Carnegie Mellon University

Department of Electrical and Computer Engineering

“Accountability for Privacy and Fairness Violations in Data-Driven Systems with Limited Access”

Ph.D. Thesis Prospectus

Amit Datta
Electrical and Computer Engineering
Carnegie Mellon University

Adviser: Prof. Anupam Datta

Contents

Abstract

Automated systems are increasingly being employed to make critical decisions about our lives. Such systems utilize the power of big data to make ever so accurate predictions. The widespread use of such systems has led to concerns over societal values like user privacy and fairness. Although these systems have policies promising to protect societal values, the blackbox nature of these systems makes it difficult to analyze whether these policies are respected. Moreover, it is difficult to hold entities accountable upon detection of a policy violation. The goal of this doctoral thesis is to enable accountability for privacy and fairness violations for automated data-driven decision making systems. To achieve this goal, we first show it is possible to evaluate some privacy and fairness properties on fully blackbox decision-making systems. Next, with limited access to the internal structure and logs of the system, we propose to enable accountability for fairness violations in the system.

To study privacy and fairness properties in blackbox systems, we translate the properties into information flow instances and develop methods to detect information flow in blackbox systems. We formally establish a connection between information flow and causal effects, and leverage this connection to detect information flow using experiments traditionally used to detect causal effects. We develop AdFisher as a general framework to perform information flow experiments on web systems and use it to evaluate societal values like discrimination, transparency, and choice on Google's advertising system. We find evidence of discrimination, a lack of transparency and respect for choice.

While detecting privacy and fairness violations in blackbox systems is a non-trivial problem, enabling accountability for such violations is even harder. We posit that providing explanations and assigning responsibility for such violations is difficult, if not impossible, without more visibility into the system. As part of an ongoing investigation with Microsoft Research, we are developing methods to evaluate fairness properties and enable accountability for found violations in deployed big data decision-making systems. Since deployed systems run in real-time affecting real people and businesses, we have limited access, i.e. we are not authorized manipulate inputs and modules in the system. By enabling accountability, we aim to assign responsibility for violations to internal modules of or inputs to the system. We propose to develop and apply these methods to detect and account for fairness violations in the Bing advertising pipeline.

Chapter 1

Introduction

Motivation Automated systems are increasingly being employed to make critical decisions about our lives. Such systems utilize the power of big data to make ever so accurate predictions. The widespread use of such systems has led to concerns over societal values like user privacy and fairness.

Privacy concerns from automated decisions systems received widespread attention when the Target department store used shopping behavior of customers to infer pregnancy status and market baby products to them [1]. The use of sensitive information for targeted advertising have raised eyebrows among privacy conscious individuals. Wills and Tatar found that Google ads related to sensitive topics like sexual orientation, health and financial matters were served on seemingly unrelated websites after interests in these topics were seemingly inferred from browsing activities [2]. Guha et al. found that ads served on Facebook may be personalized on the basis of sexual preference indicated [3]. There is also evidence showing that automated decisions may lead to unfair discrimination. Sweeney found that searches for black sounding names are accompanied with ads suggesting arrest records more than searches for white sounding names [4].

Although automated decision systems have policies promising to protect societal values, the blackbox nature of these systems makes it difficult to analyze whether these policies are respected. Moreover, it is difficult to hold entities accountable upon detection of a policy violation. The goal of this doctoral thesis is to enable accountability for privacy and fairness violations for automated data-driven decision-making systems. To achieve this goal, we first show it is possible to evaluate some privacy and fairness properties on fully blackbox decision-making systems. Next, with limited access to the internal structure and logs of the system, we propose to enable accountability for fairness violations in the system.

Completed Work We study three properties on Google's advertising system: discrimination, transparency, and choice. We evaluate these properties by demonstrating the presence of flows of information between browsing activities, Ad Settings and the ads. Ad Settings is Google's transparency and choice tool where users can view and edit inferences made by the system. [1] To study *discrimination*, we search for a

[1] `www.google.com/settings/ads`

flow of information from a sensitive attribute to non-private outcomes. To evaluate *transparency*, we seek a flow from online activities to ads served, in the absence of a flow to Ad Settings. This would indicate that a flow from online actions to ads is not reflected on the transparency tool, thereby demonstrating a complete lack of transparency (*opacity*). To examine *choice*, we look for a flow of information from the settings page to the ads.

We use the notion of noninterference to formalize information flow and prove that noninterference is equivalent to an absence of causal effect from the input to the outputs. With this theorem, 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. These experiments form the basis of our methodology to detect information flow [5].

We develop AdFisher as a general framework to perform information flow experiments on web systems. We show that the permutation test is appropriate to determine whether there is a statistically significant causal effect from an input to the outputs. The permutation test requires a statistic measuring the difference in the measurements from the two groups. Since it is difficult to choose such a statistic a priori on constantly changing web systems, we use machine learning to automatically choose one.

We use AdFisher to study Google's complex ad ecosystem. To study *discrimination*, we set up AdFisher to detect differences in ads served to simulated male and female users exhibiting job-seeking behavior. We find a flow from gender to the ads served by Google in the context of job-searching behavior. Since the use of gender is prohibited in hiring decisions, we choose this particular flow to evaluate discrimination. How the ads differ is concerning as well. The top two ads served to male units are for a career coaching service for high-paying executive positions, whereas the top two ads for the female group are for a generic job posting service and for an auto dealer. To study *transparency*, we set up AdFisher to observe whether there is flow from visiting substance-abuse websites to ads served. AdFisher finds that ads for a rehabilitation center is served in large numbers upon visiting those sites. However, we find no change in the information displayed on Ad Settings. Finally, we observe that Ad Settings respects *choice*. We find that opting out has some effect on ads and removing interests reduce behaviorally targeted ads [6].

Proposed Work We propose to develop methods for evaluating fairness properties and enabling accountability for found violations in general big data decision-making systems. By enabling accountability, we aim to assign responsibility for violations to internal modules of or inputs to the system. We propose to apply these methods to detect and account for fairness violations in the Bing advertising pipeline in collaboration with Microsoft Research. In general, advertising pipelines match ads to users. Ads served alongside search results, email and website content are targeted based on present and past actions of the user performing a search, reading an email, or browsing a website. The Bing advertising pipeline is a complex big data pipeline which delivers search ads. It takes a search query and selects ads to be served alongside the search results. The system processes query attributes (like query text, location, user demographics, etc.) and ad attributes (like ad text, targeting criteria, etc.) to arrive at a decision to serve an ad in response. This framework is similar to other advertising pipelines, such as search and display ads of Google [7,8] and ads on the Facebook news feed [9], which consume different user features to determine which ads to serve. We aim to analyze these decisions for violations of societal values. From our experiments with Google, discrimination seemed to be the most concerning aspect of automated decisions, hence we focus our attention on discrimination. We propose to leverage our access to internal logs and a description of the Bing advertising pipeline to enable accountability for fairness violations.

We look for discrimination as defined by the disparate impact theory. Disparate impact is an associative notion, which checks for associations between a sensitive attribute and the decision. To detect disparate impact, we measure such associations, which we call bottomline association. We propose to define a notion of associative responsibility and show that for appropriate measures of association, the bottomline association between a query attribute and the decision of the system is bounded by individual associations between the query attribute and intermediate computations produced by internal modules. This will allow us to trace associations to specific internal modules which are responsible.

Associations between a query attribute and the decision may also arise as a result of Simpson's paradox. Simpson's paradox is a phenomenon where an association appears in aggregated data, but disappears or even reverses in direction upon considering different subpopulations of the data. We propose that by measuring associations in subpopulations of query instances which were treated differently by the system, we can identify associations arising from Simpson's paradox. Given the complexity of Bing's advertising pipeline, it is difficult to identify subpopulations that are treated differently. However, it is possible to identify such subpopulations for smaller intermediate modules, given a description of how the inputs are used in the module. Thus, we propose to first trace bottomline association to smaller modules, then check for Simpson's paradox in these modules.

We aim to apply these methods to the Bing advertising pipeline to detect disparate impact and assign responsibility to internal modules of the pipeline for producing disparate impact. By assigning responsibility, we can enable accountability for discrimination.

Thesis Statement *It is possible to enable accountability in deployed data-driven systems with mechanisms that support detection and tracing of privacy and fairness violations with limited access to interfaces and observations of system behavior.*

Related Work

There have been several studies which study various privacy and fairness properties in big data systems, a majority among which take a blackbox approach. These works have evaluated online advertisements, personalized recommendations, search results and prices [3,4,10,11,12,13,14]. Our work on evaluating fully blackbox systems is different from most of these by virtue of that fact that we detect causal effects and not correlations. Moreover, we make minimal assumptions on the nature of the data. Not much prior work has been done in the area of tracing responsibility of violations to internal modules of a system. We suspect this is primarily because gaining access to the internal structure and logs of a big data decision making system is difficult. A related notion is accountability, which has been defined and used in other contexts [15,16,17]. We are interested in detecting and tracing discrimination from observational data on the Bing advertising pipeline. There have been many studies on detecting discrimination from observational data [18,19,20,21,22]. We plan to develop metrics to measure discrimination based on these prior studies.

Analyzing blackbox systems Wills and Tatar [10] evaluate transparency by analyzing both the ads shown by Google and the information on Google's Ad Settings (then called 'Ad Preferences'). They find the presence of opacity: that ads could change without a corresponding change in Ad Settings. However, their experiments were mostly manual, small scale, lacked any statistical analysis, and did not follow a rigorous experimental design.

Guha et al. compare ads seen by three browser instances to see whether online ad networks treat one different from the other two [3]. They find that user features like location, gender and website visits affect advertisements from Google and Facebook. They automate the collection and analysis of data, but do not provide statistical significance of their results.

Sweeney ran 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 [4]. Her study required considerable insight to determine that suggestions of an arrest was a key difference. We develop methods which can automatically identify such key differences by using machine learning.

Lécuyer et al. develop XRay [11] and Sunlight [12] to study privacy properties on online targeting systems like Google ads and Amazon recommendations. XRay looks for correlations between user activity and the ads shown to users, whereas Sunlight is able to find causal relationships under certain assumptions. They find that Google serves ads which appear to be targeted on sensitive and prohibited topics.

Hannak et al. study whether personalization on Google search leads to the *filter bubble* effect, where users are not able to access certain information that Google thinks is irrelevant [13]. In a separate study, Hannak et al. analyze e-commerce websites for price discrimination and steering [14].

Accountability By accountability, we adopt the notion of attributing violations to entities who are responsible for them (e.g. [15,16,17]). This is different from the approach taken by Feigenbaum et al. [23] who insist that accountability must ensure that violations get punished. Kuesters et al. [15] connect accountability with verifiability and show that by verifying (possibly false) claims of entities, it is possible to identify a misbehaving entity and hold them accountable. Backes et al. [16] define accountability as the ability to show evidence when an entity deviates from protocol. Datta et al. [17] take a causal approach to assigning responsibility. Instead of holding a deviating party accountable, they determine if a deviating action actually caused the violation.

We aim to enable accountability in deployed decision-making systems that use complex data processing methods like machine learning. In these settings, violations (like disparate impact) may appear from the normal (non-deviating) execution of the modules. Thus, the approach of identifying deviating entities cannot enable accountability in such systems. Moreover, since these production systems often affect real people and businesses, companies are not willing to change them. Hence, we focus on providing accountability by analyzing existing infrastructure and log files. Since we cannot run experiments without manipulating parts of the system, it is hard to detect causal relationships. While there are studies which can detect causal relationships from observational data by finding natural experiments (e.g. [24]), we do not assume that we can find natural experiments in the data. Thus, we enable accountability by uncovering associative relationships to assign responsibility for violations to internal modules and inputs to the system.

Discrimination detection We find discrimination to be the most concerning aspect of automated decision systems. We are interested in detecting discrimination from observational logs in Bing's advertising pipeline. There are several prior studies which detect discrimination from observational data. Pedreshi et al. [18], Hajian et al. [19] and Ruggieri et al. [20] measure associations between a sensitive outcome and an algorithm's output to measure discrimination. Tramer et al. present the Fairtest tool which finds such associations in different subpopulations, thereby providing an approach to address Simpson's paradox [22].

We do not try to define what it means to be fair (like [25]), nor develop methods to remove or prevent discrimination (like [26]). We adopt existing definitions of discrimination to detect its presence on the Bing advertising pipeline. We leave the problem of discrimination prevention on the Bing system for future work.

Chapter 2

Completed Work

We study privacy and fairness properties on a real-world blackbox decision making system - Google's advertising ecosystem. By considering users' past behavior and ads' past performance, Google personalizes ads for users. To provide some insight and control over ads, they provide Ad Settings,[1] a transparency and choice tool where users can view and edit inferences made by the system. We study three properties on Google's advertising system: discrimination, transparency, and choice. We encode these properties in terms of information flows through the system and show that by demonstrating the presence of certain flows of information between browsing activities, Ad Settings and the ads, we can evaluate these properties. To study *discrimination*, we search for a flow of information from a sensitive attribute to non-private outcomes. In particular, we search for a flow from gender to ads in the context of job-searching behavior. Since the use of gender is prohibited in hiring decisions, we choose this particular flow to evaluate discrimination. To evaluate *transparency*, we seek a flow from online activities to ads served, in the absence of a flow to Ad Settings. This would indicate that a flow from online actions to ads is not reflected on the transparency tool, thereby demonstrating a complete lack of transparency (*opacity*). To examine *choice*, we look for a flow of information from the settings page to the ads. This would indicate that the control tools on Ad Settings have some effect. We also evaluate whether the effect corresponds to the preference indicated by looking for a specific kind of flow.

In our system model, we can control a subset of the inputs to the system and observe a subset of the outputs. We show that the ability to control some inputs enables experiments that can discover the flow of information from a controllable input to an observable output. We use the notion of noninterference to formalize information flow and prove that interference is equivalent to a causal effect from the input to the output. With this theorem, 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. These experiments form the basis of our methodology to detect information flow.

[1] www.google.com/settings/ads

Experiments are performed on units. As units in our experiments, we use browser instances with no history, cache, or cookies, all running from the same virtual machine and IP address. We randomly assign units to two groups, each setting one of two values of the input of interest (e.g. gender set to male or female). Then, the units collect measurements (e.g. ads). A statistical test on the measurements determines if they are systematically different in the two groups. If the test finds a statistically significant difference, we infer that the input had a causal effect on the outputs, thereby concluding an information flow. While there are many choices for statistical tests, we choose the permutation test since it is a non-parametric test and does not require independence of units.

The permutation test requires a statistic measuring the difference in the measurements from the two groups. Given that web systems are in a constant state of flux, it is difficult to choose such a statistic a priori. To avoid determining a statistic a priori, we use machine learning to automatically choose one. We first divide the outputs from both groups into training and testing sets. We use the training set to train a classifier that learns to differentiate between the two groups. We then apply this classifier to generate a statistic and perform the permutation test on the testing set. We also use the predictive model learnt by this classifier to provide explanations characterizing how the ads served to the two groups differed.

To study *discrimination*, we search for a flow of information from gender to ads in the context of job-searching behavior. Our experiments reveal a causal effect of gender on ads served by Google on third-party news websites. How the ads differ is concerning as well. The top two ads served to male units are for a career coaching service for high-paying executive positions, whereas the top two ads for the female group are for a generic job posting service and for an auto dealer. To detect *opacity*, we seek a flow from online activities to ads served, in the absence of a flow to Ad Settings. This would indicate inferences from online actions that were used to determine ads is not shown on the transparency tool, thereby demonstrating opacity. We observe that visiting websites on substance abuse has a causal effect on the ads, where ads for a rehabilitation center are served after the visits. However, we find no change in the information displayed on Ad Settings. Finally, we observe that Ad Settings respects user *choice* and making changes on Ad Settings affects ads. We study two notions of choice. Google is said to respect *effective choice* if editing the settings has any effect on the ads. On the other hand, *ad choice* is respected when the effect is meaningful. We find that opting out has a significant effect on the ads served, thus demonstrating effective choice. We also observe that removing interests related to dating reduces the number of dating related ads, thereby establishing ad choice.

In the next section, we briefly discuss discrimination, transparency, and choice, and how they correspond to information flow. In Section 2.2, we discuss the formal connection between noninterference and causality and how we leverage that connection to detect information flow using experiments. Finally, we discuss how we implement the information flow experiment methodology in AdFisher and deploy it to study the three properties on Google's ad ecosystem.

Property Name	Requirement	Flow property	Finding
Nondiscrimination	Users differing only on protected attributes are treated similarly	No flow from sensitive attributes to ads	Violation
Transparency	Users can view all data about them used for ad selection	No flow from browsing activities to Ad Settings implies no flow to ads	Violation
Effectful choice	Changing a setting has an effect on ads	Flow from Ad Settings to ads	Compliance
Ad choice	Removing an interest decreases the number ads related to that interest	Flow from Ad Settings to ads using an appropriate statistic in the expected direction	Compliance

Table 2.1: Privacy and fairness properties tested on Google's ad ecosystem

2.1 Information Flow Properties

We discuss three privacy and fairness properties and show how they can be translated into information flow properties. As a fundamental limitation of science, we can only prove the existence of a flow; we cannot prove that one does not exist. Thus, experiments can only demonstrate violations of nondiscrimination and transparency, which require effects. On the other hand, we can only demonstrate that choice is complied with since compliance follows from the existence of an effect. We summarize these properties, their equivalent information flow properties, and what finding we can detect in Table 2.1.

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. This occurs when there is a flow of information from class membership to the ads served. Such discrimination is not always bad (e.g., many would be comfortable with men and women receiving different clothing ads). The determination of whether the found discrimination is unjust is out of scope of this work. We do not claim to have a scientific method for determining the morality of discrimination.

Determining whether class membership causes a change in ads is difficult since many factors not under our control or even observable 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.

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 clear. 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, where the ad network displays any inference that it uses to serve ads to the user. 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 a complete lack of transparency - *opacity.* We say that a transparency tool has opacity if there is a flow of information from some browsing activity to the ads, but none to the ad settings. If we find that prior browsing activities affected ads, we can argue that 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.

Choice

Ad Settings offers users the option of editing 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, both of which require a flow of information from the settings page to the ads. By testing for specific kinds of effects, we can test for each form of choice.

A very coarse form of choice is *effectful choice*, which requires that altering the settings has some effect on the ads seen by the user. This shows that altering the settings has a real effect on the network's ads. To check for effectful choice, we can choose any statistic with the permutation test. However, effectful choice does not capture whether the effect on ads is meaningful. For example, if a user adds interests for cars and starts receiving *fewer* ads for cars, effectful choice is satisfied.

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* and test for it by using appropriate test statistics with the permutation test. 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.

2.2 IFE: Information Flow Experiments

We first provide some background on noninterference and causality and present our theorem connecting the two concepts. We then leverage this connection to apply randomized controlled experiments, typically used to detect causal effects, for detecting information flow in blackbox systems.

2.2.1 Noninterfernce and Causality

Noninterference

Goguen and Meseguer introduced *noninterference* to formalize when a sensitive input to a system interacting with multiple users is protected from untrusted users of that system [27]. Intuitively, noninterference requires the system to behave identically from the perspective of untrusted users regardless of any sensitive inputs to the system. To define noninterference for a blackbox system, we model the system as consuming inputs through input channels and producing outputs via output channels. These channels are of two types - H and L. H corresponds to all channels which receive or produce sensitive information, while L channels receive or produce public information. A system q consumes a sequence $\vec{\imath}$ of input tuples, where each tuple contains inputs for the high and the low input channels. $q(\vec{\imath})$ represents the output sequence $\vec{o}$ that q produces upon receiving $\vec{\imath}$ as input where output sequences are defined as a sequence of tuples 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 $\vec{\imath}$. 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 $\vec{\imath}_2$,*

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

To handle systems with probabilistic transitions, we employ a probabilistic version of noninterference. To define it, we let $Q(\vec{\imath})$ denote a probability distribution over output sequences given the input $\vec{\imath}$. $\lfloor Q(\vec{\imath}){\downarrow}L \rfloor$ represents 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.

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

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

Intuitively, noninterference requires the system to draw low-level outputs from identical distributions when the inputs only differ in the high-level inputs.

Causality

In this section, we discuss a formal notion of causality as defined by Pearl. We then prove that noninterference corresponds to a lack of a causal effect. This result allows us to repose information flow detection as a problem of statistical inference from experimental data using causal reasoning.

Pearl [28] provides a formalization of *effect* using *structural equation models* (SEMs). A probabilistic SEM is a tuple $\langle \mathcal{V}_{\mathsf{en}}, \mathcal{V}_{\mathsf{ex}}, \mathcal{E}, \mathcal{P} \rangle$, where $\mathcal{V}_{\mathsf{en}}$ is the set of *endogenous* (or dependent) variables; $\mathcal{V}_{\mathsf{ex}}$ is the set of *exogenous* (or independent) variables; $\mathcal{E}$ comprises of *structural equations* $V := F_V(\vec{V})$ for each endogenous variable V, where $\vec{V}$ is a list of other variables other than V and F_V is a possibly randomized function; and $\mathcal{P}$ provides the probability distribution for each exogenous variable. 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 [28].

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 random variables other than X and Y.

Relationship between interference and causality

We connect the notions of interference and causality with the following theorem.

Theorem 1 *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.*

Intuitively, interference is an effect from a high-level input to a low-level output. We prove the theorem by providing a conversion from a probabilistic system to an SEM. A detailed proof is available in the full paper [5]. Notice that Theorem 1 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).

2.2.2 Experiments and Significance Testing

Having reduced the problem of information flow experiments to that of checking for causal effects, we can employ we start with the approach of Fisher for randomized controlled experiments and significance testing [29]. We discuss how we setup experiments and analyze the collected data using significance tests to infer information flow.

Randomized Controlled Experiments

Experiments are performed on units. For example, to test the effects of a drug on cancerous tumors in mice, each mouse is a unit. A randomized controlled experiment randomly assigns each *unit*, 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. For example, a Firefox browser instance should not be compared to an Internet Explorer browser instance since Google can detect the browser used. The experimenter then measures a *response* in each unit and determines whether the treatments have an effect on the measured responses.

To study properties in the Google ad ecosytem, 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 units. Since our goal is to study privacy and fairness properties of Google with respect to people using their services, interactions with Google at the granularity of people could be an appropriate experimental unit. However, since we desire automated studies, we substitute 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 Google. We can apply treatments to browsers by having them perform different actions via scripts that automate browsing behavior. 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.

With the units determined, we run such a randomized controlled experiment as follows: 1. Assign each browser instance 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.

Significance Testing

Significance testing examines a *null hypothesis*, in our case, that the inputs do not affect the outputs. Typically, significance tests require a number of assumptions on the experimental data. Chief among them are distributional assumptions, which require that the data be drawn from some standard underlying distribution (e.g. Gaussian distribution for Pearson's χ^2 test), and i.i.d. assumptions, which require the data to be independently drawn and identically distributed. For the setting of online web systems, these assumptions did not seem to hold.

Permutation tests (also known as randomization tests) are a class of non-parametric significance tests which allow for cross-unit interactions and non-i.i.d. responses [30], which were appropriate for our setting. At the core of a permutation test is a *test statistic* s. We select a test statistic that takes on a high value when the outputs to the two groups differ, i.e. the statistic is a measure of distance between the two groups. 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.

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.

2.3 AdFisher: Implementing IFE

We develop AdFisher as a general framework to perform information flow experiments on web systems and use it to study Google's complex ad ecosystem. We use the technique of blocking to collect and analyze data in a scalable manner. We use machine learning to automatically choose an appropriate statistic for the permutation test. We also extract explanations for how the ads are different from the trained machine learning model. Figure 2.1 shows an overview of AdFisher's workflow.

Blocking

In practice,it can be difficult to create a large number of nearly identical browser instances for performing randomized controlled experiments. In our case, we could only run ten instances in parallel given our hardware and network limitations. Comparing instances running at different times can result in additional noise since ads served to a browser instance change over time.

To avoid this limitation, we extended the above methodology to handle varying units using *blocking* [30]. To use blocking, we created *blocks* of nearly identical instances running in parallel. Each block's browser instances were randomly partitioned into the control and experimental groups. This randomization ensures that the minor differences between instances have no systematic impact upon the results: Running these blocks in a staged fashion, the experiment proceeds on block after block. For a blocked experiment, a modified permutation test now only compares the actual value of the test statistic to hypothetical values computed by reassignments of browser instances that respect the blocking structure. These reassignments do not permute labels across blocks of observations.

Selection of test statistic

This still 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. For such cases, we use machine learning to automatically select a test statistic.

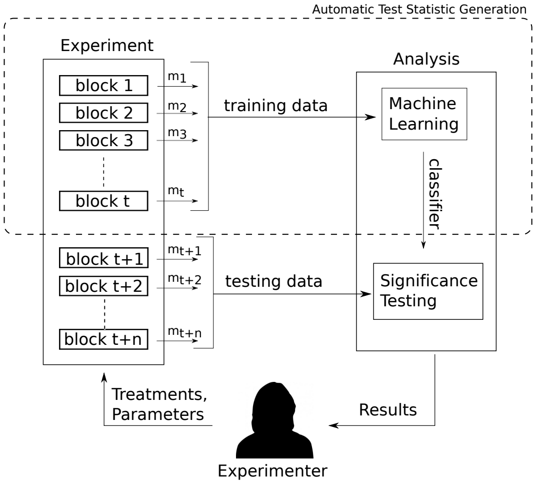

Figure 2.1: Our experimental setup with training and testing blocks. Measurements from the training blocks are used to build a classifier. The trained classifier is used to compute the test statistic on the measurements from the testing blocks for significance testing.

We partition the collected data into training and testing subsets, and use the training data to train a classifier. We perform 10-fold cross validation on the training data to select appropriate parameters. The classifier is trained to predict which treatment a browser instance received, only from the ads that get served to that instance. 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.

AdFisher evaluates the accuracy of the classifier on a testing data set that is disjoint from the training data set. 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.

Explanations

To explain how the learned classifier distinguished between the groups, we tried out several methods. We 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.

We found the most informative explanation to be the model produced by the classifier itself. We used a linear classifier (logistic regression) to select our test statistic. Recall that logistic regression weighs the various features of the instances with coefficients reflecting how predictive they are of each group. Thus, examining the ads with the most extreme coefficients identifies the ad pair most used to predict the group to which agents receiving that ad belongs.

2.4 Analyzing Google's ad ecosystem

We first provide a brief description of Google's ad ecosystem and then discuss discuss experiments to study the three privacy and fairness properties: discrimination, transparency and choice.

2.4.1 Google's ad ecosystem

The advertising ecosystem is a vast decentralized system with several actors including the online consumers, the advertisers, the publishers of web content, and ad networks. Websites publishing content integrate ads served by ad networks, such as Google, into their content by providing empty slots on their webpages into which standard-sized ads fit. Advertisers provide ads designed to fit these slots to ad networks. When an individual visits a publisher's webpage, the ad network determines which ad to place in the available slots of that webpage. To make this determination, ad networks use proprietary algorithms that among other variables take into account the expressed preferences of advertisers and publishers, and if ads are behaviorally targeted the specific user's behavior and, if communicated, preferences.

Users have some say over the advertisements Google serves them. Ad Settings is a Google tool that helps users control the ads they see on Google services and on websites that partner with Google. It allows users to independently select attributes for ad-targeting and enables them to see and modify ad-targeting inferences that Google has made about the user. Ad Settings attributes include both demographics and interests based on browsing behavior. Users can view and edit these settings at `www.google.com/settings/ads`.

Google's Ad Settings, and similar functionality provided by other advertising platforms, respond to regulators' and users' concerns about behavioral marketing on the web. They provide some transparency about how users are sliced and diced for ad-targeting, and allow users to exercise some modicum of control over the ads they see, including the ability to limit ads from Google targeted based on previous web activity and demographic details .

2.4.2 Experimental Results

We perform a total of 21 experiments to study properties of discrimination, transparency and choice on Google's ad ecosystem. Table 2.2 presents a summary of all our experiments.

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	GDN	July	53	22, 596	Inconclusive
	Gender	Jobs & Top 10	TOI	July	58	28, 738	Inconclusive
Transparency	Substance abuse	-	TOI	May	37	42, 624	Violation
	Substance abuse	-	TOI	July	41	34, 408	Violation
	Substance abuse	-	GDN	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 2.2: Summary of our experimental results. Ads are collected from the Times of India (TOI) or the Guardian (GDN), either in May or July 2014. We report how long each experiment took, how many ads were collected for it, and what result we concluded.

Discrimination

We use AdFisher to demonstrate a violation in the nondiscrimination property. If AdFisher finds a flow of information from a protected attribute to the ads served, then we have evidence that Google's ad ecosystem discriminates on that attribute. 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 show how gender information on Ad Settings flows to ads served. We detail one experiment which found evidence of discrimination.

We set up AdFisher to have the browser instances 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 instances then visited the top hundred websites related to employment and then collect ads from Times of India. AdFisher ran the experiment in 100 blocks of 10 instances each. AdFisher used the ads of 900 instances (450 from each group) for training a classifier and used the remaining 100 instances' 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, which yielded an adjusted p-value of < 0.00005.

How ads for identifying the two genders differ is also concerning from a discriminatory standpoint. The two ads 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 ads for the female group were for a generic job posting service and for an auto dealer.

Opacity

We use AdFisher to demonstrate violations of transparency. We do this by showing a flow from website visits to the ads served later on, but not to the Ad Settings. We run 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 test five interests: substance abuse, disabilities, infertility , mental disorders, and adult websites. We find a flow of information to ads in two of the experiments - substance abuse and disabilities. We examine the interests found on Ad Settings for the two cases where we found a statistically significant difference in ads. We find that they did not change at all for substance abuse and changed in an unexpected manner for disabilities.

In the experiment on substance abuse, the experimental group visited such websites while the control group idled. Then, browser instances in both groups collected information on Ad Settings and the Google ads served on the Times of India. After visiting websites about substance abuse, none of the instances had any inferences listed on their Ad Settings pages. However, the collected ads were significantly different with an adjusted p-value of < 0.00005. The top three ads for the experimental group were for an alcohol and drug rehabilitation center called the Watershed Rehab. The experimental group saw these ads a total of

3309 times (16% of the ads) while the control group never saw any of them. This is an example of opacity where there was a flow of information from the browsing activities to the ads, but not to Ad Settings.

One possible reason why Google served Watershed's ads could be *remarketing*, a marketing strategy that encourages users to return to previously visited websites. The website `thewatershed.com` was among the websites visited by the experimental group prior to ad collection. Nevertheless, this information was not reflected on Ad Settings, which constitutes as an instance of opacity.

The experiment on disabilities was identical except that the experimental group visited websites on disability. This experiment also yielded statistically significant differences in the ads, with the top ads to the experimental group being about mobility lifters and standing wheelchairs from the Ablities Expo. However, there were changes to the inferences as well, with interests about 'Reference' and 'Tourist Destinations' being inferred for a majority of the browser instances. However, we cannot conclude opacity, since there was some flow of information to Ad Settings, even though the inferences seemed unrelated to disabilities. After we communicated our opacity findings to Google, they added a notice on Ad Settings stating that 'interests listed here do not reflect ads based on a visit to a specific advertiser's page (remarketing)'.

Choice

To evaluate choice, we performed experiments to test for both effectful choice and ad choice. For effectful choice, we first tested whether opting out of tracking actually had any effect by comparing the ads shown to browser instances that opted out after visiting car-related websites to ads from those that did not opt out. We found a statistically significant difference, which suggested compliance with effectful choice. We also tested whether removing interests from the settings page had any effect. We set AdFisher to have both groups of browser instances simulate an interest in dating by visiting a website which we observed to induce the dating interest (`midsummerseve.com`). AdFisher then had the instances in the experimental group remove dating interests from Ad Settings. We found statistically significant differences in the ads, with an adjusted p-value of < 0.00003. The top ads for identifying browser instances in the experimental group are about dating with titles like 'Are You Single?' and 'Why can't I find a date?'. None of the top five for the control group that removed the interests were related to dating.

To study ad choice, we test for a more specific effect: whether is an increase or decrease in the number of relevant ads seen. In particular, after removing an interest for dating, we check for a decrease in the number of dating related ads to test for compliance, by specifying a null hypothesis stating that there is either no change or an increase in the number of dating related ads. On the other hand, to check for a violation of ad choice, we test for the null hypothesis that there was either no change or a decrease in the number of dating related ads. We found that removing dating interests resulted in a significant decrease (p-value adjusted for all six experiments: 0.0456) in the number of dating related ads, which we identified using relevant keywords.

Chapter 3

Proposed Work

We propose to develop methods for evaluating fairness properties and enabling accountability for found violations in general big data decision-making systems. By enabling accountability, we aim to assign responsibility for violations to internal modules of or inputs to the system. We propose to apply these methods to detect and account for fairness violations in the Bing advertising pipeline in collaboration with Microsoft Research. In general, an advertising pipeline matches ads to users. Ads served alongside search results, email and website content are targeted based on present and past actions of the user performing a search, reading an email, or browsing a website. The Bing advertising pipeline is a complex big data pipeline which serves search ads. It takes a search query on Bing and selects ads to be served alongside the search results. The system processes query attributes (like query text, location, user demographics, etc.) and ad attributes (like ad text, targeting criteria, etc.) to arrive at a decision to serve an ad in response. This framework is similar to other advertising pipelines, such as search and display ads of Google [7,8] and ads on the Facebook news feed [9], which consume different user features to determine which ads to serve. We aim to analyze these decisions for violations of societal values. From our experiments with Google, discrimination seemed to be the most concerning aspect of automated decisions, hence we focus our attention on discrimination. For example, if a job-related ad is served disparately based on gender, we would like to detect and explain how such disparity may have arisen in the system.

We look for discrimination as defined by the disparate impact theory. Disparate impact is an associative notion, which checks for associations between a sensitive attribute and the decision. To detect disparate impact, we measure such associations, which we call bottomline association. Next, we try to assign responsibility for the bottomline association. Since the system runs in real-time affecting real people and businesses, we are not authorized manipulate inputs to the system. As a result, we cannot run randomized controlled experiments (and hence information flow experiments) to detect flows of information inside the system. Instead, we have observational access to past logs of intermediate computations that led to final prediction, as well as a description of the sequence of computations and internal modules inside the system, either from internal documents or from the source code. We propose to define an associative notion of responsibility and show that for appropriate measures of association, bottomline association between a query attribute and the decision of is bounded by individual associations between the query attribute and intermediate computations produced by internal modules. This will allow us to trace associations to specific internal modules.

Associations between a query attribute and the decision may also arise as a result of Simpson's paradox. Simpson's paradox is a phenomenon where an association appears in aggregated data, but disappears or even reverses in direction upon considering different subpopulations of the data. We propose that by measuring associations in subpopulations of query instances which were treated differently by the system, we can identify associations arising from Simpson's paradox. Given the complexity of Bing's advertising pipeline, it is difficult to identify subpopulations that are treated differently. We hypothesize that it is possible to identify such subpopulations for smaller intermediate modules, given a description of how the inputs are used in the module. Thus, we propose to first trace bottomline association to smaller modules, then check for Simpson's paradox in these modules.

In the next Section, we provide a brief description of the Bing advertising pipeline, which serves as a running example as we develop methods for tracing associations in Section 3.2 and checking for Simpson's paradox in Section 3.3. We propose to apply these methods to the Bing advertising pipeline to detect disparate impact and assign responsibility to internal modules of the pipeline for producing disparate impact.

3.1 Bing's advertising pipeline

At the highest level, the system takes a search query and selects a set of ads to be served alongside the search results. Advertisers submit ads for delivery on the Bing platform. In addition to the text to be displayed in the ad, they specify targeting criteria (like keywords, time, location, gender, etc.) as well as a bid amount, indicating how much they are willing to pay for an impression (or click, depending on the bidding scheme). When a Bing user searches for a query, the system selects a subset of ads to serve based on various query attributes like query text, location and other demographics of the user performing the query.

There are three major modules inside the Bing pipeline that helps make the decision:

1. *Query Matching:* The query matching module parses the query that the user makes and finds a list of ads with keywords that match the query.

2. *Ad Filtration:* The ad filtration module filters ads competing for a query based based on filters like location targeting, budget availability, etc.

3. *Auction:* The ads which survive filtration participate in an auction where a number of metrics (like relevance, pClick, etc.) are computed and combined to rank the ads, the top few among which are served.

From the perspective of an individual ad, the pipeline can be modeled as a sequence of **filtering modules**. Each module takes as input a pool of units and produces a binary decision for each unit indicating whether they proceed to the next step. For example, for an individual ad, the *Query Matching* module chooses a set of query instances that match with the targeting keywords of the ad, from the entire population of query instances.

Within each of these filtering modules, there are additional submodules. These modules perform various intermediate computations to help the parent filtering module make its decision. We call these modules **computation modules**. An example of a computation module within the Auction module is the *pClick* module, which takes as input attributes of the user performing the query (like gender, the query itself) and ad attributes (like text) and computes the probability of click (pClick) for the ad impression. The pClick is one of the metrics used in the Auction module.

Technically, a filtering module is not much different from a computation module. The decision of a filtering module for a unit is binary indicating whether the unit survives for the next module, whereas there are no restrictions on the decision of a computation module. However, the computation module applies only on the set of query instances which survived the previous filtration module.

While this description is specific to the Bing advertising pipeline, we expect similar modules to exist in other advertising pipelines. For example, both Google and Facebook advertising pipelines have similar auction modules [31, 32]. However, without further visibility into these systems, we cannot be certain of the exact structure of the pipeline.

3.2 Tracing Responsibility

We show that for some specific measures of associations, we can trace responsibility of bottomline association between a decision and a sensitive attribute of a query instance. We call these *traceable* metrics of association. For these metrics, we can show that bottomline associations can be bounded (and hence explained) by associations in internal computations or inputs to the system.

3.2.1 System Model

We consider a system which takes as input a vector of inputs $\mathbf{X}$ and produces a decision Y. The system is composed of several internal modules (say $M_1, ..., M_k$) which produce intermediate computations. Each internal module M_i consumes a set of inputs $\mathcal{I}_{M_i}$ and produces a set of outputs $\mathcal{O}_{M_i}$. $\mathcal{I}_{M_i}$ may contain some or all of the inputs in $\mathbf{X}$ as well as some or all of intermediate computations that were computed by previous modules. As the system performs these computations, it maintains a log of the values of some of these computations, which we have access to. We also have a description of the sequence of computations and internal modules inside the system, either from internal documents or from the source code. This description tells which inputs were consumed by internal modules to produce computations or filtering decisions. We do not assume experimental access to the system nor do we expect natural experiments to arise automatically in the observational data.

3.2.2 Responsibility

We would like to detect an association (bottomline) between a sensitive input $S \in \mathbf{X}$ to the system and a decision Y from the system and trace responsibility for this association to internal modules. A module M may produce multiple outputs, each of which may have different degrees of responsibility for the bottomline association. Thus, we define responsibility for a module-output tuple. We first introduce a notion of causal responsibility, which is typical connotation of responsibility. We define both qualitative and quantitative versions of causal responsibility. Since we cannot detect causal relationships, we define associative responsibility as a practical compromise. We also introduce a quantitative version of associative responsibility, which we use to measure and trace responsibility.

Definition 4 (Causal Responsibility) (Informal) *If an input S causally affects a decision Y in a system, a module-output tuple (M, O) in the system is said to be responsible for it iff the output O of M lies on a causal chain from S to Y.*

A causal chain of variables is a sequence of variables which causally affect the next variable in sequence. S may causally affect Y via several causal chains. The tuple (M, O) is said to be responsible if O lies on any one of these causal chains. Due to the equivaluence of information flow and causality, this definition can also be interpreted in terms of information flow.

Definition 5 (Causal Responsibility) (Informal) *If there is a flow of information from input S to a decision Y in a system, a module-output tuple (M, O) in the system is said to be responsible for it iff M introduces or propagates the flow through O.*

If information flows from S to Y, then this flow occurs through one or more causal chains of intermediate computations. All modules which lie on a causal chain are said to be responsible for the flow of information. The first modules on these chains are said to have introduced the flow, while the others are said to have propagated the flow. We also define a quantitative notion of causal responsibility.

Definition 6 (Quantitative Causal Responsibility) (Informal) *For an input S and a decision Y, the quantitative responsibility of a module-output tuple (M, O) towards bottomline association of S and Y is the amount of causal influence that O has on the bottomline association.*

To define quantitative responsibility, we draw on the notion of Quantitative Input Influence (QII) [33]. QII is defined for a quantity of interest, which in our setting is the association between S and Y ($\mathcal{Q}_{S,Y}$). We define the Quantitative Responsibility of a module-output tuple (M, O) as the difference in the quantity of interest in the system in its original state ($\mathcal{S}$) and in the system in a counterfactual state ($\mathcal{S}_{-O}$) where O is drawn randomly from O's distribution.

However, we neither have experimental access to the system, nor do we assume natural experiments to arise in the data. Hence, we are unable to detect causal responsibility. To get around this issue, we define an associative notion of responsibility along the lines of causal responsibility (Definition 5).

Definition 7 (Associative Responsibility) (Informal) *If input S and decision Y are associated, a module-output tuple (M, O) is said to be responsible iff M introduces or propagates the association through O.*

This notion of responsibility depends on finding associations, so the lack of experimental data is not a deterrent in detecting associative responsibility. M is said to introduce association through O if one of its inputs is S, O is associated with S and that association in O propagates to Y. On the other hand, M is said to propagate association through O if one of its inputs ($\neq S$) is associated with S, O is associated with S and the association in O propagates to Y. An association in O is said to propagate to Y if there exists a chain of modules which propagate the association via intermediate computations to Y.

A quantitative notion of associative responsibility the amount of association that a module-output tuple contributes to the bottomline association. If bottomline association can be bounded by a monotonic function of the associations of intermediate module-output tuples using a certain metric of association, then that metric is said to measure the contribution of association towards bottomline association. We show in the next section that certain metrics of association can be used to measure quantitative associative responsibility.

Definition 8 (Quantitative Associative Responsibility) (Informal) *For an input S and a decision Y, the quantitative associative responsibility of a module-output tuple (M, O) towards bottomline association of S and Y is the degree of association M contributes through O towards the bottomline association.*

Task 1 *Formalize definition of quantitative associative responsibility.*

In the Bing advertising example, S can be gender and Y the decision to serve an ad. As a first step, we would like to measure the responsibility of each filtering module (e.g. *Auction*) and its corresponding binary output (decision to serve ad). As a next step, we measure the responsibility of each computation module and its output within a filtration module, for example the *pClick* module and the pClick computation within the *Auction* module.

3.2.3 Traceable metrics of association

The definition of quantitative associative responsibility (QAR) is contingent on a metric of association. QAR of a module-output tuple is the amount of association the output contributes towards bottomline association. Given an exhaustive decomposition of the system into internal modules, we posit that the bottomline association is bounded by the QAR of each internal module-output tuple. QAR of a module-output tuple is a measure of association between the output and the input attribute under study. We call metrics of association that can used to measure QAR *traceable*.

Definition 9 (Traceable metric) (Informal) *A metric of association μ is said to be traceable iff the bottomline association measured by μ between S and Y can be bounded by a monotonic function of individual associations measured by μ in module-output pairs of the system.*

Task 2 *Formalize definition and identify traceable metrics of association.*

For filtering modules, the risk ratio is a traceable metric of association. In the Bing pipeline, we can show that the bottomline risk ratio is the product of risk ratios of the three filtering modules. Thus, risk ratio is a traceable metric of association for the filtering modules and may be used to measure QAR for the filtering modules.

For the more general computation modules, the risk ratio is not a traceable measure. For example, a computation module which uses a thresholding mechanism to produce outputs can arbitrarily increase or decrease the bottomline risk ratio, while keeping the risk ratio of internal modules unchanged. We believe an information theoretic measure of association, which quantifies the amount of information leakage from S to an output is traceable for computation modules. We state the following lemmas to show that information leakage, a quantitative information flow metric, satisfies the definition of a traceable metric of association.

Lemma 1 (Sequential Composition) (Informal) *For a module M which consumes only W and produces Z, leakage from S to Z is less than or equal to the leakage from S to W.*

This theorem can be proved as a consequence of the data processing inequality. A variant was also pointed out as Theorem 5.1 in [34].

Lemma 2 (Parallel Composition) (Informal) *For a module M which consumes only W_1 and W_2 and produces Z, leakage from S to Z is less than or equal to the sum of leakages from S to W_1 and W_2, and an additional factor.*

This theorem should be provable from Corollary 5.1 in [35].

Task 3 *Consolidate theorems and proofs for an appropriate information theoretic traceable metric of association.*

Once we have identified an appropriate metric, we have to design algorithms which can apply the metric to available data. It is easy to measure the risk ratio from a 2 × 2 contingency table. However, information theoretic metrics (like mutual information) are not easy to measure, since they require knowledge of prior and posterior distributions. On the Bing advertising pipeline, we have to design and apply techniques to estimate such metrics.

Task 4 *Design algorithms to measure or estimate traceable metrics of association from observational data.*

3.3 Avoiding Simpson's Paradox

Disparity in aggregated data may arise from spurious correlations, also known as Simpson's paradox. This was exemplified in the analysis of Berkeley's admissions data by Bickel et al. [36]. They found gender based disparity in Berkeley's university wide admissions numbers. However, upon considering admission numbers on a per department basis, the disparity vanished. Since we are measuring associations on aggregated data, our results may also be susceptible to Simpson's paradox.

Once we have isolated internal modules which have traceable associations in their outcomes, we try to rule out Simpson's paradox as the source of disparity in these modules. Simpson's paradox arises if subpopulations of units are treated differently. These subpopulations are based on inputs that the module consumes. Based on different values of the inputs, the system may treat subpopulations of query instances differently. In order to rule out Simpson's paradox as the reason behind bottomline disparity, each of these subpopulations must be tested for associations.

It may be possible to craftily divide a population up into subpopulations so as to remove associations in the groups. This does not suggest that an entity can get away with disparate impact by suggesting such subpopulations. We are proposing that the analyst should consider legitimate subpopulations that were subject to different treatments based on non-sensitive user characteristics. By performing this additional step of analysis, the analyst does not falsely accuse the entity of disparate impact that arises as a result of Simpson's paradox. In the Berkeley admissions example, dividing the population on the basis of departments is an appropriate one because each subpopulation of applicants in a department is subject to different admissions criteria. We consider two levels of access to a module under analysis and show how one can identify appropriate subpopulations.

If we do not have access to the source code or if the source code is uninterpretable, then it is difficult to Identify subpopulations which are treated differently by the module. In such cases, one option would be to consider all inputs other than S and create subpopulations which have unique assignments for every input. For example, the *pClick* module is a neural network, which is hard to interpret. We know that *pClick* consumes query attributes like query text, and user demographics. Given our lack of understanding of how *pClick* uses the query text, we can consider every variant of the query text to project a different subpopulation. Moreover, if the predictive models of *pClick* are updated once in every t time units, then time is also a dimension along which subpopulations are treated differently. This can lead to a very high number of subpopulations and may render each subpopulation comprising of very few query instances. If a subpopulation comprises of a single query instance, or instances which all have the same assignment of S, then we cannot carry out any meaningful analysis. Thus, in the fully blackbox model, it may not always be possible to check for Simpson's paradox.

However, when we have a better understanding of how different inputs are used, then we can create more appropriate subpopulations. For example, the *Query Matching* module produces an intermediate set of related queries for each query instance. If this set of related queries contains at least one matching targeting keyword in the ad (assuming exact matching scheme), then the ad is matched to that query instance. Given this knowledge, we can create a class of queries which all produce at least one related query matching the targeting keywords. All query instances matching this class are treated the same by the system (i.e. they are all matched to the ad), whereas all query instances outside this class are treated similarly (i.e. they are not matched to the ad). Depending on the specific settings, we can take one of the two approaches to address Simpson's paradox.

Task 5 *Develop techniques to identify subpopulations that are treated differently by a process to rule out Simpson's paradox.*

Task 6 *Consolidate above tasks and design algorithms which can detect, trace, and rule out Simpson's paradox for violations of associative properties.*

3.4 Explanations on the Bing pipeline

Finally, we propose to apply these methods to detect and explain associative properties on the Bing advertising pipeline. Discrimination is one of the properties that we are interested in studying in the Bing pipeline. While our prior work was able to detect discrimination in a very specific setting, with access to logs of advertising data, we can detect discrimination across a wide range of users and ads. We first identify ads which are served disparately based on a sensitive attribute (like gender). This finding serves as prima facie evidence of discrimination. Next, we try to assign responsibility of the found violation to internal modules and computations inside the Bing pipeline.

Task 7 *Implement and apply algorithm to detect, trace, and rule out Simpson's paradox for discrimination based on gender in the Bing advertising pipeline.*

3.5 Proposed Timeline

Task	Expected completion date
Definitions and proofs of responsibility and traceable metrics of association (Tasks 1, 2, 3)	April 2017
Estimation of traceable metrics (Task 4)	May 2017
Address Simpson's paradox and consolidate algorithm (Tasks 5, 6)	May 2017
Implementation of methods on Bing pipeline (Task 7)	July 2017
Writing of thesis and paper	September 2017

Bibliography

[1] Charles Duhigg. Psst, you in aisle 5. *New York Times Magazine*, 2012. Online version titled "How Companies Learn Your Secrets".

[2] Craig E. Wills and Can Tatar. Understanding what they do with what they know. Technical Report WPI-CS-TR-12-03, Computer Science Department, Worcester Polytechnic Institute, 2012.

[3] 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.

[4] Latanya Sweeney. Discrimination in online ad delivery. *Commun. ACM*, 56(5):44–54, 2013.

[5] Michael Carl Tschantz, Amit Datta, Anupam Datta, and Jeannette M Wing. A methodology for information flow experiments. In *Computer Security Foundations Symposium, IEEE 28th*, 2015.

[6] Amit Datta, Michael Carl Tschantz, and Anupam Datta. Automated experiments on ad privacy settings. *Proceedings on Privacy Enhancing Technologies*, 2015(1):92–112, 2015.

[7] Google. About the google search network. `https://support.google.com/adwords/answer/1722047`. Accessed Feb. 6, 2017.

[8] Google. About the google display network. `https://support.google.com/adwords/answer/2404190`. Accessed Feb. 6, 2017.

[9] Facebook. Facebook ads. `https://www.facebook.com/about/ads/`. Accessed Feb. 6, 2017.

[10] 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.

[11] 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.

[12] Mathias Lecuyer, Riley Spahn, Yannis Spiliopolous, Augustin Chaintreau, Roxana Geambasu, and Daniel Hsu. Sunlight: Fine-grained targeting detection at scale with statistical confidence. In *Proceedings*

of the 22nd ACM SIGSAC Conference on Computer and Communications Security, pages 554–566. ACM, 2015.

[13] Aniko Hannak, Piotr Sapiezynski, Arash Molavi Kakhki, Balachander Krishnamurthy, David Lazer, Alan Mislove, and Christo Wilson. Measuring personalization of web search. In *Proceedings of the 22nd international conference on World Wide Web*, pages 527–538. ACM, 2013.

[14] Aniko Hannak, Gary Soeller, David Lazer, Alan Mislove, and Christo Wilson. Measuring price discrimination and steering on e-commerce web sites. In *Proceedings of the 2014 conference on internet measurement conference*, pages 305–318. ACM, 2014.

[15] Ralf Küsters, Tomasz Truderung, and Andreas Vogt. Accountability: Definition and relationship to verifiability. In *Proceedings of the 17th ACM Conference on Computer and Communications Security*, CCS '10, pages 526–535, New York, NY, USA, 2010. ACM.

[16] Michael Backes, Anupam Datta, Ante Derek, John C Mitchell, and Mathieu Turuani. Compositional analysis of contract-signing protocols. *Theoretical Computer Science*, 367(1-2):33–56, 2006.

[17] Anupam Datta, Deepak Garg, Dilsun Kaynar, Divya Sharma, and Arunesh Sinha. Program actions as actual causes: A building block for accountability. In *Computer Security Foundations Symposium (CSF), 2015 IEEE 28th*, pages 261–275. IEEE, 2015.

[18] Dino Pedreshi, Salvatore Ruggieri, and Franco Turini. Discrimination-aware data mining. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 560–568. ACM, 2008.

[19] Sara Hajian and Josep Domingo-Ferrer. A methodology for direct and indirect discrimination prevention in data mining. *IEEE transactions on knowledge and data engineering*, 25(7):1445–1459, 2013.

[20] Salvatore Ruggieri, Dino Pedreschi, and Franco Turini. Data mining for discrimination discovery. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 4(2):9, 2010.

[21] Binh Thanh Luong, Salvatore Ruggieri, and Franco Turini. k-nn as an implementation of situation testing for discrimination discovery and prevention. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 502–510. ACM, 2011.

[22] Florian Tramèr, Vaggelis Atlidakis, Roxana Geambasu, Daniel J. Hsu, Jean-Pierre Hubaux, Mathias Humbert, Ari Juels, and Huang Lin. Discovering unwarranted associations in data-driven applications with the fairtest testing toolkit. *CoRR*, abs/1510.02377, 2015.

[23] Joan Feigenbaum, Aaron D. Jaggard, Rebecca N. Wright, and Hongda Xiao. Systematizing "accountability" in computer science (version of feb. 17, 2012). Technical Report YALEU/DCS/TR-1452, Department of Computer Science, Yale University, 2012.

[24] Amit Sharma, Jake Hofman, and Duncan Watts. Split-door criterion for causal identification: Natural experiments with testable assumptions. 2017. Accessed Jan. 24, 2017.

[25] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. Fairness through awareness. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, ITCS '12, pages 214–226, New York, NY, USA, 2012. ACM.

[26] Richard S Zemel, Yu Wu, Kevin Swersky, Toniann Pitassi, and Cynthia Dwork. Learning fair representations. *ICML (3)*, 28:325–333, 2013.

[27] Joseph A. Goguen and Jose Meseguer. Security policies and security models. In *IEEE Symp. on Security and Privacy*, pages 11–20, 1982.

[28] Judea Pearl. *Causality*. Cambridge University Press, second edition, 2009.

[29] R. A. Fisher. *The Design of Experiments*. Oliver & Boyd, 1935.

[30] Phillip Good. *Permutation, Parametric and Bootstrap Tests of Hypotheses*. Springer, 2005.

[31] Google. About the ad auction. `https://support.google.com/adsense/answer/160525`. Accessed Feb. 6, 2017.

[32] Facebook. Ad auction — facebook help center. `https://www.facebook.com/business/help/163066663757985`. Accessed Feb. 6, 2017.

[33] 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.

[34] Geoffrey Smith. Recent developments in quantitative information flow (invited tutorial). In *Proceedings of the 2015 30th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pages 23–31. IEEE Computer Society, 2015.

[35] Yusuke Kawamoto, Konstantinos Chatzikokolakis, and Catuscia Palamidessi. Compositionality results for quantitative information flow. In *International Conference on Quantitative Evaluation of Systems*, pages 368–383. Springer, 2014.

[36] Peter J Bickel, Eugene A Hammel, J William OConnell, et al. Sex bias in graduate admissions: Data from berkeley. *Science*, 187(4175):398–404, 1975.