads served by Google on a third-party website to understand how they vary over time. Following Balebako et al.'s study [11], we used the Breaking News page of the Chicago Tribune (`http://www.chicagotribune.com/news/local/breaking/`).*

To collect ads, we simultaneously started two browser instances, and collected the ads served by Google on the webpage. Each instance reloaded the web page 200 times, with a one minute interval between successive reloads. □

Figure 2.2 shows a temporal plot of the ads served to one of these instances. (Figure A.1 of the Appendix A.5.1 shows the other.) The plots suggest that each instance received certain kinds of ads for a period of time, before being switched to receiving a different kind, which implies that *ads are not identically distributed* across time. This pattern held using other intervals for reloads. (Figure A.2 in Appendix A.5.1).

One explanation for this behavior is that Google associated users with various ad pools switching users from pool to pool over time. While hierarchical families of parametric models could capture this behavior, we are not comfortable making such an assumption and the resulting models would be more complex than those typically used in parametric tests. Thus, *parametric tests would employ models of low confidence.*

Our results do not mean that one could not reverse engineer enough of Google to find an appropriate model. However, they suggest that such reserve engineering would be difficult. Furthermore, it runs against the spirit of performing black-box information flow analysis.

Shortly after we conducted these experiments, the Chicago Tribune stopped hosting text ads from Google. Thus, when we later wanted to replicate the study, we instead looked at the Times of India, the BBC, and Fox News. The ads continued to appear to violate the i.i.d. assumption with some ads being shown over and over again in streaks (Figure A.3 in Appendix A.5.1). However, the binning behavior was gone. This difference in behavior suggests that any success at reverse engineering Google may be specific to a webpage or short lived as Google changes its behavior.

We carried out this and all other experiments using Python bindings for Selenium WebDriver, which is a browser automation framework. A test browser instance launched by Selenium uses a temporary folder that can be accessed only by the process creating it. So, two browser instances launched by different processes do not share cookies, cache, or other browsing data. All our tests were carried out with the Firefox browser. When observing Google's behavior, we first "opted-in" to receive interest-based Google Ads across the web on every test instance. This placed a Doubleclick cookie on the browser instance. No ads were clicked in an automated fashion throughout any experiment.

Experiment 2. *We studied whether multiple browser instances running in parallel affect one another. We compared the ads collected from a browser instance running alone to the ads collected by an instance running with seven additional browser instances each collecting ads from the same page.*

A primary browser instance would first establish an interest in cars by visiting car-related websites. We selected car-related sites by collecting, before the experiment, the top 10 *websites returned by Google when queried with the search terms "BMW buy", "Audi purchase", "new cars", "local car dealers", "autos and vehicles", "cadillac prices", and "best limousines". After manifesting this interest in cars, the instance would collect text ads served by Google on the International Homepage of Times of India (`http://timesofindia.indiatimes.com/international-home`). We attempted to reload the collection page* 10 *times, but occasionally it would time out. Each successful reload would have* 5 *text ads, yielding as many as* 50 *ads.*

Our experiment repeated this round of interest manifestation and ad collection 10 *times using a new primary browser instance during each round. We randomly selected* 5 *of the rounds to include seven additional browsers. When the additional browsers were present, three of them performed the same actions as the primary one. The other four would wait doing nothing instead of visiting the car-related websites and then went on to collecting ads after waiting. All instances would start collecting ads at the same time.* □

The experiment showed that the primary browsers that ran in isolation would receive a more diverse set of ads than those running in parallel with other browsers. We repeated the experiment four times (twice using 20 rounds) and found this pattern each time:

Rounds	Unique ads in isolation	Unique ads in parallel
10	37	25
10	46	33
20	58	47
20	57	52

The presence of this pattern leads us to believe that *cross-unit effects between browser instances exist*. While a statistical test could report whether the observed effect is significant, doing so would inappropriately shift the burden of proof: if a scientist would like to use a statistical analysis that requires an absence of cross-unit effects, then the onus is on her to justify the absence.

This experiment also leads us to suspect that *browser instances are not identically distributed*. In particular, the nature of the cross-unit effects suggested by the experiment raises the possibility that the one browser receiving an ad might decrease the probability for another browser, leading to non-identical distributions.

Given the results of these experiments, for information flow experiments, we find each of these assumptions to be suspect: parametric models, i.i.d. responses, and the absence of cross-unit effects. Any work employing these assumptions must take care to justify their use in their particular experimental design and setting with pilot studies. Believing that these assumptions do not hold for many such experiments, we instead choose to focus on statistical analyses that do not require making such assumptions.

2.5.2 The Permutation Test

Let us look at selecting a statistical test from the angle of security. In our setting, the system in question, not the analyst, is the adversary. From this angle, the pilot studies are reflections of the adversary's ability to violate most assumptions an analyst might wish to make about it. In a security setting, one of the few assumptions safe for the analyst to make is that the adversary cannot guess the (pseudo-)random numbers she generates. Indeed, selecting a random key is the core of many security algorithms, such as encryption.

With the security properties of randomization in mind, we should adopt a statistical test that leverages randomization rather than the types of assumptions more typically seen in statistics. Fortunately, *permutation tests* (e.g., [49]), also known as *randomization tests*, uses randomization to allow cross-unit interactions [121] and non-i.i.d. responses.

At the core of a permutation test is a *test statistic* s. A test statistic is a function from the data, represented as a vector of responses, to a number. The vector of responses $\vec{y}$ has one response for each experimental unit. The vector must be ordered by the random indices i_k used to assign each unit k a treatment from the treatment vector $\vec{x}$ prepared during the experiment. Thus, the kth entry of $\vec{y}$ received the treatment at the kth entry of $\vec{x}$. In particular, s could use the first n components of the data vector as the results of the experimental group and the remaining m as the results for the control group where the groups have n and m units, respectively.

For example, consider an experiment on whether visiting car-related websites impacts the ads one sees. In it, the experimental group visits such websites while the control group does not. The analyst could use a keyword-based test statistic s_{kw}, which looks at the number of ads that each instance received containing the keywords “bmw”, “audi”, “car”, “vehicle”, “automobile”, “cadillac”, and “limo”, words whose presence we believe to be indicative of an instance being in the experimental group. Let the value of s_{kw} be the number of ads that contained any of the keywords amongst the experimental group less the number in the control group. Intuitively, a small value would suggest no noteworthy difference between the groups whereas a large value would indicate that the experimental group saw more car ads as a result of visiting car-related websites.

To make this intuition formal, we must quantify “small” and “large” values. Since the scientist is allowed to pick any function s from response vectors to numbers for the test statistic, the permutation test needs to gauge whether an observed data vector $\vec{y}$ produces a large value with respect to s. To do so, it compares the value of $s(\vec{y})$ to the value of $s(\pi(\vec{y}))$ for every permutation π of $\vec{y}$. Intuitively, this permuting mixes the treatment groups together and compares the observed value of s to its value for these arbitrary groupings.

The significance of these comparisons is that under the null hypothesis of independence (noninterference), the groups should have remained exchangeable after treatment and there is no reason to expect $s(\vec{y})$ to differ in value from $s(\pi(\vec{y}))$. Thus, we would expect to see at least half of the comparisons succeed. Thus, we call a permutation π such that $s(\vec{y}) \leq s(\pi(\vec{y}))$ fails to hold a *rejecting permutation* since too many rejecting permutations leads to rejecting the null hypothesis.

Formally, the value produced by a (one-tailed signed) permutation test given observed responses $\vec{y}$ and a test statistic s is

$$\mathsf{pt}(s, \vec{y}) = \frac{1}{|\vec{y}|!} \sum_{\pi \in \Pi(|\vec{y}|)} I[s(\vec{y}) \leq s(\pi(\vec{y}))] \tag{2.1}$$

where $I[\cdot]$ returns 1 if its argument is true and 0 otherwise, $|\vec{y}|$ is the length of $\vec{y}$ (i.e., the sample size), and $\Pi(|\vec{y}|)$ is the set of all permutations of $|\vec{y}|$ elements, of which there are $|\vec{y}|!$.

Recall that under significance testing, a p-value is the probability of seeing results at least as extreme as the observed data under the assumption that the null hypothesis is true. $\mathsf{pt}(s, \vec{y})$ is a (one-tailed) p-value using s and $\leq$ to define *at least as extreme as* in the definition of p-value. To see this, recall that the null hypothesis H_0 is that the treatments have no effect. Thus, since the order of the responses in $\vec{y}$ is by treatment, which should not matter under H_0, and otherwise random, any permutation of them would be equally likely under H_0. Thus,

$$\mathsf{pt}(s, \vec{y}) = \sum_{\pi \in \Pi(|\vec{y}|) : s(\vec{y}) \leq s(\pi(\vec{y}))} \Pr[\vec{Y} = \vec{y} \mid H_0] \tag{2.2}$$

matches the definition of a p-value. One could use other definitions of *as extreme as* by replacing the $\leq$ in (2.1) and (2.2) by $\geq$ or by comparing the absolute values of $s(\vec{y})$ and $s(\pi(\vec{y}))$ to check for extremism in both directions (a two-tailed test).

Good discusses using sampling to make the computation of $\mathsf{pt}(s, \vec{y})$ tractable for large $\vec{y}$ [49]. Greenland provides detailed justification of using permutation tests to infer causation [57].

2.5.3 Discussion

The above method avoids some pitfalls. Most fundamentally, we use a statistical analysis whose assumptions matches those of our experimental design. Assumptions required by many statistical analyses appear unjustifiable in our setting.

Remark 1. *The permutation test provides a method of determining whether a system has interference that is probabilistically sound to a degree quantified by the p-value.*

Our use of randomization implies that many factors that could be confounding factors in an unrandomized design become noise in our design (e.g., [49]). While such noise may require us to use a large sample size to find an effect, it does not affect the soundness of our analysis. We expect our methodology to suggest that an effect exists when one does not with a probability equal to or less than the p-value.

Remark 2. *The permutation test is not sound for finding noninterference nor complete for finding interference.*

Our method might fail to detect some use of information. For example, the web service's behavior might vary by some feature not measured by the test statistic.

Furthermore, we do not claim that results generalize beyond the setting of the experiment. To do so, our method may be combined with random sampling and methods to ensure that the observed system does not attempt to evade the study.

Lastly, we do not claim that our method shows how the system in question uses information internally. Any observed effect may be the result of complex interactions between the system and other ones in its environment. In particular, as discussed in Section 2.2.3, our method finds interference not within the system in isolation, but rather for the system operating in its environment.

Work	Year	Approach	Limitations/Assumptions
Guha et al. [59]	2010	cosine similarity	lacks test of statistical significance
Balebako et al. [11]	2012	cosine similarity	lacks test of statistical significance
Wills and Tatar [148]	2012	manual examination	lacks test of statistical significance
Sweeney [135]	2013	randomized χ^2 test over browsers	requires a large sample size
Liu et al. [88]	2013	process of elimination	lacks test of statistical significance
Barford et al. [12]	2014	non-randomized χ^2 test over ads	assumes ads identically distributed
Lécuyer et al. [85]	2014	parametric model over ads	correlation; assumes i.i.d. ads and model
Englehardt et al. [37]	2014	parametric model over ads	assumes independence between ads

Table 2.2: Prior Works Experimenting on Online Marketing Systems with Other Methods.

2.6 Prior Studies of WDUD

Our work was inspired by prior WDUD studies. Table 2.2 provides an overview of those using other methods. The first four studies used non-statistical analyses, Sweeney's uses a method similar to ours, and the last four use statistical analyses under assumptions that appear not to hold in the setting we study.

The method of Guha et al. [59], which Balebako et al. adopt to study the effectiveness of web privacy tools [11], uses three browser instances. Two of them receive the same treatment and can be thought of as controls. The third receives some experimental treatment. They collect the ads Google serves each of them, which they compare using a similarity metric. Based on experimental performance, they decided to use one that only looks at the URL displayed in each ad. For each instance, they perform multiple page reloads and record the number of page reloads for which each displayed URL appears. From these counts, they construct a vector for each unit where the ith component of the vector contains the logarithm of the number of reloads during which the ith ad appears. To compare runs, they compare the vectors resulting from the instances using the cosine similarity of the vectors.

More formally, their similarity metric is $\mathsf{sim}(\vec{v}, \vec{w}) = \mathsf{coss}(\ln^*(\vec{v}), \ln^*(\vec{w}))$ where $\vec{v}$ and $\vec{w}$ are vectors that record the number of page reloads during which each displayed URL ad appears, $\ln^*$ applies a logarithm to each component of a vector, and coss computes the cosine similarity of two vectors. They conclude that a flow of information is likely if $\mathsf{sim}(\vec{v}_{\mathsf{c1}}, \vec{v}_{\mathsf{c2}})$ is much larger than $\mathsf{sim}(\vec{v}_{\mathsf{c1}}, \vec{v}_{\mathsf{e}})$ where $\vec{v}_{\mathsf{c1}}$ and $\vec{v}_{\mathsf{c2}}$ are the responses from the two control instances and $\vec{v}_{\mathsf{e}}$ is the response from the experimental instance.

Wills and Tatar also studied how Google selects ads by posing as various sorts of website visitors [148]. They drew conclusions in two ways. The first was similar to Guha et al.'s methodology of looking for differences between the ads seen by various profiles. While their study was conducted by hand and without statistics, they intuitively used the keyword test statistic similar to s_{kw} presented in Section 2.5.2. The second was to observe Google showing them ads that included sensitive information they provided to Google by interacting with a website that uses a Google service.

Liu et al. provide *AdReveal*, a system designed to determine the reasons behind the ads a user sees [88]. They consider three types of ads: (1) ads for particular products that the user has previously expressed interest in, (2) ads based on the context of the website, and (3) ads from behavioral marketing. To check for the first, they check the webpage for Javascript code that sets up re-marketing frameworks. To check for the second, they use machine learning to construct a model that intuitively judges whether an ad and a webpage cover the same topic. They consider ads that do not trigger either of these checks to be behavioral.

None of these studies performed a statistical analyses to show whether or not their results are significant. It remains possible that the differences observed are simply from random variations caused by factors other than behavioral marketing.

Sweeney examined the flow of information from a search field to ads shown alongside the search results [135]. Among other things, she found that, compared to characteristically white names, searching for characteristically black names yielded more ads for InstantCheckmate that were suggestive of the searched name having an arrest record. She randomized the order of names searched [136], which can enable statistical analyses without making questionable assumptions. She used the χ^2 test to analyze her results and found them to be significant.

Under some conditions, the χ^2 test becomes an approximation of the permutation test [89]. With the size of her data set, such approximations become not only accurate, but useful for computational reasons. Indeed, her results are also statistically significant under the permutation test [136].

We believe that our methodology with permutation testing provides a foundation for such approximations by linking them to information flow, especially considering that the traditional justification of the χ^2 test includes an assumption that the experimental units are independent [86], which seems unlikely in some cases (Experiment 2). The permutation test is also more flexible in that it works for any response variable whereas the χ^2 test is only for a single categorical (binary) response from each experimental unit (e.g., browser instance). For example, the χ^2 test cannot be used if the response is the number of times a certain ad appears to each browser since this is not a categorical response.

For each webpage, ad, and user profile, Barford et al. [12] record the number of times that webpage shows that ad to a user with that profile as they crawl the web. Using the χ^2 test, they identify those ads shown on a webpage to some profiles more often than others, suggesting behavioral targeting. However, rather than use randomization for the purposes of the χ^2 test, they counted each ad as an i.i.d. response [90] (cf. Experiments 1 and 2).

Lécuyer et al. present XRay, a tool that looks for correlations between the data that web services have about users and the ads shown to users [85]. While their tool may check many changes to a type of input to determine whether any of them has a correlation with the frequency of a single ad, it does not check for causation, as our method does. Furthermore, they assume identically distributed ads (cf. Experiment 1).

Englehardt et al. study filter bubbles with an analysis that assumes a binomial model and independence between observations [37] (cf. Experiment 2).

2.7 Comparison of Test Statistics

Given all the test statistics discussed, one might wonder how they compare. We will empirically compare a subset of the test statistics in our motivating setting of WDUD. However, we caution that our experiment should not be considered definitive since other WDUD problems may result in different results. We recommend that each experiment is preceded by a pilot study to determine the best test(s) for the experiment's needs. For example, we have found pilot studies useful for selecting distinguishing keywords to search for in ads.

The following experiment also illustrates our experimental design and statistical analysis. We find that it works as expected by running it in settings where we can be almost certain that targeting either is or is not happening.

Experiment 3. *This experiment has multiple iterations. Each iteration involved ten simultaneous browser instances, each of which represents an experimental unit. We used a sample size of ten due to the processing power and RAM restrictions of our experimental setup. In each iteration, we randomly assigns five of the instances, the experimental group, to receive the treatment of manifesting an interest in cars (a heavily marketed topic). As in Experiment 2, an instance manifests its interest by visiting the top* 10 *websites returned by Google when queried with certain automobile-related terms: "BMW buy", "Audi purchase", "new cars", "local car dealers", "autos and vehicles", "cadillac prices", and "best limousines". The remaining five instances made up our control group, which remained idle as the experimental group visited the car-related websites. Such idling is needed to remove time as a factor ensuring that the only systematic difference between the two groups was the treatment of visiting car-related websites.*

As soon as the experimental group completed visiting the websites, all ten instances began collecting text ads served by Google on the International Homepage of Times of India. As in Experiment 2, each instance attempted to collect 50 *text ads by reloading a page of five ads ten times, but page timeouts would occasionally result in an instance getting fewer. We repeated this process for* 20 *iterations with fresh instances to collect* 20 *data sets, each containing ads from each of ten instances.*

□

Across all runs of the experiment, we collected 9832 ads with 281 being unique. Instances collected between 40 and 50 ads with two outliers each collecting zero. Both outliers were in the 19th run and in the experimental group.

First, we used the permutation test with s_{sim}, an extension of Guha et al.'s cosine similarity metric sim for comparing more than two responses (Section 2.6), as the test statistic [59]. We first aggregate together multiple URL-count vectors by computing the average number of times each URL appeared across the aggregated units. Formally, let $\mathsf{avg}(\vec{u})$ compute the component-wise average of the vectors in $\vec{u}$, a vector of vectors of URL counts. We can then define the test statistic $s_{\mathsf{sim}}(\vec{y})$ as $-\mathsf{sim}(\mathsf{avg}(\vec{y}_{1:n}), \mathsf{avg}(\vec{y}_{n+1:n+m}))$ where $\vec{y}_{a:b}$ is the sub-vector consisting of the entries a though b of $\vec{y}$, the first n responses are from the experimental group, and the next m are those from the control group. We use negation since our permutation test takes a metric of difference, not similarity. Intuitively, the permutation test using the test statistic s_{sim} will compare the *between-group* dis-similarity to the dis-similarity of vectors that mix up the units by a permutation. In aggregate, the dis-similarity of these mixed up vectors provide a view on the global dis-similarity inherit in the system.

Observe that there are $10! > 3$ million different permutations for the ten instances. However, since sim treats the response vector provided to it as two sets (intuitively, the experimental and control groups) many permutations will produce the same value for s_{sim}. To speed up the calculation, we replaced comparing all permutations with comparing all partitions of the responses into equal sized sets of 5, yielding only $\binom{10}{5} = 252$ comparisons.

Second, we tested the statistic s_{kw} inspired by Wills and Tatar [148] and discussed in Section 2.5.2, which looks at the number of ads that each instance received containing a keyword. As with s_{sim}, we have at most 252 unique comparisons to make.

Third, we tested a simplified version of s_{kw}, s_{kw01}, that treats each browser instance as providing a categorical (binary) response that is 1 if it got any number of ads with a keyword and 0 if it got none. Comparing s_{kw} to s_{kw01} illustrates the power of non-categorical responses.

Lastly, we conducted the χ^2 test on a 2×2 contingency table computed from the data from each round. The type of treatment was represented in rows, while the presence or absence of keywords was represented in the columns, using an approach similar to s_{kw01} since the χ^2 test is limited to categorical variables. However, our sample size was not large enough to produce meaningful results from the χ^2 test, which requires that each outcome shows up with a certain minimum frequency. Thus, we did not consider this test further.

For comparison purposes, we re-run the above experiment without having the experimental group manifest any interests. That is, we compared two control groups against one another expecting not to find

Setup	Actual	Reported positives		
	positives	s_{sim}	s_{kw}	s_{kw01}
Experiment-control	20	18	18	0
Control-control	0	1	1	1
Experiment-experiment	0	0	1	0

Table 2.3: The number of runs out of 20 experiments that various tests consider to show information usage.

statistically significant differences.

Table 2.3 summarizes the results using the standard $\alpha = 0.05$ cutoff for statistical significance. For the experiment-control setup, in which we do expect to find a difference between the groups, both s_{sim} and s_{kw} reported a positive result 18 out of 20 times whereas the s_{kw01} reported no significant results. For s_{kw}, 13 of the p-values were less than 0.004 (Appendix A.5.3). They show, with high certainty, that Google and its ad ecosystem has interference from visiting the car related webpages to the ads that Google shows. Furthermore, these results show that s_{kw} is indeed a more powerful test statistic than s_{kw01}, which is limited to categorical responses.

As expected with a theoretical false positive rate of $\alpha = 5\%$ and 20 tests, we found that each of the permutation tests produced one or fewer statistically significant results for the experiments with no difference (control-control and experiment-experiment). We provide no minimal acceptable number of true positives for the experiment-control setup since significance testing does not guarantee any rate.

The wide range of tests might tempt one into running more than one test on a data set. However, running multiple tests increases the chance of getting a low p-value for one of them by an unlucky randomization of units rather from an effect. Thus, one cannot look just at the test that produced the lowest p-value. Rather one must report them all or apply a correction for multiple tests such as those for the false discovery rate [15].

2.8 Conclusions and Suggestions

Based upon theoretical results, we reduced finding a flow of information in WDUD and related applications to that of finding an effect using experiments. Based upon empirical observations, we recommended an experimental design and a statistical analysis, based on permutation testing, that is well suited to studying behavioral marketing. This process allows us to convert the abstract principles of experimental design and analysis into concrete suggestions:

1. *Use an appropriate statistical test.* Attempting to shoehorn data into familiar statistics can result in

incurring requirements, such as i.i.d. data, that cannot be met in our setting. Fortunately, they are *not* required for permutation testing.

2. *Randomly assign treatments to units.* Randomization provides the justification needed for permutation testing, which allows us to avoid unachievable requirements needed for other statistical tests.

3. *Use domain knowledge gained during pilot studies to select a test statistic.* Finding the correct keywords to examine in ads allowed us to not only get results that were statistically significant, but also intuitive.

While statistical analyses can be intimidating due to their complex requirements, selecting the correct test is liberating by also identifying what conditions the analyst needs not worry about. In addition to not needing i.i.d. data, permutation testing does not require complete control over the environment or a lack of cross-unit effects, none of which are achievable in our setting.

Furthermore, we do not require random samples. Acquiring units by randomly sampling from a more general population will, with high likelihood, provide a representative sample, which allows findings of effects to generalize to the population as a whole. While results need not be general to show that a marketer tracks some behavior, showing that the marketer often does is more interesting. However, given that the marketer could alter its behavior in response to the atypical patterns of access exhibited by experiments, we take comfort in knowing that our findings of information usage will hold even if they do not generalize to marketer's typical behavior. (Also, it would be odd for a marketer to only exhibit questionable behavior to those looking for it.)

In general, information flow experiments allow an analyst to exercise oversight and detect transgressions by an entity not controlled by the analyst and unwilling or unable to provide the analyst complete access to the system. We see this setting becoming ever more common: data lives in the cloud, jobs are outsourced, products licensed, and services replace infrastructure. In each of these cases, a party has ceded control of a resource for efficiency. Nevertheless, each party must ensure that the other abides by their agreement and respect privacy policies while having only limited access to the other. Thus, we envision black-box experimentation for auditing and accountability playing an increasing role in information security, privacy, and society in general.

Chapter 3

Discrimination, Opacity, and Choice in Google's Ad Ecosystem

3.1 Introduction

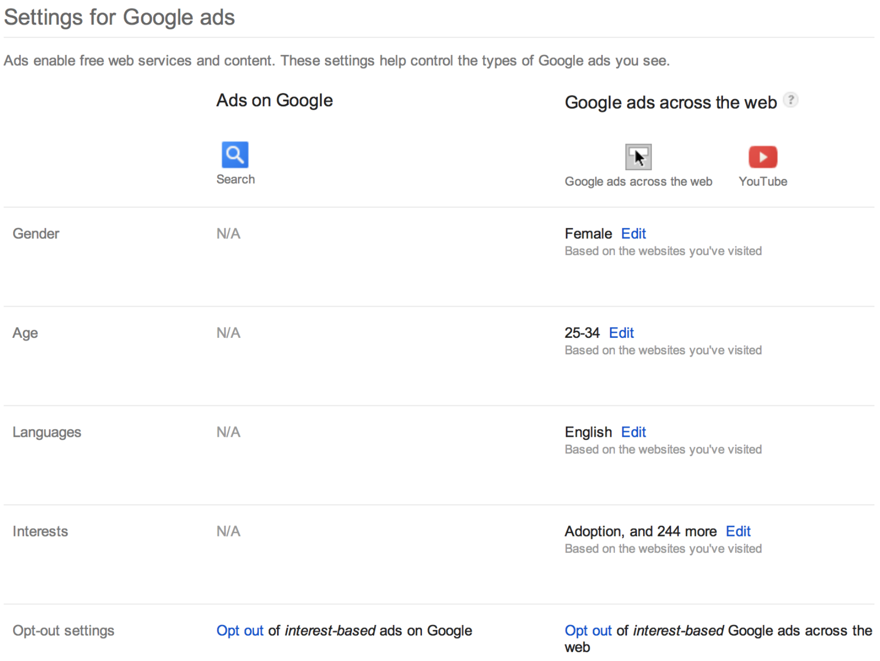
In this chapter, we apply information flow experiments to evaluate discrimination, transparency, and choice in Google's advertising ecosystem. To detect information flow to the large numbers of ads that the Google ad ecosystem produces, we extend the methodology from Chapter 2 to be scalable. To avoid missing subtle effects due to small samples, we modify the experimental setup and the nature of statistical analyses to increase the sample size. Additionally, to avoid having to guess which effect to test for, we use machine learning to automate the selection of the effect. Incorporating these two improvements to the methodology, we develop AdFisher as a general framework for automating such experiments on web-based systems and use it to study Google's advertising ecosystem,. Using AdFisher, we uncover evidence suggestive of gender-based discrimination in employment-related ads, opacity in Google's transparency tool, and respect for choice in the choice tool.

3.1.1 Problem and Overview

To increase transparency and choice into behavioral advertising on the web, Google provides Ad Settings, which is "a Google tool that helps you control the ads you see on Google services and on websites that partner with Google" [50]. It displays inferences Google has made about a user's demographics and interests based on his browsing behavior. Users can view and edit these settings at

`http://www.google.com/settings/ads`

A user can also edit these demographics. Figure 3.1 provides a screenshot. Yahoo [150] and Microsoft [98]



Settings for Google ads

Ads enable free web services and content. These settings help control the types of Google ads you see.

	Ads on Google	Google ads across the web ?
	Search	Google ads across the web YouTube
Gender	N/A	Female Edit Based on the websites you've visited
Age	N/A	25-34 Edit Based on the websites you've visited
Languages	N/A	English Edit Based on the websites you've visited
Interests	N/A	Adoption, and 244 more Edit Based on the websites you've visited
Opt-out settings	Opt out of *interest-based* ads on Google	Opt out of *interest-based* Google ads across the web

Figure 3.1: Screenshot of Google's Ad Settings webpage in 2015

also offer personalized ad settings.

However, they provide little information about how these pages operate, leaving open the question of how completely these settings describe the profile they have about a user. In this study, we explore how a user's behaviors, either directly with the settings or with content providers, alter the ads and settings shown to the user and whether these changes are in harmony. In particular, we study the degree to which the settings provides transparency and choice as well as checking for the presence of discrimination. Transparency is important for people to understand how the use of data about them affects the ads they see. Choice allows users to control how this data gets used, enabling them to protect the information they find sensitive. Discrimination is an increasing concern about machine learning systems and one reason people like to keep information private [40,155].

To conduct these studies, we developed AdFisher, a tool for automating randomized, controlled experiments for studying online tracking. Our tool offers a combination of automation, statistical rigor, scalability, and explanation for determining the use of information by web advertising algorithms and by personalized ad settings, such as Google Ad Settings. The tool can simulate having a particular interest or attribute by visiting webpages associated with that interest or by altering the ad settings provided by Google. It collects ads served by Google and also the settings that Google provides to the simulated users. It automatically analyzes the data to determine whether statistically significant differences between groups of agents exist. AdFisher uses machine learning to automatically detect differences and then executes a test of significance specialized for the difference it found.

Someone using AdFisher to study behavioral targeting only has to provide the behaviors the two groups are to perform (e.g., visiting websites) and the measurements (e.g., which ads) to collect afterwards. AdFisher can easily run multiple experiments exploring the causal connections between users' browsing activities, and the ads and settings that Google shows.

The advertising ecosystem is a vast, distributed, and decentralized system with several players including the users consuming content, the advertisers, the publishers of web content, and ad networks. With the exception of the user, we treat the entire ecosystem as a blackbox. We measure simulated users' interactions with this blackbox including page views, ads, and ad settings. Without knowledge of the internal workings of the ecosystem, we cannot assign responsibility for our findings to any single player within it nor rule out that they are unintended consequences of interactions between players. However, our results show the presence of concerning effects illustrating the existence of issues that could be investigated more deeply by either the players themselves or by regulatory bodies with the power to see the internal dynamics of the ecosystem.

3.1.2 Motivating Experiments

In one experiment, we explored whether visiting websites related to substance abuse has an impact on Google's ads or settings. We created an experimental group and a control group of agents. The browser agents in the experimental group visited websites on substance abuse while the agents in the control group simply waited. Then, both groups of agents collected ads served by Google on a news website.

Having run the experiment and collected the data, we had to determine whether any difference existed in the outputs shown to the agents. One way would be to intuit what the difference could be (e.g. more ads containing the word "alcohol") and test for that difference. However, developing this intuition can take considerable effort. Moreover, it does not help find unexpected differences. Thus, we instead used machine learning to automatically find differentiating patterns in the data. Specifically, AdFisher finds a classifier that can predict which group an agent belonged to, from the ads shown to an agent. The classifier is trained on a subset of the data. A separate test subset is used to determine whether the classifier found a statistically significant difference between the ads shown to each group of agents. In this experiment, AdFisher found a classifier that could distinguish between the two groups of agents by using the fact that only the agents that visited the substance abuse websites received ads for Watershed Rehab.

We also measured the settings that Google provided to each agent on its Ad Settings page after the experimental group of agents visited the webpages associated with substance abuse. We found no differences (significant or otherwise) between the pages for the agents. Thus, information about visits to these websites is indeed being used to serve ads, but the Ad Settings page does not reflect this use in this case. Rather than providing transparency, in this instance, the ad settings were *opaque* as to the impact of this factor.

In another experiment, we examined whether the settings provide *choice* to users. We found that removing interests from the Google Ad Settings page changes the ads that a user sees. In particular, we had both groups of agents visit a site related to online dating. Then, only one of the groups removed the interest related to online dating. Thereafter, the top ads shown to the group that kept the interest were related to dating but not the top ads shown to the other group. Thus, the ad settings do offer the users a degree of choice over the ads they see.

We also found evidence suggestive of *discrimination* from another experiment. We set the agents' gender to female or male on Google's Ad Settings page. We then had both the female and male groups of agents visit webpages associated with employment. We established that Google used this gender information to select ads, as one might expect. The interesting result was how the ads differed between the groups: during this experiment, Google showed the simulated males ads from a certain career coaching agency that promised large salaries more frequently than the simulated females, a finding suggestive of discrimination. Ours is the first study that provides statistically significant evidence of an instance of discrimination in online advertising when demographic information is supplied via a transparency-choice mechanism (i.e., the Ad Settings page).

While neither of our findings of opacity or discrimination are clear violations of Google's privacy policy [52] and we do not claim these findings to generalize or imply widespread issues, we find them concerning and warranting further investigation by those with visibility into the ad ecosystem. Furthermore, while our finding of discrimination in the non-normative sense of the word is on firm statistical footing, we acknowledge that people may disagree about whether we found discrimination in the normative sense of the word. We defer discussion of whether our findings suggest unjust discrimination until Section 3.7.

3.1.3 Chapter Contributions

In addition to the experimental findings highlighted above, we provide AdFisher, a tool for *automating* such experiments. AdFisher is structured as a Python API providing functions for setting up, running, and analyzing experiments. We use Selenium to drive Firefox browsers and the scikit-learn library [115] for implementations of classification algorithms. We use the SciPy library [69] for implementing the statistical analyses of the core methodology.

AdFisher offers *rigor* by performing a carefully designed experiment. We apply statistical analyses techniques from Chapter 2 which do not make questionable assumptions about the collected data.

Our automation, experimental design, and statistical analyses allow us to *scale* to handling large numbers of agents for finding subtle differences. In particular, we modify the analysis of Chapter 2 to allow for experiments running over long periods of time. We do so by using *blocking* (e.g., [49]), a nested statistical analysis not previously applied to understanding web advertising. The blocking analysis ensures that agents are only compared to the agents that start out like it and then aggregates together the comparisons across blocks of agents. Thus, AdFisher may run agents in batches spread out over time while only comparing those agents running simultaneously to one another.

AdFisher also provides *explanations* as to how Google alters its behaviors in response to different user actions. It uses the trained classifier model to find which features were most useful for the classifier to make its predictions. It provides the top features from each group to provide the experimenter/analyst with a qualitative understanding of how the ads differed between the groups.

To maintain statistical rigor, we carefully circumscribe our claims. We only claim statistical soundness of our results: if our techniques detect an effect of the browsing activities on the ads, then there is indeed one with high likelihood (made quantitative by a p-value). We do not claim that we will always find a difference if one exists, nor that the differences we find are typical of those experienced by users. Furthermore, while we can characterize the differences, we cannot assign blame for them since either Google or the advertisers working with Google could be responsible.

Contents. After covering prior work next, we present, in Section 3.3, privacy properties that our tool AdFisher can check: nondiscrimination, transparency, and choice. Section 3.4 explains the methodology we use to ensure sound conclusions from using AdFisher. Section 3.5 presents the design of AdFisher. Section 3.6 discusses our use of AdFisher to study Google's ads and settings. We end with conclusions and future work.

Raw data and additional details about AdFisher and our experiments can be found at

http://www.cs.cmu.edu/~mtschant/ife/

AdFisher is freely available at

https://github.com/tadatitam/info-flow-experiments/

3.2 Prior Work

We are not the first to study how Google uses information. The work with the closest subject of study to ours is by Wills and Tatar [148]. They studied both the ads shown by Google and the behavior of Google's Ad Settings (then called the "Ad Preferences"). Like us, they find the presence of opacity: various interests impacted the ads and settings shown to the user and that ads could change without a corresponding change in Ad Settings. Unlike our study, theirs was mostly manual, small scale, lacked any statistical analysis, and did not follow a rigorous experimental design. Furthermore, we additionally study choice and discrimination.

Other related works differ from us in both goals and methods. They all focus on how visiting webpages change the ads seen. While we examine such changes in our work, we do so as part of a larger analysis of the interactions between ads and personalized ad settings, a topic they do not study.

Barford et al. come the closest in that their recent study looked at both ads and ad settings [12]. They do so in their study of the "adscape", an attempt to understand each ad on the Internet. They study each ad individually and cast a wide net to analyze many ads from many websites while simulating many different interests. They only examine the ad settings to determine whether they successfully induced an interest. We rigorously study how the settings affects the ads shown (choice) and how behaviors can affect ads without affecting the settings (transparency). Furthermore, we use focused collections of data and an analysis that considers all ads collectively to find subtle causal effects within Google's advertising ecosystem. We also use a randomized experimental design and analysis to ensure that our results imply causation.

The usage study closest to ours in terms of implementation is that of Liu et al. in that they also use machine learning [88]. Their goal is to determine whether an ad was selected due to the content of a page, by using behavioral profiling, or from a previous webpage visit. Thus, rather than using machine learning to select a statistical test for finding causal relations, they do so to detect whether an ad on a webpage matches the content on the page to make a case for the first possibility. Thus, they have a separate classifier for each interest a webpage might cover. Rather than perform a statistical analysis to determine whether treatment groups have a statistically significant difference, they use their classifiers to judge the ratio of ads on a page unrelated to the page's content, which they presume indicates that the ads were the result of behavioral targeting.

Lécuyer et al. present XRay, a tool that looks for correlations between the data that web services have about users and the ads shown to users [85]. While their tool may check many changes to a type of input to determine whether any of them has a correlation with the frequency of a single ad, it does not check for causation, as ours does.

Englehardt et al. study filter bubbles with an analysis that assumes independence between observations [37], an assumption we are uncomfortable making. (See Section 3.4.4.)

Guha et al. compare ads seen by three agents to see whether Google treats differently the one that behaves differently from the other two [59]. We adopt their suggestion of focusing on the title and URL displayed on ads when comparing ads to avoid noise from other less stable parts of the ad. Our work differs by studying the ad settings in addition to the ads and by using larger numbers of agents. Furthermore, we use rigorous statistical analyses. Balebako et al. run similar experiments to study the effectiveness of privacy tools [11].

Sweeney performed an experiment to determine that searching for names associated with African-Americans produced more search ads suggestive of an arrest record than names associated with European-Americans [135]. Her study required considerable insight to determine that suggestions of an arrest was a key difference. AdFisher can automate not just the collection of the ads, but also the identification of such key differences by using its machine learning capabilities. Indeed, it found on its own that simulated males were more often shown ads encouraging the user to seek coaching for high paying jobs than simulated females.

3.3 Privacy Properties

Motivating our methodology for finding causal relationships, we present some properties of ad networks that we can check with such a methodology in place. As a fundamental limitation of science, we can only prove the existence of a causal effect; we cannot prove that one does not exist (see Section 3.4.5). Thus, experiments can only demonstrate violations of nondiscrimination and transparency, which require effects.

On the other hand, we can experimentally demonstrate that effectful choice and ad choice are complied with in the cases that we test since compliance follows from the existence of an effect. Table 3.1 summarizes these properties.

3.3.1 Discrimination

At its core, *discrimination* between two classes of individuals (e.g., one race vs. another) occurs when the attribute distinguishing those two classes causes a change in behavior toward those two classes. In our case, discrimination occurs when membership in a class causes a change in ads. Such discrimination is not always bad (e.g., many would be comfortable with men and women receiving different clothing ads).

Property Name	Requirement	Causal Test	Finding
Nondiscrimination	Users differing only on protected attributes are treated similarly	Find that presence of protected attribute causes a change in ads	Violation
Transparency	User can view all data about him used for ad selection	Find attribute that causes a change in ads, not in settings	Violation
Effectful choice	Changing a setting has an effect on ads	Find that changing a setting causes a change in ads	Compliance
Ad choice	Removing an interest decreases the number ads related to that interest	Find that changing a setting causes a decrease in relevant ads	Compliance

Table 3.1: Privacy Properties Tested on Google's Ad Settings

We limit our discussion of whether the discrimination we found is unjust to the discussion section (§3.7) and do not claim to have a scientific method of determining the morality of discrimination.

Determining whether class membership causes a change in ads is difficult since many factors not under the experimenter's control or even observable to the experimenter may also cause changes. Our experimental methodology determines when membership in certain classes causes significant changes in ads by comparing many instances of each class.

We are limited in the classes we can consider since we cannot create actual people that vary by the traditional subjects of discrimination, such as race or gender. Instead, we look at classes that function as surrogates for those classes of interest. For example, rather than directly looking at how gender affects people's ads, we instead look at how altering a gender setting affects ads or at how visiting websites associated with each gender affects ads.

3.3.2 Transparency

Transparency tools like Google Ad Settings provide online consumers with some understanding of the information that ad networks collect and use about them. By displaying to users what the ad network may have learned about the interests and demographics of a user, such tools attempt to make targeting mechanisms more transparent.

However the technique for studying transparency is not straightforward. One cannot expect an ad network to be *completely transparent* to a user. This would involve the tool displaying all other users' interests as well. A more reasonable expectation is for the ad network to display any inferred interests about that user. So, if an ad network has inferred some interest about a user and is serving ads relevant to that interest, then that interest should be displayed on the transparency tool. However, even this notion of transparency cannot be checked precisely as the ad network may serve ads about some other interest correlated with the original inferred interest, but not display the correlated interest on the transparency tool.

Thus, we only study the extreme case of the lack of transparency — *opacity*, and leave complex notions of transparency open for future research. We say that a transparency tool has opacity if some browsing activity results in a significant effect on the ads served, but has no effect on the ad settings. If there is a difference in the ads, we can argue that prior browsing activities must have been tracked and used by the ad network to serve relevant ads. However, if this use does not show up on the transparency tool, we have found at least one example which demonstrates a lack of transparency.

3.3.3 Choice

The Ad Settings page offers users the option of editing the interests and demographics inferred about them. However, the exact nature of how these edits impact the ad network is unclear. We examine two notions of choice.

A very coarse form is *effectful choice*, which requires that altering the settings has some effect on the ads seen by the user. This shows that altering settings is not merely a "placebo button": it has a real effect on the network's ads. However, effectful choice does not capture whether the effect on ads is meaningful. For example, even if a user adds interests for cars and starts receiving *fewer* ads for cars, effectful choice is satisfied. Moreover, we cannot find violations of effectful choice. If we find no differences in the ads, we cannot conclude that users do not have effectful choice since it could be the result of the ad repository lacking ads relevant to the interest.

Ideally, the effect on ads after altering a setting would be meaningful and related to the changed setting. One way such an effect would be meaningful, in the case of removing an inferred interest, is a decrease in the number of ads related to the removed interest. We call this requirement *ad choice*. One way to judge whether an ad is relevant is to check it for keywords associated with the interest. If upon removing an interest, we find a statistically significant decrease in the number of ads containing some keywords, then we will conclude that the choice was respected. In addition to testing for compliance in ad choice, we can also test for a violation by checking for a statistically significant increase in the number of related ads to find egregious violations. By requiring the effect to have a fixed direction, we can find both compliance and violations of ad choice.

3.4 Methodology

The goal of our methodology is to establish that a certain type of input to a system causes an effect on a certain type of output of the system. For example, in our experiments, we study the system of Google. The inputs we study are visits to content providing websites and users' interactions with the Ad Settings page. The outputs we study are the settings and ads shown to the users by Google. However, nothing in our methodology limits ourselves to these particular topics; it is appropriate for determining I/O properties of any web system. Here, we present an overview of our methodology; Appendix B.2 provides details of the statistical analysis.

3.4.1 Background: Significance Testing

To establish causation, we start with the approach of Fisher (our tool's namesake) for significance testing [44] as specialized in Chapter 2. Significance testing examines a *null hypothesis*, in our case, that the inputs do not affect the outputs. To test this hypothesis the experimenter selects two values that the inputs could take on, typically called the *control* and *experimental treatments*. The experimenter applies the treatments to *experimental units*. In our setting, the units are the browser agents, that is, simulated users. To avoid noise, the experimental units should initially be as close to identical as possible as far as the inputs and outputs in question are concerned. For example, an agent created with the Firefox browser should not be compared to one created with the Internet Explorer browser since Google can detect the browser used.

The experimenter randomly applies the experimental (control) treatment to half of the agents, which form the experimental (control) group. (See Figure 3.2.) Each agent carries out actions specified in the treatment applied to it. Next, the experimenter takes measurements of the outputs Google sends to the agents, such as ads. At this point, the experiment is complete and data analysis begins.

Data analysis starts by computing a *test statistic* over the measurements. The experimenter selects a test statistic that she suspects will take on a high value when the outputs to the two groups differ. That is, the statistic is a measure of distance between the two groups. She then uses the *permutation test* to determine whether the value the test statistic actually took on is higher than what one would expect by chance unless the groups actually differ. The permutation test randomly permutes the labels (control and experimental) associated with each observation, and recomputes a hypothetical test statistic. Since the null hypothesis is that the inputs have no effect, the random assignment should have no effect on the value of the test statistic. Thus, under the null hypothesis, it is unlikely that the actual value of the test statistic is larger than the vast majority of hypothetical values.

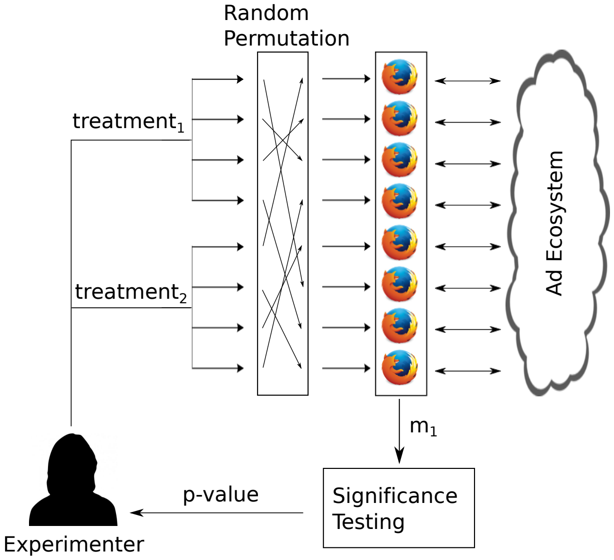


Figure 3.2: Experimental setup to carry out significance testing on eight browser agents comparing the effects of two treatments. Each agent is randomly assigned a treatment which specifies what actions to perform on the web. After these actions are complete, they collect measurements which are used for significance testing.

The *p-value* of the permutation test is the proportion of the permutations where the test statistic was greater than or equal to the actual observed statistic. If the value of the test statistic is so high that under the null hypothesis it would take on as high of a value in less than 5% of the random assignments, then we conclude that the value is *statistically significant* (at the 5% level) and that causation is likely.

3.4.2 Blocking

In practice, the above methodology can be difficult to use since creating a large number of nearly identical agents might not be possible. In our case, we could only run ten agents in parallel given our hardware and network limitations. Comparing agents running at different times can result in additional noise since ads served to an agent change over time. Thus, with the above methodology, we were limited to just ten comparable units. Since some effects that the inputs have on Google's outputs can be probabilistic and subtle, they might be missed looking at so few agents.

To avoid this limitation, we extended the above methodology to handle varying units using *blocking* [49]. To use blocking, we created *blocks* of nearly identical agents running in parallel. These agents differ in terms their identifiers (e.g., process id) and location in memory. Despite the agents running in parallel, the operating system's scheduler determines the exact order in which the agents operate. Each block's agents were randomly partitioned into the control and experimental groups. This randomization ensures that the minor differences between agents noted above should have no systematic impact upon

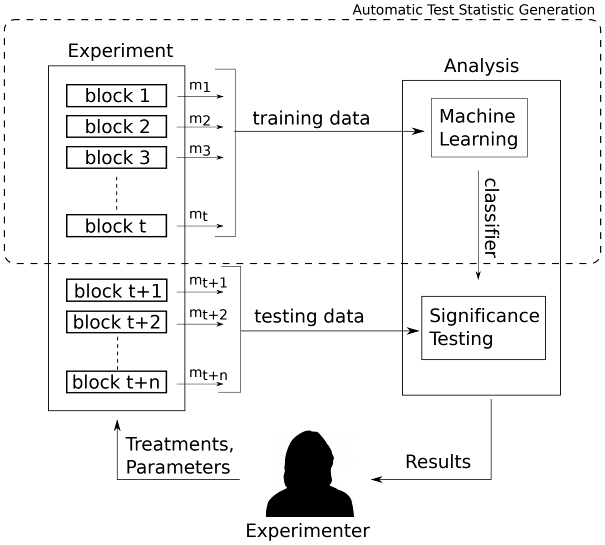


Figure 3.3: Our experimental setup with training and testing blocks. Measurements from the training blocks are used to build a classifier. The trained classifier is used to compute the test statistic on the measurements from the testing blocks for significance testing.

the results: these differences become noise that probably disappears as the sample size increases. Running these blocks in a staged fashion, the experiment proceeds on block after block. A modified permutation test now only compares the actual value of the test statistic to hypothetical values computed by reassignments of agents that respect the blocking structure. These reassignments do not permute labels across blocks of observations.

Using blocking, we can scale to any number of agents by running as many blocks as needed. However, the computation of the permutation test increases exponentially with the number of blocks. Thus, rather than compute the exact p-value, we estimate it by randomly sampling the possible reassignments. We can use a confidence interval to characterize the quality of the estimation [49]. The p-values we report are actually the upper bounds of the 99% confidence intervals of the p-values (details in Appendix B.2).

3.4.3 Selecting Test Statistics

The above methodology leaves open the question of how to select the test statistic. In some cases, the experimenter might be interested in a particular test statistic. For example, an experimenter testing ad choice could use a test statistic that counts the number of ads related to the removed interest. In other cases, the experimenter might be looking for *any* effect. AdFisher offers the ability to automatically select a test statistic. To do so, it partitions the collected data into training and testing subsets, and uses the training data to train a classifier. Figure 3.3 shows an overview of AdFisher's workflow.

To select a classifier, AdFisher uses 10-fold cross validation on the training data to select among several possible parameters. The classifier predicts which treatment an agent received, only from the ads that get served to that agent. If the classifier is able to make this prediction with high accuracy, it suggests a systematic difference between the ads served to the two groups that the classifier was able to learn. If no difference exists, then we would expect the number to be near the guessing rate of 50%. AdFisher uses the accuracy of this classifier as its test statistic.

To avoid the possibility of seeing a high accuracy due to overfitting, AdFisher evaluates the accuracy of the classifier on a testing data set that is disjoint from the training data set. That is, in the language of statistics, we form our hypothesis about the test statistic being able to distinguish the groups before seeing the data on which we test it to ensure that it has predictive power. AdFisher uses the permutation test to determine whether the degree to which the classifier's accuracy on the test data surpasses the guessing rate is statistically significant. That is, it calculates the p-value that measures the probability of seeing the observed accuracy given that the classifier is just guessing. If the p-value is below 0.05, we conclude that it is unlikely that classifier is guessing and that it must be making use of some difference between the ads shown to the two groups.

3.4.4 Avoiding Pitfalls

The above methodology avoids some pitfalls. Most fundamentally, we use a statistical analysis whose assumptions match those of our experimental design. Assumptions required by many statistical analyses appear unjustifiable in our setting. For example, many analyses assume that the agents do not interact or that the ads are independent and identically distributed (e.g., [12, 37]). Given that all agents receive ads from the same pool of possible ads governed by the same advertisers' budgets, these assumptions appear unlikely to hold. Indeed, we find empirical evidence suggesting that it does not (Experiment 2 in Section 2.5). The permutation test, which does not require this assumption, allows us to ensure statistical soundness of our analysis without making these assumptions [58].

Our use of randomization implies that many factors that could be confounding factors in an unrandomized design become noise in our design (e.g., [49]). While such noise may require us to use a large sample size to find an effect, it does not affect the soundness of our analysis.

Our use of two data sets, one for training the classifier to select the test statistic and one for hypothesis testing ensures that we do not engage in overfitting, data dredging, or multiple hypothesis testing (e.g., [99]). All these problems result from looking for so many possible patterns that one is found by chance. While we look for many patterns in the training data, we only check for one in the testing data.

Relatedly, by reporting a p-value, we provide a quantitative measure of the confidence we have that the observed effect is genuine and not just by chance [68]. Reporting simply the classifier accuracy or that some difference occurred fails to quantify the possibility that the result was a fluke.

3.4.5 Scope

We restrict the scope of our methodology to making claims that an effect exists with high likelihood as quantified by the p-value. That is, we expect our methodology to only rarely suggest that an effect exists when one does not.

We do not claim "completeness" or "power": we might fail to detect some use of information. For example, Google might not serve different ads upon detecting that all the browser agents in our experiment are running from the same IP address. Despite this limitation in our experiments, we found interesting instances of usage.

Furthermore, we do not claim that our results generalize to all users. To do so, we would need to a take a random sample of all users, their IP addresses, browsers, and behaviors, which is prohibitively expensive. We cannot generalize our results if for example, instead of turning off some usage upon detecting our experiments, Google turns it on. While our experiments would detect this usage, it might not be experienced by normal users. However, it would be odd if Google purposefully performs questionable behaviors only with those attempting to find it.

While we use webpages associated with various interests to simulate users with those interests, we cannot establish that having the interest itself caused the ads to change. It is possible that other features of the visited webpages causes change - a form of confounding called "profile contamination" [12], since the pages cover other topics as well. Nevertheless, we have determined that visiting webpages associated with the interest does result in seeing a change, which should give pause to users visiting webpages associated with sensitive interests.

Lastly, we do not attempt to determine how the information was used. It could have been used by Google directly for targeting or it could have been used by advertisers to place their bids. We cannot assign blame. We hope future work will shed light on these issues, but given that we cannot observe the interactions between Google and advertisers, we are unsure whether it can be done.

3.5 AdFisher

In this section, we describe AdFisher - a tool implementing our methodology. AdFisher makes it easy to run experiments using the above methodology for a set of treatments, measurements, and classifiers (test statistics) we have implemented. AdFisher is also extensible allowing the experimenter to implement additional treatments, measurements, or test statistics. For example, an experimenter interested in studying a different online platform only needs to add code to perform actions and collect measurements on that platform. They need not modify methods that randomize the treatments, carry out the experiment, or perform the data analysis.

To simulate a new person on the network, AdFisher creates each agent from a fresh browser instance with no browsing history, cookies, or other personalization. AdFisher randomly assigns each agent to a group and applies the appropriate treatment, such as having the browser visit webpages. Next, AdFisher makes measurements of the agent, such as collecting the ads shown to the browser upon visiting another webpage. All of the agents within a block execute and finish the treatments before moving on to collect the measurements to remove time as a factor. AdFisher runs all the agents on the same machine to prevent differences based on location, IP address, operating system, or other machine specific differences between agents.

Next, we detail the particular treatments, measurements, and test statistics that we have implemented in AdFisher. We also discuss how AdFisher aids an experimenter in understanding the results.

3.5.1 Treatments

A treatment specifies what actions are to be performed by a browser agent. AdFisher automatically applies treatments assigned to each agent. Typically, these treatments involve invoking the Selenium WebDriver to make the agent interact with webpages.

AdFisher makes it easy to carry out common treatments by providing ready-made implementations. The simplest stock treatments we provide set interests, gender, and age range in Google's Ad Settings. Another stock treatment is to visit a list of webpages stored on a file.

To make it easy to see whether websites associated with a particular interest causes a change in behavior, we have provided the ability to create lists of webpages associated with a category on Alexa. For each category, Alexa tracks the top websites sorted according to their traffic rank measure (a combination of the number of users and page views) [6]. The experimenter can use AdFisher to download the URLs of the top webpages Alexa associates with an interest. By default, it downloads the top 100 URLs. A treatment can then specify that agents visit this list of websites. While these treatments do not correspond directly to having such an interest, it allows us to study how Google responds to people visiting webpages associated with those interests.

Often in our experiments, we compared the effects of a certain treatment applied to the experimental group against the *null treatment* applied to the control group. Under the null treatment, agents do nothing while agents under a different treatment complete their respective treatment phase.

3.5.2 Measurements

AdFisher can currently measure the values set in Google's Ad Settings page and the ads shown to the agents after the treatments. It comes with stock functionality for collecting and analyzing text ads. Experimenters can add methods for image, video, and flash ads.

To find a reasonable website for ad collection, we looked to news sites since they generally show many ads. Among the top 20 news websites on `alexa.com`, only five displayed text ads served by Google: the Guardian (`theguardian.com/us`), the Times of India (`timesofindia.indiatimes.com`), BBC (`bbc.com/news`), Reuters (`reuters.com/news/us`) and Bloomberg (`bloomberg.com`). AdFisher comes with built-in functionality to collect ads from any of these websites. One can also specify for how many reloads ads are to collected (default 10), or how long to wait between successive reloads (default 5s). For each page reload, AdFisher parses the page to find the ads shown by Google and stores the ads. The experimenter can add parsers to collect ads from other websites.

We run most of our experiments on Times of India as it serves the most (five) text ads per page reload. We repeat some experiments on the Guardian (three ads per reload) to demonstrate that our results are not specific to one site.

3.5.3 Classification

While the experimenter can provide AdFisher with a test statistic to use on the collected data, AdFisher is also capable of automatically selecting a test statistic using machine learning. It splits the entire data set into training and testing subsets, and examines a training subset of the collected measurements to select a classifier that distinguishes between the measurements taken from each group. From the point of view of machine learning, the set of ads collected by an agent corresponds to an *instance* of the concept the classifier is attempting to learn.

Machine learning algorithms operate over sets of *features*. AdFisher has functions for converting the text ads seen by an agent into three different feature sets. The *URL feature set* consists of the URLs displayed by the ads (or occasionally some other text if the ad displays it where URLs normally go). Under this feature set, the feature vector representing an agent's data has a value of n in the ith entry iff the agent received n ads that display the ith URL where the order is fixed but arbitrary.

The *URL+Title feature set* looks at both the displayed URL and the title of the ad jointly. It represents an agent's data as a vector where the *i*th entry is *n* iff the agent received *n* ads containing the *i*th pair of a URL and title.

The third feature set AdFisher has implemented is the *word feature set*. This set is based on word stems, the main part of the word with suffixes such as "ed" or "ing" removed in a manner similar to the work of Balebako et al. [11]. Each word stem that appeared in an ad is assigned a unique id. The *i*th entry in the feature vector is the number of times that words with the *i*th stem appeared in the agent's ads.

We explored a variety of classification algorithms provided by the scikit-learn library [115]. We found that logistic regression with an L2 penalty over the URL+title feature set consistently performed well compared to the others. At its core, logistic regression predicts a class given a feature vector by multiplying each of the entries of the vector by its own weighting coefficient (e.g., [16]). It then takes a the sum of all these products. If the sum is positive, it predicts one class; if negative, it predicts the other.

While using logistic regression, the training stage consists of selecting the coefficients assigned to each feature to predict the training data. Selecting coefficients requires balancing the training-accuracy of the model with avoiding overfitting the data with an overly complex model. We apply 10-fold cross-validation on the training data to select the regularization parameter of the logistic regression classifier. By default, AdFisher splits the data into training and test sets by using the last 10% of the data collected for testing.

3.5.4 Explanations

To explain how the learned classifier distinguished between the groups, we explored several methods. We found the most informative to be the model produced by the classifier itself. Recall that logistic regression weighs the various features of the instances with coefficients reflecting how predictive they are of each group. Thus, with the URL+title feature set, examining the features with the most extreme coefficients identifies the URL+title pair most useful to predict the group to which agents receiving an ad with that URL+title belongs.

We also explored using simple metrics for providing explanations, like ads with the highest frequency in each group. However, some generic ads gets served in huge numbers to both groups. We also looked at the proportion of times an ad was served to agents in one group to the total number of times observed by all groups. However, this did not provide much insight since the proportion typically reached its maximum value of 1.0 from ads that only appeared once. Another choice we explored was to compute the difference in the number of times an ad appears between the groups. However, this metric is also highly influenced by how common the ad is across all groups.

3.6 Experiments

In this section, we discuss experiments that we carried out using AdFisher. In total, we ran 21 experiments, each of which created its own testing data sets using independent random assignments of treatments to agents. We analyze each test data set only once and report the results of each experiment separately. Thus, we do not test multiple hypotheses on any of our test data sets ensuring that the probability of false positives (p-value) are independent with the exception of our analyses for ad choice. In that case, we apply a Bonferroni correction.

Each experiment examines one of the properties of interest from Table 3.1. We found violations of nondiscrimination and data transparency and cases of compliance with effectful and ad choice. Since these summaries each depend upon more than one experiment, they are the composite of multiple hypotheses. To prevent false positives for these summaries, for each property, we report p-values adjusted by the number of experiments used to explore that property. We use the Holm-Bonferroni method for our adjustments, which is uniformly more powerful than the commonly used Bonferroni correction [63]. This method orders the component hypotheses by their unadjusted p-values applying a different correction to each until reaching a hypothesis whose adjusted value is too large to reject. This hypothesis and all remaining hypotheses are rejected regardless of their p-values. Appendix B.3 provides details.

Table B.1 in Appendix B.1 summarizes our findings.

3.6.1 Nondiscrimination

We use AdFisher to demonstrate a violation in the nondiscrimination property. If AdFisher finds a statistically significant difference in how Google treats two experimental groups, one consisting of members having a protected attribute and one whose members do not, then the experimenter has strong evidence that Google discriminates on that attribute. In particular, we use AdFisher's ability to automatically select a test statistic to check for possible differences to test the null hypothesis that the two experimental groups have no differences in the ads they receive.

As mentioned before, it is difficult to send a clear signal about any attribute by visiting related webpages since they may have content related to other attributes. The only way to send a clear signal is via Ad Settings. Thus, we focus on attributes that can be set on the Ad Settings page. In a series of experiments, we set the gender of one group to female and the other to male. In one of the experiments, the agents went straight to collecting ads; in the others, they simulated an interest in jobs. In all but one experiment, they collected ads from the Times of India (TOI); in the exception, they collected ads from the Guardian. In one experiment, they also visited the top 10 websites for the U.S. according to `alexa.com` to fill out their interests.[1] Table B.2 in Appendix B.1 summarizes results from these experiments.

AdFisher found a statistically significant difference in the ads for male and female agents that simulated an interest in jobs in May, 2014. It also found evidence of discrimination in the nature of the effect. In particular, it found that females received fewer instances of an ad encouraging the taking of high paying jobs than males. AdFisher did not find any statistically significant differences among the agents that did not visit the job-related pages or those operating in July, 2014. We detail the experiment finding a violation before discussing why we think the other experiments did not result in significant results.

Gender and Jobs. In this experiment, we examine how changing the gender demographic on Google Ad Settings affects the ads served and interests inferred for agents browsing employment related websites. We set up AdFisher to have the agents in one group visit the Google Ad Settings page and set the gender bit to female while agents in the other group set theirs to male. All the agents then visited the top 100 websites listed under the Employment category of Alexa.[2] The agents then collect ads from Times of India.

AdFisher ran 100 blocks of 10 agents each. (We used blocks of size 10 in all our experiments.) AdFisher used the ads of 900 agents (450 from each group) for training a classifier using the URL+title feature set, and used the remaining 100 agents' ads for testing. The learned classifier attained a test-accuracy of 93%, suggesting that Google did in fact treat the genders differently. To test whether this response was statistically significant, AdFisher computed a p-value by running the permutation test on a million randomly selected block-respecting permutations of the data. The significance test yielded an adjusted p-value of < 0.00005.

We then examined the model learned by AdFisher to explain the nature of the difference. Table B.3 shows the five URL+title pairs that the model identifies as the strongest indicators of being from the female or male group. How ads for identifying the two groups differ is concerning. The two URL+title pairs with the highest coefficients for indicating a male were for a career coaching service for "$200k+" executive positions. Google showed the ads 1852 times to the male group but just 318 times to the female group. The top two URL+title pairs for the female group was for a generic job posting service and for an auto dealer.

The found discrimination in this experiment was predominately from a pair of job-related ads for the same service making the finding highly sensitive to changes in the serving of these ads. A closer examination of the ads from the same experimental setup ran in July, 2014, showed that the frequency of these ads reduced from 2170 to just 48, with one of the ads completely disappearing. These 48 ads

[1] http://www.alexa.com/topsites/countries/US

[2] http://www.alexa.com/topsites/category/Top/Business/Employment

were only shown to males, continuing the pattern of discrimination. This pattern was recognized by the machine learning algorithm, which selected the ad as the second most useful for identifying males. However, they were too infrequent to establish statistical significance. A longer running experiment with more blocks might have succeeded.

3.6.2 Transparency

AdFisher can demonstrate violations of individual data use transparency. AdFisher tests the null hypothesis that two groups of agents with the same ad settings receives ads from the same distribution despite being subjected to different experimental treatments. Rejecting the null hypothesis implies that some difference exists in the ads that is not documented by the ad settings.

In particular, we ran a series of experiments to examine how much transparency Google's Ad Settings provided. We checked whether visiting webpages associated with some interest could cause a change in the ads shown that is not reflected in the settings.

We ran such experiments for five interests: substance abuse, disabilities, infertility[3], mental disorders[4], and adult websites[5]. Results from statistical analysis of these experiments are shown in Table B.4 of Appendix B.1.

We examined the interests found in the settings for the two cases where we found a statistically significant difference in ads, substance abuse and disability. We found that settings did not change at all for substance abuse and changed in an unexpected manner for disabilities. Thus, we detail these two experiments below.

Substance Abuse. We were interested in whether Google's outputs would change in response to visiting webpages associated with substance abuse, a highly sensitive topic. Thus, we ran an experiment in which the experimental group visited such websites while the control group idled. Then, we collected the Ad Settings and the Google ads shown to the agents at the Times of India. For the webpages associated with substance abuse, we used the top 100 websites on the Alexa list for substance abuse[6].

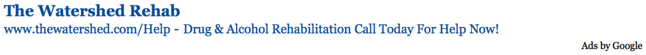
AdFisher ran 100 blocks of 10 agents each. At the end of visiting the webpages associated with substance abuse, none of the 500 agents in the experimental group had interests listed on their Ad Settings pages. (None of the agents in the control group did either since the settings start out empty.) If one expects

[3] http://www.alexa.com/topsites/category/Top/Health/Reproductive_Health/Infertility

[4] http://www.alexa.com/topsites/category/Top/Health/Mental_Health/Disorders

[5] http://www.alexa.com/topsites/category/Top/Adult

[6] http://www.alexa.com/topsites/category/Top/Health/Addictions/Substance_Abuse



The Watershed Rehab
www.thewatershed.com/Help - Drug & Alcohol Rehabilitation Call Today For Help Now!

Ads by Google

Figure 3.4: Screenshot of an ad with the top URL+title for identifying agents that visited webpages associated with substance abuse

the Ad Settings page to reflect all learned inferences, then he would not anticipate ads relevant to those website visits given the lack of interests listed.

However, the ads collected from the Times of India told a different story. The learned classifier attained a test-accuracy of 81%, suggesting that Google did in fact respond to the page visits. Indeed, using the permutation test, AdFisher found an adjusted p-value of < 0.00005. Thus, we conclude that the differences are statistically significant: Google's ads changed in response to visiting the webpages associated with substance abuse. Despite this change being significant, the Ad Settings pages provided no hint of its existence: the transparency tool is opaque!

We looked at the URL+title pairs with the highest coefficients for identifying the experimental group that visited the websites related to substance abuse. Table B.5 provides information on coefficients and URL+titles learned. The three highest were for "Watershed Rehab". The top two had URLs for this drug and alcohol rehab center. The third lacked a URL and had other text in its place. Figure 3.4 shows one of Watershed's ads. The experimental group saw these ads a total of 3309 times (16% of the ads); the control group never saw any of them nor contained any ads with the word "rehab" or "rehabilitation". None of the top five URL+title pairs for identifying the control group had any discernible relationship with rehab or substance abuse.

These results remain robust across variations on this design with statistical significance in three variations. For example, two of these ads remain the top two ads for identifying the agents that visited the substance abuse websites in July using ads collected from the Guardian.

One possible reason why Google served Watershed's ads could be *remarketing*, a marketing strategy that encourages users to return to previously visited websites [51]. The website `thewatershed.com` features among the top 100 websites about substance-abuse on Alexa, and agents visiting that site may be served Watershed's ads as part of remarketing. However, these users cannot see any changes on Google Ad Settings despite Google having learnt some characteristic (visited `thewatershed.com`) about them and serving ads relevant to that characteristic.

Disabilities. This experiment was nearly identical in setup but used websites related to disabilities instead of substance abuse. We used the top 100 websites on Alexa on the topic.[7]

[7] `http://www.alexa.com/topsites/category/Top/Society/Disabled`

For this experiment, AdFisher found a classifier with a test-accuracy of 75%. It found a statistically significant difference with an adjusted p-value of less than 0.00005.

Looking at the top ads for identifying agents that visited the webpages associated with disabilities, we see that the top two ads have the URL www.abilitiesexpo.com and the titles "Mobility Lifter" and "Standing Wheelchairs". They were shown a total of 1076 times to the experimental group but never to the control group. (See Table B.6.)

This time, Google did change the settings in response to the agents visiting the websites. None of them are directly related to disabilities suggesting that Google might have focused on other aspects of the visited pages. Once again, we believe that the top ads were served due to remarketing, as abilitiesexpo.com was among the top 100 websites related to disabilities.

3.6.3 Effectful Choice

We tested whether making changes to Ad Settings has an effect on the ads seen, thereby giving the users a degree of choice over the ads. In particular, AdFisher tests the null hypothesis that changing some ad setting has no effect on the ads.

First, we tested whether opting out of tracking actually had an effect by comparing the ads shown to agents that opted out after visiting car-related websites to ads from those that did not opt out. We found a statistically significant difference.

We also tested whether removing interests from the settings page actually had an effect. We set AdFisher to have both groups of agents simulate some interest. AdFisher then had the agents in one of the groups remove interests from Google's Ad Settings related to the induced interest. We found statistically significant differences between the ads both groups collected from the Times of India for two induced interests: online dating and weight loss. We describe one in detail below.

Online Dating. We simulated an interest in online dating by visiting the website www.midsummerseve.com/, a website we choose since it sets Google's ad setting for "Dating & Personals" (this site no longer affects the setting). AdFisher then had just the agents in the experimental group remove the interest "Dating & Personals" (the only one containing the keyword "dating"). All the agents then collected ads from the Times of India.

AdFisher found statistically significant differences between the groups with a classifier accuracy of 74% and an adjusted p-value of < 0.00003. Furthermore, the effect appears related to the interests removed. The top ad for identifying agents that kept the romantic interests has the title "Are You Single?" and the second ad's title is "Why can't I find a date?". None of the top five for the control group that removed the interests were related to dating (Table B.8). Thus, the ad settings appear to actually give users the ability to avoid ads they might dislike or find embarrassing. In the next set of experiments, we explicitly test for this ability.

We repeated this experiment in July, 2014, using dating websites relationshipsurgery.com and datemypet.com. We continued to see an effect on Ad Settings, but did not find statistically significant differences.

3.6.4 Ad Choice

Whereas the other experiments tested merely for the presence of an effect, testing for ad choice requires determining whether the effect is an increase or decrease in the number of relevant ads seen. Fortunately, since AdFisher uses a one-sided permutation test, it tests for either an increase or a decrease, but not for both simultaneously, making it usable for this purpose. In particular, after removing an interest, we check for a decrease to test for compliance using the null hypothesis that either no change or an increase occurred, since rejecting this hypothesis would imply that a decrease in the number of related ads occurred. To check for a violation, we test for the null hypothesis that either no change or a decrease occurred. Due to testing two hypotheses, we use an adjustment to the p-value cutoff considered significant to avoid finding significant results simply from testing multiple hypotheses. In particular, we use the standard Bonferroni correction, which calls for multiplying the p-value by 2 (e.g., [1]).

We ran three experiments checking for ad choice. The experiments followed the same setup as the effectful choice ones, but this time we used all the blocks for testing a given test statistic. The test statistic counted the number of ads containing keywords. In the first, we again test online dating using relationshipsurgery.com and datemypet.com. In particular, we found that removing online dating resulted in a significant decrease (p-value adjusted for all six experiments: 0.0456) in the number of ads containing related keywords (from 109 to 34). We detail the inconclusive results for weight loss below.

Weight Loss. We induced an interest in weight loss by visiting dietingsucks.blogspot.com. Afterwards, the agents in the experimental group removed the interests "Fitness" and "Fitness Equipment and Accessories", the only ones related to weight loss. We then used a test statistic that counted the number of ads containing the keyword "fitness". Interestingly, the test statistic was higher on the group with the interests removed, although not to a statistically significant degree. We repeated the process with a longer keyword list and found that removing interests decreased test statistic this time, but also not to a statistically significant degree.

3.7 Discussion and Conclusion

Using AdFisher, we conducted 21 experiments using 17,370 agents that collected over 600,000 ads. Our experiments found instances of discrimination, opacity, and choice in targeted ads of Google. Discrimination, is at some level, inherent to profiling: the point of profiling is to treat some people differently. While customization can be helpful, we highlight a case where the customization appears inappropriate taking on the negative connotations of discrimination. In particular, we found that males were shown ads encouraging the seeking of coaching services for high paying jobs more than females (§3.6.1).

We do not, however, claim that any laws or policies were broken. Indeed, Google's policies allow it to serve different ads based on gender. Furthermore, we cannot determine whether Google, the advertiser, or complex interactions among them and others caused the discrimination (§3.4.5). Even if we could, the discrimination might have resulted unintentionally from algorithms optimizing click-through rates or other metrics free of bigotry. Given the pervasive structural nature of gender discrimination in society at large, blaming one party may ignore context and correlations that make avoiding such discrimination difficult. More generally, we believe that no scientific study can demonstrate discrimination in the sense of *unjust discrimination* since science cannot demonstrate normative statements (e.g., [65])

Nevertheless, we are comfortable describing the results as "discrimination". From a strictly scientific view point, we have shown discrimination in the non-normative sense of the word. Personally, we also believe the results show discrimination in the normative sense of the word. Male candidates getting more encouragement to seek coaching services for high-paying jobs could further the current gender pay gap (e.g., [117]). Thus, we do not see the found discrimination in our vision of a just society even if we are incapable of blaming any particular parties for this outcome.

Furthermore, we know of no justification for such customization of the ads in question. Indeed, our concern about this outcome does not depend upon how the ads were selected. Even if this decision was made solely for economic reasons, it would continue to be discrimination [153]. In particular, we would remain concerned if the cause of the discrimination was an algorithm ran by Google and/or the advertiser automatically determining that males are more likely than females to click on the ads in question. The amoral status of an algorithm does not negate its effects on society.

However, we also recognize the possibility that no party is at fault and such unjust effects may be inadvertent and difficult to prevent. We encourage research developing tools that ad networks and advertisers can use to prevent such unacceptable outcomes (e.g., [156]).

Opacity occurs when a tool for providing transparency into how ads are selected and the profile kept on a person actually fails to provide such transparency. Our experiment on substance abuse showed an extreme case in which the tool failed to show any profiling but the ad distributions were significantly different in response to behavior (§3.6.2). In particular, our experiment achieved an adjusted p-value of < 0.00005, which is 1000 times more significant than the standard 0.05 cutoff for statistical significance. This experiment remained robust to variations showing a pattern of such opacity.

Ideally, tools, such as Ad Settings, would provide a complete representation of the profile kept on a person, or at least the portion of the profile that is used to select ads shown to the person. Two people with identical profiles might continue to receive different ads due to other factors affecting the choice of ads such as A/B testing or the time of day. However, systematic differences between ads shown at the same time and in the same context, such as those we found, would not exist for such pairs of people.

In our experiments testing transparency, we suspect that Google served the top ads as part of remarketing, but our blackbox experiments do not determine whether this is the case. While such remarketing may appear less concerning than Google inferring a substance abuse issue about a person, its highly targeted nature is worrisome particularly in settings with shared computers or shoulder surfing. There is a need for a more inclusive transparency-choice mechanism which encompasses remarketed ads as well. Additionally, Google states that "we prohibit advertisers from remarketing based on sensitive information, such as health information" [51]. Although Google does not specify what they consider to be "health information", we view the ads as in violation of Google's policy, thereby raising the question of how Google should enforce its policies.

Lastly, we found that Google Ad Settings does provide the user with a degree of choice about the ads shown. In this aspect, the transparency-choice tool operated as we expected.

Altogether, experiments with AdFisher may have cost advertisers a small sum of money. AdFisher never clicked on any ads, but its experiments may have caused per-impression fees, which run about $0.00069 per impression [111]. In the billion dollar ad industry, its total effect was about $400 distributed over advertisers whose ads we collected.

Our tool, AdFisher, makes it easy to run additional experiments exploring the relations between Google's ads and settings. It can be extended to study other systems. It's design ensures that it can run and analyze large scale experiments to find subtle differences. It automatically finds differences between large data sets produced by different groups of agents and explains the nature of those differences. By completely automating the data analysis, we ensure that an appropriate statistical analysis determines whether these differences are statistically significant and makes sound conclusions.

Chapter 4

An Evaluation of Privacy Enhancing Technologies against Browser Fingerprinting

4.1 Introduction

In this chapter, we consider Privacy Enhancing Techologies that aim to protect consumers from fingerprinting, an advanced form of tracking, as a possible defense against data collection. Given the numerous PETs available, it is unclear which PET consumers should adopt. Since many PETs are black boxes it is difficult to evaluate how effectively they hinder data collection. After demonstrating shortcomings of purely observational and purely experimental evaluation methods, we propose and apply a novel combination of these methods, offering the best of both worlds. We use experiments to find causal effects of a PET on various fingerprintable attributes and create a model of a PET. We then apply this model on a pre-existing data set of real-world fingerprints to produce a counterfactual PET-modified data set. By comparing the abilities of a data aggregator in tracking the original fingerprints with that of the PET-modified fingerprints, we evaluate the effectiveness of the PET. We apply our evaluations to 26 different PETs, 15 of which explicit claim to protect against fingerprinting. We find that 13 of the 15 anti-fingerprinting PETs do not provide much additional protection than using no PET at all. We find the Tor Browser Bundle to be the most effective among the ones we evaluate.

4.1.1 Motivation and Problem

Online data aggregators track consumer activities on the Internet to build behavioral profiles. Traditional forms of tracking use stateful mechanisms, where the tracker places an identifier (e.g., a HTTP cookie) with a unique value on the consumer's browser or computer. When the consumer visits webpages where the tracker has a presence, their browser automatically sends the identifier value to the tracker. This allows the tracker to link these visits to the same consumer. Two properties make an identifier good for tracking purposes: *uniqueness* and *predictability*.[1] Uniqueness requires the identifier value is sufficiently unique among the consumers to be tracked, whereas predictability requires the identifier values are predictable for a consumer across webpage visits.

Increasing awareness about stateful tracking has led consumers to take precautions against them (e.g., by blocking or clearing cookies). This has spurred the growth of stateless tracking mechanisms, also known as browser fingerprinting. A stateless tracker extracts fingerprints from a user's browser as a collection of several attributes of the browser, operating system, and hardware, typically accessed through Javascript APIs. Fingerprints collected on websites like `panopticlick.eff.org` and `amiunique.org/fp` demonstrate that they are sufficiently unique and predictable for tracking purposes [34,83]. The list of attributes that can be used in fingerprints has been rapidly increasing [4,5,21,38,43,102,132]. Studies have also uncovered fingerprinting code on popular webpages [4,5,38].

Some Privacy Enhancing Technologies (PETs) aim to protect consumers against fingerprinting by spoofing the values of attributes. For each attribute, they can either (1) *standardize* it, so that all PET users reveal the same or one of a small set of attribute values, thereby affecting the uniqueness of fingerprints, or (2) *vary*[2] the attribute, so that fingerprints of all PET users vary across webpage visits, thereby affecting their predictability and uniqueness. Our goal is to develop a methodology for evaluating the effectiveness of such PETs by comparing the predictability and uniqueness of PET-modified fingerprints with those of the original fingerprints.

Depending on the goals, the PET evaluation could depend on the context in which the PET is used, including features of other users and non-users, or be a more theoretical assessment of its potential, untied to the vagaries of today. For example, if the goal of evaluation is to determine which PET to use today, one would want to know how many other users of the PET there are since they will form the anonymity set – the group of other users one will blend in with. If instead the goal is to determine which PET to fund for further development, the user numbers of today may matter less than the technical or theoretical capabilities of the PET. Given that no one PET evaluation can match all goals, we will explore points in the space of possible evaluations.

[1] Prior work has used the term stability instead of predictability [34,81,108]. We see stability as a form of predictability where identifier values remain identical.

[2] Tor developers use the term *randomization* [116]. We see randomization as an instance of variation.

4.1.2 Overview of Methods

First, we consider a highly context-dependent, observational method. Websites like panopticlick.eff.org and amiunique.org/fp obtain large sets of real-world fingerprints, revealing which are the most trackable (i.e., unique and predictable). In principle, these data sets can be studied to evaluate a PET by selecting the fingerprints generated by users of that PET and, for each such fingerprint, checking how trackable they are compared to other fingerprints in the data set. In practice, such observational data sets may contain too few users of a PET, particularly for new ones, to evaluate it. Furthermore, in some cases, it may be difficult to determine which fingerprints correspond to which PETs. Thus, while such data sets can provide valuable information about the fingerprints appearing in the wild, they are of limited use in the evaluation of PETs.

Second, we consider a more theoretical, experimental analysis that instead looks at a PET's ability to *mask* attributes. This method runs browsers with and without a PET installed to determine which attributes the PET masks, either by standardizing or varying its value. For this purpose, we develop an experimental framework, PETInspector, which has three components: a *fingerprinting server*, which collects fingerprints from visitors, a *client simulator*, which simulates consumers and drives them to FS with and without PETs, and an *analysis engine*, which compares fingerprints across clients to characterize PET behaviors. This tool can be applied to new PETs that currently lack a user base, and, by being under the evaluator's control, determining which fingerprints correspond to which PET is trivial. Unlike static analyses, this experimental method does not require access to the source code of PETs. However, it does not tell us which attributes are the most important to mask.

Third, we develop a novel hybrid method combining data from the aforementioned observational and experimental methods to enable the evaluation PETs with low or no usage within the context of the browsers used today but without source code. It contextualizes the mask model produced by the experimental evaluation by applying it to an observational data set of real-world fingerprints to produce a counterfactual data set representing what the browsers would look like to trackers had everyone used the PET. We determine the trackability of the original data set and the PET-modified data set in a manner similar to the first purely observational method. By comparing the trackabilities on the two data sets, we can evaluate the effectiveness of the PET. The hybrid method is outlined in Figure 4.1. By parametrically leveraging data from ongoing, large-scale measurement studies, our results may be updated for the ever changing landscape of browsers with little additional work.

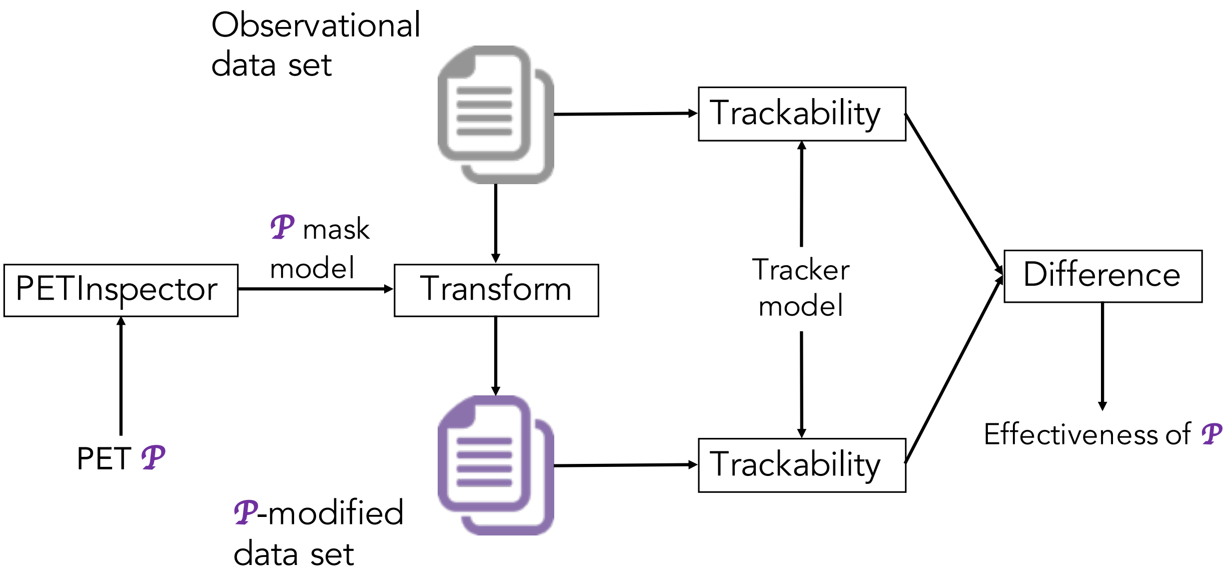


Figure 4.1: Hybrid method for PET evaluation

4.1.3 Overview of Results

Using PETInspector, we characterize how 26 different PETs, 15 explicitly claiming to protect against fingerprinting and 11 other popular ones, mask 53 different attributes. Our experimental evaluations uncover undocumented behaviors and inconsistencies in how some PETs modify different fingerprintable attributes:

- The Brave browser spoofs the `User-Agent` to appear like the Chrome browser. However, it modifies the `Accept-Language` header, `language` and `plugins` differently than Chrome. To a tracker, this can make Brave users stand out from other Chrome users. We have raised the issue with Brave developers[3] and have received comments from the developers acknowledging the issue.
- Privacy Badger and Firefox's tracking protection configuration implement Do Not Track differently. While both send the `Dnt` header, only the latter sets the `doNotTrack` variable in JavaScript's navigator object. As a result, web-services which only use JavaScript to detect the Do Not Track choice will not be able to do so for Privacy Badger users. We have raised this issue with Privacy Badger developers[4] and have received comments from the developers acknowledging the issue.
- HideMyFootprint randomizes the `User-Agent` header, while not modifying the `platform`. This leads to inconsistencies like the `User-Agent` containing *Windows NT 10.0* on a *Linux x86_64* `platform`. Moreover it sends an additional `Pragma` header, which can make users easily distinguishable.

[3] https://github.com/brave/muon/issues/429 and https://github.com/brave/browser-laptop/issues/12479
[4] https://github.com/EFForg/privacybadger/issues/1835

- While the Tor Browser is able to conceal the operating system by spoofing attributes like the `User-Agent` and `cpu class`, the `javascript fonts` revealed by different browsing platforms can reveal that information.

For the hybrid method, we use a pre-existing data set of over 25,000 real world fingerprints collected on the website `amiunique.org`.[5] Due to constraints on how the client simulator can simulate consumers, experiments provide a complete characterization of a PET's behavior only for 20 of the 53 attributes. Of these, only 12 appear in the `amiunique.org` data set. For these 12 attributes, we generate a set of PET-modified fingerprints from the original fingerprints and measure effectiveness of 15 anti-fingerprinting PETs.

Our hybrid methods finds that even with just 12 attributes, 13 of the 15 anti-fingerprinting PETs do not provide much additional protection than using no PET at all, decreasing the entropy revealed from about 13 bits to 11 bits. It places the Tor Browser Bundle (TBB) as the most effective PET, which reveals under 3 bits of entropy. The only source of entropy for TBB fingerprints is the revealed screen resolution. TBB reveals partial information about the screen resolution of its users using a spoofing strategy which depends on the true resolution for usability reasons. We explore a space of alternate spoofing strategies and find some which utilize more pixels on average for browsing than TBB, but are just as effective.

4.1.4 Chapter Contributions

We make the following main contributions in this chapter:

- We propose a purely observational method for evaluating PETs and point out hurdles in its application (Section 4.4).

- We develop an experimental framework (PETInspector) to verify how 26 PETs spoof 53 different attributes. In addition to obtaining a more complete picture of PETs' behaviors, uncovering some inconsistencies and peculiarities (Section 4.5).

- We develop a hybrid method for evaluating PETs from observational data set of real-world fingerprints and apply it to evaluate 15 anti-fingerprinting PETs. We find TBB to be the most effective PET among the ones we evaluate (Section 4.6).

- We explore a space of alternate spoofing strategies for screen resolution by TBB and uncover some which have higher screen utilization than TBB, but are just as effective. (Section 4.7).

[5] The data set was graciously provided to us by Pierre Laperdrix, one of the creators of `amiunique.org`.

4.2 Prior Work

Prior work finds that various attributes are trackable by measuring the uniqueness and predictability of fingerprints collected from real world browsing platforms [34, 83, 151]. However, few studies evaluate the effectiveness of PETs against fingerprinting.

Many prior studies have focused on PETs which use blacklists to block known tracking domains and scripts. Since these blocking PETs try to prevent the consumer's browser from interacting with trackers, metrics suggestive of successful interactions (e.g., third-party requests sent, cookies placed, etc.) are good indicators of PET effectiveness. These studies evaluate a blocking PET by comparing these metrics between browsers with and without the PET when visiting popular websites [38, 62, 66, 75, 76, 91, 97, 120]. FPGuard takes a blacklisting approach to protect against fingerprinting: it uses various heuristics to identify fingerprinting domains and blocks them [42]. However, many PETs protect against fingerprinting by spoofing browser, operating system and hardware characteristics, without blocking specific domains and scripts. For example, PETs like the Tor Browser standardize various attribute values [116], whereas others like PriVaricator [108], FP-Block [138], Blink [82], and FPRandom [81] vary them. Metrics used for evaluating blocking PETs would not be able to meaningfully evaluate these PETs. Some studies have evaluated attribute varying PETs by observing variations in fingerprints when using these PETs (e.g., [81, 108]). While variations are good indicators of effectiveness against simple trackers, they aren't so against more capable yet practical trackers which can detect variations introduced by PETs.

4.3 Trackers and PETs

When a user visits a webpage, trackers can have the user's browser execute code that requests information about the user's browsing platform, including their hardware, operating system, and the browser itself.[6] The leftmost column of Table 4.2 provides a list of 53 attributes known to be good candidates for fingerprinting. The tracker can combine multiple attributes $a_1, \ldots, a_n$ to compute a *fingerprint* $id(b) = \langle a_1(b), \ldots, a_n(b) \rangle$ of the browsing platform b where $a_i(b)$ represents the value of attribute a_i for the platform b. A tracker can use fingerprints to identify browsing platforms visiting two websites as being the same one. The more unique the fingerprint is for each user, fewer false matches the tracker will produce linking two different users by accident. The more predictable (ideally, unchanging) the fingerprint is as a user goes from website to website, the fewer matches the tracker will miss.

To protect themselves from fingerprinting, consumers can install PETs on their browsing platform,

[6] We do not consider more sophisticated cross-device [158] and cross-browser [21] trackers, which aim to link together different browsing platforms originating from the same consumer.

which can decrease the uniqueness or predictability of the platform's fingerprints. Upon installing a PET $\mathcal{P}$, the consumer's browsing platform b is modified to $\mathcal{P}(b)$. As a result, the tracker now interacts with $\mathcal{P}(b)$ and extracts the fingerprint $id(\mathcal{P}(b))$.

In this study, we look at three types of PETs:

I. **Attribute standardizing.** These PETs reveal one (full standardization) or one of a small set of possible values (partial standardization) for an attribute. Full standardization makes all PET users appear identical, whereas partial standardization makes them appear from a small number of groups, with respect to that attribute. A PET may choose partial over full standardization if spoofing the attribute value has usability implications.

II. **Attribute varying.** These PETs vary the value of an attribute so that the values of all PET users vary across browsing activities. Such variations may affect both the predictability and uniqueness of the revealed attribute. Laperdrix et al. [81] show that variation PETs can vary attributes in a manner that reduces the usability impact.

III. **Interaction blocking.** These PETs block some or all interactions between the browsing platform and trackers. They rely on a blacklist (e.g., EasyPrivacy) to block interactions matching known tracking patterns. Trackers interacting with browsing platforms with these PETs receive an error message instead of the true fingerprints.

We are primarily interested in evaluating and comparing PETs that modify the attribute values either by standardizing (I) or varying (II) their values. In some places, we comment on PETs that block interactions with known trackers (III), since they are popular and have been the subject of past evaluation studies. However, we do not directly compare them to the other PETs since they do not purport to modify any attributes explicitly, and their quality depends upon the quality of their blacklists, necessitating a different form of evaluation.

We leave completely out of scope PETs that protect against fingerprinting by blocking scripts (e.g., NoScript [67], ScriptSafe [7]) since these have a considerable impact on usability [66]. We also leave out PETs like Noiszy (`noiszy.com`), Internet Noise (`makeinternetnoise.com`), and AdNauseum (`adnauseam.io`) that do not attempt to prevent tracking but rather to make it pointless by injecting noise into the user's history with fake clicks and website visits.

In this paper, we consider a total of 26 PETs — 23 of which are extensions for Chrome and Firefox[7] (the two most popular desktop browsers at the time of writing), two are full browsers, and one is a browser

[7] Extensions for Firefox are called add-ons.

Table 4.1: List of PETs we study, their abbreviation, and approach to protection. Most PETs are browser extensions, * indicates full browsers, and ** indicates browser configurations.

PET	Abbr.	Approach
Chrome PETs		
CanvasFingerprintBlock [10]	CFB	I
Privacy Extension [131]	PE	I
Brave [18]	BR*	I+III
Canvas Defender [105]	CD$_C$	II
Glove [106]	GL	II
HideMyFootprint [2]	HMF	II+III
Trace [3]	TR	II+III
Adblock Plus [41]	AP$_C$	III
Disconnect [32]	D$_C$	III
Ghostery [25]	GH$_C$	III
Privacy Badger [36]	PB$_C$	III
uBlock Origin [61]	UO$_C$	III
Firefox PETs		
Blend In [119]	BI	I
Blender [96]	BL	I
No Enum. Extensions [125]	NE	I
Stop Fingerprinting [109]	SF	I
Tor Browser Bundle [116]	TBB*	I
TotalSpoof [45]	TO	I
Canvas Defender [105]	CD$_F$	II
CanvasBlocker [73]	CA	II
Adblock Plus [41]	AP$_F$	III
Disconnect [32]	D$_F$	III
Ghostery [25]	GH$_F$	III
Privacy Badger [36]	PB$_F$	III
Tracking Protection [103]	TP**	III
uBlock Origin [61]	UO$_F$	III

configuration. Among browser extensions, 11 are for Chrome, and 12 are for Firefox.[8] 12 of the 24 extensions are pairs of 6 extensions available for both Chrome and Firefox. We assign each PET a unique abbreviation which we use in the remainder of the paper, adding an extra letter for PETs with versions for both browsers. We present the full list of PETs, their abbreviations, baseline browser and approach in Table 4.1. 15 of the 26 PETs are anti-fingerprinting and purport to either standardize or vary attribute values, while 11 others are popular but provide protection primarily by blocking interactions. Some PETs assume mixed approaches. For example, BR, HMF and TR modify some attributes in addition to blocking some types of interactions.

We go over the documentation of PETs to uncover how they purport to modify attributes. For all PETs which explicitly document masking an attribute, we place a □ in the corresponding cell in Table 4.2.

[8] Several Firefox extensions have been rendered incompatible with Firefox 57.0+ due to a transition in their add-on policies.

This approach is similar to how Torres et al. produce their comparison table [138, Table 1]. However, the documentation is not always clear about which attributes are masked. One can obtain additional clarity from the programs themselves for open-sourced PETs, but program analyses cannot be applied to proprietary PET software. As a result, the □s in Table 4.2 may not reflect the full picture of how PETs mask attributes. In Section 4.5 we demonstrate how we use our experimental method to obtain a more complete picture of the masking behavior.

4.4 Observational Evaluation of PETs

In this section we discuss a simple straw-man evaluation method. Before explaining the difficulty of applying it in practice, we discuss how it could work in principle since its components will be useful for the hybrid method we introduce later.

In principle, a highly-context dependent, completely observational method could function as follows. A server could attract all the world's people and record their fingerprints for all known attributes. The server could then compare the fingerprints produced by users of each PET to determine which are the least trackable. Websites like panopticlick.eff.org and amiunique.org/fp have set up such a fingerprinting server. When users visit these websites, a copy of their fingerprint is stored on the server. The collected fingerprinting data serves as a good starting point for the observational evaluation of PETs.

In practice, we must use a sample of the world's population and pick a concrete metric of trackability. Even after making these adjustments, we are still faced with the problem of determining which users run which PETs and obtain a representative sample of PET users. Working on a slice of the fingerprinting data collected on amiunique, even after making adjustments required to apply the observational method, these obstacles make us abandon this method.

4.4.1 Sampling

We cannot, in practice, see all the world's browsing platforms and instead must work with a sample. The quality of the metrics computed from the sample depends upon both the nature of the metric and the sample. For example, a random sample will provide a reasonable estimation of the entropy (e.g., [112]). However, estimating the proportion of users in small anonymity sets from even a random sample proves difficult since the length of the tail of the distribution may be unclear from a random sample.

Furthermore, in practice, we must approximate truly random samples of browser platforms from available data sets since we cannot force all users to participate. We do so by using a convenience sample provided to us by the amiunique website, a website performing an evaluation approach along the lines of the one discussed here but without considering PET usage. This sample comprises of $25,984$ real-world fingerprints collected over a period of 30 days (10/02/2017 to 11/02/2017). Each fingerprint comprises of 32 different attributes (full list in Table C.7 in Appendix C.1). Determining the representativeness of this sample is difficult since it can only be compared to other possibly unrepresentative samples. We compare our sample's distributions to GlobalStat's for desktop users [133]. We find that our sample has a higher proportion of Firefox users (42% vs. 12%) and of Linux users (19% vs. 1%). One explanation for the greater shares of the Firefox browser as well as the Linux platform is that it is more likely that perhaps people visiting browser fingerprinting websites have more technical knowledge and a preference for open-source technologies. Table C.1 in the Appendix provides details.

4.4.2 Metrics of trackability

In our observational method of evaluation, it is not clear what we mean by *trackability*. Is a tracker that can determine with 10% certainty 90% of the time that you visited a website worse than one that can determine it with 90% certainty 10% of the time? This depends upon both the tracker's and the evaluator's goals. With this in mind, we do not argue for a single metric, but rather consider a few.

To measure trackability of the fingerprints, we look to metrics used in prior work for measuring uniqueness [34,151]. These metrics are various functions of the distribution of anonymity sets of browsing platforms. An anonymity set comprises of browsing platforms with identical fingerprints and thus are indistinguishable from each other. Thus, the smaller and numerous the anonymity sets, the higher the uniqueness.

The first metric which we use to measure uniqueness is entropy. For a set of browser platforms $\mathcal{D} = \{b_i\}_i$, such as those using a particular PET, let $\mathcal{D}[id(\cdot)]$ denote the multiset fingerprints $\{id(b_i)\}_i$ where $id(\cdot)$ is the fingerprinting mechanism. The entropy of these fingerprints is given by

$$\mathsf{entropy}(\mathcal{D}[id(\cdot)]) = -\sum_{id_k \in \mathcal{D}[id(\cdot)]} Pr[id_k] \log_2(Pr[id_k])$$

where $Pr[id_k]$ is the probability of observing the fingerprint id_k, which we estimate from the frequency of id_k in $\mathcal{D}[id(\cdot)]$. The higher the entropy, the higher the uniqueness of the fingerprints.

We also measure the proportion of users in anonymity sets of size less than 1 ($\mathsf{prop\_less1}$) and 10 ($\mathsf{prop\_less10}$). These metrics measure the proportion of browsing platforms hiding in anonymity sets of sizes less than 1 and 10. The higher $\mathsf{prop\_less1}$ is, the higher is the fraction of browsing platforms that can be uniquely identified. Similarly, higher $\mathsf{prop\_less10}$ indicates a higher fraction of browsing platforms that can be uniquely identified to a set of size at most 10. Thus, higher values of these metrics indicate higher uniqueness of the fingerprints.

We measure effectiveness of a PET $\mathcal{P}$ against fingerprinting mechanism $id(\cdot)$ in terms of a metric $\mathsf{f} \in \{\mathsf{entropy}, \mathsf{prop\_less1}, \mathsf{prop\_less10}\}$ from the data set of fingerprints $\mathcal{D}[id(\cdot)]$ as

$$\mathsf{eff}_{\mathsf{f}}(\mathcal{P}, id, \mathcal{D}_{\mathcal{P}}, \mathcal{D}_{\bar{\mathcal{P}}}) := \mathsf{f}(\mathcal{D}_{\bar{\mathcal{P}}}[id(\cdot)]) - \mathsf{f}(\mathcal{D}_{\mathcal{P}}[id(\cdot)]) \tag{4.1}$$

where $\mathcal{D}_{\mathcal{P}}$ is the subset of $\mathcal{D}$ using the PET and $\mathcal{D}_{\bar{\mathcal{P}}}$ is the rest of $\mathcal{D}$.

4.4.3 Application to data set

We cannot directly measure these metrics on the `amiunique` data set without a mapping of fingerprints to unique browsing platforms. We approximate this mapping using cookie IDs associated with each fingerprint. We approximate fingerprints with different cookie IDs as being produced by different browsing platform. While one may clear their cookies and revisit the website, we assume the fraction of such consumers to be low to significantly affect the metrics. Eckersley also uses cookies in his Panopticlick study to approximate returning visitors [34]. In the data set, $21,395$ have a cookie associated with them, of which, $18,295$ are unique.

To obtain $\mathcal{D}_{\bar{\mathcal{P}}}[id(\cdot)]$, we sanitize the data set to remove fingerprints with obvious signs of PET use, specifically those with JavaScript disabled and illegitimate screen resolutions. Additionally, we only retain fingerprints from desktop browsers (with Windows, Mac, or Linux platforms) since all PETs we study are for desktops. These sanitizations leave $17,109$ fingerprints. Finally, we separate this set into two sets of fingerprints, one from Chrome and another from Firefox browsers by looking at the `User-Agent` attribute in each fingerprint. This results in $9,493$ Chrome and $6,516$ Firefox browser fingerprints, which we use to simulate the tracker's view of the original fingerprints for evaluating Chrome and Firefox PETs respectively. We find that the original fingerprints reveal 13.002 and 12.359 bits of entropy for Chrome and Firefox browsers respectively. These and other metrics are presented in Table 4.3 corresponding to the 'no mask' row.

Obtaining $\mathcal{D}_{\mathcal{P}}[id(\cdot)]$ is considerably harder. Ideally, for our purposes, we would know for each visitor to `amiunique` which, if any PETs, the user has installed so we can obtain a sample of fingerprints generated by a PET. That would allow use to consider how the metrics are correlated with using each PET and to compute (4.1). However, the PET information is missing from the `amiunique` fingerprints. We try to guess what PET is installed based on the fingerprints themselves. For example, some fingerprints indicate whether JavaScript was turned, which we use as a way to identify PETs disabling JavaScript execution. We find that only 22 Chrome and 93 Firefox fingerprints indicate that JavaScript was blocked, which is far smaller than the set of original fingerprints for meaningfully comparing metrics of trackability. All 22 Chrome fingerprints are unique revealing 4.459 bits of entropy. The 93 Firefox fingerprints however are not all unique and reveal 4.608 bits of entropy. However, the set of fingerprints Although a total of 3913 fingerprints indicate JavaScript blocking, 3497 do not have any cookies associated with them, which drastically reduces the set of fingerprints for which we can compute trackability.

4.4.4 Limitations

Even after making adjustments for measuring trackability and obtaining a sample of fingerprints, some major limitations remain.

PET determination. Determining PET use from fingerprints not containing the information explicitly is difficult. This limitation can be overcome by a fingerprinting server designed to collect information about PET use. One approach is to ask visitors about their PETs, but users can be unaware of their own browser's configurations. In some cases, PETs have a distinctive fingerprint that gives away their use, but this would only help us with a subset of PETs. Alternatively, automated methods to detect installed browser extensions may also be used to detect PETs (e.g., [132], [126]).

PET sampling. Even with a fingerprinting server collecting PET information, driving a representative sample of real users with PETs to visit the website may be difficult, since there are few PET users. This is especially true for new and not yet popular PETs. Furthermore, users of PETs may be systematically different from non-PET-users, thereby introducing confounding factors influencing the trackability metrics. To remove or minimize the effect of these confounding factors, one may have to identify matched pairs of users, one using a PET and another not.

For these limitations, we cannot apply this evaluation method to our data set. Moreover, the PET sampling limitation may prohibit application of this method directly to collected data on fingerprinting servers in spite of being designed for PET evaluation. However, as we will see in Section 4.6, the techniques discussed above will find use in an hybrid evaluation method that avoids the PET determination and sampling problems. But first we will discuss the other evaluation method that is a prerequisite to our hybrid approach.

4.5 Experimental Evaluation of PETs

We now consider a radically different method of PET evaluation. Rather than being a highly-context specific, observational approach tied to the browsing platforms of today, it is an experimental approach conducted on artificial users. We compare the attribute values of browser platforms only differing in whether we add a PET to the platform. From these comparisons, we infer which attributes the PET is masking. We use the degree of masking by PET as an evaluation metric.

Below, we discuss this method and our experimental framework implementing it. We then explain an experiment we ran with it and the results. The results show that while one could instead look to a PET's documentation for information on which attributes it masks, the documentation sometimes provides an incomplete picture of a PET's behavior. We end with a discussion of this method's limitations.

4.5.1 Method

We develop an experimental framework, PETInspector. PETInspector comprises of (1) a *client simulator* (CS), which creates and drives experimental browsing platforms, with and without various PETs installed, to visit a server; (2) a *fingerprinting server* (FS), which collects fingerprints when the browsing platforms, driven by CS, visit it; (3) an *analysis engine* (AE), which compares fingerprints across clients to detect whether a PET varies, standardizes, or does not spoof the value of an attribute. To observe these behaviors, AE compares the value of the attribute on the browsing platform without any PET (i.e., on the baseline browser) with the value when a PET is installed.

Roughly speaking, FS corresponds to the `amiunique` server mentioned for the purely observational approach (Section 4.4) with the browsers and FS interacting to simulate fingerprinting in the wild. The new components, surrounding this simulation, produce an omniscient view with CS telling AE which fingerprints correspond to which PETs, thereby side-stepping the PET-determination problem.

Client Simulator. CS drives simulated clients using browsing platforms in different configurations to visit to FS. For each base configuration and PET, CS simulates clients with no differences other than whether the PET is installed, to allow the isolation of the PET's effects.

We choose the base configurations to exercise a wide range of attribute values in hopes of triggering a PET's masking behavior even when the masking is partial. For attributes that differ across the browsing platforms, we can detect whether a PET was standardizing them. Thus, we setup CS to exercise control over as many attributes as possible. CS simulates browsing platforms either locally on a computer or on pre-configured VirtualBox virtual machines [147] to exercise control over many of these attributes. It can also set up a virtual machine and configure it according to stated preferences. As of now, CS can configure a virtual machine to simulate different fonts, timezones, languages and screen properties.

CS installs different fonts by adding a TrueType font file (`.ttf` file) to the `.fonts` folder. Both Firefox and Chrome allow fonts from this folder to be rendered by on a webpage. To set timezones, CS uses the `timedatectl` command available by default on Linux. CS specifies the language using the `locale-gen` and by changing the LANG environment variable. Moreover, CS installs the corresponding Firefox language pack. Chrome does not have different installers for different languages, instead renders fonts based on the LANG environment variable. For screen attributes, specifically Height, Width and Depth, CS uses the display server Xvfb.

Exercising control over all attributes is difficult. Some attributes require modifications to hardware (e.g., max touch points) or operating system libraries (e.g., math attributes). Screen attributes other than Height, Width and Depth cannot be simulated using Xvfb. Moreover, we restrict CS to configuring attributes in the operating system while leaving browser settings intact. We do this to prevent reconfigurations every time a simulated platform launches a new browser instance. As a result, CS does not exercise several header attributes, storage attributes, openDB and indexedDB. CS does not configure plugins since they are gradually being phased out. Most common plugins no longer work on Firefox [104] or Chrome [24]. We do not exercise the cookies enabled and DNT enabled attributes since they conflict with CB and TP, and adBlock installed and has lied with attributes since they are aimed at detecting various PET behaviors.

After setting up a simulated browsing platform, CS drives browser instances on them using Selenium Webdriver [124] to FS. These browser instances interact with FS in a specified pattern of reloads and idling to provide insights about the modification behavior of PETs. In hopes of triggering a PET's ability to mask by varying attribute values, CS drives its browsers across various boundaries that may cause the PET to refresh its spoofed value: *reloads* of a single domain, visits to different *domains* (we give FS two domain names), and browsing across *sessions* (browsing separated by 45 minutes of down time).

Fingerprinting Server. FS collects attributes known to be helpful for fingerprinting. Specifically, we setup FS to collect attributes collected by the open-source fingerprinting programs used by FPCentral [80] and Panopticlick [35], often by re-using their code. We list these attributes in the first column of Table 4.2. Similar to websites like panopticlick.eff.org and amiunique.org/fp, any browser visiting these domains can view their fingerprint, while a copy is stored on the server.

FS has minor modifications in the attributes it collects and how it collects them. FS detects additional *Noto* fonts, which ship by default on TBB. Moreover, FS does not place cookies on the browsers visiting our domains, which FPCentral and Panopticlick use to identify returning visitors.

Analysis Engine. To check for masking by a PET, AE uses both the fingerprints produced by FS of the browsers driven by CS and information directly from CS relating which browser used which PETs and which configurations. Figure 4.2 provides an overview of AE.

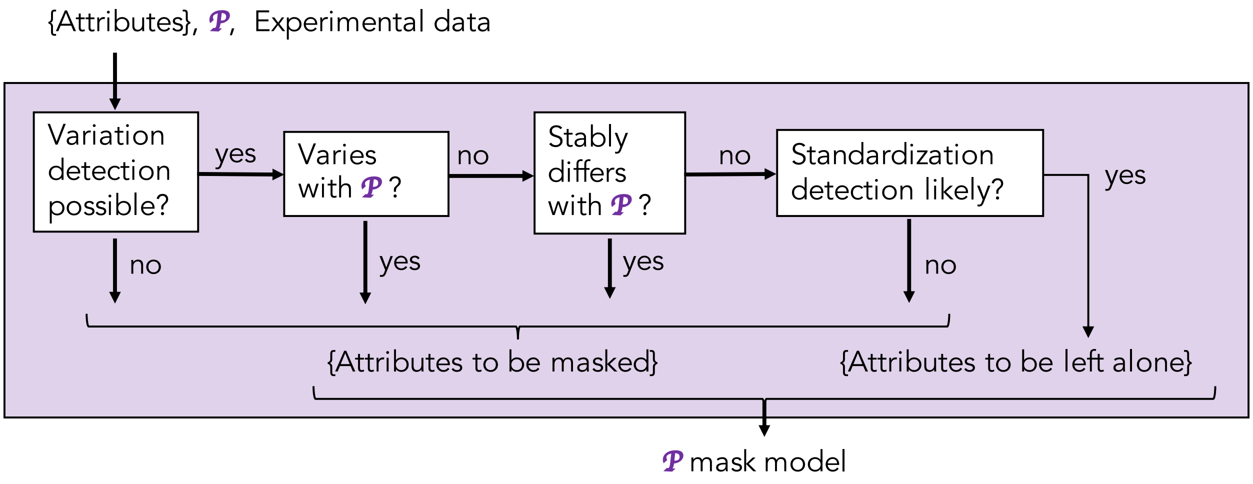

{Attributes}, 𝓟, Experimental data
Variation detection possible?
yes
Varies with 𝓟 ?
no
Stably differs with 𝓟 ?
no
Standardization detection likely?
yes
no
yes
yes
no
{Attributes to be masked}
{Attributes to be left alone}
𝓟 mask model

Figure 4.2: Analysis Engine in PETInspector

For each attribute, AE first checks whether its value varies from the baseline browser configuration as it crosses the three boundaries mentioned above. If so, it can't detect whether the PET varies that attribute since it is already varying and the analysis of this attribute stops flagging this limitation. If not, AE goes on to check whether the attribute's value varies for the browser configuration under the addition of the PET and, if so, stops and flags the attribute as masked.

If the attribute is not flagged under either variation check, AE checks whether the attribute is masked by standardization. To do so, it checks whether the value differs between browsers whose only change in configuration is the addition of a PET. If so, AE concludes that the PET is standardizing the attribute given that the change is not from the browser itself or the PET varying its value per the variation checks already performed. If not, then we can rule out with certainty full standardization but not partial standardization.

In general, ruling out partial standardization with experiments requires testing for all possible attribute values, a prohibitively expensive if not impossible task for many attributes. However, AE can, in reasonable time and with reasonable confidence, rule out *impactful* partial standardization, that is, standardization that affects at least a fraction f of the values. To do so, AE estimates the probability of seeing at least one standardized value given that at least a fraction of them f are standardized using the geometric distribution with f as the success probability. If this probability is less than some threshold, then AE reports that the attribute is probably not impactfully partially standardized. We use 0.1 for this threshold and 0.75 for f, but these are adjustable.

This approach is an estimation in two senses. First, for attributes with a finite number of values, the hypergeometric distribution would give a more accurate probability of seeing at least one standardized value, but would require knowing the number of possible values. The geometric distribution underestimates this probability, making AE conservative in ruling out standardization.

Second, using these distributions assumes that either the test attributes are drawn uniformly at random. We instead craft them to be extreme values in hopes of triggering standardization away from outlying values. While this makes computing the exact probably of finding standardization impossible, it should improve the odds of doing so except for pathologically behaving PETs.

4.5.2 Experiment and Results

Using PETInspector, we run an experiment finding that driving browsers across the sessions do not find any additional variations from the PETs (See Appendix C.2). Our main experiment does not include this check to save time.

We use CS to simulate seven browsing platforms. Six of these are virtual machines, two running 64-bit Ubuntu 14.04 (Trusty Tahr), two running 64-bit Ubuntu 16.04 (Xenial Xerus), and two running Debian 8.10 (Jessie). We introduce additional changes into these virtual machines to simulate differences in the system configurations. Specifically, we install different fonts and browser versions, setup different timezones, and simulate different screen resolutions and languages, The seventh browsing platform is simulated on a Macbook Pro. We perform measurements on Firefox and Chrome. More details about the configuration are available in Table C.2 in Appendix C.1.

CS drives these experimental browsing platforms to FS in the following pattern: five reloads of each domain name for FS. It does so a total of 28 times — 26 times for each PETs in our list, and twice for the two baseline browsers. All PETs are left in their default configuration.

While we did not think of the choice of browsers as affecting privacy, it turns out that comparing our baseline measurements for the two browsers reveals small differences in the attributes shared by them. Among the simulated platforms, Chrome sets the `cpu class` to *unknown*, the `screen.Depth` to 24, and the `buildID` to *Undefined*, unlike Firefox which reveals different values across browsing platforms. On the other hand, Firefox does not reveal any plugins, while Chrome does. Chrome's plugins differ across Ubuntu, Debian, and macOS. Table C.5 in Appendix C.1 provides more details on the attribute values revealed by each baseline browser.

Table 4.2 was automatically produced by PETInspector and displays which attributes were masked or not by which anti-fingerprinting PETs. We also tested blocking PETs, but only comment upon them in text. We provided PETInspector with the masks that each tool's documentation purports, which it uses to facilitate comparing documented behaviors with observed behaviors. Table C.3 in Appendix C.1 provides a lower level description of the finding in terms of the type of masking found.

Table 4.2: PET masks as purported and observed by PETInspector. □ indicates the PET's documentation claiming the attribute is masked. The remaining symbols represent the possible outputs of PETInspector: ✓ indicates observed masking, × indicates no masking found even when our experiment is likely to detect it, and · indicates inconclusive results (not shown for □s).

	Chrome							Firefox							
Attribute	BR	CD$_C$	CFB	GL	HMF	PE	TR	BI	BL	CD$_F$	CA	NE	SF	TBB	TO
DNT enabled	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
IE addBehavior	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
adBlock installed	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
buildID	·	·	·	·	·	·	·	☑	☑	×	×	×	×	☑	✓
canvas fingerprint	☑	☑	☑	☑	☑	☒	☒	×	×	☑	☑	×	×	☑	×
cookies enabled	·	·	·	·	·	□	·	·	·	·	·	·	·	·	·
cpu class	·	·	·	·	·	·	·	☑	☑	×	×	×	✓	☑	✓
h.Accept	·	·	·	·	·	□	·	·	·	·	·	·	·	□	·
h.Accept-Encoding	·	·	·	·	·	·	·	·	·	·	·	·	·	□	·
h.Accept-Language	✓	×	×	×	×	×	×	×	☑	×	×	×	×	☑	×
h.Connection	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
h.Dnt	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
h.Pragma	·	·	·	·	✓	·	·	·	·	·	·	·	·	·	·
h.Up.-Ins.-Req.	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
h.User-Agent	☑	×	×	×	☑	☒	☒	☑	☑	×	×	×	×	☑	☑
indexedDB	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
javascript fonts	×	×	×	×	×	×	×	×	×	×	×	×	☒	☑	×
language	✓	×	×	×	×	×	×	×	☑	×	×	×	×	☑	×
lied with browser	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
lied with language	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
lied with os	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
lied with res.	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
local storage	·	·	·	·	·	□	·	·	·	·	·	·	·	·	·
math.acosh(1e300)	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
math.asinh(1)	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
math.atanh(05)	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
math.cbrt(100)	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
math.cosh(10)	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
math.expm1(1)	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
math.log1p(10)	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
math.sinh(1)	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
math.tanh(1)	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
openDB	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
platform	×	×	×	×	×	×	×	☑	☑	×	×	×	×	☑	✓
plugins	☑	×	×	×	×	×	×	·	·	·	·	□	□	□	·
screen.AvailHeight	×	×	×	×	×	×	×	×	×	×	×	×	☑	☑	×
screen.AvailLeft	·	·	·	·	·	·	·	·	·	·	·	·	□	□	·
screen.AvailTop	×	×	×	×	×	×	×	×	×	×	×	×	☑	☑	×
screen.AvailWidth	×	×	×	×	×	×	×	×	×	×	×	×	☑	☑	×
screen.Depth	·	·	·	·	·	·	·	×	×	×	×	×	☑	☑	×
screen.Height	×	×	×	×	×	×	×	×	×	×	×	×	☑	☑	×
screen.Left	·	·	·	·	·	·	·	·	·	·	·	·	□	□	·
screen.Pixel Ratio	×	×	×	×	×	×	×	×	×	×	×	×	☑	☑	×
screen.Top	·	·	·	·	·	·	·	·	·	·	·	·	□	□	·
screen.Width	×	×	×	×	×	×	×	×	×	×	×	×	☑	☑	×
session storage	·	·	·	·	·	□	·	·	·	·	·	·	·	·	·
timezone	×	×	×	×	×	×	×	×	×	×	×	×	×	☑	×
touch.event	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
touch.max points	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
touch.start	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
webGL.Data Hash	☑	✓	✓	✓	✓	×	×	×	×	✓	✓	×	×	☑	×
webGL.Renderer	☑	✓	✓	✓	✓	×	×	×	×	✓	×	×	×	☑	×
webGL.Vendor	☑	✓	✓	✓	✓	×	×	×	×	✓	×	×	×	☑	×

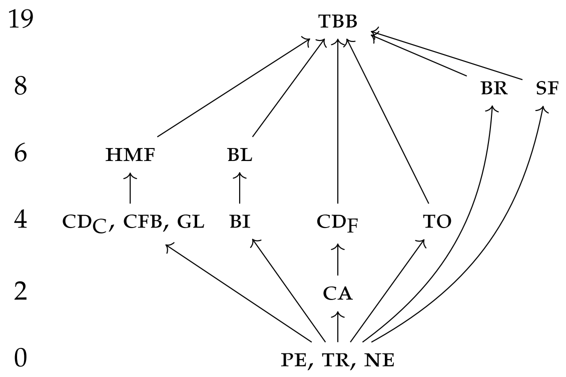


Figure 4.3: Ranking of PETs by the experimental method. Arrows show the preorder, with PETs at an equivalent order being grouped together. The y-axis shows the number of attributes masked.

The results show that PETInspector was able to check all the shown attributes by creating baseline configurations that differed in that attribute's value. Among the 15 anti-fingerprinting PETs, three (TR, PE, NE) do not lead to any detectable masking. The remaining 12 PETs mask at least one of the collected attributes.

Our experiments also detect undocumented masking of attributes by PETs. For example, while CD$_C$, CD$_F$, CFB, GL, CA claim to spoof only the `canvas fingerprint`, we also find them spoofing `webGL` attributes. We also find undocumented modifications of attributes by BR, SF, and TO. We also find inconsistencies in the behavior of BR, PB$_C$ and PB$_F$, HMF, and TBB which we discuss in the Introduction.

Among the 11 blocking PETs, one (D$_F$) fails to install, three (D$_C$, GH$_C$, GH$_F$) do not lead to any detectable modifications of attributes, four (AP$_C$, AP$_F$, UO$_C$, UO$_F$) modify the attribute `adBlock installed`, and three (PB$_C$, PB$_F$, TP) modify Do Not Track attributes.

Given the difficulty of taking in Table 4.2 (or Table C.3), for the purpose of ranking PETs relative to one another, we will consider each of these masking behaviors as equally valuable for reducing trackability. This level of abstraction in modeling PETs seems reasonable given our belief that trackers are foiled by any of these methods given the complexity of, for example, using a varying attribute for tracking. We produce a pre-ordering of PETs where one PET $\mathcal{P}_1$ is above or equal to another PET $\mathcal{P}_2$ iff $\mathcal{P}_1$ masks every attribute that $\mathcal{P}_2$ does. Those desiring a finer gradation can look at the number of attributes masked, but must bear in mind that not all attributes are equally important to mask. Figure 4.3 shows these rankings.

4.5.3 Limitations

As mentioned, we may miss some masking of attributes due to not testing the values that the PET standardizes away. Furthermore, we may not detect a PET varying an attribute across a boundary that we do not test. Thus, while we can be sure of masking when we find it, we cannot be sure we have found all masking.

We study a PET's behavior on 53 attributes known to be useful for fingerprinting. We can not detect masking of other attributes.

FS extracts fingerprints by running first-party fingerprinting scripts on browsing platforms. Thus, we do not detect masking that only happens for third-party scripts.

We test PETs in their default configuration. Some PETs may mask more attributes with stronger configurations, but users find it difficult to change the defaults [87], suggesting that our experiments may capture typical use.

Where our experiments dispute claimed masking (☒ in Table 4.2), it may be due to the above limitations rather than spurious claims. For example, PE has all its privacy features disabled by default.

To an extent, these limitations can be reduced with more comprehensive experiments using PETInspector. For example, one can modify FS to collect additional attributes in both first-party and third-party contexts. Moreover, one can modify CS to detect variations across other boundaries and use more diverse experimental browsing platforms to be more confident not missing standardization modifications. We will make PETInspector freely available for more extensive experimentation. Our current evaluations demonstrate the benefits of an experimental evaluation method for PETs within the current boundaries.

4.6 Hybrid Evaluation of PETs

Our experimental method provides a model of how various PETs mask fingerprints as well as provides a ranking of PETs based on the number of attributes they mask. However, it does not consider how important masking each attribute is.

As a result, we develop a hybrid method which combines the benefits of the experimental method with the observational method. We start with the *mask model* of each PET provided by the experimental method. For each attribute, we model the PET as masking the attribute if the model indicates so or if the experiment was inconclusive, thereby overestimating the PET's abilities. We use this mask model to transform a set of original fingerprints collected on the amiunique website into a counterfactual PET-modified set, which simulates the browsing platforms in the original data set visiting amiunique with a PET installed. To determine the effectiveness of the PET, we compare trackability in the two data sets.

4.6.1 Method

We first measure the trackability of original fingerprints (i.e., $\mathcal{D}_{\bar{\mathcal{P}}}[id(\cdot)]$) in the `amiunique` data set. To evaluate a PET $\mathcal{P}$, we compare the trackability of original fingerprints with the corresponding set of $\mathcal{P}$-modified fingerprints. The mask model from the experimental method provides a way to transform the original fingerprints.

We apply the mask model of a PET $\hat{\mathcal{P}}$ to generate PET-modified fingerprints (i.e., $\mathcal{D}_{\hat{\mathcal{P}}}[id(\cdot)]$) corresponding to the original fingerprints and recalculate the trackability metrics of the modified fingerprints. By comparing the metrics of the original and PET-modified fingerprints, we estimate the effectiveness of the PET $\mathcal{P}$.

Of the 53 original attributes, PETInspector provides conclusive characterization for 17 attributes on Chrome browsers and 19 attributes on Firefox browsers. Of these, only 12 appear in the `amiunique.org` data set. For a given PET, we mask these 12 attributes according to the model generated by PETInspector and fully mask all the remaining attributes for which the experiment is inconclusive. By fully masking inconclusive attributes, we overestimate the effectiveness of PETs. Thus, we generate a set of PET-modified fingerprints (i.e., $\mathcal{D}[id(\hat{\mathcal{P}}(\cdot))]$) from the original fingerprints and measure effectiveness of the 15 anti-fingerprinting PETs. The mask models of these PETs are shown in Table C.4 of Appendix C.1.

4.6.2 Results

We present the metrics of trackability from Section 4.4.2 for both Chrome and Firefox PETs in Table 4.3. The original fingerprints reveal 13.002 and 12.359 bits of entropy for Chrome and Firefox browsers respectively. Applying a base mask comprising of all inconclusive attributes reduces the entropies to 12.892 and 12.177 bits. The base mask corresponds to using no PET in our evaluations.

Our evaluations reveal that all PETs but BR and TBB reveal over 11 bits of entropy and hence are marginally better than not using any PET at all. For these PETs, fewer than 20% of the fingerprints are in anonymity sets of size greater than 10. BR appears to do better, leaking just over 8 bits of entropy and having over 70% of fingerprints in anonymity sets of size greater than 10. TBB appears to perform best since it modifies all the 12 attributes we consider.

We perform all evaluations on the same set of fingerprints and thus do not account for the difference in the popularity of PETs. To observe the consequences of having user bases of different sizes, we also evaluate the PETs taking into account their popularity. We draw random samples of fingerprints from the data set with samples of size equal to the number of PET users and estimate uniqueness metrics on the samples. Table C.6 in Appendix C.1 displays the number of users of each PET in our list as of

Table 4.3: Trackability metrics for PETs. TBB* denotes an evaluation done with a handcrafted, more accurate model of TBB's masking.

PET	entropy	prop_less1	prop_less10
	Chrome PETs		
no mask	13.002	0.892	0.983
base mask	12.892	0.816	0.982
CD$_C$, CFB, GL	12.172	0.602	0.875
HMF	11.065	0.371	0.708
BR	8.117	0.073	0.265
	Firefox PETs		
no mask	12.359	0.875	0.960
base mask	12.177	0.797	0.949
BI, TO	12.049	0.747	0.936
CA	12.002	0.700	0.941
BL	11.875	0.678	0.924
SF	11.778	0.726	0.919
CD$_F$	11.263	0.483	0.833
TBB	0.000	0.000	0.000
TBB*	2.902	0.001	0.010

Table 4.4: Entropy-based effectiveness of PETs on fingerprint samples scaled according to the PET's popularity, sorted according to the effectiveness

PET	#users	entropy($\mathcal{D}[id(\cdot)]$)	entropy($\mathcal{D}[id(\mathcal{P}(\cdot))]$)	eff$_{\text{entropy}}$
		Chrome PETs		
GL	342	8.364 ± 0.002	8.279 ± 0.003	0.085 ± 0.004
HMF	177	7.440 ± 0.002	7.336 ± 0.004	0.104 ± 0.004
CFB	7630	12.073 ± 0.001	11.573 ± 0.002	0.500 ± 0.002
		Firefox PETs		
TO	265	7.967 ± 0.003	7.913 ± 0.004	0.054 ± 0.005
BI	858	9.518 ± 0.003	9.410 ± 0.004	0.108 ± 0.005
BL	1816	10.412 ± 0.002	10.187 ± 0.003	0.225 ± 0.004
SF	1754	10.375 ± 0.003	10.000 ± 0.005	0.375 ± 0.006
CD$_F$	5274	11.452 ± 0.002	10.659 ± 0.003	0.793 ± 0.004

Dec. 2017. We are unable to perform these evaluations for PETs with an undisclosed number of users (like BR, TP). We also do not perform these evaluations for PETs with a user base greater than 17,109 (like TBB, CD$_C$ and CA), since repeated fingerprints do not change the bits of entropy revealed. For all other PETs, we compute the mean and the standard error of mean (mean ± sem) of the trackability metrics from 100 random samples. Table 4.4 displays the entropy-based effectiveness metrics for these PETs, sorted according to the effectiveness. We can see that CFB scores better than HMF due to its high popularity, contrary to the original evaluations. We also see that the effectiveness of tools with identical effects increases with popularity. For example, TO and BI both have identical modifications to the 12 attributes, but the effectiveness increases with popularity. Table C.8 in Appendix C.1 provides estimates of all trackability metrics for these PETs.

4.6.3 Limitations

While this hybrid method enables us to perform a fine-grained evaluation of PETs with few users, it inherits some of the limitations of the methods on which it builds. For example, from the observational method comes limitations that samples can be biased and that not no one metric can completely capture the quality of a PET. From the experimental method, we inherit the rough nature of the mask model, which does not model how partial standardization is performed.

In particular, our analysis overestimates the effectiveness of all PETs, since we assume any modifications of an attribute by a PET renders that attribute useless to a tracker. This may not be the case. For example, BR modifies the `User-Agent` and the `Accept-Language` headers to different values than Chrome. The revealed values may be continue to reveal bits of entropy. Similarly, TBB also reveals spoofed values of screen resolution.

We can carry out a tighter evaluation by considering a tracker which can take advantage of the spoofed values. This evaluation requires knowledge of how a PET spoofs the attribute. For TBB, we performed a manual code analysis to determine how exactly TBB deals with screen resolution attributes. We rerun the hybrid analysis on a hand crafted mask model capturing this behavior instead of using the rough model produced by PETInspector.[9] Table 4.3 shows this analysis as TBB*. This provides a tighter evaluation for TBB showing it reveals non-zero entropy after all.

4.7 Application: Informing PET design

Table 4.5: Comparison of effectiveness and loss of TBB's original spoofing strategies with alternate strategies. Trackability metrics and losses are rounded to three decimal places and to three significant digits respectively.

Threshold	Quanta	entropy	prop_less1	prop_less10	Abs. Loss	% Loss
1000×1000	200×100	2.902	0.001	0.010	870*k*	50.3%
1350×1000	200×193	2.715	0.001	0.009	729*k*	42.6%
1350×1000	269×160	2.901	0.000	0.009	728*k*	42.3%
1550×1000	222×197	2.899	0.000	0.009	666*k*	40.3%
1550×1000	295×160	2.882	0.000	0.010	636*k*	37.2%

[9] We create the handcrafted mask model of TBB from the Firefox patch at `https://gitweb.torproject.org/tor-browser.git/commit/?h=tor-browser-45.8.0esr-6.5-2&id=7b3e68bd7172d4f3feac11e74c65b06729a502b2`.

With the ability to accept handcrafted mask models, our hybrid method can help PET developers make an informed choice while designing PETs. By measuring effectiveness of hypothetical designs, PET developers can compare different masking strategies to tradeoff utility with trackability. We carry out such an exploration comparing alternate designs of TBBby applying our hybrid method on various hypothetical versions of TBBthat mask attributes differently.

Among the 12 attributes we consider, TBB only leaks bits through the partially standardized screen resolution. Specifically, it resizes new browser windows to a multiple of 200×100 pixels, while capping the window size at 1000×1000 pixels, and uses the client content window size as screen dimensions [116]. As a result all TBB users get placed into one of 50 anonymity sets based on the revealed screen dimensions, as long as they do not change the window dimensions manually. We explore the impact of the choices of threshold and quanta parameters on the effectiveness of TBB.

We use the number of unutilized screen pixels due to a spoofing strategy as a measure of utility-loss. We measure two variants: the total number of unutilized pixels (average absolute loss), as well as the number of unutilized pixels as a percentage of the available pixels (average percentage loss). Increasing the threshold parameters and decreasing the quanta parameters reduces this loss. We first measure the effectiveness of alternate strategies with strictly lower loss (i.e., higher threshold and lower quanta parameters) than TBB's. An exhaustive search of all $19,999$ quanta parameters less than TBB's (i.e., 200×100), while fixing the threshold parameters at 1000×1000 finds no strategy achieving higher effectiveness in all metrics than TBB. Similarly, fixing the quanta parameters at 200×100, while increasing the threshold parameters in steps of 50 pixels from 1000×1000 to 2000×2000 does not uncover any strategy with higher effectiveness either. We perform these explorations on the Firefox fingerprints in the `amiunique.org` data set.

Next, we explore strategies that tradeoff losses resulting from one set of parameters (e.g., quanta) with gains from another (e.g., threshold) with the goal of finding a strategy that reduces the loss while increasing the effectiveness. We find that threshold width parameter in the most need of improvement since less than 13% of `amiunique.org` fingerprints have a screen width less than 1000 pixels. We consider alternative threshold widths of 1350, 1550, and 1600 since a higher percentage of fingerprints (25%, 47%, and 51% respectively) have screen widths less than these thresholds. We retain the threshold height of 1000 pixels as more than 50% of the fingerprints remain below that threshold. We exhaustively search for all 10201 quantas in the range 200×100 to 300×200 for all three threshold parameters. We set an upper bound of 300×200 as the loss may be too high for low-resolution displays for very high quanta parameters. We find 786 and 291 quanta parameters for threshold widths of 1350 and 1550 respectively for which the losses are lower than TBB's, but the effectiveness is higher. We display strategies with the least quanta parameters in Table 4.5. However, as we increase the threshold width to 1600, none of the quanta parameters lead to a higher measure of effectiveness than TBB.

4.8 Conclusion and Discussion

We carry out an evaluation of 26 different Privacy Enhancing Technologies against fingerprinting using two different methods. We develop PETInspector and use it for experiments to determine how these PETs spoof 53 different attributes. In addition to uncovering inconsistencies, it provides a model of PETs' behaviors. While the experimental method provides an evaluation in terms of the number of attributes that a PET masks, it cannot distinguish between the relative importance of masking different attributes. Our hybrid method leverages a real world fingerprinting data set to provide a finer grained view into the impact of modifying different attributes. We find TBB to be the most effective PET among the ones we evaluate using both methods. It standardizes the most attributes and reduces the trackability of revealed fingerprints by the highest margins among the PETs we evaluate. We also apply our hybrid method to find some hypothetical spoofing strategies which have a smaller utility loss than Tor, yet are just as effective.

It is not entirely surprising that TBB protects well against known fingerprinting attacks. The Tor Project is part of the team behind the FPCentral fingerprinting repository, which spans a comprehensive collection of fingerprinting techniques. Being aware of these fingerprinting techniques, developers of TBB are in a position to build defenses for fingerprinting. TBB also benefits from a huge user base. At over 3M daily users, it is the highest among the PETs we evaluate. This however does not mean that TBB users are protected against all possible fingerprinting attacks. Developers must be on the lookout for new fingerprinting techniques and build in fresh defenses.

Chapter 5

A Legal Analysis of Discrimination in Online Advertising

5.1 Introduction

In this chapter, we consider legal action against corporations behind personalization systems as a possible deterrent for discriminatory outcomes. We start from the 'Gender and Jobs' experiment in Chapter 3, which finds that simulated male and female users receive employment-related advertisements in differing rates along gender lines despite identical web browsing patterns. We find that the language of Title VII makes it applicable only to specific categories of entities (like employer or employment agency) and may not apply to an advertising platform like Google. The Fair Housing Act, however, does not have the same restrictions and may consider Google as a covered entity. This is where another law, Section 230 of the Communications Decency Act, comes into play. It provides immunity to online intermediaries for illegal content created or developed by a third party. Section 230 provides Google with immunity if discrimination came about solely from the advertiser's inputs. However, the law does not provide immunity if the online intermediary "materially contributes" to the illegality of the content. We argue that the targeting of ads by an ad platform towards or away from protected classes without explicit instructions from the advertiser constitutes material contribution. As a result, the immunity should not extend to the ad platform in such scenarios. The threat of legal liability may incentivize companies behind personalization systems to take precautionary measures to avoid discrimination.

5.1.1 Motivation

In addition to experiments in Chapter 3, several recent studies demonstrate that computer systems can discriminate, including by gender [17, 20, 71, 79] and race [8, 9, 135]. Although much scholarship exists on the legal consequences of discrimination, little work has explored the legal status of these concrete cases ([13] is the only one we are aware of). The consideration of such concrete cases, instead of abstract hypotheticals, forces us to confront the difficulties of proving a case based upon the limited evidence practically available to investigators. Such careful consideration can show what empirical evidence could aid the crafting of a case, which suggests new studies, and how laws might not be enforceable in practice. Furthermore, they have the potential to show that liability can lie with an advertising platform, not just in theory, but in practice. Such a finding can promote positive change and guide regulators to the interesting questions to ask.

An example of a real world difficulty is that while the existence of discrimination might be clear, the cause might not be. Computers may use factors associated with, but distinct from, protected attributes. This not only complicates the detection of discrimination, but also provides those intending to discriminate with a gloss of statistical rationality and leads fair-minded individuals to unwittingly discriminate via models that redundantly encode gender, race, or other protected attributes.

5.1.2 Chapter Contributions

This chapter provides a legal analysis of our discrimination findings ('Gender and Jobs' experiment in Section 3.6.1). We explore the operation of Google's advertising network to understand the various decision points that could contribute to the gender-skewed placement of such ads (Section 5.4). In doing so, we find that advertisers can use Google's advertising platform to target and serve employment and housing ads based on gender. While we explore possible reasons that could have contributed to the discriminatory placement of ads, these explorations are not exhaustive. Uncovering the cause behind the discriminatory placement of ads requires further visibility into the advertising ecosystem or assumptions over how the ecosystem operates, and is beyond the scope of this work.

We then explore legal questions and policy concerns raised by these results. Focusing on employment-related ads, we consider potential liability for advertisers and ad networks under Title VII, which makes it unlawful for employers and employment agencies "to print or publish or cause to be printed or published any ... advertisement relating to employment...indicating any preference, limitation, specification, or discrimination, based on ... sex".[1]

[1] §704(b) of Title VII of the Civil Rights Act of 1964, codified at 42 USC §2000e-3(b).

Due to the limited coverage of Title VII we conclude that a generic advertising platform, like Google's, is unlikely to incur liability under Title VII's prohibitions regardless of any contributions they make to the illegality of an advertisement. Advertisements that run afoul of the Fair Housing Act's (FHA's) prohibition on indicating a preference however could create liability as unlike Title VII the FHA provision is of general applicability. In a case under the FHA, a court would need to consider how the advertising prohibition interacts with Section 230 of the Communications Decency Act (CDA),[2] which provides interactive computer services with immunity for providing access to information created or developed by a third party. Thus, we focus on the interaction between the prohibition on discriminatory advertising in the FHA and Section 230. We argue that despite the broad immunity generally afforded by Section 230, interactive computer services can lose that immunity if they target ads toward or away from protected classes. The loss of immunity is based on the act of targeting itself rather than any content that is contained within the four corners of the advertisement. We focus our analysis on Google, its system, documentation, consumer and advertising interfaces, and empirical research looking at it to provide useful details for our legal analysis. However, throughout, we generalize our analysis to generic machine learning systems where appropriate.

Our main contribution to the existing scholarship examining discrimination in automated decision-making is the analysis of the application of the discriminatory advertising prohibition in Title VII and the FHA in the light of Section 230. Our main novelty is drawing on the relevant regulations and case law under the parallel, but broader, provision in the Fair Housing Act, which has been more aggressively and creatively used.

We show the potential for ad platforms to face liability for algorithmic targeting in some circumstances under the FHA despite Section 230. Given the limited scope of Title VII we conclude that Google is unlikely to face liability on the results from the 'Gender and Jobs' experiment. Thus, the advertising prohibition of Title VII, like the prohibitions on discriminatory employment practices, is ill equipped to advance the aims of equal treatment in a world where algorithms play an increasing role in decision making.

5.2 Related Work

We are not the first to consider possible causes of discrimination in behavioral advertising. Todd interviewed the parties involved looking for, but not finding, definitive answers [137]. Lambrecht et al. conduct a study similar to ours, but with more control, to analyze possible causes [79]. Sweeney considers possible causes of discrimination in contextual advertising [135]. We further discuss these works when we consider

[2] 47 USC §230.

the causes they find likely.

Several law review articles have looked at the legal and policy implications of such outcomes and how policies can help prevent them. Barocas and Selbst discuss the difficulties in applying traditional antidiscrimination law as a remedy to discrimination caused by data mining (automated pattern finding) [13]. Kim explores the application of antidiscrimination norms of Title VII to computers making employment decisions and argues that this requires reassessment of the laws [72]. Kroll et al. explore how computational tools can ensure that automated decision making avoids unjust discrimination and conform with legal standards [77] .

The most similar to our own work, Tremble applies Section 230 of the Communications Decency Act to content served by Facebook [140]. While Section 230 of the Communications Decency Act frees interactive computer services like Facebook of liability for user generated content, Tremble argues that personalized content, like that on Facebook, constitutes content generated by Facebook and as such does not qualify for exclusion under Section 230.

5.3 A primer on Google's web advertising

The advertising ecosystem is a vast, distributed, and decentralized system with several parties. First, there are *publishers*, websites which host content. Typical examples of publishers include news websites, social networks, and blogs. As most publishers provide content for free, they earn revenue by serving advertisements on their pages. Second, there are *advertisers*, who seek to place their ads on publishers' webpages. Advertisers include online shopping portals, firms providing web-services, and traditional brick-and-mortar shops trying to reach their customers online. Third, there are *ad networks* which connect advertisers and publishers. Fourth, there are *consumers* who consume online content. Publishers sign up with an ad network to register available spaces on their webpage. Advertisers also sign up with the ad network to register their ads. When a consumer visits a publisher website, the ad network determines which ad to place on the page for the consumer to view. To make this determination, ad networks use proprietary algorithms that among other variables take into account the expressed preferences of advertisers and publishers, and the specific user's behavior, and, if communicated, preferences.

In this ecosystem, Google dominates as the largest ad network connecting advertisers to publishers. With nearly a third of all digital ad revenue, Google has more than three times the market share of the next largest ad network (Facebook) [95]. Google serves many different types of ads including those beside Google search results (search ads), on YouTube (video ads), and on third-party websites (display ads). As our prior experiments found evidence of discrimination on display advertisements, we focus on them for the remainder of this section.

Below, we provide more details about how publishers, advertisers, Google (acting as an ad network), and consumers interact for display ads. Our description is largely based upon Google's own public-facing descriptions. We do not, and often cannot, verify that Google's descriptions of its internal behaviors are accurate.

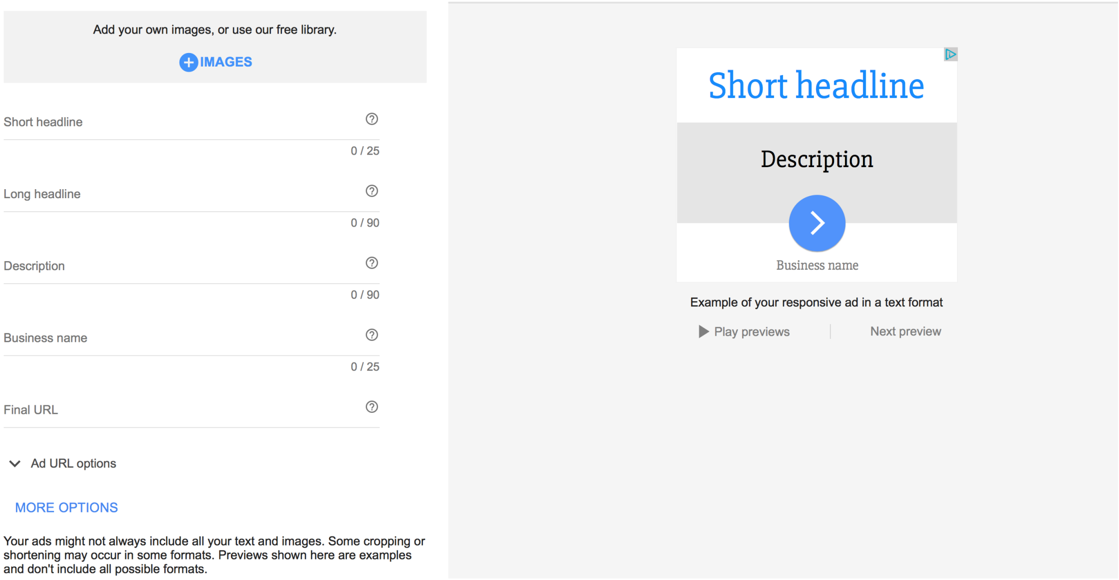
Publishers. Publishers who serve content on the web for free often rely on advertisements to earn revenue. Most such publishers defer the task of identifying potential advertisers for their site to ad networks. To use Google's ad network, publishers register their site on AdSense[3] and indicate *ad spaces* on their website where they would like to show ads. These could be locations on the lateral margins, at the top of the page, or inline with content. Google provides small pieces of code that publishers add to their webpage's source code, thereby enabling Google to serve ads within the ad spaces.

Advertisers. Advertisers are entities who are willing to pay to have their ads shown to consumers. Advertisers wanting to serve advertisements through the Google ad network sign up on Google's AdWords[4] interface and communicate to Google their ads. While there are as many as five different types of ad campaigns as of this writing, we focus on 'Display Network Only' campaigns, since it is the closest to the ad of interest from our 'Gender and Jobs' experiment. In such campaigns, Google allows serving four types of ads, namely responsive ads, image ads, gallery ads, and app/digital content ads. For simplicity, we focus on responsive ads, which modern variants of the text ads that were analyzed in the experiment. A responsive ad includes two headlines (a short and a long one), a description, a business name a URL, to which a consumer is directed upon clicking the ad, and an image. Figure 5.1 shows the interface for creating an ad on Google AdWords.

In addition, Google also enables advertisers to specify targeting criteria for their ads. This helps advertisers reach desired consumers more effectively. While advertisers can target their advertisement based on several criteria established by Google, we are interested in targeting based on demographics (like gender, age group, and parental status) and interests (like Cat Lovers, Mobile Enthusiasts, etc.) since they allow tailoring advertising campaigns based on data Google knows or believes about specific users. This knowledge may be based on user specified settings as discussed below, inferences drawn by Google, and data from other sources. The advertiser must also select a bid, which indicates how much they are willing to pay Google when consumers view or interact with their ads. Google provides a portion of the revenue to the publisher on whose page the ad appears.

[3] adsense.google.com

[4] adwords.google.com



Add your own images, or use our free library.

Your ads might not always include all your text and images. Some cropping or shortening may occur in some formats. Previews shown here are examples and don't include all possible formats.

Example of your responsive ad in a text format

Figure 5.1: Creating responsive ads on Google AdWords

Ad Network. Once the advertiser has submitted the ad and targeting criteria, Google reviews it to make sure it is safe and appropriate for consumers and complies with Google's advertising policies, which include a requirement to "comply with all applicable laws and regulations." [53] Google's policies restrict certain advertising content, while establishing special procedures for other content. While some gender specific advertising would violate the provision requiring compliance with applicable law, we could find no independent provision limiting advertising content that discriminates based on sex even though they have restrictions that apply to other sensitive categories (like race, sexual orientation, political affiliation, etc.) [54]. Once Google approves the ad and the advertiser has entered their billing details, the ad becomes eligible to be served.

Once the ad is eligible, it is up to Google to determine when, where, and to whom to serve the ad. When a consumer visits a publisher's website and ads from multiple advertisers in its network are eligible for a limited number of ad spaces, Google uses a real-time auction to determine which ad appears and in what order. For each competing ad, Google calculates a metric called the *Ad Rank* which determines how competitive an ad is for a given ad space for a particular consumer. Ad Rank depends upon several factors, but the most important for our purposes, is the *expected click-through rate* (CTR), which is Google's prediction of how likely the consumer is to click on the ad if shown. Google bases this prediction on what it knows about the current consumer and on its experience from prior click behavior of millions of other consumers.

The Ad Rank determines the position of the ad in the page and whether it shows up at all. Google computes the Ad Rank for an ad every time it competes in an auction and serves the top ranking ads in that order to the consumer. An ad with the same Ad Rank may or may not be served in subsequent auctions depending on the competition at the time. Thus, competing ads may influence which consumers an ad gets served to.

Consumers. Consumers browse the Internet and consume publisher content. Google provides consumers some insight and control over what types of advertisements Google serves them. Ad Settingsis a Google tool that helps consumers view and control the ads they see on Google services and on websites that partner with Google. It allows a consumer to select attributes for ad-targeting and to see and modify ad-targeting inferences that Google has made about the consumer. Consumers can view and edit both demographics and interests based on browsing behavior on Ad Settings.

Google's Ad Settings respond to regulators' and consumers' concerns about behavioral marketing on the web [141]. They provide some transparency about how consumers are profiled for ad-targeting, and allow consumers to exercise some modicum of control over the ads they see, including the ability to limit ads targeted based on previous web activity and demographic details. In the summer of 2015, Google updated the Ad Settings page to require being logged in to their Google account to access most of its features, including viewing and editing gender, age, languages and interests.

5.4 Possible Causes of Discrimination

We will now consider possible ways that the results of the 'Gender and Jobs' experiment can manifest in an online advertising ecosystem. The advertising ecosystem is a vast, distributed, and decentralized system with several actors. There are *publishers* who host online content, *advertisers* who seek to place their ads on publishers' websites, *ad networks* who connect advertisers and publishers, and *consumers* who consume online content and ads.

Each actor has a set of primary mechanisms through which they can introduce a difference in how men and women are treated (Factor I in Table 5.1). Thus, we can view the first factor as saying *who* creates the inputs that might contribute to a discriminatory outcome. In all cases, the impact of the input, and in some instances its availability, is ultimately determined by Google. Indeed, by being the central player connecting the parties, Google always plays a role. While the simulated users surely played a role in the selection of ads by indicating their gender, this is not included in our analysis because it would suggest that, by admitting one's gender, a consumer bore some responsibility for the potentially discriminatory result. We do not believe this position to be technically accurate, nor legally defensible.

With respect to each actor we consider *how* the results may have occurred (Factor II in Table 5.1). Where

Table 5.1: Possible causes of the 'Gender and Jobs' experiment finding organized around four actors

Factor I: (Who) Possible mechanisms leading to males seeing the ads more often include:

1. (Google alone) Explicitly programming the system to show the ad less often to females, e.g., based on independent evaluation of demographic appeal of product (explicit and intentional discrimination);
2. (The advertiser) The advertiser's targeting of the ad through explicit use of demographic categories (explicit and intentional discrimination), the pretextual selection of demographic categories and/or keywords that encode gender (hidden and intentional), or through those choices without intent (unconscious selection bias), and Google respecting these targeting criteria;
3. (Other advertisers) Other advertisers' choice of demographic and keyword targeting and bidding rates, particularly those that are gender specific or divergent, that compete with the ad under question in Google's auction, influencing its presentation;
4. (Other consumers) Male and female consumers behaving differently to ads
   (a) Google learned that males are more likely to click on this ad than females,
   (b) Google learned that females are more likely to click other ads than this ad, or
   (c) Google learned that there exists ads that females are more likely to click than males are; and
5. (Multiple parties) Some combination of the above.

Factor II: (How) The mechanisms can come in multiple favors based on how the targeting was done

1. on gender directly
2. on a proxy for gender, i.e. on a known correlate of gender because it is a correlate),
3. on a known correlate of gender, but not because it is a correlate, or
4. on an unknown correlate of gender.

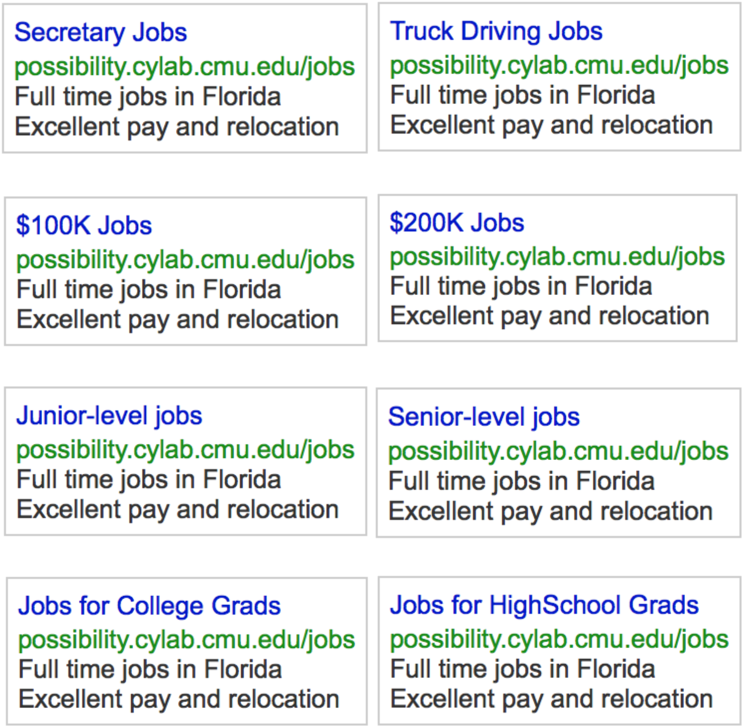
appropriate we consider the use of gender as a targeting criteria, the intentional and unintentional use of features that correlate with gender and the impact of the bidding system.[5]

5.4.1 Google's Actions Alone

Google created the entire advertising platform. It designed the AdWords interface that allows advertisers to target ads based on inputs including gender. Its terms of use admonishes advertisers to comply with all applicable laws and regulations. Through examples it specifies areas where advertisers have in the past run afoul of the law.

However, bans on sex-based targeting of employment, housing, and credit are not specifically addressed. Google has a set of policies for interest-based advertising that prohibit using any "sensitive information" about site or app visitors to create ads. While race, ethnicity, sexual orientation, and religion are considered "sensitive information", gender is not.

[5] Since correlation is the most familiar form of statistical association, we use correlations here, but all our statements may generalize to other forms of association.



Secretary Jobs
possibility.cylab.cmu.edu/jobs
Full time jobs in Florida
Excellent pay and relocation

Truck Driving Jobs
possibility.cylab.cmu.edu/jobs
Full time jobs in Florida
Excellent pay and relocation

$100K Jobs
possibility.cylab.cmu.edu/jobs
Full time jobs in Florida
Excellent pay and relocation

$200K Jobs
possibility.cylab.cmu.edu/jobs
Full time jobs in Florida
Excellent pay and relocation

Junior-level jobs
possibility.cylab.cmu.edu/jobs
Full time jobs in Florida
Excellent pay and relocation

Senior-level jobs
possibility.cylab.cmu.edu/jobs
Full time jobs in Florida
Excellent pay and relocation

Jobs for College Grads
possibility.cylab.cmu.edu/jobs
Full time jobs in Florida
Excellent pay and relocation

Jobs for HighSchool Grads
possibility.cylab.cmu.edu/jobs
Full time jobs in Florida
Excellent pay and relocation

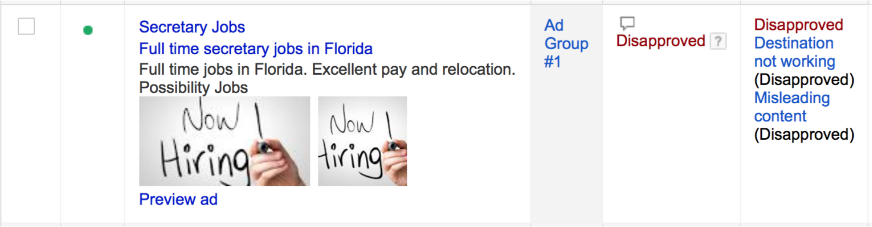
Figure 5.2: Ads approved by Google in 2015. Ads in the left (right) column were targeted to women (men).

Given its control over the platform there are many ways in which Google could have caused or contributed to the difference in advertisements directed to men and women (Case 1 of Factor I). A Google employee could have manually set the ad to target by gender or a feature associated with gender. While presumably the advertising system is largely autonomously driven by programs, researchers have documented that even in highly automated systems, such as search, a sizable amount of manual curation occurs [47].

5.4.2 Direct Targeting of Gender by Advertisers

Advertisers, including The Barrett Group, which showed the ad in question, can make multiple decisions through the AdWords interface that could steer their ads toward or away from women. The simplest way gender-skewed advertising could have emerged is if the advertiser directly targeted on gender (i.e. Factor I.2+Factor II.1). AdWords offers the ability to set demographic parameters to explicitly target ads toward, or away from, a single sex. While such explicit intentional gender targeting is supported by the AdWords interface, we wanted to explore whether the Barrett Group could actually use this feature to target their advertisement. To do so we performed another study in three phases.

First, in 2015, we constructed several ad campaigns that targeted job-related ads on the basis of gender



□	●	Secretary Jobs Full time secretary jobs in Florida Full time jobs in Florida. Excellent pay and relocation. Possibility Jobs Preview ad	Ad Group #1	Disapproved ?	Disapproved Destination not working (Disapproved) Misleading content (Disapproved)

Figure 5.3: Ad disapproved by Google in 2017.

using Google's advertising platform, AdWords. Figure 5.2 shows all ads that were approved by Google. Ad 5.2a is for a secretary job targeted towards women, while ad 5.2b is for a truck-driving job targeted towards men. The other pairs of differentially targeted ads varied by pay, seniority level, and educational requirements.

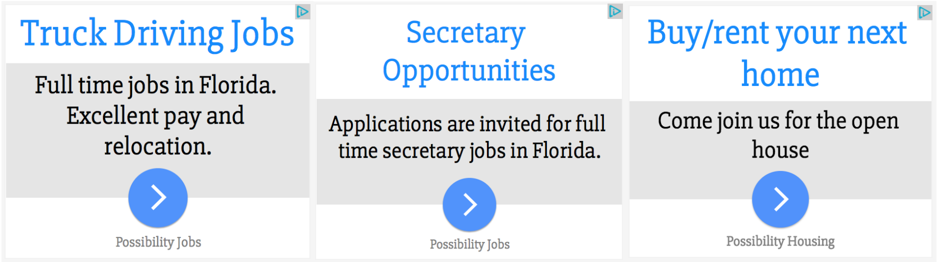
Our ads all had the same display and destination URLs [6]. This page has the words "Test ad. No jobs here." We also verified that Google rejects some advertisements at this stage by intentionally submitting ads with broken links or excessive exclamation points and found these were not approved.

Second, in 2017, we again tested Google's ad approval procedure and, this time, found it to be somewhat more sophisticated. While we were able to get one ad approved with the same destination URL and ad text as in Figure 5.2b, the other ads were disapproved. In particular, Google AdWords reported the destination was not working and the content was misleading (Figure 5.3). However, by changing the ad text and destination URL as well as adding more text to the destination webpage, we got the ad approved.

Third, while these explorations make it clear that Google AdWords allows creation of discriminatory job ad campaigns, it leaves open the possibility that Google would prevent the gender-targeted employment ads from being delivered at a later point in the process. As our last step, to check whether this is the case, we enabled both the ad campaigns at the same time (differing by a few seconds) for about 12 hours in 2017. We observe that both the campaigns receive several thousands of impressions, with the truck driver campaign receiving over 70k impressions and the secretary campaign receiving over 55k impressions. The campaigns collectively cost less than $100. The demographics of the users receiving the impressions exactly matched the targeting criteria. All the truck driver ad impressions were to men (or consumers who Google believes are men) and those for the secretary ad were all to women. This finding demonstrates that an advertiser with discriminatory intentions can use the AdWords platform to serve employment related ads disparately based on gender.

We also had ads for housing approved, targeted and served disparately (Figure 5.4c). The ad was suggestive of attending a open house for buying or renting a house. The final destination, however, had

[6] possibility.cylab.cmu.edu/jobs



Truck Driving Jobs

Full time jobs in Florida. Excellent pay and relocation.

Possibility Jobs

Secretary Opportunities

Applications are invited for full time secretary jobs in Florida.

Possibility Jobs

Buy/rent your next home

Come join us for the open house

Possibility Housing

(a) (b) (c)

Figure 5.4: Ads approved and served by Google in 2017: truck driver jobs only to men, secretary jobs only to women, and housing disparately.

text indicating that the ad was created and served as part of a study.[7] This ad was targeted to both male and female demographics who were *American Football Fans* or *Baseball Fans*. These interests were chosen intentionally to target the male demographic more. With these interests, Google's AdWords estimated that the ad would be targeted to between 1B and 5B women and between 5B and 10B men. We kept the ad active for about 24 hours. During this period, it received over 23k impressions, 75% of them being to men. This again demonstrates that an advertiser can intentionally use proxies for gender to target housing ads disparately based on gender.

We look at this scenario in detail to explore whether an employer or employment agency using Google's ad network can engage in explicit, intentional discrimination. Using AdWords, the career coaching service, Barrett Group, could have intentionally targeted their ad toward males, or limited targeting to females. Our small study also suggests that Google's review process does not weed out employment-related advertisements explicitly targeted by gender. While our study shows that such direct targeting as one possible explanation for the advertising outcome, it cannot tell us whether whether the Barrett Group actually used the demographic choices to target their advertisement.

5.4.3 Other Possibilities for the Advertiser

In fact the Barrett Group denied targeting on gender and claimed to have sought those older than 45, receiving high pay, and of executive-level experience [137], pointing to another possibility: the Barrett Group could have made other choices that led indirectly to targeting on gender. For purposes of this analysis we set aside the issue of targeting based on age which is an independently prohibited act by employers and employment agencies under Title VII. This explanation points to the possibility of the advertiser choosing interests or keywords correlated with gender (i.e. Factor I.2 + Factor II.{2 or 3 or 4}). Given the targeting criteria, it seems reasonable to assume that on average ad placement would skew toward men.

[7] possibility.cylab.cmu.edu/housing

On average, men earn more than women. Numerous studies have documented the under representation of women in the executive suite (e.g., in 2016, only 4% of Fortune 500 CEOs were women [154]). Thus, one could conclude that The Barrett Group intended to use the attributes as a proxy for gender to target males without appearing to do so—a practice called *masking* (Factor II.2). However, it may be that these were among a broader set of attributes and keywords that The Barrett Group selected, which reduces our concern with masking, that still redundantly encoded gender. Even a series of attributes that alone or in consort do not appear gender-specific may be found through statistical techniques to correlate with a gender (Factor II.4). Selecting such correlates could result in Google showing the ads more to men by attempting to satisfy The Barrett Group's request to target the correlates.

The conditions of the 'Gender and Jobs' experiment, however, allow us to probe this issue further. The simulated users lacked other attributes which could be correlated with gender. They all engaged in the same behavior. As a result, we can rule out that the difference in received advertisements resulted from differences in age, affluence, or prior work experience. If Google inferred these attributes from user behavior, all thousand users should have resulted in identical inferences. The only differentiating data in these experimental conditions was gender. If Google did use correlations with gender, it used correlations found in other populations of real consumers and applied them to synthetic population in the 'Gender and Jobs' experiment.

5.4.4 Decisions of Other Advertisers

Other advertisers can influence the targeting of an advertiser, such as The Barrett Group, through the selection of their ad auction bids. This possibility was raised by Google itself [137]. If advertisers other than The Barrett Group were willing to pay more to reach women users, The Barrett Group's advertisement may have ended up predominantly being served to men (Factor I.3). If advertisers in general consider female consumers to be a more valuable demographic, they would set higher bids to advertise to them. As a result, if an advertiser, like The Barrett Group, sets equal bids for men and women, it could end up only reaching men if it is outbid by other ads for female users. The real-time auction makes it difficult for advertisers to figure out how to treat protected classes similarly.

In their study of Facebook ads, [79] suspect that the higher competition for reaching younger women was the reason behind lower impressions of job-related ads for women than men, in spite of gender-neutral targeting criteria and bids.

5.4.5 Behavior of Other Consumers

User behavior could also play a role in the disparate ad results in the 'Gender and Jobs' experiment. Google's understanding of which users are likely to respond to an ad is built off observations of millions of users' behavior (Factor I.4). For example, Google could have found that (a) males are more likely to click this ad than females are, (b) females are more likely to click other ads than this ad, or (c) there exist other contemporary ads that females are more likely to click than males are. Google's computers may have targeted The Barrett Group's ads in response to one of the above findings to increase revenue. For example, suppose The Barrett Group pays Google per click (i.e. using the cost-per-click bidding strategy), then ad serving models that are developed over time to maximize Google's revenue may end up serving the ads to more men and fewer women.

To the extent user behavior over all expresses sex stereotypical responses to ads about job opportunities, using their behavior as an input risks building the product of sexist hiring practices, and general employment inequality, into the targeting. For example we know there are fewer women currently in the executive tier of companies so they may self-select away from The Barrett Group ad, while males who are over-represented in the executive tier may aggressively click on it. Using these inputs to target ads constrains women's access to job advertisements based on prior patterns of discrimination and inequality reflected in the stereotypical responses of women as a whole. [135], who found disparate serving of ads indicating arrest records based on the race-affiliation of first names, also suggested that user inputs may have resulted in the disparity. After a conversation with the advertiser who claimed to have provided the same ad text to Google for groups of last names, she hypothesized that the bias in served ads might result from a society that "clicked ads suggestive of arrest more often for black identifying names".

5.4.6 Limitations

The above possibilities are by no means exhaustive. In addition to variations of the above, there exist also completely different possibilities, such as the difference arising solely due to an error in Google's code or even from malicious outsiders (e.g., hackers) purposefully manipulating Google's computer systems.

We have seen that each actor in the advertising ecosystem may have contributed inputs that produced the discrepancy in ads in the 'Gender and Jobs' experiment. We make these guesses based on our model the ad ecosystem developed from Google's help pages. Without additional information it is impossible for us to know what actors—other than the users receiving the ads—did or did not do. It is also impossible to assess whether the advertisers or Google intended to target advertisements based on gender, or whether they were unaware such gendered distribution was occurring. In several instances, answers to these questions would be necessary to assess the extent, if any, of legal liability. As we discuss in Section 5.5 below, in two instances, however, we can draw conclusions about legal liability without assessing intent or knowledge. Liability for violating the advertising prohibition does not turn on intent; it is essentially a strict liability standard.[8] Second, Section 230 of the Communications Decency Act limits the liability interactive computer services face for content that they have not developed or created. The size of the shield against liability §230 provides to Google can be assessed without consideration of intent, and, combined with the text of Section 704(b), creates an important limit on Google's exposure.

5.5 Title VII and Prohibitions on Discriminatory Advertising

Title VII of the 1964 Civil Rights Act makes it unlawful to discriminate on the basis of race, color, national origin, religion, or sex in several stages of employment. Title VII also prohibits employers, labor organizations, employment agencies, and joint labor-management committees from engaging in advertising that indicates a preference based on sex:

> It shall be an unlawful employment practice . . . to print or publish or cause to be printed or published any notice or advertisement relating to employment by such an employer [or other entity covered by the statute], or relating to any classification or referral for employment by [such entity] . . . indicating any preference, limitation, specification, or discrimination based on race, color, religion, sex, or national origin.[9]

Despite its clarity, Section 704(b) did not put an end to discriminatory advertisements. This was exemplified by the fact that sex-specific help wanted columns in newspapers persisted for nearly a decade after the law took effect. During these years, the EEOC revised its interpretation of the law twice until finally, under pressure from women's organizations,[10] it adopted a flat prohibition on the use of sex-specific advertising columns by covered entities.[11]

[8] Housing Statements and §3604(c): A New Look at the Fair Housing Act's Most Intriguing Provision, 29 Fordham Urb. L.J. 187, 215-16 (2001) (describing parallel advertising prohibition as "essentially a strict liability" standard)

[9] Section 704(b) of Title VII of the Civil Rights Act of 1964, codified at 42 USC §2000e-3(b) (gender alone may be used where it is a bona fide occupational qualification for employment).

[10] The EEOC's initial advertising guidelines were one impetus for founding the National Organization for Women, and one of NOW's first actions was to challenge them. See Pedriana and Abraham, "Now You See Them, Now You Don't: The Legal Field and Newspaper Desegregation of Sex & Segregated Help Wanted Ads 1965-75", op. cit. See also NOW's history, http://now.org/about/history/highlights/ In addition to challenging the EEOC's guidelines, women and women's organizations sought to bring newspaper job advertisements fully under the purview of the statute arguing that they acted as covered "employment agencies" under Title VII. This argument was rejected in *Brush v. San Francisco Publishing Co.*, 31.5 F.Supp. 577 (N.D. Cal. 1970) based largely on legislative history. Kerman, Peter W. "Sex Discrimination in Help Wanted Advertising." 15 Santa Clara L. Rev. 183, 194 (1974)

[11] 29 C.F.R. §1604.5 "It is a violation of title VII for a help-wanted advertisement to indicate a preference, limitation, specification, or discrimination based on sex unless sex is a bona fide occupational qualification for the particular job involved. The placement of an advertisement in columns classified by publishers on the basis of sex, such as columns headed "Male" or "Female," will be considered an expression of a preference, limitation, specification, or discrimination based on sex." 33 Fed. Reg. 11539 (1968). Note

States and localities were busy during this period as well, adopting non-discrimination laws containing aiding and abetting provisions that were aimed in part at prohibiting newspapers from segregating help wanted advertisements into sex-specific sections. These laws complimented Title VII, which did not regulate newspapers, and therefore could not limit them from creating such columns. In *Pittsburgh Press Co. v. Human Rel. Comm'n*, the Supreme Court ultimately upheld the application of the aiding and abetting provisions of these state laws to newspapers as a permissible limitation on commercial speech.

While there is limited case law interpreting the advertising prohibition in Title VII, the significant case law and guidance informing the application of a similar provision of the Fair Housing Act (FHA)[12] offers guidance on its scope. Both the statutory parallels and shared objectives of Title VII and the FHA suggest that the FHA case law and the guidance documents issued by the US Department of Housing and Urban Development (HUD) interpreting the FHA's prohibition on discriminatory advertisements,[13] provide a reasonable resource for contemplating the interpretation of Section 704(b), and its potential application to online behavioral advertising.

There are many ways to indicate improper preferences through advertising. These include not only the written or visual text of the ads, but also the ways in which advertisements are distributed or targeted. The explicit prohibition on sex-specific advertising columns in Title VII are one example of the way in which improper preferences can be revealed outside the text of the advertisement itself. In the context of Title VII, courts have found that a city's "refusal to publicize jobs outside [a] racially homogenous [white] county" was evidence of discrimination.[14] In the fair housing context,[15] regulations issued by HUD confirm that such targeting can indicate an illegal preference, stating that "selecting media or locations for advertising. . .which deny particular segments of the housing market information" or "refusing to publish advertising for the sale or rental of dwellings. . ." because of a protected class indicates a discriminatory preference.[16] Other activities that can indicate a discriminatory preference include publishing advertisements exclusively in a language other than English[17] and indicating a language preference, which could mask a preference for people of a specific national origin.[18]

this prohibition did not extend to newspapers as publishers of advertisements, but focused on the behavior of regulated employers and employment agencies.

[12] 42 USC §3601 et seq.

[13] 42 USC §3608.

[14] United States v. City of Warren, MI, 138 F.3d 1083 (6th Cir. 1998).

[15] Given the statutory parallels and shared objectives of Title VII and the FHA, an examination of the FHA case law and the guidance documents issued by HUD interpreting the parallel prohibition on advertisements (42 USC §3608 (2014)), provides useful insight on how 2000e-3 could be interpreted, and its potential application in the context of online behavioral advertising.

[16] 24 C.F.R. §100.75

[17] Hous. Rights Ctr. v. Sterling, 404 F. Supp. 2d 1179, 1193-94 (C.D. Cal. 2004) (notices and banners in Korean would suggest to the ordinary reader a racial preference for Korean tenants.)

[18] Holmgren v. Little Village Community Rptr., 342 F. Supp. 512, 513 (N.D. Ill. 1971)

5.5.1 Scope of Title VII

Before turning to an analysis of Section 704(b), it is important to note that the law creates a significant limitation on avenues for relief under the results of the 'Gender and Jobs' experiment. Section 704(b) only prohibits certain kinds of entities from printing or publishing discriminatory advertisements: employers, labor organizations, employment agencies, and joint labor-management committees. For this prohibition to apply to the ads, The Barrett Group, Google, or both would have to fall within the definition of one of these entities.

The Barret Group describes itself as an executive career coaching service and it does not appear to be affiliated with or promise to procure opportunities to work for particular firms, so it seems unlikely to be considered an employment agency. Additionally, there is no evidence that The Barret Group is affiliated with a joint labor-management committee. Although Google's vastly complex structure and the difficulty of knowing exactly what ads run through the platform make it difficult to be certain, we believe it is unlikely that Google would be considered an "employment agency"—an entity "regularly undertaking with or without compensation to procure employees for an employer or to procure for employees opportunities to work for an employer"[19]—given the availability of specialized platforms such as Craigslist's online classifieds, LinkedIn's professional networking platform, and Monster.com's job boards. We therefore set aside this question, but we note that in addition to Federal civil rights law, laws in several states including California, New York and Pennsylvania, prohibit any person from aiding, abetting, inciting, compelling, or coercing discriminatory employment practices. These laws create potential liability for Google if its services are used by covered entities to target ads based on gender or other protected class.[20]

Despite our conclusion that Title VII is unlikely to reach The Barrett Group or Google under the facts of the 'Gender and Jobs' experiment, we believe it is useful to consider whether the law could create liability for advertising platforms under the similar but broadly applicable provision in the FHA, which we discuss in more detail below. This analysis requires exploring the various ways in which an illegal preference could be communicated to the public and how those variations interact with the prohibition in Section 230 on holding Interactive Computer Services, such as ad platforms, liable for content.

[19] 42 USC §2000e(c).

[20] Nat'l Org. for Women v. State Div. of Human Rights, 314 N.E.2d 867, 870–71 (Ct. App. N.Y. 1974); Pittsburgh Press Co. v. Pittsburgh Comm'n on Human Relations, 287 A.2d 161, 169 (Pa. Cmmwlth. 1972); Alch v. Superior Court, 122 Cal. App. 4th 339, 389 n.48 (2004).

5.5.2 Ad Content and Ad Targeting

Courts analyze advertisements based on whether a reader (or listener or viewer) would interpret the advertisement to convey a preference based on a protected class.[21] Because the statute focuses on the perspective of the recipient, the intent behind the content or targeting of the ad is not relevant to whether it violates Section 704(b).[22] This is an important factor in the online behavioral advertising environment. The FHA case law connects the ordinary reader standard to the prohibition on sex-designated advertising columns by explaining that advertisements that exclusively feature white models "may discourage black people from pursuing housing opportunities by conveying a racial message in much the same way that the sex-designated columns. . .furthered illegal employment discrimination."[23] While informational models used to target specific populations make the expression of preference more difficult to see in one sense—the advertisements are literally withheld from the undesired class—the 'Gender and Jobs' experiment reveals that such targeting communicates a preference more effectively than "subtle methods of indicating racial preferences"[24] already barred by courts.

In sum, the ban on sex-specific advertising columns, case law, and guidance provided under Title VII and the FHA aim to limit both *content* and *activities* that target advertisements based on protected attributes. The regulations and case law limit activities that direct information about employment and housing opportunities to or away from individuals based on membership in a protected class. Advertisements can run afoul of Title VII both substantively through content choices, and procedurally through publishing decisions that affect the literal availability of advertisements to different recipients, or otherwise indicate an illegal preference.

Our focus in this analysis is not on the content of the ads identified in the 'Gender and Jobs' experiment. These ads appear neutral using phrases such as "$200k+ Jobs—Execs Only" or "Goodwill—Hiring".

Instead we explore how advertising platforms create new risks that access to information about job or housing opportunities will vary based on protected status, regardless of the intent of advertisers, and consider how such targeting would be dealt with differently under two key civil rights laws. Such targeted advertising—whether it involves placing neutral ads in sex-segregated columns[25] or advertising

[21] Rodriguez v. Vill. Green Realty, Inc., 788 F.3d 31, 53 (2d Cir. 2015) ("What matters is whether the challenged statements convey a prohibited preference or discrimination to the ordinary listener."); Ragin v. New York Times Co., 923 F.2d 995, 999-1000 (2d Cir. 1991) (explaining that readers can "infer a racial message from advertisements that are more subtle than the hypothetical swastika or burning cross").

[22] Capaci v. Katz & Besthoff, Inc., 711 F.2d 647, 660 (5th Cir. 1983) (finding an ad violated the act although the "practices in composing and placing ads were not to carry out any policy of discrimination against women, but to achieve the best results from the ads in light of her experience as to the gender which would be more interested in the job vacancy being advertised")

[23] Ragin v. New York Times Co., 923 F.2d 995, 1003 (emphasis added).

[24] Ragin v. New York Times Co., 923 F.2d 995, 1000.

[25] Pittsburgh Press Co. v. Pittsburgh Comm'n on Human Relations, 413 U.S. 376 (1973) (Supreme Court affirmed that sex-segregated columns for employment ads in a newspaper were in and of themselves discriminatory, even if the specific text of the ads was sex-neutral).

only to a certain demographic—can be just as damaging to equal opportunity as an employment ad that says "Women Need Not Apply." Thus, our concern is with dissemination choices that convey unlawful preferences regardless of an ad's content.

5.5.3 Online Ads and Civil Rights

The world of *Pittsburgh Press Co.* and similar cases, where prohibited preferences and discrimination were painfully obvious in the form of sex-segregated help-wanted columns, is long gone. Increasingly, advertisements are moving online and are being handled by large advertising platforms such as Google and Facebook.[26] These companies are generally considered to be "interactive computer services" and protected from liability as a publisher or speaker of content created and developed by others under Section 230 of the Communications Decency Act (CDA).[27]

The involvement of interactive computer services in distributing ads raises the question of the relationship between activities civil rights law prohibits—dissemination choices, venue selection, and/or steering (rather than textual indications of preference)—and the prohibition in Section 230 on holding people liable as publishers of content they did not create or develop.[28] In particular, like other recent cases involving civil rights statutes and Section 230, it raises a question of whether any of the functionality offered to third parties to indicate preferences, or independent activities conducted by advertising platforms that determine who sees advertisements, rise to co-development of the advertisement.

To answer this question, we must examine the connection between publishing and advertising. Section 230(c)(1) protects interactive computer services from publisher liability even where those services might be engaging in activity traditionally associated with publishers, such as editing or removing content. Interpretation of the term "publish" in the Fair Housing context suggests that targeting advertisements is publishing activity, and can independently indicate an illegal preference. For example, in *Mayers v. Ridley*, "publishing" of a discriminatory statement was found where a Recorder of Deeds collected restrictive covenants "in a manner that facilitates access to them by prospective buyers."[29] More broadly, the court noted that "the proscription against 'publication' should therefore be read...to bar all devices for making public racial preferences in the sale of real estate, whether or not they involve the printing

[26] Between 2000 and 2015, print newspaper advertising revenue fell 65% (from around $60 billion to around $20 billion). Derek Thompson, The Print Apocalypse and How to Survive It, The Atlantic (Nov. 3, 2016).

[27] 47 USC §230(c)(1).

[28] There is also a critical question of whether Section 704(b) would apply to Google at all, given that it probably does not fall within any of the categories listed in the statute (employer, labor organization, employment agency, or joint labor-management committee). For purposes of this analysis we will set this question aside. Courts and the EEOC eventually concluded that newspapers were prohibited from displaying sex-segregated ads, despite the fact that newspapers are not included as a category in Section 704(b). Most corresponding state statutes prohibit aiding and abetting discriminatory ads, which provides another potential avenue for arguing that Google is subject to civil rights obligations.

[29] Mayers v. Ridley, 465 F.2d 630, 633 (D.C. Cir. 1972).

process."[30]

As we demonstrate, The Barrett Group's ads disproportionately targeted men. But, did that targeting indicate a preference for men? Unlike the gendered help wanted columns, the classifier used to target ads was not revealed in written text to the recipients. It is possible that a recipient of a Barrett Group advertisement might have noted an "about this ad" symbol next to it, clicked on it, and received some information about why they received the advertisement. Another possibility is a user might have looked at their ad preferences and noted that they were identified as male and assumed that The Barrett Group advertisement was being targeted to them based on that criteria.

As discussed above, indications of illegal preference can be conveyed to users in more subtle and less literal ways. Targeting that has the effect of limiting an audience in a discriminatory way, even though it does not convey a preference within the advertisement itself, is addressed by both regulations and case law. The Barrett Group claims to have targeted their advertisement to those older than 45[31], receiving high pay, and of executive-level experience [137]. It is less clear whether they indicated a preference for men over women. As discussed above, the compliance manual states that "employers are prohibited from structuring their job advertisements in such a way as to indicate that a group or groups of people would be excluded from consideration for employment."[32]

It seems that the choices The Barrett Group made could be viewed as an indication that men were preferred over women for certain jobs. We noted that HUD regulations state that "selecting media or locations for advertising which deny particular segments of the housing market information" because of a protected class indicates a discriminatory preference.[33] The input selections made by The Barrett Group denied particular segments of the market, women, information about a job-related opportunity. However, assuming The Barrett Group was truthful about the inputs, it is unclear whether the EEOC would find that in doing so they indicated a discriminatory preference.

It would seem an odd outcome if employers prohibited from advertising in gender specific help wanted columns that signaled gender preference but were at least practically available for all readers to peruse, could engage in a similar practice only with the classifier obscured.[34]

[30] Mayers v. Ridley, 465 F.2d 630, 633 (D.C. Cir. 1972). See also United States v. City of Warren, MI, 138 F.3d 1083 (6th Cir. 1998) (city violated Title VII by purchasing recruitment ads in publications with disproportionately white readers); Hous. Rights Ctr. v. Sterling, 404 F. Supp. 2d 1179, 1193-94 (C.D. Cal. 2004) (notices and banners in Korean would suggest to the ordinary reader a racial preference for Korean tenants); Holmgren v. Little Village Community Rptr., 342 F. Supp. 512, 513 (N.D. Ill. 1971) (defendant newspapers violated FHA prohibition on discriminatory advertising by publishing ads indicating a preference for buyers or tenants that spoke a particular language).

[31] We note, but don't address, that the targeting criteria on its face expresses an age preference which is an independent violation of the Age Discrimination in Employment Act, which prohibits publishing an "advertisement indicating preference. . . based on age" (29 U.S.C.A. §623).

[32] EEOC Compliance Manual Vol. 2, Sec. 632.2(a).

[33] 24 C.F.R. §100.75.

[34] The concept of steering under the FHA provides another way to articulate concerns with the outputs of online behavioral advertising systems. It addresses issues such as withholding information from certain groups of individuals.

Google's targeting might also suggest a discriminatory preference to ad recipients. For example, if gender is the sole attribute in a user's ad settings, the user might conclude that it is the feature on which job-related ads (and all others) are targeted to them. Even when an ad is delivered to everyone on the advertising platform, an ordinary user might perceive it to be targeted to their gender given the limited transparency they have into its full functioning. Admittedly, this may argue too much, but the standard focuses on the perception of the ordinary reader or listener. But again, the existing law addresses targeting that more subtly conveys a preference. Whether an online ad platform targeted on gender explicitly or on attributes that correlated to it, that targeting would skew who learned about an opportunity.

5.5.4 CDA §230 and Google's Ad Platform

Assuming then that holding an entity liable for targeting advertisements is holding them liable as a publisher where the entity at issue is an interactive computer service, Section 230 comes into play. Interactive computer services are protected from liability for content created by others. However, if an interactive computer service materially contributes to the development of discriminatory content they are treated like an "information content provider,"[35] and lose the protection §230 offers.[36]

Generally, entities are treated as an interactive computer service (ICS) if they provide "neutral tools" that others use to create discriminatory content. For example, Craigslist was protected against claims under the Fair Housing Act based on user-generated ads that violated the FHA because "[n]othing in the service Craigslist offers induces anyone to post any particular listing or express a preference for discrimination."[37] Similarly, where a defendant website provided an online questionnaire that was used to publish allegedly defamatory content, but left the selection of content exclusively to a third party, Section 230 provided immunity.[38]

In contrast, if the ICSes materially contribute to the discriminatory aspect of content, they are not protected by Section 230. Thus a website designed to match people with available housing was considered a content provider because it required each user to answer a series of questions disclosing his sex, sexual orientation and whether he would bring children to a household and compiled this information into a profile page that displayed the descriptions and preferences of users gleaned from the questions.[39] By forcing "subscribers to provide the information as a condition of accessing its service" and providing

[35] Information content providers are defined as "any person or entity that is responsible, in whole or in part, for the creation or development of information provided through the Internet or any other interactive computer service."

[36] See Fair Housing Council of San Fernando Valley v. Roommates.com, LLC, 521 F.3d 1157 (9th Cir. 2008).

[37] Chicago Lawyers' Comm. for Civil Rights Under Law, Inc. v. Craigslist, Inc., 519 F.3d 666 (7th Cir. 2008), as amended (May 2, 2008).

[38] Carafano v. Metrosplash.com, Inc., 339 F.3d 1119, 1124 (9th Cir. 2003).

[39] Fair Hous. Council of San Fernando Valley v. Roommates.com, LLC, 521 F.3d 1157, 1164 (9th Cir. 2008).

"a limited set of pre-populated answers," the website became "much more than a passive transmitter of information provided by others; it becomes the developer, at least in part, of that information." [40]

Courts have found that Google's ad serving platform meets the definition of an ICS.[41] With regard to Google's advertising platform, the question whether Google's actions go beyond those typical of an ICS and into those that would be associated with an information content provider is highly contextual. The AdWords platform is a black box mechanism that makes it difficult to identify who is responsible for discriminatory outputs. Below we discuss four potential scenarios which reveal how potential legal liability shifts depending upon how targeting occurs.

Possible Causes of Ad Targeting

We now go through the possible causes of disparate ad targeting outlined in Section 5.4 and explore the legal ramifications of each of them.

(1) Disparate targeting was fully a product of the advertiser selecting gender segmentation. In this scenario, Google is probably not creating or developing content. Instead, by allowing, but not requiring, advertisers to choose to target their ads to men or women, Google is providing a "neutral tool" that is protected by Section 230.[42] This tool allows third parties to determine who receives their ads, which is likely protected as a publisher function. Policies that allow advertisers to control who sees their ads are "precisely the sort of website policies and practices" to which "section 230(c)(1) extends."[43] In sum, in this scenario, the ads and who they target is information "provided [to Google] by another information content provider",[44] making Google not liable even if misused under a generally applicable provision like that of the FHA.[45]

(2) Disparate targeting was fully a product of machine learning—Google alone selects gender. In this scenario, Google, and not the advertiser, is doing the targeting. Google, using programs that are part of its AdWords platform, decides who sees an ad based on Google's opinion of who is most likely to click

[40] Fair Hous. Council of San Fernando Valley v. Roommates.com, LLC, 521 F.3d 1157, 1166 (9th Cir. 2008).

[41] Rosetta Stone Ltd. v. Google Inc., 732 F. Supp. 2d 628, 632 (E.D. Va. 2010) (claim against Google for unjust enrichment based on its practice of allowing trademarks to appear on its AdWords advertising platform was barred because in that context "Google is no more than an interactive computer service provider"); Goddard v. Google, Inc., 640 F. Supp. 2d 1193, 1198 (N.D. Cal. 2009) (allegations based on keywords in Google's AdWords advertisements were barred because Keyword Tool was a neutral tool permitted within the scope of CDA immunity).

[42] See Carafano v. Metrosplash.com, Inc., 339 F.3d 1119, 1121 (9th Cir. 2003); Fair Hous. Council of San Fernando Valley v. Roommates.com, LLC, 521 F.3d 1157, 1169 (9th Cir. 2008).

[43] Jane Doe No. 1 v. Backpage.com, LLC, 817 F.3d 12, 20 (1st Cir. 2016).

[44] 47 USC §230(c)(1).

[45] This conclusion is troubling in light of the increasing stratification of recipients that ad platforms, such as Google and Facebook, are able to achieve. For example, ProPublica reported in 2016 that Facebook provides advertisers with the option to exclude groups using "Ethnic Affinities," which included categories such as "African American" and "Asian American" [8]. Under current interpretations of Section 230, Facebook may avoid liability for providing these options if they are considered a neutral tool.

on it. Advertisers are not part of the decision, and in fact they may be unaware that such a decision is being made. This is critical for understanding the application of Section 230 to an ad platform's activities.

Courts have adopted a "material contribution test to determine whether a website operator is 'responsible, in whole or in part, for the creation or development of'" unlawful information.[46] Under this test, a "material contribution to the alleged illegality of the content . . . means being responsible for what makes the displayed content allegedly unlawful."[47] As a court in the fair housing context has put it, "[c]ausation in a statute such as § 3604(c) must refer to causing a particular statement to be made, or perhaps the discriminatory content of a statement."[48]

In this scenario, the "content" that would implicate Section 704(b) is the targeting of the advertisement. And if the advertiser has not selected for gender, Google decides to target the ads toward individuals based on whether they are men or women. This is different from providing a neutral tool such as allowing the advertiser the option to select for gender or suggesting keywords. Instead, Google alone is responsible for targeting certain employment ads toward men and away from women, and it is the targeting itself—irrespective of the content of the ad—that indicates a preference that would violate Section 704(b). As a result, Google is making a material contribution to the publishing enterprise. It is responsible, at least in part, for the creation or development of information and therefore would not be protected by Section 230. Because Google is likely outside the scope of Title VII, there is no risk of liability for the results in the 'Gender and Jobs' experiment. However, under the FHA, targeting that arose in this way would be the material basis for the illegality of an otherwise facially neutral ad.

(3) Disparate targeting was fully a product of the advertiser selecting keywords. Courts have examined Google's keyword suggestions in the context of its Sponsored Link ad program and determined that it is a neutral tool that does not rise to the level of information creation or development.[49] In Goddard for example the court determined that the choice of keyword ultimately falls to the advertiser, and Google "does nothing more than provide options that advertisers may adopt or reject at their discretion."[50] That does not mean, however, that the "keyword tool" is per se neutral. In a situation where Google uses

[46] Jones v. Dirty World Entm't Recordings LLC, 755 F.3d 398, 413 (6th Cir. 2014) (quoting 47 USC § 230(f)(3)).

[47] Jones v. Dirty World Entm't Recordings LLC, 755 F.3d 398, 410 (6th Cir. 2014).

[48] Chicago Lawyers' Comm. for Civil Rights Under Law, Inc. v. Craigslist, Inc., 519 F.3d 666, 671 (7th Cir. 2008).

[49] Goddard v. Google, Inc., 640 F. Supp. 2d 1193, 1197 (N.D. Cal. 2009); Jurin v. Google, Inc., 695 F. Supp. 2d 1117, 1119, 1123 (E.D. Cal. 2010). Rosetta Stone Ltd. v. Google Inc., 732 F. Supp. 2d 628, 630 (E.D. Va. 2010). But see 800-JR Cigar, Inc. v. GoTo.com Inc., 437 F. Supp. 2d 273 (D.N.J. 2006) (holding that a keyword tool was not entitled to Section 230 immunity "because the alleged fraud is the use of the trademark name in the bidding process, and not solely the information from third parties that appears on the search results page").

[50] Goddard v. Google, Inc., 640 F. Supp. 2d 1193, 1198, 1199 (N.D. Cal. 2009). That "neutral tool" concept was used extensively in Fair Hous. Council of San Fernando Valley v. Roommates.Com, LLC, 521 F.3d 1157, 1169 (9th Cir. 2008) to determine whether a website has engaged in "development" that would negate §230 protection: "providing neutral tools to carry out what may be unlawful or illicit searches does not amount to 'development' for purposes of the immunity exception". "To be sure, the website provided neutral tools, which the anonymous dastard used to publish the libel, but the website did absolutely nothing to encourage the posting of defamatory content – indeed, the defamatory posting was contrary to the website's express policies."

keywords to target ads (as opposed to serving them up in Sponsored Links), such targeting may rise to the level of a material contribution along the lines of Scenario (2) above. The specific keywords chosen and the role they play in targeting would be key in determining whether a material contribution had been made.

(4) Disparate targeting was fully the product of the advertiser being outbid for women. Another possible situation is where a job advertisement does not reach women because other advertisers win the auctions for those ad placements. In this scenario, third parties are involved to some extent because they are selecting the price they are willing to pay for an ad placement. Nevertheless, Google would bear the same responsibility as if the bidding did not occur at all. That is because ultimately, the decision about where to place the ad is made by Google. The advertisers make a decision about how much they will pay, but they have no say over who finally sees an ad. This is true even if an advertiser tried to target its employment ads toward women. Google is therefore still in the position of doing the targeting that makes the most material contribution to employment ads that express a preference for men by virtue of the fact that it decides to show more of these ads to men than to women.

Again, although there is no Title VII liability, under the FHA this could be a basis for liability, and existing case law suggests that Section 230 would not provide a barrier.

5.5.5 Limitations

Our analysis is limited due to our black box access to the workings of the system. We consider hypothetical scenarios that represent guesses at how the AdWords system might work. This is a non-exhaustive list of ways in which the results in the 'Gender and Jobs' experiment may have come about. In fact, as discussed in depth in Section 5.4 above, it is likely that the arose from a combination of advertiser and platform choices. Identifying how various actors produced the results requires inside information that we do not possess. As a result, we are forced to make assumptions about whether liability would arise and how it would be apportioned.

Ultimately, given the parameters of the research in this case and the applicable statutes, it does not appear that any violations of law have occurred. Google would first have to fall under the coverage of Section 704(b), and, as currently drafted, that does not appear to be the case. However, in other circumstances, such as housing, we argue that the act of targeting itself could be considered a contribution to development of illegal content and under other statutes, specifically the FHA, could create a risk of liability.

5.6 Conclusions

Using information flow experiments, we found discriminatory outputs from Google's advertising system. Less clear is why or how it happened. We have presented a selection of possible causes, but cannot without further access to Google's advertising system determine which is the actual cause. Analyzing potential legal liability under civil rights law requires an understanding of the entities covered by the law, as well as how discriminatory outputs arose.

Our analysis of existing case law concludes that Section 230 may not immunize advertising platforms from liability under the FHA for algorithmic targeting of advertisements that indicate a preference for or against a protected class. We argue that, in cases where an advertising platform, rather than the advertiser, makes the key decisions resulting in the ad being shown in different rates to members and non-members of a protected class, the ad platform becomes a co-developer of the ad, thereby losing its immunity. However, only some of possible targeting scenarios would satisfy this condition.

Although Section 230 poses the most obvious hurdle for holding online platforms such as Google liable, it turns out that in the setting of the 'Gender and Jobs' experiment, the narrow scope of Title VII itself is a more formidable hurdle. By only applying to a tightly scoped set of employment-related entities, none of which Google appears to be acting as while serving ads, Title VII would not apply.

Our analysis reveals that advertisers covered by Title VII and the FHA using online algorithmically driven black-box advertising platforms face a dilemma: on the one hand they are bound to avoid advertising that infers a preference, but on the other, they cannot independently control the demographic makeup of those receiving their advertisements. The assumption has been that the advertising platforms which have the capacity to control the demographics of an advertising campaign were beyond the reach of antidiscrimination law due to Section 230's preclusion of holding interactive computer service providers liable for content created and developed by others. Our analysis reveals that Section 230 may not preclude liability in all instances. This is because targeting produced by platform algorithms that contributes to the illegality of an advertisement—its expression of a preference for or against a protected class—could be considered development under existing case law, thereby opening up the possibility of advertising platforms being found liable under the FHA. However, the inherent coverage limits in Title VII constrain the types of advertising platforms that might face liability (they would need to meet the statutory definition of an employment agency or other entity listed in Section 704(b)). Advertisers should be aware of the ways in which advertising platform algorithms can introduce bias into advertising campaigns, advertising platforms should provide ways to ensure advertisers can reach demographically diverse audiences where the law demands that they do so, and policymakers should consider whether the narrow scope of Title VII's advertising provision is fit for purpose in today's advertising ecosystem.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

In this dissertation, we develop a rigorous methodology to study of fairness and privacy violations in personalization systems. We apply the methods to study discrimination, transparency, and choice on Google's advertising system. Our experiments find evidence of gender-based discrimination in employment-related ads and opacity in Google's transparency tool. Our findings demonstrate the potential loss of societal values in a world where we are surrounded by automated decision systems. Our results renewed research interest in discrimination (e.g., [17,78]) and transparency in automated systems (e.g., [19,30]). Consumers seeking an immediate defense against data collection by personalization systems can look to the Tor Browser, which we found to be the most effective privacy enhancing technology against advanced forms of tracking, specifically fingerprinting. Our legal analyses reveal that an advertising platform like Google is unlikely to incur liability under Title VII for serving employment-related ads disparately to different genders. We identify possible revisions to the language of Title VII to make it more broadly applicable to personalization systems in this age of extreme personalization. We find that when laws are applicable, such as the Fair Housing Act to targeted housing ads, these systems may be held liable.

6.2 Future Work

We first discuss some possible directions for future work in three main categories: We discuss possible directions for future work in three main categories: the first direction involves extending information flow experiments to other settings, the second involves studying different personalization systems, possibly under different access models, the third informs better alternatives to evaluate privacy enhancing technologies.

6.2.1 Information Flow Experiments

Demonstrating Noninterference. The permutation test requires that the null hypothesis be that the system has noninterference. Thus, it can only provide a quantitative measure of the evidence *against* noninterference. Conceptually, proving noninterference would require looking at every test statistic under every input sequence. Since examining an infinite set of sequences is impossible, using the scientific method to show that a system has noninterference would require building a theory of the system's operation and then proving noninterference in that theory.

Monitoring and Observational Studies. Passive monitoring in IFA corresponds to observational studies. A wide range of work deals with the cases under which one can infer causation from a correlation learned from an observational study (see, e.g., [114]). Future work can import these results to IFA showing how monitoring could be useful in some cases despite its inherit unsoundness [94,127,145].

6.2.2 Study of personalization systems

Extensions of AdFisher. We would like to extend AdFisher to study information flow on other advertising systems like Facebook, Bing, or Gmail. We would also like to analyze other kinds of ads like image or flash ads. We also plan to use the tool to detect price discrimination on sites like Amazon or Kayak, or find differences in suggested posts on blogs and news websites, based on past user behavior. We have already mentioned the interesting problem of how ad networks can ensure that their policies are respected by advertisers (§3.7).

Blame Assignment in Personalization Violations. We also like to assign blame for personalization violations. However, doing so is often difficult, especially in black box settings. We explore blame assignment from a legal point of view for some possible causes of discriminatory results in Chapter 5. However, personalized results may be based on many different signals coming from a variety of sources and the decision made through a series internal computations. A white box analysis of these personalization platforms may enable more fine-grained blame assignment to specific internal computations.

6.2.3 PET Evaluation

Evaluation of predictability. Our evaluations do not evaluate the impact of fluctuation strategies on the predictability of fingerprints. For poorly designed fluctuation strategies, the spoofed values may retain certain patterns which may be predictable and unique. The predictability also depends on the tracker abilities and how well it can extract patterns from the fluctuating fingerprints.

Comparison of standardization and fluctuation strategies. It is not clear whether standardization or fluctuation of attributes is the better approach for a defense. Our evaluations equate fluctuating an attribute with fully standardizing it to one value, but may not be the case. While fluctuation strategies may leave fingerprints predictable as well as unique, they have more flexibility in choosing a value to minimize usability impact (e.g. [81]). On the other hand, standardized attributes have less flexibility and may have a higher usability impact. It would be interesting to see how these various strategies compare on uniqueness, predictability, and usability.

Evaluation of dynamic attributes. Many fingerprintable attributes like the audio context and the battery status are dynamic in nature. These attributes do not have identical values, rather vary predictably over time. Trackers try to extract consistent patterns from devices from these dynamic attributes. How different PET strategies stand up against dynamic attributes remains to be seen.

Appendix A

Appendices for A Methodology for Information Flow Experiments

A.1 System Formalism

For a finite set, let $\Delta(X)$ be the set of distributions over X. Let $\delta(x)$ be the degenerate distribution assigning probability 1 to x. Let $[]$ be the empty list. Let $\vec{\imath}{\cdot}i$ be the list created by appending i to $\vec{\imath}$, and let $i{\cdot}\vec{\imath}$ be the list created by prepending i to $\vec{\imath}$.

Let a probabilistic Moore Machine be $Q = \langle \mathcal{S}, s_0, \mathcal{I}, \mathcal{O}, \tau, \sigma \rangle$ where S is a finite set of states, s_0 is the initial state, $\mathcal{I}$ is a finite input set, $\mathcal{O}$ is a finite output set, $\tau : \mathcal{S} \times \mathcal{I} \to \Delta(\mathcal{S})$ is the state transition function, and $\sigma : \mathcal{S} \to \mathcal{O}$ is the output function.

Let $Q(s, \vec{\imath})(\vec{o}, \vec{s})$ be the probability of the seeing the trace $\vec{s}[1], \vec{o}[1], \vec{\imath}[1], \vec{s}[2], \vec{o}[2], \vec{\imath}[2], \ldots, \vec{s}[k], \vec{o}[k], \vec{\imath}[k]$, $\vec{s}[k+1], \vec{o}[k+1]$:

$$Q(s, [])([\sigma(s)], [s]) = 1 \tag{A.1}$$

$$Q(s, i{\cdot}\vec{\imath})(\sigma(s){\cdot}\vec{o}, s{\cdot}\vec{s}) = \sum_{s'} \tau(s, i)(s') * Q(s', \vec{\imath})(\vec{o}, \vec{s}) \tag{A.2}$$

$$Q(s, \vec{\imath})(\vec{o}, \vec{s}) = 0 \qquad \text{otherwise} \tag{A.3}$$

We take the distribution $Q(\vec{\imath})$ over outputs to such that $Q(\vec{\imath})(\vec{o}) = \sum_{\vec{s}} Q(s_0, \vec{\imath})(\vec{o}, \vec{s})$.

The following lemma provides a closed form for $Q(s, \vec{\imath})$, which will become useful later.

Lemma 2. *For all Q, s, and $\vec{\imath}$, $\vec{o}$, and $\vec{s}$ of equal lengths $k \geq 0$, $k+1$, and $k+1$, respectively,*

$$Q(s, \vec{\imath})(\vec{o}, \vec{s}) = \delta(s)(\vec{s}[1]) * \delta(\sigma(\vec{s}[1]))(\vec{o}[1]) * \prod_{\kappa=1}^{k} \tau(\vec{s}[\kappa], \vec{\imath}[\kappa])(\vec{s}[\kappa+1]) * \delta(\sigma(\vec{s}[\kappa+1]))(\vec{o}[\kappa+1]) \tag{A.4}$$

Proof. Proof by induction. Base Case: $k = 0$. $Q(s, [])([\sigma(s)], [s]) = \delta(s)(\vec{s}[1]) * \delta(\sigma(\vec{s}[1]))(\vec{o}[1])$, which is 1 if $\vec{s}[1] = s$ and $\sigma(\vec{s}[1]) = \vec{o}[1]$ and 0 otherwise as needed.

Inductive Case: $k > 0$.

$$Q(s, i{\cdot}\vec{\imath})(o{\cdot}\vec{o}, s'{\cdot}\vec{s}) \tag{A.5}$$

$$= \delta(s)(s') * \delta(\sigma(s))(o) * \sum_{s''} \tau(s,i)(s'') * Q(s'', \vec{\imath})(\vec{o}, \vec{s}) \tag{A.6}$$

$$= \begin{array}{ll} & \delta(s)(s') * \delta(\sigma(s))(o) * \sum_{s''} \tau(s,i)(s'') * \delta(s'')(\vec{s}[1]) * \delta(\sigma(\vec{s}[1]))(\vec{o}[1]) \\ * & \prod_{\kappa=1}^{k-1} \tau(\vec{s}[\kappa], \vec{\imath}[\kappa])(\vec{s}[\kappa+1]) * \delta(\sigma(\vec{s}[\kappa+1]))(\vec{o}[\kappa+1]) \end{array} \tag{A.7}$$

$$= \begin{array}{ll} & \delta(s)(s') * \delta(\sigma(s))(o) * \tau(s,i)(\vec{s}[1]) * \delta(\vec{s}[1])(\vec{s}[1]) * \delta(\sigma(\vec{s}[1]))(\vec{o}[1]) \\ * & \prod_{\kappa=1}^{k-1} \tau(\vec{s}[\kappa], \vec{\imath}[\kappa])(\vec{s}[\kappa+1]) * \delta(\sigma(\vec{s}[\kappa+1]))(\vec{o}[\kappa+1]) \end{array} \tag{A.8}$$

$$= \delta(s)(s') * \delta(\sigma(s))(o) * \tau(s,i)(\vec{s}[1]) * \delta(\sigma(\vec{s}[1]))(\vec{o}[1]) * \prod_{\kappa=1}^{k-1} \tau(\vec{s}[\kappa], \vec{\imath}[\kappa])(\vec{s}[\kappa+1]) * \delta(\sigma(\vec{s}[\kappa+1]))(\vec{o}[\kappa+1]) \tag{A.9}$$

$$= \begin{array}{ll} & \delta(s)(s') * \delta(\sigma(s))(o) * \tau(s,i)(s'{\cdot}\vec{s}[1+1]) * \delta(\sigma(s'{\cdot}\vec{s}[1+1]))(o{\cdot}\vec{o}[1+1]) \\ * & \displaystyle\prod_{\kappa=1}^{k-1} \tau(s'{\cdot}\vec{s}[\kappa+1], i{\cdot}\vec{\imath}[\kappa+1])(s'{\cdot}\vec{s}[\kappa+1+1]) * \delta(\sigma(s'{\cdot}\vec{s}[\kappa+1+1]))(o{\cdot}\vec{o}[\kappa+1+1]) \end{array} \tag{A.10}$$

$$= \begin{array}{ll} & \delta(s)(s'{\cdot}\vec{s}[1]) * \delta(\sigma(s'{\cdot}\vec{s}[1]))(o{\cdot}\vec{o}[1]) * \tau(s'{\cdot}\vec{s}[1], i{\cdot}\vec{\imath}[1])(s'{\cdot}\vec{s}[1+1]) * \delta(\sigma(s'{\cdot}\vec{s}[1+1]))(o{\cdot}\vec{o}[1+1]) \\ * & \displaystyle\prod_{\kappa=1}^{k-1} \tau(s'{\cdot}\vec{s}[\kappa+1], i{\cdot}\vec{\imath}[\kappa+1])(s'{\cdot}\vec{s}[\kappa+1+1]) * \delta(\sigma(s'{\cdot}\vec{s}[\kappa+1+1]))(o{\cdot}\vec{o}[\kappa+1+1]) \end{array} \tag{A.11}$$

$$= \begin{array}{ll} & \delta(s)(s'{\cdot}\vec{s}[1]) * \delta(\sigma(s'{\cdot}\vec{s}[1]))(o{\cdot}\vec{o}[1]) \\ * & \tau(s'{\cdot}\vec{s}[1], i{\cdot}\vec{\imath}[1])(s'{\cdot}\vec{s}[1+1]) * \delta(\sigma(s'{\cdot}\vec{s}[1+1]))(o{\cdot}\vec{o}[1+1]) \\ * & \displaystyle\prod_{\kappa=1+1}^{(k-1)+1} \tau(s'{\cdot}\vec{s}[\kappa], i{\cdot}\vec{\imath}[\kappa])(s'{\cdot}\vec{s}[\kappa{+}1]) * \delta(\sigma(s'{\cdot}\vec{s}[\kappa{+}1]))(o{\cdot}\vec{o}[\kappa{+}1]) \end{array} \tag{A.12}$$

$$= \delta(s)(s'{\cdot}\vec{s}[1]) * \delta(\sigma(s'{\cdot}\vec{s}[1]))(o{\cdot}\vec{o}[1]) * \prod_{\kappa=1}^{k} \tau(s'{\cdot}\vec{s}[\kappa], i{\cdot}\vec{\imath}[\kappa])(s'{\cdot}\vec{s}[\kappa+1]) * \delta(\sigma(s'{\cdot}\vec{s}[\kappa+1]))(o{\cdot}\vec{o}[\kappa+1]) \tag{A.13}$$

where (A.7) comes from the inductive hypothesis, (A.8) follows since $\delta(s'')(\vec{s}[1])$ will be 0 for all other values of s'', (A.11) follows since unless $s' = s$, the value will be zero due the $\delta(s)(s')$ term, (A.12) changes the indexing of the product so that (A.13) can roll the two terms before the product into the product by starting the indexing from 1 instead of $1 + 1$. □

A.2 Universal Unsoundness and Incompleteness Proofs

A.2.1 Proof of Theorem 1

The theorem states:

> Any black-box analysis that ever returns a positive result from interference for H to L is unsound for interference from H to L.

Proof. Assume that analysis A can return a positive result for interference from interacting with a system. Then, there must exist a system $q_+ = \langle \mathcal{S}_+, s_{0+}, \mathcal{I}_+, \mathcal{O}_+, \tau_+, \sigma_+ \rangle$ and $\vec{\imath}_+$ such that the output $q_+(\vec{\imath}_+)$ leads to A returning a positive result. $q_+(\vec{\imath}_+)$ leads to a trace $[s_1, o_1, i_1, s_2, o_2, i_2, \ldots, s_k, o_k, i_k, s_{k+1}, o_{k+1}]$ where $s_1 = s_{0+}$, $o_j = \sigma_+(s_j)$, $i_j = \vec{\imath}_+[j]$, $s_j = \tau(s_{j-1}, i_{j-1})$, and $|\vec{\imath}_+| = k$.

Assume system q_{N} has noninterference but behaves like q_+ on $\vec{\imath}_+$, i.e., let q_{N} be $\langle \mathcal{S}_{\mathrm{N}}, s_{0\mathrm{N}}, \mathcal{I}_+, \mathcal{O}_+, \tau_{\mathrm{N}}, \sigma_{\mathrm{N}} \rangle$ where

- $\mathcal{S}_{\mathrm{N}} = \{s^{\mathrm{N}}_1, \ldots, s^{\mathrm{N}}_k, s^{\mathrm{N}}_{k+1}\}$,
- $s_{0\mathrm{N}} = s^{\mathrm{N}}_1$,
- $\tau_{\mathrm{N}}(s^{\mathrm{N}}_j, i) = s^{\mathrm{N}}_{j+1}$ for all $j \leq k$ and $\tau_{\mathrm{N}}(s^{\mathrm{N}}_{k+1}, i) = s^{\mathrm{N}}_{k+1}$ for all i, and
- $\sigma_{\mathrm{N}}(s^{\mathrm{N}}_j) = o_j$ for all $j \leq k+1$.

Since the behavior of q_{N} does not depend upon any inputs, it has noninterference. However, by construction, $q_{\mathrm{N}}(\vec{\imath}_+) = q_+(\vec{\imath}_+)$. Thus, A cannot tell them apart even with the ability to observe every input and output to the system. Thus, it must produce an unsound positive result for interference on q_{N}. □

A.2.2 Proof of Theorem 2

The theorem states:

> Any black-box analysis that ever returns a positive result for noninterference from H to L is unsound for noninterference from H to L if H has two inputs and L has two outputs.

Proof. Assume that A can return a positive result from interacting with a system. Then, there must exist a system q_- and $\vec{\imath}_-$ such that the output $q_-(\vec{\imath}_-)$ lead A to return positive for noninterference. Let the trace of q_- on $\vec{\imath}_-$ be $[s_1, o_1, i_1, s_2, o_2, \ldots]$.

Let q_{I} be a system that has interference but behaves like q_- on $\vec{\imath}_-$. That is, q_{I} be $\langle \mathcal{S}_{\mathrm{I}}, s_{0\mathrm{I}}, \mathcal{I}_-, \mathcal{O}_-, \tau_{\mathrm{I}}, \sigma_{\mathrm{I}} \rangle$ where

- $\mathcal{S}_{\mathrm{I}} = \mathcal{S}_{\text{-}} \cup \{s_{00}, s_{01}\}$;
- $s_{0\mathrm{I}} = s_{00}$;
- $\tau_{\mathrm{I}}(s,i) = \tau_{\text{-}}(s,i)$ for all i and s in $\mathcal{S}_{\text{-}}$, $\tau_{\mathrm{I}}(s_{00}, \vec{\imath}_{\text{-}}[1]) = \tau(s_0, \vec{\imath}_{\text{-}}[1])$, $\tau_{\mathrm{I}}(s_{00}, i) = s_{01}$ for all $i \neq \vec{\imath}_{\text{-}}[1]$, and $\tau_{\mathrm{I}}(s_{01}, i) = s_{01}$ for all i;
- $\sigma_{\mathrm{I}}(s) = \sigma_{\text{-}}(s)$ for all s other than s_{01} and $\sigma_{\mathrm{I}}(s_{01}) = o_{01}$ where $o_{01} \neq o_2 = \sigma_{\text{-}}(s_2)$.

Note that since $|\mathcal{O}| \geq 2$, such an o_{01} exists, and since $|\mathcal{I}| \geq 2$, an $i \neq \vec{\imath}_{\text{-}}[1]$ exists making s_{01} reachable.

The behavior of q_{I} at the state s_{01} versus s_1 shows that it has interference when we consider an input i such that $i \neq \vec{\imath}_{\text{-}}[1]$ and i differs from $\vec{\imath}_{\text{-}}[1]$ by just high-level information. However, by construction, $q_{\mathrm{I}}(\vec{\imath}_{\text{-}}) = q_{\text{-}}(\vec{\imath}_{\text{-}})$. Thus, A cannot tell them apart even with the ability to observe every input and output to the system. Thus, it must produce an unsound result for q_{I} having noninterference. □

A.3 Background: Causality

In this section, we review Pearl's formalism of causality [114]. In particular, we use notation and results found in Chapters 1 and 7 of [114].

Background on Probability. Recall that for any two propositions A_1 and A_2, $\mathcal{P}(A_1 \wedge A_2) = \mathcal{P}(A_2 \,|\, A_1) * \mathcal{P}(A_1)$ if $\mathcal{P}(A_1) > 0$. If $\mathcal{P}(A_1) = 0$, then $\mathcal{P}(A_2 \,|\, A_1)$ is not defined. We adopt the convention that the product of an undefined term by zero will be zero (which is similar to [74]). Under this convention, $\mathcal{P}(A_1 \wedge A_2) = \mathcal{P}(A_2 \,|\, A_1) * \mathcal{P}(A_1)$ holds in general. Under this convention, the chain rule of probability iterates the above equation:

$$\mathcal{P}(\wedge_{j=1}^{J} A_j) = \prod_{j=1}^{J} \mathcal{P}(A_j \,|\, \wedge_{k=1}^{j-1} A_k) \tag{A.14}$$

SEMs. Recall that a probabilistic SEM M is a tuple $\langle \mathcal{V}_{\mathsf{en}}, \mathcal{V}_{\mathsf{ex}}, \mathcal{E}, \mathcal{P} \rangle$ where $\mathcal{V}_{\mathsf{en}}$ is the endogenous variables, $\mathcal{V}_{\mathsf{ex}}$ is the exogenous variables, $\mathcal{E}$ provides a *structural equation* for each endogenous variable V, and $\mathcal{P}$ is a probability distribution.

To define $\mathcal{E}$ in more detail, let the space of functions $\mathcal{F}_V$ be (possibly randomized) functions from the ranges of a subset of the variables other than V to the range of V. $\mathcal{E}$ maps a variable V in $\mathcal{V}_{\mathsf{en}}$ to a function in $\mathcal{F}_V$. If V is mapped to a function F_V that does not include the range of the variable V', then V does not have a direct dependence upon V'. We write $V := F_V(\vec{V})$ where $\vec{V}$ is a list of other variables not equal to V if $\mathcal{E}$ maps V to a function F_V that directly depends upon the variables $\vec{V}$. Let $\mathsf{par}(V)$ denote the variables $\vec{V}$, called the *parents* of V. Let $\mathsf{par}(V)$ be the empty set for exogenous variables V.

To define $\mathcal{P}$ in more detail, let $\mathcal{P}$ map each exogenous variable V to a probability distribution $\mathcal{P}_V$ over the range of V. Note that exogenous variables are assumed to be independent and, thus, these marginal distributions suffice for explaining their behavior.

We call a SEM *recursive* if the graph of variables created by including a directed edge from every parent to every child variable (node) is acyclic. We will limit our discuss to recursive SEMs. We will implicitly order their variables by the topology created by this graph.

Assigning Probabilities: Factorization. We can use the topological ordering on the variables to extend to $\mathcal{P}$ to assign probabilities to assignments of values to variables. To do so, we define some notation. For a vector $\vec{V}$, we use $\vec{V}[j]$ to denote its jth component. We take $\vec{V} = \vec{v}$ be shorthand for $\bigwedge_{j=1}^{t} \vec{V}[j] = \vec{v}[j]$ where $\vec{V}$ is a vector of length t holding variables. Similarly, let $\vec{V}^{j:k} = \vec{v}$ be shorthand for $\bigwedge_{t=j}^{k} \vec{V}[t] = \vec{v}[t]$. We use $\mathsf{par}(V) = \vec{w}$ as sort hand for $\bigwedge_{W_j \in \mathsf{par}(V)} W_j = \vec{w}[j]$ where there is some implicit ordering on variables associating the jth element of $\mathsf{par}(V)$ to the jth component of $\vec{w}$.

We start by assigning a probability to a variable given its parents in the SEM M. For exogenous variables V, let $\mathcal{P}^M(V = v \mid \mathsf{par}(V) = \vec{v})$ be $\mathcal{P}_V(v)$. (Recall that $\mathsf{par}(V)$ is the empty set for exogenous variables. Thus, the vector $\vec{v}$ of values is empty as well.) For endogenous variables V defined by a deterministic function f_V, let $\mathcal{P}^M(V = v \mid \mathsf{par}(V) = \vec{v})$ be 1 if $v = f_V(\vec{v})$ and be 0 otherwise. For randomized functions F_V, let $\mathcal{P}^M(V = v \mid \mathsf{par}(V) = \vec{v})$ be the probability that $v = F_V(\vec{v})$.

For a vector of all the variables $\vec{V}$ and a vector of values $\vec{v}$ they can take on, we determine $\mathcal{P}^M(\vec{V}{=}\vec{v})$ using a *factorization* created by the chain rule:

$$\mathcal{P}^M(\vec{V}{=}\vec{v}) = \prod_{j=1}^{|\vec{V}|} \mathcal{P}^M(\vec{V}[j]{=}\vec{v}[j] \mid \vec{V}^{1:j-1} = \vec{v}^{1:j-1}) \tag{A.15}$$

$$= \prod_{j=1}^{|\vec{V}|} \mathcal{P}^M(\vec{V}[j]{=}\vec{v}[j] \mid \mathsf{par}(\vec{V}[j]) = \vec{v}_{\mathsf{par}(\vec{V}[j])}) \tag{A.16}$$

where j ranges over $\vec{V}$ in a manner that respects the variables' topology, $\vec{v}_{\mathsf{par}(\vec{V}[j])}$ is $\vec{v}$ restricted to just these components corresponding to elements of $\mathsf{par}(\vec{V}[j])$, and we take $\mathsf{par}(V)$ to be the empty set for exogenous variables V. (A.16) follows since $\vec{V}^{1:j-1} = \vec{v}^{1:j-1}$ includes all the parents of $\vec{V}[j]$ by using the topological ordering and $\vec{V}[j]$ is independent of its non-parents given its parents.

For $\vec{W} = \vec{w}$ involving a subset of the variables, we use the following:

$$\mathcal{P}^M(\vec{W}{=}\vec{w}) = \sum_{\vec{u}} \mathcal{P}^M(\vec{W} = \vec{w}, \vec{U} = \vec{u}) = \sum_{\vec{u}} \prod_{j=1}^{|\vec{V}|} \mathcal{P}^M(\vec{V}[j]{=}\vec{v}[j] \mid \mathsf{par}(\vec{V}[j]) = v(\vec{w}, \vec{u})_{\mathsf{par}(\vec{V}[j])}) \tag{A.17}$$

where $\vec{U}$ are the remaining variables, $\vec{V}$ is a vector consisting of the components of $\vec{W}$ and $\vec{U}$ put into order, and $v(\vec{w}, \vec{u})$ is the vector $\vec{v}$ that results from combining the components of $\vec{w}$ and $\vec{u}$ in order.

Sub-Models and Truncated Factorization. Recall that for an SEM M, endogenous variable X, and value x that X can take on, the *sub-model* $M[X{:=}x]$ is the SEM that results from replacing the equation $X := F_X(\vec{V})$ in $\mathcal{E}$ with the equation $X := x$. That is, for $M = \langle \mathcal{V}_{\mathsf{en}}, \mathcal{V}_{\mathsf{ex}}, \mathcal{E}, \mathcal{P} \rangle$, $M[X{:=}x] = \langle \mathcal{V}_{\mathsf{en}}, \mathcal{V}_{\mathsf{ex}}, \mathcal{E}[X := x], \mathcal{P} \rangle$ where $\mathcal{E}[X := x](X) = \lambda.x$ (the function that takes no arguments and always returns x) and $\mathcal{E}[X := x](V) = \mathcal{E}(V)$ for $V \neq X$.

$\mathcal{P}^M$ and $\mathcal{P}^{M[X:=x]}$ are related by *truncated factorization*. To define it, let X be the kth variable in the topological order. For $\vec{V}{=}\vec{v}$ that assigns X the value x (i.e., $\vec{v}[k] = x$),

$$\mathcal{P}^{M[X:=x]}(\vec{V}{=}\vec{v}) = \prod_{j=1}^{|\vec{V}|} \mathcal{P}^{M[X:=x]}(\vec{V}[j]{=}\vec{v}[j] \mid \mathsf{par}(\vec{V}[j]) = \vec{v}_{\mathsf{par}(\vec{V}[j])}) \tag{A.18}$$

$$= \prod_{j=1:j\neq k}^{|\vec{V}|} \mathcal{P}^{M}(\vec{V}[j]{=}v(\vec{w},\vec{u})[j] \mid \mathsf{par}(\vec{V}[j]) = \vec{v}_{\mathsf{par}(\vec{V}[j])}) \tag{A.19}$$

where the produce in (A.19) skips X, the kth variable. For $\vec{V}{=}\vec{v}$ that assigns X a value other than x, $\mathcal{P}^{M[X:=x]}(\vec{V}{=}\vec{v})$ is 0. The above extends to subsets of all variables as in (A.17).

Henceforth, for readability, we adopt Pearl's *do* notation. We will drop the M from $\mathcal{P}^M$ when M is clear from context. We will denote $\mathcal{P}^{M[X:=x]}(\vec{V}{=}\vec{v})$ as $\mathcal{P}(\vec{V}{=}\vec{v} \mid \mathsf{do}(X{:=}x))$. We take $\mathsf{do}(\vec{X} := \vec{x})$ be shorthand for $\bigwedge_{j=1}^{|\vec{X}|} \mathsf{do}(\vec{X}[j] := \vec{x}[j])$. We understand $\mathcal{P}(\vec{V}{=}\vec{v} \mid \mathsf{do}(\vec{X}{:=}\vec{x}))$ to be iterative application of taking a sub-model with

$$\mathcal{P}(\vec{V}{=}\vec{v} \mid \mathsf{do}(\vec{X}{:=}\vec{x})) = \prod_{j=1:j\notin K}^{|\vec{V}|} \mathcal{P}(\vec{V}[j]{=}\vec{v}[j] \mid \mathsf{par}(\vec{V}[j]) = \vec{v}_{\mathsf{par}(\vec{V}[j])}) \tag{A.20}$$

where K is the set containing the indexes of the variables in $\vec{X}$.

Pearl presents two useful properties [114, pg 24]. The first allows converting normal conditional statements to *do* statements when conditioning upon all of a variable's parents. The second allows dropping irreverent *do* statements when conditioning upon all of a variable's parents.

Lemma 3 (Pearl's Property 1)**.**

$$\mathcal{P}(Y{=}y \mid \mathsf{par}(Y){=}\vec{x}) = \mathcal{P}(Y{=}y \mid \mathsf{do}(\mathsf{par}(Y){:=}\vec{x}))$$

Lemma 4 (Pearl's Property 2)**.**

$$\mathcal{P}(Y{=}y \mid \mathsf{do}(\mathsf{par}(Y){:=}\vec{x}), \mathsf{do}(\vec{Z}{:=}\vec{z})) = \mathcal{P}(Y{=}y \mid \mathsf{do}(\mathsf{par}(Y){:=}\vec{x}))$$

A.4 Interference and Causation

A.4.1 Model

The following table shows how we define these variables and functions:

V		$\mathsf{par}(V)$	F_V	
HU_{t+1}	high user	$\emptyset$	(exogenous)	$\forall t \geq 0$
LU_{t+1}	low user	$\emptyset$	(exogenous)	$\forall t \geq 0$
S_0	initial state	$\emptyset$	$F_{\mathsf{s},0}() = \delta(s_0)$	
S_{t+1}	state	$\{S_t, \mathsf{HI}_t, \mathsf{LI}_t\}$	$F_{\mathsf{s},t+1}(s_t, \mathsf{hi}_t, \mathsf{li}_t)(s') = \tau(s_t, \langle \mathsf{hi}_t, \mathsf{li}_t \rangle\rangle)$	$\forall t \geq 0$
HI_{t+1}	high input	$\{\mathsf{HU}_{t+1}, \mathsf{LU}_{t+1}, \mathsf{HO}_1, \ldots, \mathsf{HO}_t, \mathsf{LO}_1, \ldots, \mathsf{LO}_t\}$	$F_{\mathsf{hi},t+1}(\mathsf{HU}_{t+1}, \mathsf{LU}_{t+1}, \vec{\mathsf{HO}}^t, \vec{\mathsf{LO}}^t)$	$\forall t \geq 0$
LI_{t+1}	low input	$\{\mathsf{HU}_{t+1}, \mathsf{LU}_{t+1}, \mathsf{HO}_1, \ldots, \mathsf{HO}_t, \mathsf{LO}_1, \ldots, \mathsf{LO}_t\}$	$F_{\mathsf{li},t+1}(\mathsf{HU}_{t+1}, \mathsf{LU}_{t+1}, \vec{\mathsf{HO}}^t, \vec{\mathsf{LO}}^t)$	$\forall t \geq 0$
HO_t	high output	$\{S_t\}$	$F_{\mathsf{ho},t}(s_t) = \delta(\lfloor \sigma(s_t) {\downarrow} H \rfloor)$	$\forall t \geq 0$
LO_t	low input	$\{S_t\}$	$F_{\mathsf{lo},t}(s_t) = \delta(\lfloor \sigma(s_t) {\downarrow} L \rfloor)$	$\forall t \geq 0$

The form of $\mathcal{P}(V{=}v \mid \mathsf{par}(V))$ depends upon the type of variable that V is. Here are the options based on the above table:

V	$\mathcal{P}(V{=}v \mid \mathsf{par}(V))$		
HU_{t+1}	$\mathcal{P}(\mathsf{HU}_{t+1}{=}\mathsf{hu}_{t+1})$		$\forall t \geq 0$
LU_{t+1}	$\mathcal{P}(\mathsf{LU}_{t+1}{=}\mathsf{lu}_{t+1})$		$\forall t \geq 0$
S_0	$\mathcal{P}(S_0{=}s)$	$= \delta(s_0)(s)$	
S_{t+1}	$\mathcal{P}(S_{t+1} = s_{t+1} \mid S_t{=}s_t, \mathsf{HI}_t{=}\mathsf{hi}_t, \mathsf{LI}_t{=}\mathsf{li}_t)$	$= \tau(s_t, \langle \mathsf{hi}_t, \mathsf{li}_t \rangle\rangle)(s_{t+1})$	$\forall t \geq 0$
HI_{t+1}	$\mathcal{P}(\mathsf{HI}_{t+1}{=}\mathsf{hi}_{t+1} \mid \mathsf{HU}_{t+1}{=}\mathsf{hu}_{t+1}, \mathsf{LU}_{t+1}{=}\mathsf{lu}_{t+1}, \vec{\mathsf{HO}}^t{=}\vec{\mathsf{ho}}, \vec{\mathsf{LO}}^t{=}\vec{\mathsf{lo}})$	$= F_{\mathsf{hi},t+1}(\mathsf{hu}_{t+1}, \mathsf{lu}_{t+1}, \vec{\mathsf{ho}}, \vec{\mathsf{lo}})(\mathsf{hi}_{t+1})$	$\forall t \geq 0$
LI_{t+1}	$\mathcal{P}(\mathsf{LI}_{t+1}{=}\mathsf{li}_{t+1} \mid \mathsf{HU}_{t+1}{=}\mathsf{hu}_{t+1}, \mathsf{LU}_{t+1}{=}\mathsf{lu}_{t+1}, \vec{\mathsf{HO}}^t{=}\vec{\mathsf{ho}}, \vec{\mathsf{LO}}^t{=}\vec{\mathsf{lo}})$	$= F_{\mathsf{li},t+1}(\mathsf{hu}_{t+1}, \mathsf{lu}_{t+1}, \vec{\mathsf{ho}}, \vec{\mathsf{lo}})(\mathsf{li}_{t+1})$	$\forall t \geq 0$
HO_t	$\mathcal{P}(\mathsf{HO}_t{=}\mathsf{ho}_t \mid S_t{=}s_t)$	$= \delta(\lfloor \sigma(s_t) {\downarrow} H \rfloor)(\mathsf{ho}_t)$	$\forall t \geq 0$
LO_t	$\mathcal{P}(\mathsf{LO}_t{=}\mathsf{lo}_t \mid S_t{=}s_t)$	$= \delta(\lfloor \sigma(s_t) {\downarrow} L \rfloor)(\mathsf{lo}_t)$	$\forall t \geq 0$

Let M_Q consist of the variables and equations defined above plus an unknown probability distribution $\mathcal{P}$.

A.4.2 Lemma for Showing the Relation of Models

Let $\vec{V}^{j:k} = \vec{v}$ be shorthand for $\bigwedge_{t=j}^{k} \vec{V}[t] = \vec{v}[t]$. Let $\mathsf{do}(V := v)$ be Pearl's *do* operation denoting an intervention fixing a value, such as by applying a treatment to an experimental unit [114]. Let $\mathsf{do}(\vec{V}^{j:k} := \vec{v})$ be short hand for $\bigwedge_{t=j}^{k} \mathsf{do}(\vec{V}[t] := \vec{v}[t])$. Let $\vec{O}^{j:k} = \vec{o}$ be shorthand for $\lfloor \mathsf{HO}^{j:k} {\downarrow} H \rfloor = \lfloor \vec{o} {\downarrow} H \rfloor \wedge \lfloor \mathsf{LO}^{j:k} {\downarrow} L \rfloor =$

$\lfloor \vec{o}{\downarrow}L \rfloor$. Let $\vec{I}^{j:k} = \vec{\imath}$ be shorthand for $\lfloor \mathsf{HI}^{j:k}{\downarrow}H \rfloor = \lfloor \vec{\imath}{\downarrow}H \rfloor \wedge \lfloor \mathsf{LI}^{j:k}{\downarrow}L \rfloor = \lfloor \vec{\imath}{\downarrow}L \rfloor$. We define $\mathsf{do}(\vec{O}^{j:k} := \vec{o})$ and $\mathsf{do}(\vec{I}^{j:k} := \vec{\imath})$ similarly.

We define the equivalent of $Q(s,\vec{\imath})(\vec{s},\vec{o})$ for an SEM M_Q as follows: let

$$\mathsf{fix}^t(M_Q)(s,\vec{\imath})(\vec{s},\vec{o}) \quad = \quad \mathcal{P}(\vec{S}^{t:t+k}{=}\vec{s} \wedge \vec{O}^{t:t+k}{=}\vec{o} \mid \mathsf{do}(S_t{:=}s), \mathsf{do}(\vec{I}^{t:t+k-1}{:=}\vec{\imath})) \tag{A.21}$$

where $\vec{\imath}$, $\vec{o}$, and $\vec{s}$ are of lengths $k \geq 0$, $k+1$, and $k+1$, respectively. The time $t \geq 0$ represents the time at which M_Q starts operating. Note that when $k = 0$, $\vec{I}^{t:t+k-1}$ is $\vec{I}^{t:t-1}$, which is an empty sequence, as is $\vec{\imath}$. Thus, $\mathsf{do}(\vec{I}^{t:t+k-1}{:=}\vec{\imath})$ is vacuously true when $k = 0$. On the other hand, $\vec{S}^{t:t+k}$ is $\vec{S}^{t:t} = [\vec{S}_t]$, a sequence with a single component, which is compared to the single component of $\vec{s} = [s_1]$.

Lemma 5. *For all Q, s, and $t \geq 0$, and $\vec{\imath}$, $\vec{o}$, and $\vec{s}$ of lengths $k \geq 0$, $k+1$, and $k+1$, respectively,*

$$\mathsf{fix}^t(M_Q)(s,\vec{\imath})(\vec{s},\vec{o}) = Q(s,\vec{\imath})(\vec{o},\vec{s})$$

Proof.

$$\mathsf{fix}^t(M_Q)(s,\vec{\imath})(\vec{s},\vec{o}) \tag{A.22}$$

$$= \mathcal{P}(\vec{S}^{t:t+k}{=}\vec{s} \wedge \vec{O}^{t:t+k}{=}\vec{o} \mid \mathsf{do}(S_t{:=}s), \mathsf{do}(\vec{I}^{t:t+k-1}{:=}\vec{\imath})) \tag{A.23}$$

$$= \mathcal{P}(\bigwedge_{\kappa=0}^{k} \vec{S}[t+\kappa]{=}\vec{s}[1+\kappa] \wedge \vec{O}[t+\kappa]{=}\vec{o}[1+\kappa] \mid \mathsf{do}(S_t{:=}s), \mathsf{do}(\vec{I}^{t:t+k-1}{:=}\vec{\imath})) \tag{A.24}$$

$$= \prod_{\kappa=0}^{k} \mathcal{P}(\vec{S}[t+\kappa]{=}\vec{s}[1+\kappa] \wedge \vec{O}[t+\kappa]{=}\vec{o}[1+\kappa] \mid \vec{S}^{t:t+\kappa-1}{=}\vec{s}^{1:\kappa}, \vec{O}^{t:t+\kappa-1}{=}\vec{o}^{1:\kappa}, \mathsf{do}(S_t{:=}s), \mathsf{do}(\vec{I}^{t:t+k-1}{:=}\vec{\imath})) \tag{A.25}$$

$$= \prod_{\kappa=0}^{k} \begin{array}{l} \mathcal{P}(\vec{S}[t+\kappa]{=}\vec{s}[1+\kappa] \mid \vec{S}^{t:t+\kappa-1}{=}\vec{s}^{1:\kappa}, \vec{O}^{t:t+\kappa-1}{=}\vec{o}^{1:\kappa}, \mathsf{do}(S_t{:=}s), \mathsf{do}(\vec{I}^{t:t+k-1}{:=}\vec{\imath})) \\ *\mathcal{P}(\vec{O}[t+\kappa]{=}\vec{o}[1+\kappa] \mid \vec{S}[t+\kappa]{=}\vec{s}[1+\kappa], \vec{S}^{t:t+\kappa-1}{=}\vec{s}^{1:\kappa}, \vec{O}^{t:t+\kappa-1}{=}\vec{o}^{1:\kappa}, \mathsf{do}(S_t{:=}s), \mathsf{do}(\vec{I}^{t:t+k-1}{:=}\vec{\imath})) \end{array} \tag{A.26}$$

where (A.24) expands $\vec{S}^{t:t+k}{=}\vec{s} \wedge \vec{O}^{t:t+k}{=}\vec{o}$ into $\bigwedge_{\kappa=0}^{k} \vec{S}[t+\kappa]{=}\vec{s}[1+\kappa] \wedge \vec{O}[t+\kappa]{=}\vec{o}[1+\kappa]$. Since κ ranges from 0 to k while we index the sequences $\vec{s}$ and $\vec{o}$ from 1 to $k+1$, we add 1 to κ while indexing into $\vec{s}$ and $\vec{o}$. Both (A.25) and (A.26) follow from the chain rule of probability. Note that when κ is 0, the term $\vec{S}^{t:t+\kappa-1}{=}\vec{s}^{1:\kappa}$ becomes $\vec{S}^{t:t-1}{=}\vec{s}^{1:0}$, which compares the empty sequence to the empty sequence. This comparison is vacuously true as it should be since no state precedes the first state $\vec{s}[1+\kappa] = \vec{s}[1+0] = \vec{s}[1]$ and, thus, the probability of this state should not be conditioned on a preceding state. The same holds for the output $\vec{o}[1]$.

In (A.26), $\mathcal{P}(\vec{S}[t+\kappa]{=}\vec{s}[1+\kappa] \mid \vec{S}^{t:t+\kappa-1}{=}\vec{s}^{1:\kappa}, \vec{O}^{t:t+\kappa-1}{=}\vec{o}^{1:\kappa}, \mathsf{do}(S_t{:=}s), \mathsf{do}(\vec{I}^{t:t+k-1}{:=}\vec{\imath}))$ is looking at the probability of $\vec{S}[t+\kappa]{=}\vec{s}[1+\kappa]$ conditional upon every term on which $\vec{S}[t+\kappa]$ depends in the model M_Q

(i.e., all of the variables in $\mathsf{par}(S_{t+\kappa})$). The same holds for the outputs $\vec{O}[t+\kappa]$. Thus, Pearl's Property 1 [114, pg 24] applies to (A.26) and justifies (A.28) in the following:

$$\mathsf{fix}^t(M_Q)(s,\vec{\imath})(\vec{s},\vec{o}) \tag{A.27}$$

$$= \prod_{\kappa=0}^{k} \begin{array}{l} \mathcal{P}(\vec{S}[t+\kappa]{:=}\vec{s}[1+\kappa] \mid \mathsf{do}(\vec{S}^{t:t+\kappa-1}{:=}\vec{s}^{1:\kappa}), \mathsf{do}(\vec{O}^{t:t+\kappa-1}{:=}\vec{o}^{1:\kappa}), \mathsf{do}(S_t{:=}s), \mathsf{do}(\vec{I}^{t:t+k-1}{:=}\vec{\imath})) \\ *\mathcal{P}(\vec{O}[t+\kappa]{=}\vec{o}[1+\kappa] \mid \mathsf{do}(\vec{S}[t+\kappa]{:=}\vec{s}[1+\kappa]), \mathsf{do}(\vec{S}^{t:t+\kappa-1}{:=}\vec{s}^{1:\kappa}), \mathsf{do}(\vec{O}^{t:t+\kappa-1}{:=}\vec{o}^{1:\kappa}), \\ \hfill \mathsf{do}(S_t{:=}s), \mathsf{do}(\vec{I}^{t:t+k-1}{:=}\vec{\imath})) \end{array} \tag{A.28}$$

$$= \begin{array}{l} \mathcal{P}(\vec{S}[t+0]{:=}\vec{s}[0+1] \mid \mathsf{do}(\vec{S}^{t:t+0-1}{:=}\vec{s}^{1:0}), \mathsf{do}(\vec{O}^{t:t+0-1}{:=}\vec{o}^{1:0}), \mathsf{do}(S_t{:=}s), \mathsf{do}(\vec{I}^{t:t+k-1}{:=}\vec{\imath})) \\ *\mathcal{P}(\vec{O}[t+0]{=}\vec{o}[0+1] \mid \mathsf{do}(\vec{S}[t+0]{:=}\vec{s}[0+1]), \mathsf{do}(\vec{S}^{t:t+0-1}{:=}\vec{s}^{1:0}), \mathsf{do}(\vec{O}^{t:t+0-1}{:=}\vec{o}^{1:0}), \\ \hfill \mathsf{do}(S_t{:=}s), \mathsf{do}(\vec{I}^{t:t+k-1}{:=}\vec{\imath})) \\ * \displaystyle\prod_{\kappa=1}^{k} \begin{array}{l} \mathcal{P}(\vec{S}[t+\kappa]{:=}\vec{s}[1+\kappa] \mid \mathsf{do}(\vec{S}^{t:t+\kappa-1}{:=}\vec{s}^{1:\kappa}), \mathsf{do}(\vec{O}^{t:t+\kappa-1}{:=}\vec{o}^{1:\kappa}), \mathsf{do}(S_t{:=}s), \mathsf{do}(\vec{I}^{t:t+k-1}{:=}\vec{\imath})) \\ *\mathcal{P}(\vec{O}[t+\kappa]{=}\vec{o}[1+\kappa] \mid \mathsf{do}(\vec{S}[t+\kappa]{:=}\vec{s}[1+\kappa]), \mathsf{do}(\vec{S}^{t:t+\kappa-1}{:=}\vec{s}^{1:\kappa}), \mathsf{do}(\vec{O}^{t:t+\kappa-1}{:=}\vec{o}^{1:\kappa}), \\ \hfill \mathsf{do}(S_t{:=}s), \mathsf{do}(\vec{I}^{t:t+k-1}{:=}\vec{\imath})) \end{array} \end{array} \tag{A.29}$$

$$= \begin{array}{l} \mathcal{P}(\vec{S}[t]{:=}\vec{s}[1] \mid \mathsf{do}(S_t{:=}s), \mathsf{do}(\vec{I}^{t:t+k-1}{:=}\vec{\imath})) * \mathcal{P}(\vec{O}[t]{=}\vec{o}[1] \mid \mathsf{do}(\vec{S}[t]{:=}\vec{s}[1]), \mathsf{do}(S_t{:=}s), \mathsf{do}(\vec{I}^{t:t+k-1}{:=}\vec{\imath})) \\ * \displaystyle\prod_{\kappa=1}^{k} \begin{array}{l} \mathcal{P}(\vec{S}[t+\kappa]{:=}\vec{s}[1+\kappa] \mid \mathsf{do}(\vec{S}^{t:t+\kappa-1}{:=}\vec{s}^{1:\kappa}), \mathsf{do}(\vec{O}^{t:t+\kappa-1}{:=}\vec{o}^{1:\kappa}), \mathsf{do}(S_t{:=}s), \mathsf{do}(\vec{I}^{t:t+k-1}{:=}\vec{\imath})) \\ *\mathcal{P}(\vec{O}[t+\kappa]{=}\vec{o}[1+\kappa] \mid \mathsf{do}(\vec{S}[t+\kappa]{:=}\vec{s}[1+\kappa]), \mathsf{do}(\vec{S}^{t:t+\kappa-1}{:=}\vec{s}^{1:\kappa}), \mathsf{do}(\vec{O}^{t:t+\kappa-1}{:=}\vec{o}^{1:\kappa}), \\ \hfill \mathsf{do}(S_t{:=}s), \mathsf{do}(\vec{I}^{t:t+k-1}{:=}\vec{\imath})) \end{array} \end{array} \tag{A.30}$$

$$= \begin{array}{l} \mathcal{P}(\vec{S}[t]{:=}\vec{s}[1] \mid \mathsf{do}(S_t{:=}s), \mathsf{do}(\vec{I}^{t:t+k-1}{:=}\vec{\imath})) * \mathcal{P}(\vec{O}[t]{=}\vec{o}[1] \mid \mathsf{do}(\vec{S}[t]{:=}\vec{s}[1])) \\ * \displaystyle\prod_{\kappa=1}^{k} \begin{array}{l} \mathcal{P}(\vec{S}[t+\kappa]{:=}\vec{s}[1+\kappa] \mid \mathsf{do}(\vec{S}[t+\kappa-1]{:=}\vec{s}[\kappa]), \mathsf{do}(\vec{I}[t+\kappa-1]{:=}\vec{\imath}[\kappa])) \\ *\mathcal{P}(\vec{O}[t+\kappa]{=}\vec{o}[1+\kappa] \mid \mathsf{do}(\vec{S}[t+\kappa]{:=}\vec{s}[1+\kappa])) \end{array} \end{array} \tag{A.31}$$

$$= \delta(s)(\vec{s}[1]) * \delta(\sigma(\vec{s}[1]))(\vec{o}[1]) * \prod_{\kappa=1}^{k} \tau(\vec{s}[\kappa], \vec{\imath}[\kappa])(\vec{s}[1+\kappa]) * \delta(\sigma(\vec{s}[1+\kappa]))(\vec{o}[1+\kappa]) \tag{A.32}$$

$$= Q(s,\vec{\imath})(\vec{o},\vec{s}) \tag{A.33}$$

where (A.29) simply pulls out the case where $\kappa = 0$; (A.30) just removes terms that are vacuously true; (A.31) follows from Pearl's Property 2 [114, pg 24], which removes *do* terms that are not parents in M_Q of the term whose probability we are computing; (A.32) comes from how we construct the model M_Q; and (A.33) comes from Lemma 2. □

A.4.3 Proof of Lemma 1

The lemma states:

> For all Q, t, $\vec{\imath}$, and $\vec{\mathsf{lo}}$ of lengths t and $t+1$, respectively, $\mathcal{P}(\vec{\mathsf{LO}}^{t+1}{=}\vec{\mathsf{lo}} \mid \mathsf{do}(\vec{I}^{t}{:=}\vec{\imath})) \;=\; \lfloor Q(\vec{\imath}){\downarrow}L\rfloor(\vec{\mathsf{lo}})$.

Proof.

$$\mathcal{P}(\vec{\mathsf{LO}}^{1:t+1}{=}\vec{\mathsf{lo}} \mid \mathsf{do}(\vec{I}^{1:t}{:=}\vec{\imath})) = \sum_{\vec{o}:\lfloor\vec{o}\downarrow L\rfloor=\vec{\mathsf{lo}}} \mathcal{P}(\vec{O}^{1:t+1}{=}\vec{o} \mid \mathsf{do}(\vec{I}^{1:t}{:=}\vec{\imath})) \tag{A.34}$$

$$= \sum_{\vec{s}\in\mathcal{S}^{t+1}} \sum_{\vec{o}:\lfloor\vec{o}\downarrow L\rfloor=\vec{\mathsf{lo}}} \mathcal{P}(\vec{S}^{1:t+1} = \vec{s} \wedge \vec{O}^{1:t+1}{=}\vec{o} \mid \mathsf{do}(\vec{I}^{1:t}{:=}\vec{\imath})) \tag{A.35}$$

$$= \sum_{\vec{s}\in\mathcal{S}^{t+1}} \sum_{\vec{o}:\lfloor\vec{o}\downarrow L\rfloor=\vec{\mathsf{lo}}} \mathcal{P}(\vec{S}^{1:t+1} = \vec{s} \wedge \vec{O}^{1:t+1}{=}\vec{o} \mid S_0{=}s_0, \mathsf{do}(\vec{I}^{1:t}{:=}\vec{\imath})) \tag{A.36}$$

$$= \sum_{\vec{s}\in\mathcal{S}^{t+1}} \sum_{\vec{o}:\lfloor\vec{o}\downarrow L\rfloor=\vec{\mathsf{lo}}} \mathcal{P}(\vec{S}^{1:t+1} = \vec{s} \wedge \vec{O}^{1:t+1}{=}\vec{o} \mid \mathsf{do}(S_0{:=}s_0), \mathsf{do}(\vec{I}^{1:t}{:=}\vec{\imath})) \tag{A.37}$$

$$= \sum_{\vec{s}\in\mathcal{S}^{t+1}} \sum_{\vec{o}:\lfloor\vec{o}\downarrow L\rfloor=\vec{\mathsf{lo}}} Q(s_0, \vec{\imath})(\vec{o}, \vec{s}) \tag{A.38}$$

$$= \sum_{\vec{o}:\lfloor\vec{o}\downarrow L\rfloor=\vec{\mathsf{lo}}} Q(\vec{\imath})(\vec{o}) \tag{A.39}$$

$$= \lfloor Q(\vec{\imath}){\downarrow}L\rfloor(\vec{\mathsf{lo}}) \tag{A.40}$$

where (A.34) and (A.35) hold since output sequences and state sequences are mutually exclusive, (A.36) follows since S_0 is known to be s_0, (A.37) follows from Pearl's Property 1 [114, pg 24], and (A.38) follows from Lemma 5. □

A.4.4 Proof of Theorem 3

The theorem states:

> Q has probabilistic interference iff there exists low inputs $\vec{\mathsf{li}}$ of length t such that $\vec{\mathsf{HI}}^{t}$ has an effect on $\vec{\mathsf{LO}}^{t}$ given $\vec{\mathsf{LI}}^{t} := \vec{\mathsf{li}}$ in M_Q.

Proof. Under this notation, we must show that **(1)** there exist input sequences $\vec{\imath}_1$ and $\vec{\imath}_2$ such that $\lfloor\vec{\imath}_1{\downarrow}L\rfloor = \lfloor\vec{\imath}_2{\downarrow}L\rfloor$ and $\lfloor Q(\vec{\imath}_1){\downarrow}L\rfloor \neq \lfloor Q(\vec{\imath}_2){\downarrow}L\rfloor$ if and only if **(2)** there exist low inputs $\vec{\mathsf{li}}$ of length t and high inputs $\vec{\mathsf{hi}}_1$ and $\vec{\mathsf{hi}}_2$ of length t such that the probability distribution of $\vec{\mathsf{LO}}^{1:t}$ in $M_Q[\vec{\mathsf{HI}}^{1:t} := \vec{\mathsf{hi}}_1][\vec{\mathsf{LI}}^{1:t} := \vec{\mathsf{li}}]$ is not equal to its distribution in $M_Q[\vec{\mathsf{HI}}^{1:t} := \vec{\mathsf{hi}}_2][\vec{\mathsf{LI}}^{1:t} := \vec{\mathsf{li}}]$.

The distribution of $\vec{\mathsf{LO}}^{1:t}$ in $M_Q[\vec{\mathsf{HI}}^{1:t} := \vec{\mathsf{hi}}][\vec{\mathsf{LI}}^{1:t} := \vec{\mathsf{li}}]$ is $\mathcal{P}(\vec{\mathsf{LO}}^{1:t}{=}\vec{\mathsf{lo}} \mid \mathsf{do}(\vec{\mathsf{HI}}^{1:t}{:=}\vec{\mathsf{hi}}), \mathsf{do}(\vec{\mathsf{LI}}^{1:t}{:=}\vec{\mathsf{li}}))$ for various values of $\vec{\mathsf{lo}}$. Thus, Condition (2) is equivalent to **(3)** there exists $\vec{\imath}_1$ and $\vec{\imath}_2$ of length t such that $\lfloor\vec{\imath}_1{\downarrow}L\rfloor = \lfloor\vec{\imath}_2{\downarrow}L\rfloor$ and there exists $\vec{\mathsf{lo}}$ such that $\mathcal{P}(\vec{\mathsf{LO}}^{1:t}{=}\vec{\mathsf{lo}} \mid \mathsf{do}(\vec{I}^{1:t}{:=}\vec{\imath}_1)) \neq \mathcal{P}(\vec{\mathsf{LO}}^{1:t}{=}\vec{\mathsf{lo}} \mid \mathsf{do}(\vec{I}^{1:t}{:=}\vec{\imath}_2))$.

By Property 1 [114, pg 24], Condition (3) is equivalent to **(4)** there exist $\vec{\imath}_1$ and $\vec{\imath}_2$ of length t such that $\lfloor \vec{\imath}_1{\downarrow}L \rfloor = \lfloor \vec{\imath}_2{\downarrow}L \rfloor$ and there exists $\vec{\mathsf{lo}}$ such that $\mathcal{P}(\vec{\mathsf{LO}}^{1:t}{=}\vec{\mathsf{lo}} \mid \mathsf{do}(\vec{I}^{1:t-1}{:=}\vec{\imath}_1)) \neq \mathcal{P}(\vec{\mathsf{LO}}^{1:t}{=}\vec{\mathsf{lo}} \mid \mathsf{do}(\vec{I}^{1:t-1}{:=}\vec{\imath}_2))$ since the output at time t does not depend upon the input at time t in M_Q.

By Lemma 1, $\mathcal{P}(\vec{\mathsf{LO}}^{1:t}{=}\vec{\mathsf{lo}} \mid \mathsf{do}(\vec{I}^{1:t-1}{:=}\vec{\imath})) = \lfloor Q(\vec{\imath}){\downarrow}L \rfloor(\vec{\mathsf{lo}})$. Thus, Condition (4) is equivalent to **(5)** there exists input sequences $\vec{\imath}_1$, $\vec{\imath}_2$, and $\vec{\mathsf{lo}}$ such that $\lfloor \vec{\imath}_1{\downarrow}L \rfloor = \lfloor \vec{\imath}_2{\downarrow}L \rfloor$ and $\lfloor Q(\vec{\imath}_1){\downarrow}L \rfloor(\vec{\mathsf{lo}}) \neq \lfloor Q(\vec{\imath}_2){\downarrow}L \rfloor(\vec{\mathsf{lo}})$.

Condition (5) is equivalent to Condition (1). Thus, Conditions (1) and (2) are equivalent as needed. □

A.5 Details of Experiments

When observing Google's behavior, we first 'opted-in' to receive interest-based Google Ads across the web on every test instance by visiting the Google Ad Settings page and clicking the **Opt-in** link. This placed a Doubleclick cookie on the browser instance. No ads were clicked in an automated fashion throughout any experiment.

A.5.1 Experiment 1

This experiment suggested that Google associates users with various ad pools switching users from pool to pool over time. Figure A.1 shows the ads for the second instance. We also ran the same experiment with different intervals between successive reloads. We tested intervals of 0s, 5s, 15s, 30s, 60s, and 120*s*, the ad-plots of which are shown in Figure A.2. All the above experiments were conducted in 2013. Figure A.3 shows the results for other webpages, conducted in 2015: Times of India (`http://timesofindia.indiatimes.com/international-home`), BBC (`http://www.bbc.com/news/`), and Fox News (`http://www.foxnews.com/us/index.html`).

Table A.1: For Experiments 2 and 3, the list of websites returned by Google upon searching with corresponding term. These websites were used for creating the profile of an auto enthusiast.

"BMW buy": www.bmwusa.com/, www.autotrader.com/find/BMW-328i-cars-for-sale.jsp, www.autotrader.com/find/used-BMW-cars-for-sale.jsp, www.bmw.com/, www.bmwmotorcycles.com/, autos.aol.com/new-cars/, www.exchangeandmart.co.uk/used-cars-for-sale/bmw, en.wikipedia.org/wiki/BMW, www.cars.com/bmw/, autos.aol.com/bmw/

"Audi purchase": www.audiusa.com/inventory/european-delivery, www.audiusa.com/help/leasing, www.audiusa.com/myaudi/finance, www.audiusa.com/myaudi/offers-programs, www.audiusa.com/inventory/certified-pre-owned, www.audisupplier.com/, townhall-talk.edmunds.com/direct/view/.f1cc6d7, www.autotrader.com/find/Audi-A3-cars-for-sale.jsp, en.wikipedia.org/wiki/Audi, jalopnik.com/5903083/why-audi-just-bought-ducati

"new cars": www.edmunds.com/new-cars/, www.edmunds.com/car-reviewsautos.yahoo.com/new-cars.html, autos.yahoo.com/new-cars.html, www.kbb.com/new-cars/, www.autotrader.com/research/new-cars/, www.autotrader.com/buy-a-new-car.jsp, www.cars.com/, autos.aol.com/new-cars/, www.newcars.com/, www.motortrend.com/new_cars/

"local car dealers": www.edmunds.com/dealerships/, www.cars.com/dealers/search.action, www.cochran.com/, www.autotrader.com/find/pittsburgh.jsp, www.baierl.com/, www.kbb.com/car-dealers-and-inventory/, www.enterprisecarsales.com/location/.../Enterprise_Car_Sales_Pittsburgh, autos.aol.com/new-cars, www.toyota.com/dealers/

"autos and vehicles": www.youtube.com/channel/HCLfhQGBR0ujg, www.youtube.com/channel/HCHXCPGmshRz4, www.youtube.com/live/autos, en.wikipedia.org/wiki/Automobile, www.veoh.com/list/videos/autos_and_vehicles, vidstatsx.com/most-popular-autos-vehicles-videos-today, www.savevid.com/category/auto-vehicles, www.smbiz.com/sbrl003.html, www.pinterest.com/hasaniqbal/autos-and-vehicles/, www.justluxe.com/lifestyle/car/articles-2.php

"cadillac prices": www.truecar.com/prices-new/cadillac/, www.motortrend.com/new_cars, autos.msn.com/browse/Cadillac.aspx, www.nadaguides.com/Cars/Cadillac, autos.yahoo.com/new-cars.html, www.gizmag.com/cadillac-elr-plug-in-hybrid-price/29389/, www.autonews.com/article/20131011/RETAIL03/131019967/, www.cadillac.com/, usnews.rankingsandreviews.com/cars-trucks/browse/cadillac, www.automobilemag.com/car_prices/01/cadillac/

"best limousines": www.medialightbox.com/blog/.../the-10-best-limousines-in-the-world/, www.bestlimousines.com/, www.celebritylimospittsburgh.com/, www.tdflimo.com/, www.limo.com/limo-pittsburgh-limousines.php, www.pittsburghluxurylimoservice.com/, www.angieslist.com/companylist/, www.forbes.com/2005/03/10/cx_dl_0310feat_bill05.html, www.thebestlimousine.com/, www.youtube.com/watch?v=0iqi6jHviJ0

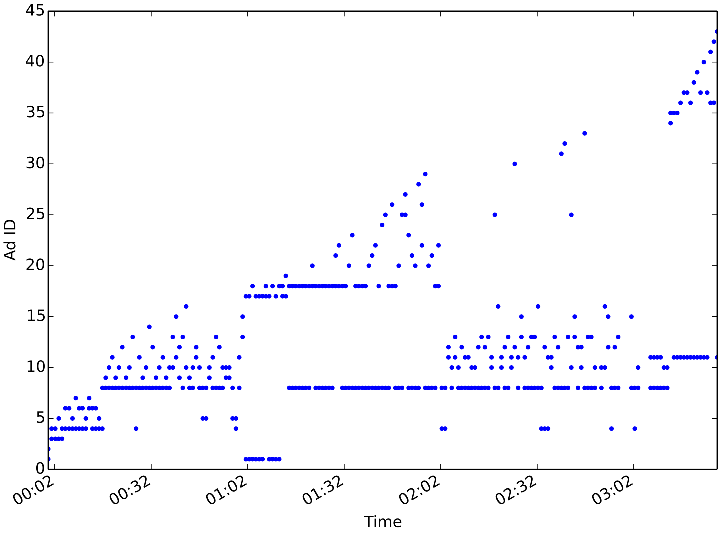
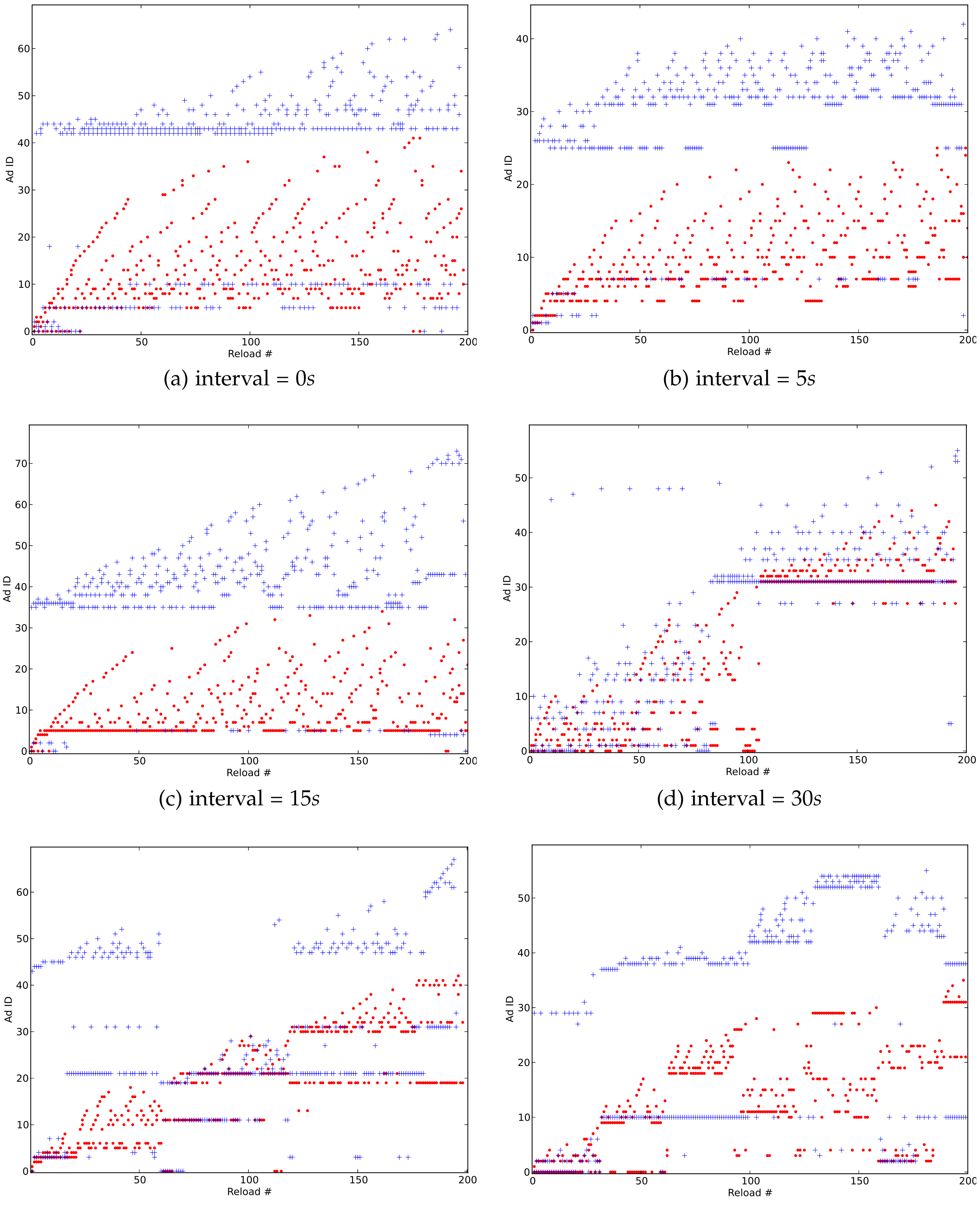


Figure A.1: The second browser instance visiting the Chicago Tribune for Experiment 1 The time interval for collection was one minute. The x-axis is time measured in hh:mm.

A.5.2 Experiment 2

A primary browser instance would first establish an interest in cars by visiting car-related websites. The car-related sites selected by collecting the top 10 websites excluding images, news articles or ads returned by Google when queried with the search terms “BMW buy”, “Audi purchase”, “new cars”, “local car dealers”, “autos and vehicles”, “cadillac prices”, and “best limousines” are shown in Table A.1. Note that the results from “local car dealers” has only 9 results because the page `local.yahoo.com/pittsburgh/Automotive/Dealers/Used+Car+Dealers` took a long time to load and was manually removed from the training pages.

During the 10 rounds of ad collections, each round would attempt to reload the International Homepage of Times of India (`http://timesofindia.indiatimes.com/international-home`) 10 times. Occasionally it would time out instead of reloading. We set the page-load-timeout to be 60 seconds. We repeated the experiment four times (twice using 10 rounds and twice using 20 rounds) and found that the page would not always load completely resulting in fewer ads being collected. Details on the number of ads collected by the primary browser instance in each round are shown in Table A.2.



(a) interval = 0*s*

(b) interval = 5*s*

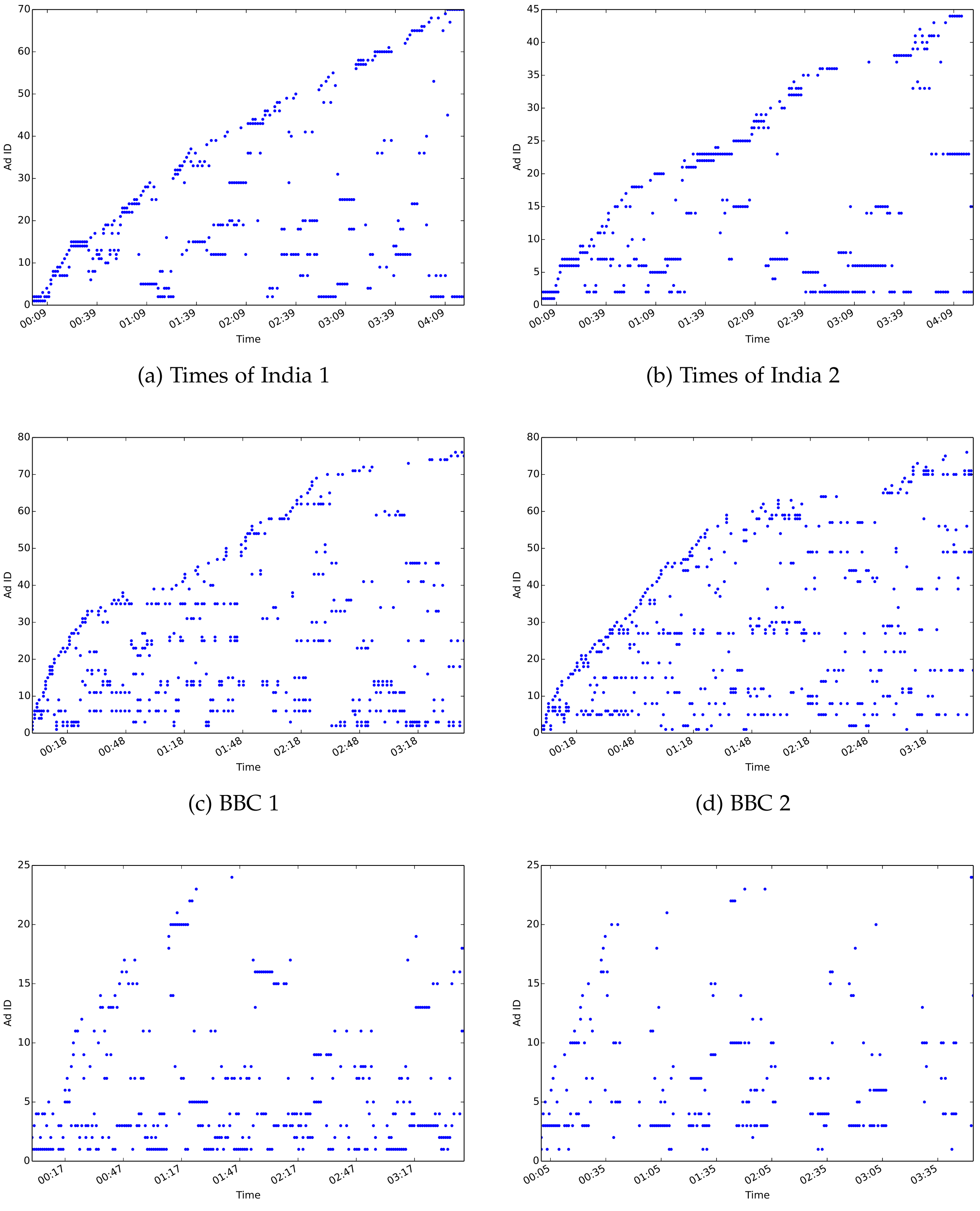
(c) interval = 15*s*

(d) interval = 30*s*

(e) interval = 60*s*

(f) interval = 120*s*

Figure A.2: For Experiment 1, plots of ads from Instances 1 and 2 in of the six experiments with varying time intervals between reloads. Observe that the pooling behavior appears for the first time in A.2(d), where the pool seems to switch somewhere around the 80th reload. After that the number of these switches keep increasing in successive plots with the reload interval.



(a) Times of India 1

(b) Times of India 2

(c) BBC 1

(d) BBC 2

(e) Fox News 1

(f) Fox News 2

Figure A.3: For Experiment 1, plots of ads from the Times of India, BBC, and Fox News. The time interval for collection was one minute for each plot. The x-axis is time measured in hh:mm.

Table A.2: For Experiment 2, the number of unique ads collected. I denotes the set of all ads collected from the primary browser instance running in isolation, while P denotes the same collected from the primary browser instance running in parallel. This table shows the number of ads collected in each round as well as the total number of ads and the number of unique ads in I and P. The stars represent numbers from the instances running in isolation.

Data set	#rounds	ads (unique) collected by primary browser per round	total (unique) in I	total (unique) in P
1	10	*50(13), *50(13), 50(8), 50(10), *50(10), 50(12), *50(13), 50(11), 50(7), *50(17)	250(37)	250(25)
2	10	50(11), *50(14), 50(15), 50(11), 50(13), *50(19), *50(13), *50(14), 45(11), *50(14)	250(46)	245(33)
3	20	*50(12), *50(12), 42(11), 50(14), *50(12), 50(11), 50(13), *50(18), *50(15), 50(15), 50(14), 50(9), 50(17), *50(17), 50(10), *45(10), *50(12), 50(13), *50(16), *45(13)	490(58)	492(47)
4	20	50(10), 50(10), 50(15), *50(14), 50(10), *50(17), 50(13), *40(11), 50(10), 50(16), *50(14), *50(11), *50(14), 50(13), 45(11), *50(14), *50(14), 50(12), *50(12), *45(16)	485(57)	495(52)

A.5.3 Experiment 3

As in Experiment 2, an instance manifests its interest by visiting the top 10 websites returned by Google when queried with certain automobile-related terms: "BMW buy", "Audi purchase", "new cars", "local car dealers", "autos and vehicles", "cadillac prices", and "best limousines". Thus, they visited the same websites as in Experiment 2 (see Table A.1).

Across all runs of the experiment, we collected 9832 ads with 281 being unique. Table A.3 shows the number of ads collected by each instance. Notice that both outliers were in the 19th run and in the experimental group.

Across all runs of the control-control experiment, we collected 9304 ads with 295 being unique. Table A.5 shows the number of ads collected by each instance. The p-values that the permutation tests yielded for the control-control experiment are shown in Table A.6. We can see that each of the statistics produced one statistically significant result except for the χ^2, which produced 12. This seems to indicate that the χ^2-test is more prone to showing false-positives than the permutation tests.

Table A.3: For Experiment 3, how the ads were distributed over the 10 different instances. T denotes the set of all ads collected from the trained instances, while U denotes the same collected from the untrained instances. The number of ads collected by each instance in $\{i_1 \ldots i_{10}\}$ is shown in the left half of the table. The right half of the table shows the total number of ads and the number of unique ads in T and U.

Data set	i_1	i_2	i_3	i_4	i_5	i_6	i_7	i_8	i_9	i_{10}	Total(T)	Unique(T)	Total(U)	Unique(U)
1	45	50	45	50	45	50	45	50	45	50	235	28	240	44
2	50	50	50	49	50	50	50	50	50	50	250	28	249	38
3	50	50	50	50	50	50	50	50	50	50	250	38	250	30
4	50	50	50	50	50	50	50	50	50	50	250	28	250	34
5	50	50	50	50	50	50	50	50	50	50	250	36	250	31
6	50	50	50	50	50	50	46	50	50	50	250	31	246	37
7	42	50	50	50	50	50	50	50	50	50	242	25	250	39
8	50	50	50	50	50	50	50	50	50	50	250	27	250	22
9	50	50	45	50	50	50	50	48	50	50	250	29	243	52
10	50	50	50	50	50	50	50	50	49	50	249	27	250	30
11	50	50	50	50	50	50	50	50	50	50	250	29	250	38
12	50	50	50	48	50	49	50	50	50	50	250	35	247	38
13	50	50	50	50	50	50	48	50	50	50	250	37	248	30
14	50	50	50	50	50	50	50	50	50	50	250	52	250	28
15	50	50	50	50	50	50	50	50	50	50	250	40	250	35
16	50	50	50	50	50	50	50	50	50	50	250	24	250	40
17	50	50	41	50	50	48	49	50	50	50	250	39	238	38
18	50	50	45	50	50	50	50	50	50	50	250	26	245	44
19	50	50	50	50	0	50	0	50	50	50	150	24	250	53
20	50	50	50	50	50	50	50	50	50	50	250	46	250	34

Table A.4: p-values for the permutation tests in the experiment-control experiment

Data set	s_{sim}	s_{kw}	s_{kw01}	χ^2
1	0.007937	0.003968	0.222222	0.113846
2	0.007937	0.003968	1.000000	nan
3	0.015873	0.019841	0.500000	0.291841
4	0.007937	0.003968	0.083333	0.038434
5	0.007937	0.099206	1.000000	nan
6	0.007937	0.003968	0.500000	0.291841
7	0.007937	0.003968	0.222222	0.113846
8	0.007937	0.003968	1.000000	nan
9	0.007937	0.003968	1.000000	nan
10	0.007937	0.003968	0.222222	0.113846
11	0.460317	0.015873	0.500000	0.291841
12	0.023810	0.023810	1.000000	nan
13	0.007937	0.003968	1.000000	nan
14	0.015873	0.003968	1.000000	nan
15	0.039683	0.011905	1.000000	nan
16	0.007937	0.003968	0.500000	0.291841
17	0.031746	0.003968	1.000000	nan
18	0.015873	0.007937	0.500000	0.291841
19	0.007937	0.087302	1.000000	0.113846
20	0.111111	0.003968	0.500000	0.291841
Number $< 5\%$	18	18	0	1

Across all runs of the treatment-treatment experiment, we collected 9741 ads with 243 being unique. Table A.7 shows the number of ads collected by each instance. The p-values for the treatment-treatment experiments are shown in Table A.8. Here too, we would expect not to find statistically significant results. The χ^2-test once again shows more false-positives than the permutation tests. These numbers indicate that the $\mathsf{pt}(s_{\mathsf{sim}})$ and $\mathsf{pt}(s_{\mathsf{kw}})$ are good indicators of statistical significance in our setting.

Table A.5: For Experiment 3, how the ads were distributed over the 10 different instances in the *control-control* experiment. 5 out these 10 were randomly assigned to T, while the remaining to U. Observe that data-set 8 is an outlier because the instances in that round returned much fewer ads.

Data set	i_1	i_2	i_3	i_4	i_5	i_6	i_7	i_8	i_9	i_{10}	Total(T)	Unique(T)	Total(U)	Unique(U)
1	45	30	30	45	25	40	45	45	35	29	190	31	179	44
2	50	50	50	50	50	50	50	23	50	50	223	33	250	50
3	45	40	45	50	45	45	45	45	45	45	225	37	225	39
4	50	50	50	50	50	45	50	50	50	50	245	39	250	39
5	50	50	50	50	46	50	50	50	50	45	245	33	246	57
6	50	50	45	50	50	50	50	45	45	50	250	45	235	38
7	50	47	50	50	50	50	50	50	50	50	250	42	247	34
8	25	15	25	0	9	30	19	0	0	15	114	15	24	16
9	50	50	50	50	50	45	50	45	50	50	245	37	245	33
10	45	45	50	50	50	45	45	45	50	50	245	36	230	47
11	50	50	45	50	45	50	50	50	44	50	239	35	245	37
12	50	49	50	50	50	40	45	50	50	50	235	33	249	36
13	50	50	50	50	45	50	50	50	50	50	245	36	250	24
14	50	50	50	50	50	50	50	50	50	50	250	31	250	28
15	50	50	50	50	46	50	50	50	50	47	246	45	247	43
16	50	50	50	50	50	50	50	50	50	50	250	36	250	35
17	50	50	50	49	50	50	50	50	50	50	249	37	250	36
18	50	50	50	50	50	50	50	50	50	50	250	26	250	27
19	50	50	50	50	50	50	50	37	50	50	237	36	250	33
20	50	50	50	50	50	50	50	50	49	50	249	37	250	34

Table A.6: For Experiment 3, p-values for the for control-control experiment. Note that the significant p-values are from data-set 8, which we showed in Table A.5 to an outlier.

.

Data set	s_{sim}	s_{kw}	s_{kw01}	χ^2
1	0.373016	0.857143	0.777778	1.000000
2	0.063492	0.293651	0.261905	0.196706
3	0.603175	0.920635	0.777778	1.000000
4	0.436508	0.440476	0.500000	0.291841
5	0.071429	0.869048	1.000000	nan
6	0.309524	0.158730	0.500000	0.490153
7	0.103175	0.527778	1.000000	nan
8	0.007937	0.003968	0.003968	0.001565
9	0.547619	0.134921	0.222222	0.113846
10	0.119048	1.000000	1.000000	nan
11	0.936508	0.234127	0.222222	0.113846
12	0.285714	0.769841	0.222222	0.113846
13	0.761905	0.440476	0.896825	0.527089
14	0.642857	0.408730	1.000000	nan
15	0.468254	0.738095	1.000000	nan
16	0.476190	0.095238	0.500000	0.291841
17	0.984127	0.186508	0.500000	0.490153
18	0.746032	0.440476	0.896825	0.527089
19	0.611111	0.420635	0.500000	0.490153
20	0.071429	0.936508	0.777778	1.000000
Number $<$ 5%	1	1	1	1

Table A.7: For Experiment 3, how the ads were distributed over the 10 different instances in the *treatment-treatment* experiment. 5 out these 10 were randomly assigned to T, while the remaining to U.

Data set	i_1	i_2	i_3	i_4	i_5	i_6	i_7	i_8	i_9	i_{10}	Total(T)	Unique(T)	Total(U)	Unique(U)
1	50	50	50	50	50	50	45	50	50	50	245	31	250	33
2	50	50	50	50	50	50	45	50	50	43	250	33	238	42
3	50	50	50	50	50	50	50	45	50	50	245	37	250	37
4	50	50	50	50	50	50	50	50	50	50	250	31	250	45
5	49	49	50	50	50	50	50	50	50	49	248	34	249	46
6	50	50	50	50	50	50	50	50	45	45	240	40	250	32
7	45	50	50	50	50	50	50	36	50	50	250	40	231	40
8	50	40	50	50	50	50	50	50	50	50	240	36	250	35
9	50	50	45	50	45	40	50	50	40	45	230	26	235	33
10	50	50	50	50	50	50	50	50	50	50	250	33	250	32
11	50	49	50	50	50	50	50	50	50	50	249	35	250	41
12	45	45	50	50	50	0	50	50	45	50	195	25	240	43
13	45	50	50	50	50	50	50	50	45	0	195	21	245	37
14	50	50	50	49	46	50	50	50	50	50	250	37	245	28
15	50	50	50	50	50	50	50	49	50	50	250	39	249	28
16	50	50	50	50	50	50	50	50	50	50	250	28	250	33
17	50	50	50	50	50	50	50	50	50	50	250	23	250	45
18	50	50	50	50	50	50	50	50	50	50	250	38	250	37
19	50	45	47	45	50	50	50	50	50	50	247	34	240	26
20	45	45	50	50	50	50	50	45	50	50	240	44	245	34

Table A.8: For Experiment 3, p-values for the for treatment-treatment experiment

Data set	s_{sim}	s_{kw}	s_{kw01}	χ^2
1	0.634921	0.821429	1.000000	nan
2	0.722222	0.357143	1.000000	nan
3	0.134921	0.202381	1.000000	nan
4	0.492063	0.468254	1.000000	nan
5	0.103175	0.281746	1.000000	nan
6	0.952381	0.650794	1.000000	nan
7	0.515873	0.384921	1.000000	nan
8	0.547619	0.571429	1.000000	nan
9	0.492063	0.829365	1.000000	nan
10	0.523810	0.162698	1.000000	nan
11	0.198413	0.007937	1.000000	nan
12	0.515873	0.781746	1.000000	0.291840545144
13	0.222222	0.734127	1.000000	0.291840545144
14	0.563492	0.825397	1.000000	nan
15	0.103175	0.396825	1.000000	nan
16	0.674603	0.146825	0.500000	0.291840545144
17	0.063492	0.880952	0.500000	0.291840545144
18	0.325397	0.357143	1.000000	nan
19	0.119048	0.992063	1.000000	nan
20	0.476190	0.690476	1.000000	0.291840545144
Number $< 5\%$	0	1	0	0

Appendix B

Appendices for Experiments on Google's Ad Ecosystem

B.1 Tables

Table B.1 summarizes the results. Table B.2 covers the discrimination experiments with Table B.3 showing the top ads for experiment on gender and jobs. Table B.4 covers the opacity experiments with Table B.5 showing the top ads for the substance-abuse experiment and Table B.6 showing them for the disability experiment. Table B.7 show the experiments for effectful choice with Table B.8 showing the tops ads for online dating. Tables B.9 and B.10 cover ad choice.

B.2 Details of Methodology

Let the units be arranged in a vector $\vec{u}$ of length n. Let $\vec{t}$ be a *treatment vector*, a vector of length n whose entries are the treatments that the experimenter wants to apply to the units. In the case of just two treatments, $\vec{t}$ can be half full of the first treatment and half full of the second. Let a be an *assignment* of units to treatments, a bijection that maps each entry of $\vec{u}$ to an entry in $\vec{t}$. That is, an assignment is a permutation on the set of indices of $\vec{u}$ and $\vec{t}$.

The result of the experiment is a vector of observations $\vec{y}$ where the ith entry of $\vec{y}$ is the response measured for the unit assigned to the ith treatment in $\vec{t}$ by the assignment used. In a randomized experiment, such as those AdFisher runs, the actual assignment used is selected at random uniformly over some set of possible assignments $\mathcal{A}$.

Let s be a test statistic of the observations of the units. That is $s : \mathcal{Y}^n \to \mathcal{R}$ where $\mathcal{Y}$ is the set of possible observations made over units, n is the number of units, and $\mathcal{R}$ is the range of s. We require $\mathcal{R}$

Table B.1: Summary of our experimental results. Ads are collected from the Times of India (TOI) or the Guardian (G). We report how long each experiment took, how many ads were collected for it, and what result we concluded.

Property	Treatment	Other Actions	Source	When	Length (hrs)	# ads	Result
Nondiscrimination	Gender	-	TOI	May	10	40,400	Inconclusive
	Gender	Jobs	TOI	May	45	43,393	Violation
	Gender	Jobs	TOI	July	39	35,032	Inconclusive
	Gender	Jobs	G	July	53	22,596	Inconclusive
	Gender	Jobs & Top 10	TOI	July	58	28,738	Inconclusive
Data use transparency	Substance abuse	-	TOI	May	37	42,624	Violation
	Substance abuse	-	TOI	July	41	34,408	Violation
	Substance abuse	-	G	July	51	19,848	Violation
	Substance abuse	Top 10	TOI	July	54	32,541	Violation
	Disability	-	TOI	May	44	43,136	Violation
	Mental disorder	-	TOI	May	35	44,560	Inconclusive
	Infertility	-	TOI	May	42	44,982	Inconclusive
	Adult websites	-	TOI	May	57	35,430	Inconclusive
Effectful choice	Opting out	-	TOI	May	9	18,085	Compliance
	Dating interest	-	TOI	May	12	35,737	Compliance
	Dating interest	-	TOI	July	17	22,913	Inconclusive
	Weight loss interest	-	TOI	May	15	31,275	Compliance
	Weight loss interest	-	TOI	July	15	27,238	Inconclusive
Ad choice	Dating interest	-	TOI	July	1	1,946	Compliance
	Weight loss interest	-	TOI	July	1	2,862	Inconclusive
	Weight loss interest	-	TOI	July	1	3,281	Inconclusive

Table B.2: Results from the discrimination experiments sorted by unadjusted p-value. TOI stands for Times of India. G stands for the Guardian. * denotes statistically significant results under the Holm-Bonferroni method. All these experiments had 100 blocks

Treatment	Other visits	Measurement	# ads (# unique ads)		Acc.	Unadj. p-value	Adj. p-value
			female	male			
Gender	Jobs	TOI, May	21,766 (545)	21,627 (533)	93%	0.0000053	0.0000265*
Gender	Jobs	G, July	11,366 (410)	11,230 (408)	57%	0.12	0.48
Gender	Jobs & Top 10	TOI, July	14,507 (461)	14,231 (518)	56%	0.14	n/a
Gender	Jobs	TOI, July	17,019 (673)	18,013 (690)	55%	0.20	n/a
Gender	-	TOI, May	20,137 (603)	20,263 (630)	48%	0.77	n/a

Table B.3: Top URL+titles for the gender and jobs experiment on the Times of India in May.

Title	URL	Coeff.	# agents		# impr.	
			f.	m.	f.	m.
Top ads for identifying the simulated female group (f.)						
Jobs (Hiring Now)	www.jobsinyourarea.co	0.34	6	3	45	8
4Runner Parts Service	www.westernpatoyotaservice.com	0.281	6	2	36	5
Criminal Justice Program	www3.mc3.edu/Criminal+Justice	0.247	5	1	29	1
Goodwill - Hiring	goodwill.careerboutique.com	0.22	45	15	121	39
UMUC Cyber Training	www.umuc.edu/cybersecuritytraining	0.199	19	17	38	30
Top ads for identifying agents in the simulated male group (m.)						
\$200k+ Jobs - Execs Only	careerchange.com	−0.704	60	402	311	1816
Find Next \$200k+ Job	careerchange.com	−0.262	2	11	7	36
Become a Youth Counselor	www.youthcounseling.degreeleap.com	−0.253	0	45	0	310
CDL-A OTR Trucking Jobs	www.tadrivers.com/OTRJobs	−0.149	0	1	0	8
Free Resume Templates	resume-templates.resume-now.com	−0.149	3	1	8	10

Table B.4: Results from transparency experiments. TOI stands for Times of India, G stands for the Guardian. Every experiment for this property ran with 100 blocks. * denotes statistically significant results under the Holm-Bonferroni method.

Treatment	Other visits	Measurement	# ads (# unique ads)		Acc.	Unadj. p-value	Adj. p-value
			experimental	control			
Substance abuse	-	TOI, May	20,420 (427)	22,204 (530)	81%	0.0000053	0.0000424*
Substance abuse	-	TOI, July	16,206 (653)	18,202 (814)	98%	0.0000053	0.0000371*
Substance abuse	Top 10	TOI, July	15,713 (603)	16,828 (679)	65%	0.0000053	0.0000318*
Disability	-	TOI, May	19,787 (546)	23,349 (684)	75%	0.0000053	0.0000265*
Substance abuse	-	G, July	8,359 (242)	11,489 (319)	62%	0.0075	0.03*
Mental disorder	-	TOI, May	22,303 (407)	22,257 (465)	59%	0.053	0.159
Infertility	-	TOI, May	22,438 (605)	22,544 (625)	57%	0.11	n/a
Adult websites	-	TOI, May	17,670 (602)	17,760 (580)	52%	0.42	n/a

to be ordered numbers such as the natural or real numbers. We allow s to treat its arguments differently, that is, the order in which the observations are passed to s matters.

If the null hypothesis is true, then we would expect the value of s to be the same under every permutation of the arguments since the assignment of units to treatments should not matter under the null hypothesis. This reasoning motivates the permutation test. The value produced by a (one-tailed signed) permutation test given observed responses $\vec{y}$ and a test statistic s is

$$\frac{|\{\, a \in \mathcal{A} \mid s(\vec{y}) \leq s(a(\vec{y}))\,\}|}{|\mathcal{A}|} = \frac{1}{|\mathcal{A}|} \sum_{a \in \mathcal{A}} I[s(\vec{y}) \leq s(a(\vec{y}))] \tag{B.1}$$

where the assignments in $\mathcal{A}$ only swaps nearly identical units and $I[\cdot]$ returns 1 if its argument is true and 0 otherwise.

Table B.5: Top URL+titles for substance abuse experiment on the Times of India in May.

Title	URL	Coeff.	# agents		# impr.	
			cont.	exp.	cont.	exp.
Top ads for identifying agents in the experimental group						
The Watershed Rehab	www.thewatershed.com/Help	−0.888	0	280	0	2276
Watershed Rehab	www.thewatershed.com/Rehab	−0.670	0	51	0	362
The Watershed Rehab	Ads by Google	−0.463	0	258	0	771
Veteran Home Loans	www.vamortgagecenter.com	−0.414	13	15	22	33
CAD Paper Rolls	paper-roll.net/Cad-Paper	−0.405	0	4	0	21
Top ads for identifying agents in control group						
Alluria Alert	www.bestbeautybrand.com	0.489	2	0	9	0
Best Dividend Stocks	dividends.wyattresearch.com	0.431	20	10	54	24
10 Stocks to Hold Forever	www.streetauthority.com	0.428	51	44	118	76
Delivery Drivers Wanted	get.lyft.com/drive	0.362	22	6	54	14
VA Home Loans Start Here	www.vamortgagecenter.com	0.354	23	6	41	9

Table B.6: Top URL+titles for disability experiment on the Times of India in May.

Title	URL	Coeff.	# agents		# impr.	
			cont.	exp.	cont.	exp.
Top ads for identifying agents in the experimental group						
Mobility Lifter	www.abilitiesexpo.com	−1.543	0	84	0	568
Standing Wheelchairs	www.abilitiesexpo.com	−1.425	0	88	0	508
Smoking MN Healthcare	www.stillaproblem.com	−1.415	0	24	0	60
Bike Prices	www.bikesdirect.com	−1.299	0	24	0	79
$19 Car Insurance - New	auto-insurance.quotelab.com/MN	−1.276	0	6	0	9
Top ads for identifying agents in control group						
Beautiful Women in Kiev	anastasiadate.com	1.304	190	46	533	116
Melucci DDS	Ads by Google	1.255	4	2	10	6
17.2% 2013 Annuity Return	advisorworld.com/CompareAnnuities	1.189	30	5	46	6
3 Exercises To Never Do	homeworkoutrevolution.net	1.16	1	1	3	1
Find CNA Schools Near You	cna-degrees.courseadvisor.com	1.05	22	0	49	0

Table B.7: Results from effectful choice experiments using the Times of India sorted by unadjusted p-value. * denotes statistically significant results under the Holm-Bonferroni method.

Experiment	Blocks	# ads (# unique ads)			Acc.	Unadj.	Adj.
		rem./opt-out	kept/opt-in	total		p-value	p-value
Opting out	54	9,029 (139)	9,056 (293)	18,085 (366)	83%	0.0000053	0.0000265*
Dating (May)	100	17,975 (518)	17,762 (457)	35,737 (669)	74%	0.0000053	0.0000212*
Weight Loss (May)	83	15,826 (367)	15,449 (427)	31,275 (548)	60%	0.041	0.123
Dating (July)	90	11,657 (727)	11,256 (706)	22,913 (1,014)	59%	0.070	n/a
Weight Loss (July)	100	14,168 (917)	13,070 (919)	27,238 (1,323)	52%	0.41	n/a

Table B.8: Top URL+titles for the dating experiment on Times of India in May.

Title	URL	Coeff.	# agents		# impr.	
			kept	rem.	kept	rem.
Top ads for identifying the group that kept dating interests						
Are You Single?	www.zoosk.com/Dating	1.583	367	33	2433	78
Top 5 Online Dating Sites	www.consumer-rankings.com/Dating	1.109	116	10	408	13
Why can't I find a date?	www.gk2gk.com	0.935	18	3	51	5
Latest Breaking News	www.onlineinsider.com	0.624	2	1	6	1
Gorgeous Russian Ladies	anastasiadate.com	0.620	11	0	21	0
Top ads for identifying agents in the group that removed dating interests						
Car Loans w/ Bad Credit	www.car.com/Bad-Credit-Car-Loan	−1.113	5	13	8	37
Individual Health Plans	www.individualhealthquotes.com	−0.831	7	9	21	46
Crazy New Obama Tax	www.endofamerica.com	−0.722	19	31	22	51
Atrial Fibrillation Guide	www.johnshopkinshealthalerts.com	−0.641	0	6	0	25
Free $5 - $25 Gift Cards	swagbucks.com	−0.614	4	11	5	32

Table B.9: Setup for and ads from ad choice experiments. All experiments used 10 blocks. The same keywords are used to remove ad interests, as well as create the test statistic for permutation test.

Experiment	Keywords	# ads (# unique ads)		appearances	
		removed	kept	removed	kept
Dating	dating, romance, relationship	952 (117)	994 (123)	34	109
Weight Loss (1)	fitness	1,461 (259)	1,401 (240)	21	16
Weight Loss (2)	fitness, health, fat, diet, exercise	1,803 (199)	1,478 (192)	2	15

Table B.10: P-values from ad choice experiments sorted by the (unflipped) p-value. The Bonferroni adjusted p-value is only adjusted for the two hypotheses tested within a single experiment (row). The Holm-Bonferroni adjusts for all 6 hypotheses. * denotes statistically significant results under the Holm-Bonferroni method.

Experiment	Unadjusted p-value	Bonferroni p-value	Holm-Bonferroni p-value	Unadjusted flipped p-value	Bonferroni flipped p-value	Holm-Bonferroni flipped p-value
Dating	0.0076	0.0152	0.0456*	0.9970	1.994	n/a
Weight Loss (2)	0.18	0.36	0.9	0.9371	1.8742	n/a
Weight Loss (1)	0.72	1.44	n/a	0.3818	0.7636	n/a

Blocking. For the blocking design, the set of units $\mathcal{U}$ is partitioned into k blocks $\mathcal{B}_1$ to $\mathcal{B}_k$. In our case, all the blocks have the same size. Let $|\mathcal{B}_i| = m$ for all i. The set of assignments $\mathcal{A}$ is equal to the set of functions from $\mathcal{U}$ to $\mathcal{U}$ that are permutations not mixing up blocks. That is, a such that for all i and all u in $\mathcal{B}_i$, $a(u) \in \mathcal{B}_i$. Thus, we may treat $\mathcal{A}$ as k permutations, one for each $\mathcal{B}_i$. Thus, $\mathcal{A}$ is isomorphic to $\times_{i=1}^{k} \Pi(\mathcal{B}_i)$ where $\Pi(\mathcal{B}_i)$ is the set of all permutations over $\mathcal{B}_i$. Thus, $| \times_{i=1}^{k} \Pi(\mathcal{B}_i)| = (m!)^k$. Thus, (B.1) can be computed as

$$\frac{1}{(m!)^k} \sum_{a \in \times_{i=1}^{k} \Pi(\mathcal{B}_i)} I[s(\vec{y}) \leq s(a(\vec{y}))] \tag{B.2}$$

Sampling. Computing (B.2) can be difficult when the set of considered arrangements is large. One solution is to randomly sample from the assignments $\mathcal{A}$. Let $\mathcal{A}'$ be a random subset of $\mathcal{A}$. We then use the approximation

$$\frac{1}{|\mathcal{A}'|} \sum_{a \in \mathcal{A}'} I[s(\vec{y}) \leq s(a(\vec{y}))] \tag{B.3}$$

Confidence Intervals. Let $\hat{P}$ be this approximation and p be the true value of (B.2). p can be understood as the frequency of arrangements that yield large values of the test statistic where *largeness* is determined to be at least as large as the observed value $s(\vec{y})$. That is, the probability that a randomly selected arrangement will yield a large value is p. $\hat{P}$ is the frequency of seeing large values in the $|\mathcal{A}'|$ sampled arrangements. Since the arrangements in the sample were drawn uniformly at random from $\mathcal{A}$ and each draw has probability p of being large, the number of large values will obey the binomial distribution. Let us denote this value as L. and $|\mathcal{A}'|$ as n. Since $\hat{P} = L/n$, $\hat{p} * n$ also obeys the binomial distribution. Thus,

$$\Pr[\hat{P} = \hat{p} \,|\, n, p] = \binom{n}{\hat{p}n} p^{\hat{p}n} (1-p)^{(1-\hat{p})n} \tag{B.4}$$

Thus, we may use a binomial proportion confidence interval. We use the Clopper-Pearson interval [26].

Test Statistic. The statistic we use is based on a classifier c. Let $c(y_i) = 1$ mean that c classifiers the ith observation as having come from the experimental group and $c(y_i) = 0$ as from the control group. Let $\neg(0) = 1$ and $\neg(1) = 0$. Let $\vec{y}$ be ordered so that all of the experimental group comes first. The statistic we use is

$$s(\vec{y}) = \sum_{i=1}^{n/2} c(y_i) + \sum_{i=n/2+1}^{n} \neg c(y_i)$$

This is the number correctly classified.

B.3 Holm-Bonferroni Correction

The Holm-Bonferroni Correction starts by ordering the hypotheses in a family from the hypothesis with the smallest (most significant) p-value p_1 to the hypothesis with the largest (least significant) p-value p_m [63]. For a hypothesis H_k, its unadjusted p-value p_k is compared to an adjusted level of significance $\alpha'_k = \frac{\alpha}{m+1-k}$ where α is the unadjusted level of significance (0.05 in our case), m is the total number of hypotheses in the family, and k is the index of hypothesis in the ordered list (counting from 1 to m). Let $k^\dagger$ be the lowest index k such that $p_k > \alpha'_k$. The hypotheses H_k where $k < k^\dagger$ are accepted as having statistically significance evidence in favor of them (more technically, the corresponding null hypotheses are rejected). The hypotheses H_k where $k \geq k^\dagger$ are not accepted as having significant evidence in favor of them (their null hypotheses are not rejected).

We report adjusted p-values to give an intuition about the strength of evidence for a hypothesis. We let $p'_k = p(m + 1 - k)$ be the adjusted p-value for H_k provided $k < k^\dagger$ since $p_k > \alpha'_k$ iff $p'_k > \alpha$. Note that the adjusted p-value depends not just upon its unadjusted value but also upon its position in the list. For the remaining hypotheses, we provide no adjusted p-value since their p-values are irrelevant to the correction beyond how they order the list of hypotheses.

Appendix C

Appendices for An Evaluation of PETs against Fingerprinting

C.1 Additional Tables

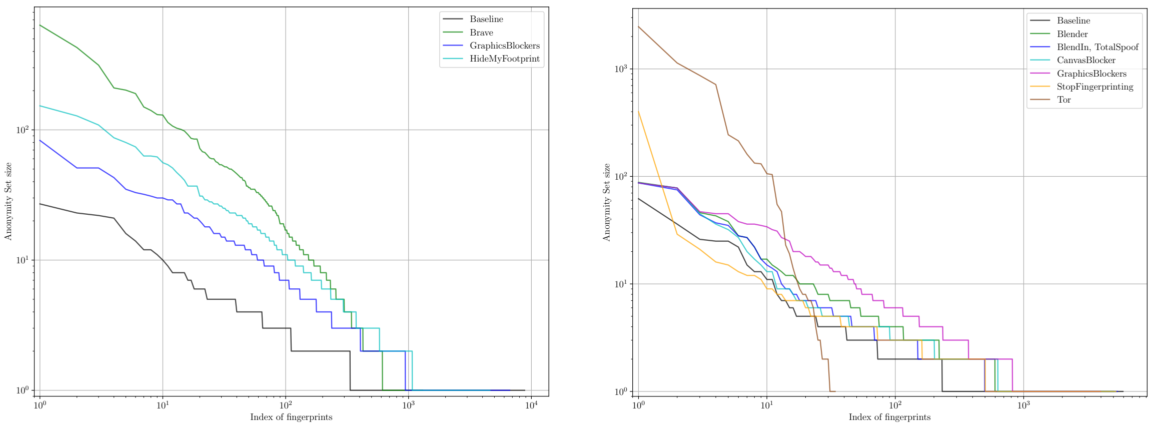


Figure C.1: The distribution of anonymity set sizes, plotted in log-scale, of fingerprints revealed by Chrome (left) and Firefox (right) PETs against aware-exact-match. Fingerprints are arranged in the decreasing order of the corresponding anonymity set size. The fingerprint with index 0 has the largest anonymity set size for the corresponding PET. Longer tails indicate higher numbers of browsing platforms residing in small anonymity sets.

Table C.1: Comparison of the samples. The sample of users we use is compared GlobalStat's sample and publically available information about the AmIUnique data set, of which our sample is a subset. Percentages of users with various attributes shown.

	Our Sample	GlobalStat Desktop	AmIUnique
Browser			
Chrome	45.6	65.98	39.4
Firefox	42.2	11.87	42.9
IE	2	7.28	3
Safari	7.74	5.87	3.5
Others	2.26	4.11	11.2
Operating System			
Windows	49.26	82.69	56.51
Mac	11.94	12.8	13.37
Linux	18.96	1.43	14.68
Others	19.84	2.17	15.44
Screen Resolution			
1920 × 1080	25.5	17.95	N/A
1366 × 768	8.8	28.75	N/A
1440 × 900	5	7.05	N/A
1536 × 864	3.18	4.46	N/A
1600 × 900	3	5.97	N/A
Others	54.52	35.82	N/A

Table C.2: Configurations of simulated browsing platforms in our main experiment. Font types Mordred, OldLondon, and OldLondonAlternate have been shortened as M, OL, and OLA respectively. (⋆) Browsing platform # 7 is simulated on a regularly used MacBook and has over 150 different fonts installed.

#	Type	OS	Addl. Fonts	Resolution	Locale & LANG	Timezone	Browser versions	
							Firefox	Chrome
1	VM	Ubuntu 16.04	M	450×721×24	ru_RU.UTF-8	GMT+6	56.0	63.0
2	VM	Ubuntu 16.04	M, OL	300×200×16	ja_JP.UTF-8	GMT+11	56.0	63.0
3	VM	Debian 8.10	-	320×256×8	vi_VN.UTF-8	GMT+3	56.0b3	62.0
4	VM	Debian 8.10	OL	2000×2000×16	de_DE.UTF-8	GMT-3	56.0	63.0
5	VM	Ubuntu 14.04	OLA	850×550×24+32	fr_FR.UTF-8	GMT-6	56.0	63.0
6	VM	Ubuntu 14.04	-	6000×3000×24+64	ar_SA.UTF-8	GMT-11	56.0	63.0
7	Local	macOS 10.13	(⋆)	1440×900×24	en_EN.UTF-8	GMT-8	56.0	63.0

Table C.3: Observed effects of PETs on attribute values in our main experiment. For brevity, we remove attributes not modified by any PET.

Attribute	Chrome							Firefox							
	BR	CD_C	CFB	GL	HMF	PE	TR	BI	BL	CD_F	CA	NE	SF	TBB	TO
buildID	·	·	·	·	·	·	·	FV	FV	×	×	×	×	FV	FV
canvas fingerprint	PV	SR	FV	SR	SR	×	×	×	×	TV	SR	×	×	FV	×
cpu class	·	·	·	·	·	·	·	FV	FV	×	×	×	CV	FV	FV
h.Accept-Language	CV	×	×	×	×	×	×	×	FV	×	×	×	×	FV	×
h.Pragma	·	·	·	·	CV	·	·	·	·	·	·	·	·	·	·
h.User-Agent	CV	×	×	×	SR	×	×	FV	FV	×	×	×	×	FV	FV
javascript fonts	×	×	×	×	×	×	×	×	×	×	×	×	×	PV	×
language	PV	×	×	×	×	×	×	×	FV	×	×	×	×	FV	×
platform	×	×	×	×	×	×	×	FV	FV	×	×	×	×	FV	FV
plugins	FV	×	×	×	×	×	×	·	·	·	·	·	·	·	·
screen.AvailHeight	×	×	×	×	×	×	×	×	×	×	×	×	FV	PV	×
screen.AvailTop	×	×	×	×	×	×	×	×	×	×	×	×	FV	FV	×
screen.AvailWidth	×	×	×	×	×	×	×	×	×	×	×	×	FV	PV	×
screen.Depth	·	·	·	·	·	·	·	×	×	×	×	×	FV	FV	×
screen.Height	×	×	×	×	×	×	×	×	×	×	×	×	FV	PV	×
screen.Pixel Ratio	×	×	×	×	×	×	×	×	×	×	×	×	FV	FV	×
screen.Width	×	×	×	×	×	×	×	×	×	×	×	×	FV	PV	×
timezone	×	×	×	×	×	×	×	×	×	×	×	×	×	FV	×
webGL.Data Hash	CV	SR	FV	FV	FV	×	×	×	×	FV	SR	×	×	PV	×
webGL.Renderer	CV	SR	FV	FV	FV	×	×	×	×	FV	×	×	×	FV	×
webGL.Vendor	CV	SR	FV	FV	FV	×	×	×	×	FV	×	×	×	FV	×

'SR': revealed values vary across reloads, 'SD': revealed values vary across domains, 'FV': differing original values fully standardize to one exposed value, 'PV': differing original values partially standardize to a smaller set of exposed values, 'CV': original values change without reduction in set of revealed values, 'TV': original values change with an increases in set of revealed values. For clarity of presentation, we only include attributes which were modified by at least one PET, and PETs which modified at least one attribute.

Table C.4: Mask models of PETs generated by our experimental method and used as input to the hybrid method

Attribute	Chrome					Firefox						
	BR	CD_C	CFB	GL	HMF	BI	BL	CD_F	CA	SF	TBB	TO
canvas fingerprint	●	●	●	●	●	·	·	●	●	·	●	·
h.Accept-Language	●	·	·	·	·	·	●	·	·	·	●	·
h.User-Agent	●	·	·	·	●	●	●	·	·	·	●	●
platform	·	·	·	·	·	●	●	·	·	·	●	●
plugins	●	·	·	·	·	●	●	●	●	●	●	●
screen.Depth	●	●	●	●	●	·	·	·	·	●	●	·
screen.Height	·	·	·	·	·	·	·	·	·	●	●	·
screen.Width	·	·	·	·	·	·	·	·	·	●	●	·
timezone	·	·	·	·	·	·	·	·	·	·	●	·
webGL.Data Hash	●	●	●	●	●	·	·	●	●	·	●	·
webGL.Renderer	●	●	●	●	●	·	·	●	·	·	●	·
webGL.Vendor	●	●	●	●	●	·	·	●	·	·	●	·

Table C.5: List of attribute values revealed by baseline Chrome and Firefox on the seven simulated browsing platforms in our main experiment. For clarity of presentation, we replace lists with long values by the length of the list.

Attribute	Chrome	Firefox
`DNT enabled`	[‘NC’]	[‘NC’]
`IE addBehavior`	[‘no’]	[‘no’]
`adBlock installed`	[‘no’]	[‘no’]
`buildID`	[‘Undefined’]	[‘20170926190823’, ‘20170815141045’]
`canvas fingerprint`	4 unique values	6 unique values
`cookies enabled`	[‘yes’]	[‘yes’]
`cpu class`	[‘unknown’]	[‘Linux x86_64’, ‘Intel Mac OS X 10.13’]
`h.Accept`	[‘text/html,application/xhtml+xml, application/xml;q=0.9,image/webp, image/apng,*/*;q=0.8’]	[‘text/html,application/xhtml+xml, application/xml;q=0.9,*/*;q=0.8’]
`h.Accept-Encoding`	[‘gzip, deflate’]	[‘gzip, deflate’]
`h.Accept-Language`	7 unique values	7 unique values
`h.Connection`	[‘keep-alive’]	[‘keep-alive’]
`h.Dnt`	[‘not sent’]	[‘not sent’]
`h.Pragma`	[‘not sent’]	[‘not sent’]
`h.Up.-Ins.-Req.`	[‘1’]	[‘1’]
`h.User-Agent`	3 unique values	2 unique values
`indexedDB`	[‘yes’]	[‘yes’]
`javascript fonts`	7 unique values	6 unique values
`language`	[‘ru’, ‘fr’, ‘en-US’, ‘vi’, ‘de’, ‘ar’, ‘ja’]	[‘fr’, ‘en-US’, ‘ru-RU’, ‘de’, ‘vi-VN’, ‘ar’, ‘ja’]
`lied with browser`	[‘yes’]	[‘no’]
`lied with language`	[‘no’]	[‘no’]
`lied with os`	[‘no’]	[‘no’]
`lied with res.`	[‘no’]	[‘no’]
`local storage`	[‘yes’]	[‘yes’]
`math.acosh(1e300)`	[‘Infinity’]	[‘Infinity’]
`math.asinh(1)`	[0.8813735870195429]	[0.8813735870195429]
`math.atanh(05)`	[0.5493061443340548]	[0.5493061443340548]
`math.cbrt(100)`	[4.641588833612778]	[4.641588833612778]
`math.cosh(10)`	[11013.232920103324]	[11013.232920103324]
`math.expm1(1)`	[1.718281828459045]	[1.7182818284590455]
`math.log1p(10)`	[2.3978952727983707]	[2.3978952727983707]
`math.sinh(1)`	[1.1752011936438014]	[1.1752011936438016]
`math.tanh(1)`	[0.7615941559557649]	[0.7615941559557649]
`openDB`	[‘yes’]	[‘no’]
`platform`	[‘Linux x86_64’, ‘MacIntel’]	[‘Linux x86_64’, ‘MacIntel’]
`plugins`	3 unique values	1 unique values
`screen.AvailHeight`	[256, 550, 200, 810, 2000, 721, 3000]	[256, 550, 200, 810, 2000, 721, 3000]
`screen.AvailLeft`	[0]	[0]
`screen.AvailTop`	[0, 23]	[0, 23]
`screen.AvailWidth`	[320, 6000, 450, 300, 2000, 850, 1440]	[320, 6000, 450, 300, 2000, 850, 1440]
`screen.Depth`	[24]	[24, 16, 8]
`screen.Height`	[256, 900, 550, 200, 2000, 721, 3000]	[256, 900, 550, 200, 2000, 721, 3000]
`screen.Left`	[‘undefined’]	[0]
`screen.Pixel Ratio`	[1, 2]	[1, 2]
`screen.Top`	[‘undefined’]	[0]
`screen.Width`	[320, 6000, 450, 300, 2000, 850, 1440]	[320, 6000, 450, 300, 2000, 850, 1440]
`session storage`	[‘yes’]	[‘yes’]
`timezone`	[480, 360, -660, -180, 180, -360, 660]	[480, 360, -660, -180, 180, -360, 660]
`touch.event`	[‘false’]	[‘false’]
`touch.max points`	[0]	[0]
`touch.start`	[‘false’]	[‘false’]
`webGL.Data Hash`	3 unique values	5 unique values
`webGL.Renderer`	2 unique values	5 unique values
`webGL.Vendor`	[‘Google Inc.’, ‘NVIDIA Corporation’]	[‘Not supported’, ‘VMware, Inc.’, ‘NVIDIA Corporation’]

Table C.6: Popularity of PETs in our list as of Dec 2017. The popularity of extensions were obtained from the Firefox add-on library and the Chrome extensions webstore. Tor's popularity was obtained from the Tor Metrics webpage.

PET	# users
Chrome	
Adblock Plus (ap$_{C}$)	10,000,000+
Brave (br)	NA
Canvas Defender (cd$_{C}$)	19,769
CanvasFingerprintBlock (cfb)	7,630
Glove (gl)	342
HideMyFootprint (hmf)	177
Ghostery (gh$_{C}$)	2,788,951
Privacy Extension (pe)	915
uBlock Origin (uo$_{C}$)	10,000,000+
Firefox	
Adblock Plus (ap$_{F}$)	13,760,128
Blend In (bi)	858
Blender (bl)	1,816
Canvas Defender (cd$_{F}$)	5,274
CanvasBlocker (ca)	27,170
Cookie Blocking (cb)	NA
Ghostery (gh$_{F}$)	1,064,473
Stop Fingerprinting (sf)	1,754
Tor (tbb)	4,000,000
Totalspoof (to)	265
Tracking Protection (tp)	NA
uBlock Origin (uo$_{F}$)	4,837,884

Table C.7: Attributes in the `amiunique` dataset

Attributes considered for evaluation
Accept-Language
User-Agent
canvas fingerprint
platform
plugins (Chrome only)
screen.Depth (Firefox only)
screen.Height
screen.Width
timezone
webGL.Data Hash
webGL.Vendor
webGL.Renderer
Attributes that do not differ in our simulated browsing platforms
adBlock installed
Accept
Accept-Encoding
local storage
session storage
IE addBehavior
cookies enabled
DNT enabled
Attributes we do not collect
orderHttp
connectionHttp
fontsFlash
resolutionFlash
languageFlash
platformFlash
octaneScore
sunspiderTime
Attributes which are functions of other attributes
pluginsHashed
canvasHashed
webGLHashed
fontsFlashHashed

Table C.8: Uniqueness metrics for different PETs on samples scaled according to their popularity

Chrome PETs				
PET	#users	entropy	prop_less1	prop_less10
HMF	177	7.340 ± 0.004	0.892 ± 0.003	1.000 ± 0.000
GL	342	8.285 ± 0.003	0.889 ± 0.002	1.000 ± 0.000
CFB	7630	11.568 ± 0.002	0.314 ± 0.001	0.905 ± 0.001
Firefox PETs				
PET	#users	entropy	prop_less1	prop_less10
TO	265	7.914 ± 0.004	0.900 ± 0.002	1.000 ± 0.000
BI	858	9.411 ± 0.003	0.785 ± 0.002	0.983 ± 0.001
SF	1754	9.998 ± 0.005	0.648 ± 0.001	0.937 ± 0.001
BL	1816	10.191 ± 0.003	0.617 ± 0.002	0.956 ± 0.001
CD$_F$	5274	10.658 ± 0.003	0.250 ± 0.001	0.847 ± 0.001

Table C.9: Configurations of simulated browsing platforms from our session experiment in Appendix C.2. FF represents Firefox, while C represents Chrome. Font types Mordred, OldLondon, and OldLondonAlternate have been shortened as M, OL, and OLA respectively. (⋆) Browsing platform # 7 is simulated on a regularly used MacBook and has over 150 different fonts installed.

#	Type	OS	Addl.	Resolution	Locale	Timezone	Browser ver.	
			Fonts		& LANG		FF	C
1	VM	Ubuntu 16.04	M	450×721×24	ru_RU.UTF-8	GMT+6	56.0	63.0
2	VM	Ubuntu 16.04	M, OL	300×200×16	ja_JP.UTF-8	GMT+11	56.0	63.0
3	VM	Debian 8.10	M, OL, OLA	320×256×8	vi_VN.UTF-8	GMT+3	56.0b3	62.0
4	VM	Debian 8.10	OL	2000×2000×16	de_DE.UTF-8	GMT-3	56.0	63.0
5	VM	Ubuntu 14.04	OLA	850×550×24+32	fr_FR.UTF-8	GMT-6	56.0b1	61.0
6	VM	Ubuntu 14.04	-	6000×3000×24+64	ar_SA.UTF-8	GMT-11	56.0	63.0

C.2 Additional Experiment

We use CS to simulate six browsing platforms. They are all virtual machines, two running 64-bit Ubuntu 14.04 (Trusty Tahr), two running 64-bit Ubuntu 16.04 (Xenial Xerus), and two running Debian 8.10 (Jessie). We introduce additional changes into these virtual machines to simulate differences in the system configurations. Specifically, we install different fonts and browser versions, setup different timezones, and simulate different screen resolutions and languages, The seventh browsing platform is simulated on a Macbook Pro. We perform measurements on Firefox and Chrome. More details about the configuration are available in Table C.9. CS drives these simulated browsing platforms to FS in the following pattern: five reloads of $\texttt{d}_1$; 45 minutes of idle time; five reloads of $\texttt{d}_1$; five reloads of $\texttt{d}_2$. It does so a total of 31 times—29 times for each PETs in our list, and twice for the two baseline browsers. All PETs are left in their default configuration.

We choose an idle time of 45 minutes following Mozilla's definition of a session as a continuous period of user activity in the browser, where successive events are separated by no more than 30 minutes[1] Table C.11 displays effects of PETs on various attributes. None of the PETs produced any variations across sessions.

[1] https://blog.mozilla.org/metrics/2010/12/22/browsing-sessions/

Table C.10: Comparison of purported and observed PET behaviors on attributes from our session experiment in Appendix C.2. □ indicates documented spoofing, ✓ indicates observed modification.

	Chrome							Firefox							
Attribute	BR	CD$_{\mathrm{C}}$	CFB	GL	HMF	PE	TR	BI	BL	CD$_{\mathrm{F}}$	CA	NE	SF	TBB	TO
DNT enabled	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
IE addBehavior	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
adBlock installed	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
buildID	·	·	·	·	·	·	·	☑	☑	×	×	×	×	☑	✓
canvas fingerprint	☑	☑	☑	☑	☑	☒	☒	×	×	☑	☑	×	×	☑	×
cookies enabled	·	·	·	·	·	□	·	·	·	·	·	·	·	·	·
cpu class	·	·	·	·	·	·	·	☑	☑	·	·	·	✓	☑	·
h.Accept	·	·	·	·	·	□	·	·	·	·	·	·	·	□	·
h.Accept-Encoding	·	·	·	·	·	·	·	·	·	·	·	·	·	□	·
h.Accept-Language	✓	×	×	×	×	×	×	×	☑	×	×	×	×	☑	×
h.Connection	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
h.Dnt	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
h.Pragma	·	·	·	·	✓	·	·	·	·	·	·	·	·	·	·
h.Up.-Ins.-Req.	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
h.User-Agent	☑	×	×	×	☑	☒	☒	☑	☑	·	·	·	·	☑	☑
indexedDB	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
javascript fonts	×	×	×	×	×	×	×	×	×	×	×	×	☒	☑	×
language	✓	×	×	×	×	×	×	×	☑	×	×	×	×	☑	×
lied with browser	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
lied with language	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
lied with os	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
lied with res.	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
local storage	·	·	·	·	·	□	·	·	·	·	·	·	·	·	·
math.acosh(1e300)	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
math.asinh(1)	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
math.atanh(05)	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
math.cbrt(100)	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
math.cosh(10)	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
math.expm1(1)	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
math.log1p(10)	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
math.sinh(1)	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
math.tanh(1)	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
openDB	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
platform	·	·	·	·	·	·	·	☑	☑	·	·	·	·	☑	·
plugins	☑	×	×	×	×	×	×	·	·	·	·	□	□	□	·
screen.AvailHeight	×	×	×	×	×	×	×	×	×	×	×	×	☑	☑	×
screen.AvailLeft	·	·	·	·	·	·	·	·	·	·	·	·	□	□	·
screen.AvailTop	·	·	·	·	·	·	·	·	·	·	·	·	□	□	·
screen.AvailWidth	×	×	×	×	×	×	×	×	×	×	×	×	☑	☑	×
screen.Depth	·	·	·	·	·	·	·	×	×	×	×	×	☑	☑	×
screen.Height	×	×	×	×	×	×	×	×	×	×	×	×	☑	☑	×
screen.Left	·	·	·	·	·	·	·	·	·	·	·	·	□	□	·
screen.Pixel Ratio	·	·	·	·	·	·	·	·	·	·	·	·	□	□	·
screen.Top	·	·	·	·	·	·	·	·	·	·	·	·	□	□	·
screen.Width	×	×	×	×	×	×	×	×	×	×	×	×	☑	☑	×
session storage	·	·	·	·	·	□	·	·	·	·	·	·	·	·	·
timezone	×	×	×	×	×	×	×	×	×	×	×	×	×	☑	×
touch.event	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
touch.max points	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
touch.start	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
webGL.Data Hash	☑	✓	✓	✓	✓	×	×	×	×	✓	✓	×	×	☑	×
webGL.Renderer	☑	✓	✓	✓	✓	·	·	×	×	✓	×	×	×	☑	×
webGL.Vendor	☑	✓	✓	✓	✓	·	·	×	×	✓	×	×	×	☑	×

Table C.11: Observed effects of PETs on attribute values from our session experiment in Appendix C.2.

Attribute	Chrome							Firefox							
	BR	CD_C	CFB	GL	HMF	PE	TR	BI	BL	CD_F	CA	NE	SF	TBB	TO
`buildID`	·	·	·	·	·	·	·	FV	FV	×	×	×	×	FV	FV
`canvas fingerprint`	FV	SR	FV	SR	SR	×	×	×	×	TV	SR	×	×	FV	×
`cpu class`	·	·	·	·	·	·	·	CV	CV	·	·	·	CV	CV	·
`h.Accept-Language`	CV	×	×	×	×	×	×	×	FV	×	×	×	×	FV	×
`h.Pragma`	·	·	·	·	CV	·	·	·	·	·	·	·	·	·	·
`h.User-Agent`	FV	×	×	×	SR	×	×	CV	CV	·	·	·	·	CV	CV
`javascript fonts`	×	×	×	×	×	×	×	×	×	×	×	×	×	PV	×
`language`	PV	×	×	×	×	×	×	×	FV	×	×	×	×	FV	×
`platform`	·	·	·	·	·	·	·	CV	CV	·	·	·	·	CV	·
`plugins`	FV	×	×	×	×	×	×	·	·	·	·	·	·	·	·
`screen.AvailHeight`	×	×	×	×	×	×	×	×	×	×	×	×	FV	PV	×
`screen.AvailWidth`	×	×	×	×	×	×	×	×	×	×	×	×	FV	PV	×
`screen.Depth`	·	·	·	·	·	·	·	×	×	×	×	×	FV	FV	×
`screen.Height`	×	×	×	×	×	×	×	×	×	×	×	×	FV	PV	×
`screen.Width`	×	×	×	×	×	×	×	×	×	×	×	×	FV	PV	×
`timezone`	×	×	×	×	×	×	×	×	×	×	×	×	×	FV	×
`webGL.Data Hash`	FV	SR	FV	FV	FV	×	×	×	×	FV	SR	×	×	FV	×
`webGL.Renderer`	CV	SR	CV	CV	CV	·	·	×	×	FV	×	×	×	FV	×
`webGL.Vendor`	CV	SR	CV	CV	CV	·	·	×	×	FV	×	×	×	FV	×

'SR': exposed values vary across reloads, 'SD': exposed values vary across domains, 'FV': differing original values fully standardize to one exposed value, 'PV': differing original values partially standardize to a smaller set of exposed values, 'CV': original values change without reduction in set of revealed values, 'TV': original values change with an increases in set of revealed values. For clarity of presentation, we only include attributes which were modified by at least one PET, and PETs which modified at least one attribute.

Bibliography

[1] H. Abdi. Bonferroni and Šidák corrections for multiple comparisons. In N. J. Salkind, editor, *Encyclopedia of Measurement and Statistics*. Sage, 2007. 54

[2] Absolute Double. HideMyFootprint: Protect your privacy. https://hmfp.absolutedouble.co.uk. Accessed Dec. 25, 2017. 64

[3] Absolute Double. Trace: Browse online without leaving a trace. https://absolutedouble.co.uk/trace/. Accessed Jan. 12, 2017. 64

[4] 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*, pages 674–689. ACM, 2014. 58

[5] 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 & communications security*, pages 1129–1140. ACM, 2013. 58

[6] Alexa. Is popularity in the top sites by category directory based on traffic rank? https://support.alexa.com/hc/en-us/articles/200461970. Accessed Nov. 21, 2014. 46

[7] Andrew. Scriptsafe: andryou. https://www.andryou.com/scriptsafe/. Accessed Dec. 25, 2017. 63

[8] Julia Angwin and Terry Parris Jr. Facebook lets advertisers exclude users by race. https://www.propublica.org/article/facebook-lets-advertisers-exclude-users-by-race, October 2016. 82, 101

[9] Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. Machine Bias: There's software used across the country to predict future criminals. and it's biased against blacks. https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing, May 2016. 1, 82

[10] appodrome.net. CanvasFingerprintBlock: Chrome Web Store. https://chrome.google.com/webstore/detail/canvasfingerprintblock/ipmjngkmngdcdpmgmiebdmfbkcecdndc?hl=en. Accessed Dec. 25, 2017. 64

[11] R. Balebako, P. Leon, R. Shay, B. Ur, Y. Wang, and L. Cranor. Measuring the effectiveness of privacy tools for limiting behavioral advertising. In *Web 2.0 Security and Privacy Wksp.*, 2012. 6, 10, 21, 26, 38, 48

[12] Paul Barford, Igor Canadi, Darja Krushevskaja, Qiang Ma, and S. Muthukrishnan. Adscape: Harvesting and analyzing online display ads. In *23rd Intl. Conf. on World Wide Web*, pages 597–608. International World Wide Web Conferences Steering Committee, 2014. 6, 10, 26, 27, 37, 44, 45

[13] Solon Barocas and Andrew Selbst. Big data's disparate impact. *California Law Review*, 104:671, 2016. 82, 84

[14] Gilles Barthe, Pedro R. D'Argenio, and Tamara Rezk. Secure information flow by self-composition. In *CSFW '04: 17th IEEE Computer Security Foundations Wksp.*, page 100, 2004. 12

[15] Y. Benjamini and Y. Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *J. the Royal Statistical Society Series B*, 57:289âĂŞ300, 1995. 30

[16] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006. 48

[17] Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In *Advances in Neural Information Processing Systems*, pages 4349–4357, 2016. 82, 106

[18] Brave Browser. Fingerprint protection mode. https://github.com/brave/browser-laptop/wiki/Fingerprinting-Protection-Mode. Accessed Dec 19, 2017. 64

[19] Jenna Burrell. How the machine âĂŸthinksâĂŹ: Understanding opacity in machine learning algorithms. *Big Data & Society*, 3(1):2053951715622512, 2016. 106

[20] Aylin Caliskan, Joanna J Bryson, and Arvind Narayanan. Semantics derived automatically from language corpora contain human-like biases. *Science*, 356(6334):183–186, 2017. 82

[21] Yinzhi Cao, Song Li, and Erik Wijmans. (cross-)browser fingerprinting via os and hardware level features. In *24th Annual Network and Distributed System Security SymposiumNDSS*, 2017. 58, 62

[22] Roberto Capizzi, Antonio Longo, V. N. Venkatakrishnan, and A. Prasad Sistla. Preventing information leaks through shadow executions. In *2008 Annual Computer Security Applications Conf.*, pages 322–331. IEEE Computer Society, 2008. 12

[23] Benny Chor, Amos Fiat, and Moni Naor. Tracing traitors. In *14th Annual International Cryptology Conf. on Advances in Cryptology*, pages 257–270. Springer-Verlag, 1994. 7

[24] Chrome: Developer. NPAPI Plugins. `https://developer.chrome.com/apps/npapi`. Accessed Jan. 12, 2018. 70

[25] Cliqz International GmbH. Ghostery makes the web cleaner, faster and safer! `https://www.ghostery.com`. Accessed Dec. 27, 2017. 64

[26] C. J. Clopper and E. S. Pearson. The use of confidence or fiducial limits illustrated in the case of the binomial. *Biometrika*, 26(4):404–413, 1934. 136

[27] US Federal Trade Commission. Privacy online: A report to congress. Technical report, US Federal Trade Commission, 1998. 2

[28] US Federal Trade Commission et al. Protecting consumer privacy in an era of rapid change: A proposed framework for businesses and policymakers. Technical report, US Federal Trade Commission, 2010. 3

[29] D. R. Cox. *Planning of Experiments*. Wiley, 1958. 20

[30] Anupam Datta, Shayak Sen, and Yair Zick. Algorithmic transparency via quantitative input influence: Theory and experiments with learning systems. In *Security and Privacy (SP), 2016 IEEE Symposium on*, pages 598–617. IEEE, 2016. 106

[31] Dominique Devriese and Frank Piessens. Noninterference through secure multi-execution. In *2010 IEEE Symp. on Security and Privacy*, pages 109–124, 2010. 12

[32] Disconnect. Disconnect. `https://disconnect.me`. Accessed Jan. 12, 2017. 64

[33] Charles Duhigg. How companies learn your secrets. *The New York Times Magazine*, February 2012. Accessed Jan 6, 2018. 1, 3

[34] Peter Eckersley. How unique is your web browser? In *Privacy Enhancing Technologies*, volume 6205, pages 1–18. Springer, 2010. 3, 58, 62, 66, 67

[35] Electronic Frontier Foundation. Panopticlick. https://panopticlick.eff.org. Accessed Dec 12, 2017. 70

[36] Electronic Frontier Foundation. Privacy Badger. https://www.eff.org/privacybadger. Accessed Jan. 13, 2017. 64

[37] Steven Englehardt, Christian Eubank, Peter Zimmerman, Dillon Reisman, and Arvind Narayanan. Web Privacy Measurement: Scientific principles, engineering platform, and new results. http://randomwalker.info/publications/WebPrivacyMeasurement.pdf, 2014. Accessed Nov. 22, 2014. 6, 10, 26, 28, 38, 44

[38] Steven Englehardt and Arvind Narayanan. Online tracking: A 1-million-site measurement and analysis. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pages 1388–1401. ACM, 2016. 58, 62

[39] Executive Office of the President. Big data: Seizing opportunities, preserving values. http://www.whitehouse.gov/sites/default/files/docs/big_data_privacy_report_5.1.14_final_print.pdf, 2014. 1

[40] Executive Office of the President. Big data: Seizing opportunities, preserving values. http://www.whitehouse.gov/sites/default/files/docs/big_data_privacy_report_may_1_2014.pdf, 2014. Accessed Jan. 26, 2014. 33

[41] eyeo GmbH. Adblock Plus: Surf the web without annoying ads! https://adblockplus.org. Accessed Dec. 27, 2017. 64

[42] Amin FaizKhademi, Mohammad Zulkernine, and Komminist Weldemariam. Fpguard: Detection and prevention of browser fingerprinting. In *IFIP Annual Conference on Data and Applications Security and Privacy*, pages 293–308. Springer, 2015. 62

[43] David Fifield and Serge Egelman. Fingerprinting web users through font metrics. In *International Conference on Financial Cryptography and Data Security*, pages 107–124. Springer, 2015. 58

[44] R. A. Fisher. *The Design of Experiments*. Oliver & Boyd, 1935. 20, 41

[45] fonk. TotalSpoof add-on homepage. http://fonk.wz.cz/totalspoof. Accessed Dec. 25, 2017. 64

[46] Deepak Garg, Limin Jia, and Anupam Datta. Policy auditing over incomplete logs: theory, implementation and applications. In *18th ACM Conf. on Computer and Communications Security*, pages 151–162, 2011. 13

[47] Tarleton Gillespie. The relevance of algorithms. *Media technologies: Essays on communication, materiality, and society*, 167, 2014. 89

[48] Joseph A. Goguen and Jose Meseguer. Security policies and security models. In *IEEE Symp. on Security and Privacy*, pages 11–20, 1982. 1, 8, 10

[49] Phillip Good. *Permutation, Parametric and Bootstrap Tests of Hypotheses*. Springer, 2005. 9, 23, 25, 36, 42, 43, 44

[50] Google. About ads settings. https://support.google.com/ads/answer/2662856. Accessed Nov. 21, 2014. 32

[51] Google. Google privacy and terms. http://www.google.com/policies/technologies/ads/. Accessed Nov. 22, 2014. 52, 56

[52] Google. Privacy policy. https://www.google.com/intl/en/policies/privacy/. Accessed Nov. 21, 2014. 35

[53] Google. Adwords policies. https://support.google.com/adwordspolicy/answer/6008942, 2017. Accessed Oct. 2, 2017. 86

[54] Google. Personalized advertising. https://support.google.com/adwordspolicy/answer/143465, 2017. Accessed Oct. 2, 2017. 86

[55] James W. Gray, III. Probabilistic interference. In *IEEE Symp. on Research in Security and Privacy*, pages 170–179, 1990. 11

[56] James W. Gray, III. Toward a mathematical foundation for information flow security. In *IEEE Computer Society Symp. on Research in Security and Privacy*, pages 21–34, 1991. 11

[57] Sander Greenland. The logic and philosophy of causal inference: A statistical perspective. In *Philosophy of Statistics*, pages 813–830. Elsevier, 2011. 25

[58] Sander Greenland and James M. Robins. Identifiability, exchangeability, and epidemiological confounding. *International J. Epidemiology*, 15(3):413–419, 1986. 44

[59] Saikat Guha, Bin Cheng, and Paul Francis. Challenges in measuring online advertising systems. In *10th ACM SIGCOMM Conf. on Internet Measurement*, pages 81–87, 2010. 1, 3, 6, 10, 26, 29, 38

[60] Saikat Guha, Bin Cheng, and Paul Francis. Privad: Practical privacy in online advertising. In *USENIX conference on Networked systems design and implementation*, pages 169–182, 2011. 3

[61] Raymond Hill. uBlock Origin: An efficient blocker for Chromium and Firefox. https://github.com/gorhill/uBlock. Accessed Dec. 27, 2017. 64

[62] Raymond Hill. uBlock and others: Blocking ads, trackers, malwares. https://github.com/gorhill/uBlock/wiki/uBlock-and-others%3A-Blocking-ads%2C-trackers%2C-malwares, May 2015. Accessed July 5, 2017. 62

[63] Sture Holm. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2):65–70, 1979. 49, 137

[64] Rick H. Hoyle, editor. *Handbook of Structural Equation Modeling*. The Guilford Press, 2012. 15

[65] David Hume. *A Treatise of Human Nature: Being an Attempt to Introduce the Experimental Method of Reasoning into Moral Subjects*. John Noon, 1738. Book III, part I, section I. 55

[66] Muhammad Ikram, Hassan Jameel Asghar, Mohamed Ali Kaafar, Anirban Mahanti, and Balachandar Krishnamurthy. Towards seamless tracking-free web: Improved detection of trackers via one-class learning. *Proceedings on Privacy Enhancing Technologies*, 2017(1):79–99, 2017. 62, 63

[67] InformAction. NoScript: JavaScript/Java/Flash blocker for a safer Firefox experience! https://noscript.net. Accessed Dec. 27, 2017. 63

[68] David D. Jensen. *Induction with Randomization Testing: Decision-oriented Analysis of Large Data Sets*. PhD thesis, Sever Institute of Washington University, 1992. 45

[69] Eric Jones, Travis Oliphant, Pearu Peterson, et al. SciPy: Open source scientific tools for Python, 2001–. 35

[70] Jaeyeon Jung, Anmol Sheth, Ben Greenstein, David Wetherall, Gabriel Maganis, and Tadayoshi Kohno. Privacy Oracle: A system for finding application leaks with black box differential testing. In *ACM Conf. on Computer and Communications Security*, pages 279–288. ACM, 2008. 12

[71] Matthew Kay, Cynthia Matuszek, and Sean A Munson. Unequal representation and gender stereotypes in image search results for occupations. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pages 3819–3828. ACM, 2015. 82

[72] Pauline T Kim. Data-driven discrimination at work. *Wm. & Mary L. Rev.*, 58:857, 2016. 84

[73] kkapsner. CanvasBlocker: A Firefox plugin to block the canvas-API. https://github.com/kkapsner/CanvasBlocker/. Accessed Dec. 25, 2017. 64

[74] Donald E. Knuth. Two notes on notation. *Am. Math. Monthly*, 99(5):403–422, 1992. 112

[75] Georgios Kontaxis and Monica Chew. Tracking protection in Firefox for privacy and performance. *arXiv preprint arXiv:1506.04104*, 2015. 62

[76] Balachander Krishnamurthy and Craig E Wills. Generating a privacy footprint on the internet. In *Proceedings of the 6th ACM SIGCOMM conference on Internet measurement*, pages 65–70. ACM, 2006. 62

[77] Joshua A Kroll, Solon Barocas, Edward W Felten, Joel R Reidenberg, David G Robinson, and Harlan Yu. Accountable algorithms. *U. Pa. L. Rev.*, 165:633, 2016. 84

[78] Juhi Kulshrestha, Motahhare Eslami, Johnnatan Messias, Muhammad Bilal Zafar, Saptarshi Ghosh, Krishna P Gummadi, and Karrie Karahalios. Quantifying search bias: Investigating sources of bias for political searches in social media. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing*, pages 417–432. ACM, 2017. 106

[79] Anja Lambrecht and Catherine E Tucker. Algorithmic bias? An empirical study into apparent gender-based discrimination in the display of STEM career ads. *Social Science Research Network (SSRN)*, August 2017. 82, 83, 92

[80] Pierre Laperdrix. Fingerprint central. `https://fpcentral.irisa.fr/`. Accessed Oct 31, 2017. 70

[81] Pierre Laperdrix, Benoit Baudry, and Vikas Mishra. Fprandom: Randomizing core browser objects to break advanced device fingerprinting techniques. In *9th International Symposium on Engineering Secure Software and Systems (ESSoS 2017)*, 2017. 3, 58, 62, 63, 108

[82] Pierre Laperdrix, Walter Rudametkin, and Benoit Baudry. Mitigating browser fingerprint tracking: multi-level reconfiguration and diversification. In *Proceedings of the 10th International Symposium on Software Engineering for Adaptive and Self-Managing Systems*, pages 98–108. IEEE Press, 2015. 62

[83] Pierre Laperdrix, Walter Rudametkin, and Benoit Baudry. Beauty and the beast: Diverting modern web browsers to build unique browser fingerprints. In *Security and Privacy (SP), 2016 IEEE Symposium on*, pages 878–894. IEEE, 2016. 58, 62

[84] Gurvan Le Guernic. Information flow testing: The third path towards confidentiality guarantee. In *Annual Asian Computing Science Conf.*, 2007. 12

[85] Mathias Lécuyer, Guillaume Ducoffe, Francis Lan, Andrei Papancea, Theofilos Petsios, Riley Spahn, Augustin Chaintreau, and Roxana Geambasu. XRay: Increasing the web's transparency with differential correlation. In *USENIX Security Symp.*, 2014. 3, 6, 10, 26, 28, 37

[86] E. L. Lehmann and Joseph P. Romano. *Testing Statistical Hypotheses*. Springer, third edition, 2005. 27

[87] Pedro Leon, Blase Ur, Richard Shay, Yang Wang, Rebecca Balebako, and Lorrie Cranor. Why johnny can't opt out: a usability evaluation of tools to limit online behavioral advertising. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 589–598. ACM, 2012. 75

[88] Bin Liu, Anmol Sheth, Udi Weinsberg, Jaideep Chandrashekar, and Ramesh Govindan. AdReveal: Improving transparency into online targeted advertising. In *Twelfth ACM Wksp. on Hot Topics in Networks*, pages 12:1–12:7. ACM, 2013. 6, 10, 26, 27, 37

[89] John Ludbrook. Analysis of 2 × 2 tables of frequencies: Matching test to experimental design. *International J. Epidemiology*, 37:1430–1435, 2008. 27

[90] Qiang Ma. personal communication, 2014. 27

[91] Jonathan R Mayer and John C Mitchell. Third-party web tracking: Policy and technology. In *Security and Privacy (SP), 2012 IEEE Symposium on*, pages 413–427. IEEE, 2012. 3, 62

[92] Stephen McCamant and Michael D. Ernst. A simulation-based proof technique for dynamic information flow. In *2007 Wksp. on Programming Languages and Analysis for Security*, pages 41–46. ACM, 2007. 12

[93] John McLean. Security models and information flow. In *IEEE Computer Society Symp. on Research in Security and Privacy*, pages 180–187, 1990. 7, 9

[94] John McLean. A general theory of composition for trace sets closed under selective interleaving functions. In *1994 IEEE Symp. on Security and Privacy*, page 79, 1994. 13, 107

[95] Media Buying. Microsoft to surpass Yahoo in global digital ad market share this year: Google, Facebook continue on as the market's leaders. *eMarketer*, July 25 2014. http://www.emarketer.com/Article/Microsoft-Surpass-Yahoo-Global-Digital-Ad-Market-Share-This-Year/1011012. 84

[96] meh. Blender: Blend in the crowd by faking to be the most common Firefox browser version, operating system and other stuff. https://github.com/meh/blender. Accessed Dec. 25, 2017. 64

[97] Georg Merzdovnik, Markus Huber, Damjan Buhov, Nick Nikiforakis, Sebastian Neuner, Martin Schmiedecker, and Edgar Weippl. Block me if you can: A large-scale study of tracker-blocking tools. In *Proceedings of the 2nd IEEE European Symposium on Security and Privacy (IEEE EuroS&P)*, 2017. 62

[98] Microsoft. Microsoft Privacy Dashboard. `https://choice.microsoft.com/`. Accessed Nov. 21, 2014. 32

[99] Tom M. Mitchell. *Machine Learning*. McGraw-Hill, 1997. 44

[100] Mark Monmonier and H. J. de Blij. *How to Lie with Maps*. University of Chicago Press, 2 edition, 1996. 7

[101] Miranda Mowbray. Causal security. In *Computer Security Foundations Wksp.*, pages 54–62, 1992. 7, 9

[102] Keaton Mowery and Hovav Shacham. Pixel perfect: Fingerprinting canvas in HTML5. *Proceedings of W2SP*, pages 1–12, 2012. 58

[103] Mozilla Support. Tracking protection. `https://support.mozilla.org/en-US/kb/tracking-protection`. Accessed Dec. 27, 2017. 64

[104] Mozilla Support. Why do Java, Silverlight, Adobe Acrobat and other plugins no longer work? `https://support.mozilla.org/en-US/kb/npapi-plugins`. Accessed Jan. 12, 2018. 70

[105] Multiloginapp. How canvas fingerprint blockers make you easily trackable. `https://multiloginapp.com/how-canvas-fingerprint-blockers-make-you-easily-trackable/`. Accessed Dec 19, 2017. 64

[106] Net-Comet. Glove: Chrome Web Store. `https://chrome.google.com/webstore/detail/glove/abdgoalibdacpnmknnpkgnfllphboefb?hl=en`. Accessed Dec. 25, 2017. 64

[107] James Newsome and Dawn Xiaodong Song. Dynamic taint analysis for automatic detection, analysis, and signature generation of exploits on commodity software. In *Network and Distributed System Security Symp.* The Internet Society, 2005. 12

[108] Nick Nikiforakis, Wouter Joosen, and Benjamin Livshits. Privaricator: Deceiving fingerprinters with little white lies. In *Proceedings of the 24th International Conference on World Wide Web*, pages 820–830. International World Wide Web Conferences Steering Committee, 2015. 3, 58, 62

[109] NiklasG. Stop Fingerprinting: Add-ons for Firefox. `https://addons.mozilla.org/en-US/firefox/addon/stop-fingerprinting/`. Accessed Dec. 25, 2017. 64

[110] Office of the Privacy Commissioner of Canada. Google ads sparked by web surfing on health sites violate privacy rights, investigation finds. *OPC News Release*, January 2014. Accessed Feb 19, 2018. 1

[111] L. Olejnik, T. Minh-Dung, and C. Castelluccia. Selling off privacy at auction. In *Network and Distributed System Security Symposium (NDSS)*. The Internet Society, 2013. 56

[112] Liam Paninski. Estimation of entropy and mutual information. *Neural computation*, 15(6):1191–1253, 2003. 65

[113] Panagiotis Papadimitriou and Hector Garcia-Molina. Data leakage detection. *IEEE Trans. on Knowl. and Data Eng.*, 23(1):51–63, 2011. 7

[114] Judea Pearl. *Causality*. Cambridge University Press, second edition, 2009. 8, 10, 15, 16, 107, 112, 114, 115, 117, 118, 119

[115] 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. *J. Machine Learning Research*, 12:2825–2830, 2011. 35, 48

[116] Mike Perry, Erinn Clark, Steven Murdoch, and Georg Koppen. The design and implementation of the tor browser. `https://www.torproject.org/projects/torbrowser/design/#privacy`. Accessed Jul 21, 2017. 58, 62, 64, 79

[117] Pew Research Center's Social and Demographic Trends Project. On pay gap, millennial women near parity — for now: Despite gains, many see roadblocks ahead, 2013. 55

[118] Roberta Rampton. White House looks at how 'Big Data' can discriminate. *Reuters*, April 2014. Accessed Feb 19, 2018. 1

[119] Reşat. Blend In: Add-ons for Firefox. `https://addons.mozilla.org/en-US/firefox/addon/blend-in/`. Accessed Dec. 25, 2017. 64

[120] Franziska Roesner, Tadayoshi Kohno, and David Wetherall. Detecting and defending against third-party tracking on the web. In *Proceedings of the 9th USENIX Conference on Networked Systems Design and Implementation*, NSDI'12, pages 12–12, Berkeley, CA, USA, 2012. USENIX Association. 3, 62

[121] Paul R. Rosenbaum. Interference between units in randomized experiments. *J. the American Statistical Association*, 102(477):191–200, 2007. 23

[122] Joseph R. Ruthruff, Sebastian Elbaum, and Gregg Rothermel. Experimental program analysis: A new program analysis paradigm. In *2006 Intl. Symp. on Software Testing and Analysis*, pages 49–60. ACM, 2006. 9

[123] Andrei Sabelfeld and Andrew C. Myers. Language-based information-flow security. *IEEE Journal on Selected Areas in Communications*, 21(1):5–19, 2003. 7, 12

[124] Sagar Shivaji Salunke. *Selenium Webdriver in Python: Learn with Examples*. CreateSpace Independent Publishing Platform, USA, 1st edition, 2014. 70

[125] Samy Sadi. No Enumerable Extensions: Firefox addon that lets you hide installed extensions and avoid being fingerprinted based on them. https://github.com/samysadi/no-enumerable-extensions. Accessed Jan. 13, 2017. 64

[126] Iskander Sanchez-Rola, Igor Santos, and Davide Balzarotti. Extension breakdown: Security analysis of browsers extension resources control policies. In *26th USENIX Security Symposium (USENIX Security 17)*, pages 679–694, Vancouver, BC, 2017. USENIX Association. 68

[127] Fred B. Schneider. Enforceable security policies. *ACM Trans. Inf. Syst. Secur.*, 3(1):30–50, 2000. 13, 107

[128] P. Sewell and J. Vitek. Secure composition of untrusted code: wrappers and causality types. In *Computer Security Foundations Workshop, 2000. CSFW-13. Proceedings. 13th IEEE*, pages 269–284, 2000. 10

[129] Mallory Simon. HP looking into claim webcams can't see black people. *CNN*, December 2009. Accessed Feb 19, 2018. 1

[130] Lance Spitzner. Honeytokens: The other honeypot. http://www.symantec.com/connect/articles/honeytokens-other-honeypot, July 2003. Accessed Jun 2013. 7

[131] Martin Springwald. Privacy-Extension-Chrome: Provides privacy for Chrome. https://github.com/marspr/privacy-extension-chrome. Accessed Dec. 25, 2017. 64

[132] Oleksii Starov and Nick Nikiforakis. Xhound: Quantifying the fingerprintability of browser extensions. In *Security and Privacy (SP), 2017 IEEE Symposium on*, pages 941–956. IEEE, 2017. 58, 68

[133] StatCounter. StatCounter global stats. http://gs.statcounter.com/. Accessed Feb. 12, 2018. 66

[134] M.D. Swanson, M. Kobayashi, and A.H. Tewfik. Multimedia data-embedding and watermarking technologies. *IEEE*, 86(6):1064–1087, 1998. 7

[135] Latanya Sweeney. Discrimination in online ad delivery. *Commun. ACM*, 56(5):44–54, 2013. 1, 3, 6, 10, 26, 27, 38, 82, 83, 93

[136] Latanya Sweeney. personal communication, 2015. 27

[137] Deborah M. Todd. CMU researchers see disparity in targeted online job ads. *Pittsburgh Post-Gazette*, July 2015. http://www.post-gazette.com/business/career-workplace/2015/07/08/Carnegie-Mellon-researchers-see-disparity-in-targeted-online-job-ads/stories/201507080107. 83, 91, 92, 99

[138] Christof Ferreira Torres, Hugo Jonker, and Sjouke Mauw. Fp-block: usable web privacy by controlling browser fingerprinting. In *European Symposium on Research in Computer Security*, pages 3–19. Springer, 2015. 62, 65

[139] Vincent Toubiana, Arvind Narayanan, Dan Boneh, Helen Nissenbaum, and Solon Barocas. Adnostic: Privacy preserving targeted advertising. In *17th Annual Network and Distributed System Security SymposiumNDSS*, 2010. 3

[140] Catherine A Tremble. Wild westworld: The application of Section 230 of the Communications Decency Act to social networks' use of machine-learning algorithms. *Social Science Research Network (SSRN)*, 2017. 84

[141] Blase Ur, Pedro Giovanni Leon, Lorrie Faith Cranor, Richard Shay, and Yang Wang. Smart, useful, scary, creepy: Perceptions of online behavioral advertising. In *Eighth Symp. on Usable Privacy and Security*, pages 4:1–4:15. ACM, 2012. 6, 87

[142] Neil Vachharajani, Matthew J. Bridges, Jonathan Chang, Ram Rangan, Guilherme Ottoni, Jason A. Blome, George A. Reis, Manish Vachharajani, and David I. August. RIFLE: An architectural framework for user-centric information-flow security. In *37th Annual IEEE/ACM Intl. Symp. on Microarchitecture*, pages 243–254, 2004. 12

[143] V. N. Venkatakrishnan, Wei Xu, Daniel C. DuVarney, and R. Sekar. Provably correct runtime enforcement of non-interference properties. In *8th Intl. Conf. on Information and Communications Security*, pages 332–351. Springer-Verlag, 2006. 12

[144] Dennis Volpano, Cynthia Irvine, and Geoffrey Smith. A sound type system for secure flow analysis. *J. Comput. Secur.*, 4(2-3):167–187, 1996. 12

[145] Dennis M. Volpano. Safety versus secrecy. In *6th Intl. Symp. on Static Analysis*, pages 303–311. Springer-Verlag, 1999. 13, 107

[146] Neal R. Wagner. Fingerprinting. In *1983 IEEE Symp. on Security and Privacy*, page 18, 1983. 7

[147] Jon Watson. Virtualbox: Bits and bytes masquerading as machines. *Linux J.*, 2008(166), February 2008. 69

[148] Craig E. Wills and Can Tatar. Understanding what they do with what they know. In *2012 ACM Wksp. on Privacy in the Electronic Society*, pages 13–18, 2012. 1, 3, 6, 10, 26, 29, 37

[149] Peter Wright. *Spycatcher: The Candid Autobiography of a Senior Intelligence Officer*. Viking Adult, 1987. 7

[150] Yahoo! Ad Interest Manager. `https://aim.yahoo.com/aim/us/en/optout/index.htm`. Accessed Nov. 21, 2014. 32

[151] Ting-Fang Yen, Yinglian Xie, Fang Yu, Roger Peng Yu, and Martin Abadi. Host fingerprinting and tracking on the web: Privacy and security implications. In *NDSS*, 2012. 62, 66

[152] Aydan R. Yumerefendi, Benjamin Mickle, and Landon P. Cox. Tightlip: keeping applications from spilling the beans. In *4th USENIX Conf. on Networked Systems Design and Implementation*, pages 12–12, 2007. 12

[153] Tal Z. Zarsky. Understanding discrimination in the scored society. *Washington Law Review*, 89:1375–1412, 2014. 55

[154] Valentina Zarya. The Percentage of Female CEOs in the Fortune 500 Drops to 4%. *Fortune*, June 2016. `http://fortune.com/2016/06/06/women-ceos-fortune-500-2016/`. 92

[155] Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. Learning fair representations. In Sanjoy Dasgupta and David Mcallester, editors, *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, volume 28, pages 325–333. JMLR Workshop and Conference Proceedings, May 2013. 33

[156] Richard S. Zemel, Yu Wu, Kevin Swersky, Toniann Pitassi, and Cynthia Dwork. Learning fair representations. In *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *JMLR: W&CP*, pages 325–333. JMLR.org, 2013. 55

[157] Maggie Zhang. Google Photos tags two african-americans as gorillas through facial recognition software. *Forbes*, July 2015. Accessed Feb 19, 2018. 1

[158] Sebastian Zimmeck, Jie S Li, Hyungtae Kim, Steven M Bellovin, and Tony Jebara. A privacy analysis of cross-device tracking. In *26th USENIX Security Symposium USENIX Security 17)*, pages 1391–1408. USENIX Association, 2017. 62