JOURNAL OF DIGITAL SOCIAL RESEARCH

WWW.JDSR.IO ISSN: 2003-1998

VOL. 3, NO. 2, 2021, 89–113

FEATURE ANALYSIS: A METHOD FOR ANALYZING THE ROLE OF IDEOLOGY IN APP DESIGN

Amy A. Hasinoff[a] and Rena Bivens[b]

ABSTRACT

Many apps are designed to solve a problem or accomplish a task, such as managing a health condition, creating a to-do-list, or finding work. The solutions that app developers offer reflects how they believe that users and other stakeholders understand the problem. Each individual developer may have different ideas but analyzing many apps together can reveal the average or typical ways that developers in the set think about the problems that their apps are designed to solve. Building on content analysis, interface analysis, the concept of affordances, and speculative design, this article offers a new method that we call "feature analysis" to analyze what a set of apps designed to solve the same problem can tell us about the relationship between app design and ideology. By counting and classifying the features in a set of apps, feature analysis enables researchers to systematically answer questions about how app developers' design choices reflect existing cultural norms, assumptions, and ideologies.

Keywords: app studies; mobile apps; software studies; content analysis; affordances.

[a] University of Colorado Denver, United States.
[b] Carleton University, Canada.

INTRODUCTION

"Want to Stop Rape? There's an App for That," claimed a headline in the politics section of *Mother Jones* magazine (Pasulka, 2012). The article focused on an app called Circle of 6, which has also been described in *Wired* as "the 21st century rape whistle" (Lapowsky, 2014). We saw these headlines and wondered, could apps really help prevent rape? We were curious how developers'[1] assumptions about the nature of sexual violence might show up in the design of these kinds of apps. When we found over 200 of them, we designed a study to answer these questions (Bivens and Hasinoff, 2018). Through that process, we developed a method we call "feature analysis" to systematically answer questions about how app developers' design choices reflect existing cultural norms, assumptions, and ideologies. The method involves counting and classifying the features in a set of apps intended to address the same social problem. This article describes the method of feature analysis by first situating it in the literature on apps and other technologies and then describing the four phases of the method. We conclude by discussing some complementary methods and feature analysis' unique contributions.

A range of scholars demonstrate that despite often appearing neutral and objective, many technologies reproduce and reinforce social hierarchies and systemic inequalities, including, for example, Google search algorithms (Noble, 2018), facial recognition software (Buolamwini and Gebru, 2018), airport security body scanners (Costanza-Chock, 2018; Magnet and Rodgers, 2012), and automated systems for determining public service benefits such as welfare (Eubanks, 2018). This kind of work often examines the discriminatory outcomes of these technologies and focuses on investigating and describing how they impact users. Feature analysis offers one way to gain insight into the design choices—specifically in apps—underlying the kinds of discriminatory outcomes that these other scholars have demonstrated.

Research on how racism, sexism, and other forms of systemic discrimination are embedded in the design of technologies reinforces the insight from science and technology studies that "culture has always been technological and technologies have always been cultural" (Slack and Wise, 2005, p. 9). "Culture" here means a "socially shared symbolic system of signs and meanings" (Balsamo, 2011, p. 5). This shared symbolic system includes dominant ideologies – the widely held beliefs that typically justify the status quo and existing social arrangements—as well as residual norms and habits and emergent ideas. Our method of feature analysis is built on

[1] In this paper we use the term "developer" to mean anyone who is involved in the creation and marketing of an app. For some apps, the coding, interface design, and marketing are handled by separate teams of people while for others one person completes all these tasks.

the understanding that culture—particularly the dominant, taken-for-granted ideas about a particular social problem—shapes technological development, becomes embedded within technologies, and suggests intended users and uses. For example, the cultural notion of a gender binary is built into airport security body scanners, thus employees operating the devices must select "male" or "female" for each passenger. Because this design does not anticipate trans and nonbinary travelers, they face higher rates of false positives and undue burden passing through airport security systems (Costanza-Chock 2018). As Balsamo (2011) argues, "Through the practices of designing, cultural beliefs are materially reproduced, identities are established, and social relations are codified" (p. 3). Feature analysis stresses the interdependence of culture and technology by offering a technique for uncovering the role of ideology in one practice of technological design: app development. It does this by allowing researchers to analyze how a set of app developers working to create solutions to the same issue translate ideas about social problems and assumptions about users into designed objects.

While there are many studies of apps, most of the work on large sets of apps is not intended to investigate the relationship between app design and dominant ideologies. Many primarily quantitative studies examine health-related apps, and their goals are generally to assess the apps' quality and effectiveness, including which ones people should use, what features are missing in the available apps, and how some proportion of health apps are inaccurate or fail to follow most or any of the clinical recommendations. One review of research on health apps found that 79% of studies assess apps' clinical or scientific basis (Grundy, Wang, and Bero, 2016), and researchers taking this approach typically focus on how well apps' design features fulfill established treatment guidelines.

Meanwhile, the studies of apps that do focus on the relationships between design and ideology typically use qualitative approaches to examine just one or a few apps. Researchers use methods from internet studies, cultural studies, and critical discourse analysis to analyze individual apps, interfaces, and websites in detail. For example, one study of the features and marketing of seven self-tracking sexual and reproductive health apps finds that they reflect normative assumptions about gender and sexuality (Lupton, 2015). Further, in Morris and Murray's (2018) preeminent collection of qualitative research on apps, many authors take media and cultural studies approaches to examine individual apps "as vectors for the production, transmission, and interactions of culture" (p. 3). For example, Schüll's chapter on the weight-loss app LoseIt! examines "the assumptions about human agency and the technological mediation of health that inhere in the app's design logic, marketing appeals, functional affordances, and user practices" (2018, p. 103). Feature analysis shares this

collection's fundamental orientation to apps as "vectors of culture" but allows researchers to investigate a large set of apps.

One common qualitative approach to studying apps is to examine their interfaces. As Dieter and coauthors (2019) explain, "Enquiries into interfaces can tell us not only about the apps but also about the expectations that those interfaces have of users and how certain ideas about users are designed into those apps" (p. 4). In particular, the "walkthrough" (Light, Burgess, and Duguay, 2018) is a qualitative method that allows researchers to systematically document the interface in one or more apps to investigate design from the perspective of an app user. As Light, Burgess, and Duguay (2018) explain: "By walking through the app's registration, everyday use and deletion, this technique allows for recognition of embedded cultural values in an app's features and functions" (p. 896). For Light, Burgess, and Duguay (2018), culture can manifest in design in a variety of ways, which they analyze primarily as the characteristics of the "mediators"[2] in an app. These characteristics include the arrangement of the interface, such as "how the app guides users through activities via the placement of buttons and menus," the functions and features of an app "that mandate or enable an activity," the textual content of the app, and its symbolic aspects, including "the look and feel of the app and its likely connotations and cultural associations" (Light, Burgess, and Duguay, 2018, pp. 891-892). They explain that these characteristics "are embedded with culture because their meanings exist in reference to cultural texts and understandings that exist outside the app" (Light, Burgess, and Duguay, 2018, p. 891).

Researchers have examined how apps and website interfaces imagine and assume particular users, uses, and contexts in a range of ways. For example, Stanfill's (2015) discursive interface analysis investigates the "functionalities, menu options, and page layouts" (p. 1059) of website interfaces to consider "how technologies both arise from particular beliefs about what [u]sers ought to do and reinforce them by constraining the actions of site visitors" (p. 1071). Other studies investigate app interfaces and features alongside the materials produced by and about apps to consider how apps imagine their users and uses. A study of the family location tracking app Life360, for example, analyzes the apps' accompanying website, press releases, FAQs, and user reviews of the app (Hasinoff, 2017). Some researchers also investigate how developers choose to market their product to users and to potential advertisers. Bivens and Haimson (2016), for example, took on the role of "potential advertiser" to explore how the identity categories programmed into the interface built for users of an app differ from those in the interface built for advertisers.

[2] Here they refer to Latour's (2005) concept of a "mediator" as an actor that transforms meaning; in contrast, an "intermediary" passes along meaning unchanged.

Further, Light, Burgess, and Duguay's walkthrough method involves analyzing the "environment of expected use," which includes the intent, monetization, and rules of an app. They suggest that researchers analyze the "vision" of an app, which includes "its purpose, target user base and scenarios of use, which are often communicated through the app provider's organisational materials" (Light, Burgess, and Duguay, 2018, p. 889). They also suggest that researchers interested in how apps imagine users examine the "governance" (Light, Burgess, and Duguay, 2018) of apps by examining their terms of service and other policies and guidelines. This "governance" perspective can provide insight on how apps construct their legal relationship with users, including how users' personal data is retained and distributed, what constitutes acceptable use, and how developers attempt to limit their legal liability.

Feature analysis builds on the close attention other scholars pay to app interfaces and the ways culture is embedded in app design by specifically concentrating on features and applying an analysis of the mechanisms and conditions of those features' affordances (Davis, 2020; Davis and Chouinard, 2017). This allows researchers to focus on analyzing developers' assumptions about the range of appropriate or reasonable actions an app can take to solve a problem, which we discuss in detail in Phase Three below. Using this framework of "affordances" to quantify the imagined conditions for a feature to effectively work against a social problem, researchers can examine developers' choices in aggregate in a large set of apps. As Davis explains:

> Technologies are designed, implemented, and used through webs of choices. Some of these choices are explicit and reflect a clear intention for the technology to affect human action in some specific way. Other choices are implicit and may not ever enter the conscious minds of designers, distributors, or end users. Each choice—explicit or implicit—reflects and affects value orientations, sociostructural arrangements, and social dynamics. (Davis, 2020, p. 14)

Feature analysis demonstrates the choices developers have made to create features that they think will be compelling solutions to users and/or to investors and granting agencies. Further, by using speculative design in the final phase of feature analysis to imagine fictional apps that use existing technologies in different ways, researchers can explore the proportional influence of cultural, technological, and other constraints on design.

In what follows, we explain how feature analysis builds on and intervenes in a range of existing methods for studying media in general and for investigating the cultural and ideological aspects of design and technology. We divide feature analysis into four phases: (1) identify a social problem and find apps that attempt to address it; (2) identify the features

that each app offers; (3) categorize how the features address the problem; and (4) use speculative design to imagine alternative apps. We describe each phase by first situating feature analysis in the existing literature and then explaining the concrete steps for applying the method in that phase. Throughout the article, we use our study of anti-rape apps (Bivens and Hasinoff, 2018) as an illustrative example.

1 PHASE ONE: CHOOSE A PROBLEM AND USE KEYWORDS TO FIND APPS

In the first phase of feature analysis, researchers select an issue or problem and find the apps designed to address it. Feature analysis builds on work in media studies investigating how cultural objects reflect and uphold taken-for-granted ideas and existing social orders. For example, a content analysis of reality TV shows about policing can demonstrate that people of color are disproportionately represented as criminals (Oliver, 1994). In particular, feature analysis builds on cultural and media studies' traditions of investigating how popular media construct social problems. For example, one of the foundational works of cultural studies is a study of how journalists (and police) misrepresented the problem of "mugging" (Hall et al., 1978). Framing is a related model for analyzing how news promotes and reflects particular understandings of the world:

> To frame is to select some aspects of a perceived reality and make them more salient in a communicating text, in such a way as to promote a particular problem definition, causal interpretation, moral evaluation, and/or treatment recommendation for the item described. (Entman, 1993, p. 52)

In this model, journalists "select" and "highlight" aspects of a "perceived reality" according to professional and organizational norms. In the context of apps, feature analysis builds on framing by considering app developers as authors of cultural texts that define problems, offer interpretations, and propose solutions. These models investigate and highlight how social values shape communication choices—and with feature analysis, design choices as well—which in turn shape social values. As Carey (1989) explains: "Communication is a symbolic process whereby reality is produced, maintained, repaired, and transformed" (p. 23). Feature analysis investigates how app design is a form of communication in these terms.

Building on these frameworks, feature analysis can help illuminate which social problems and what type of solutions designers believe will be marketable in an app. Indeed, Lupton (2014) urges researchers studying health apps to view them as sociocultural artefacts. She observes that "the content and use of health and medical apps can tell us much about which

types of illnesses and health conditions are considered important at a particular moment and what medical or health promotion strategies are privileged to prevent or treat these conditions" (Lupton, 2014, p. 611). One example of this approach compares the actual prevalence of health conditions in the population to the number of apps addressing those conditions, finding that while "diabetes and depression have an overwhelming number of apps and research, there is a lack of apps related to other conditions, such as anemia, hearing loss, or low vision" (Martínez-Pérez, de la Torre-Díez, and López-Coronado, 2013). Indeed, researchers would expect that a set of cultural objects does not mirror actual or measurable cultural, social, or material conditions—such as disease prevalence in a population or the racial demographics of the people who commit burglaries—but instead reflects the beliefs and assumptions about these issues. As such, the value of this kind of work, including feature analysis, is that it can assess and interpret the nuances of what these inaccuracies or inconsistencies can tell us about society, ideology, and dominant cultural assumptions.

1.1 Steps for Phase One

To create a set of apps, researchers can begin with an initial list of apps from any source, such as media coverage or a funders' campaign, or simply with a list of common terms that describe the problem in question. For our study, we began with a list of apps from the US Department of Health and Human Services' 2011 "Apps Against Abuse" initiative and developed a list of search terms from their descriptions. We used those terms to search the app store, and eventually developed a longer list of keywords (ultimately 13 in total) by examining the titles and descriptions for apps from our ever-growing list. We also found some apps in media coverage (ie. "The 10 best safety apps for women") as well as directory sites like App Annie. Gerlitz and coauthors (2019) also suggest that researchers should carefully consider app titles because developers are strongly encouraged by app store guidelines to write "accurate and focused titles and descriptions to cover what the app is about" (p. 6). Researchers should describe their process and rationale for the chosen search terms, given that neither their choices nor the search results will be neutral or objective (Noble, 2018; Rogers, 2013).

Next, develop a set of inclusion criteria that meet the study's goals (such as language, region, or current availability) and whether the app's title and/or description states or implies that it is intended to address the problem in question. Our study analyzed all the English anti-rape apps we could find in the Apple and Android app stores because our aim was to collect as complete a set as possible. Researchers could develop a method for limiting the set of apps (e.g. most downloaded, highest search-ranked,

etc.) as long as they provide a rationale for each choice and consider the resulting implications for their analysis.

Searching app stores on a mobile device replicates how many users might find apps, but these search engine results likely do not actually capture all existing apps designed to address a particular problem. Indeed, there is no public information about how the Apple or Android app stores rank search results. Web-based app stores and Google search results for apps are inconsistent so we recommend using mobile devices, primarily, to search app stores instead. We note that a mobile device's location and IP address will also influence search results, which meant that our study was limited by the use of only one location. Based on our experiences searching for particular keywords, we discovered that alongside app names and descriptions, Google Play's search results also seemed to include reviews from the app store page; for some results we could not find our keyword anywhere in the title or description.

We recommend systematically collecting data about the apps in a database. Consider including: app name, developer, app store category, app store description, cost, in-app purchases, operating system, release and update dates, and ideally, all of the app store images. We used Excel spreadsheets to create our database and do not recommend this approach because it becomes onerous to recategorize and organize the data. Content analysis software packages such as NVivo, Quirkos, or open source options such as Taguette, CATMA, or AQUAD would be useful for calculating frequencies and relationships between classification systems in the following phases.

2 PHASE TWO: IDENTIFY THE FEATURES THAT EACH APP OFFERS

In the second phase, researchers identify and classify the features in each app in their set. Feature analysis builds on qualitative studies of app interfaces and extends these with content analysis, specifically taking an inductive grounded theory approach. The variations on content analysis are well developed in various handbooks (e.g. Krippendorff, 2012; Neuendorf, 2017), which offer detailed guidance on developing classification systems as well as other aspects of content analysis such as sampling techniques (if the entire set of apps is too large to analyze) and intra- and inter-coder reliability.[3]

Following desciptions of the term "feature" from human-computer interaction and science and technology studies, we define a "feature" as a function that users control or are likely aware of. Some examples of features

[3] Reviewers' and journals' needs for such precise methodological techniques vary.

in our set included: tracking location, sounding an alarm, contacting the police, and taking a quiz. A feature can be any function that the user executes in the app, including anything the user can access, modify, or control, as well as functions performed on the user's behalf. In particular, we follow the definition of the term "feature" that is described in an article on the communication features of Facebook: "a technical tool on the site that enables activity on the part of the user" (Smock et al., 2011). Feature analysis asks researchers to analyze features separately rather than using individual apps as the units of analysis. Breaking apart each app into its component features helps the researcher gain a more fine-grained data set that can yield persuasive findings about the range and frequency of design choices made across the entire sample of apps.

Because we define features as functions that developers publicize or make visible to users, we do not view functions that are hidden from users as "features." Determining whether a function is a feature or not depends on the context of each app, which is necessarily a subjective interpretation. For example, tracking location would be a "feature" according to our definition in a safety app that advertises location-tracking as a safety feature. On the other hand, in a sexual-violence prevention app that provides educational materials and quizzes but also tracks user location to serve tailored advertisements, location-tracking would be a function but not a feature. While such an app may have informed users that it collects and uses location information, such as in its terms of service or by requesting location access to install the app, this function is not publicized to users so we would not consider it a "feature."

Given that the focus of our study was on how features address a problem, we did not examine every feature in each app. Instead, we limited the set of features to those that directly relate to solving the problem—this required our interpretation and assessment of each app as a whole. We rarely found features that were ambiguous in whether or not they were intended to directly address the problem, and if we had any doubt we included the feature. In our study, that meant we were only interested in features that were specifically designed to prevent rape because our research question was to examine and compare the various strategies these features used to achieve this goal. For example, we did not include actions related to settings, documentation, or cosmetic features, such as "view the privacy policy" or "update profile photo," because these features were not designed to directly help prevent sexual assault, even if they were functions that were required or suggested to the user. Researchers could, of course, choose to analyze all the features in their set of apps if it serves their research questions.

2.1 Steps for Phase Two

There are two ways of finding features in apps: (a) rely on the app store page description and screenshots for each app or (b) download and "walkthrough" (Light, Burgess, and Duguay, 2018) each app individually. We chose the app-store based approach because it was more efficient, though we did "walkthrough" a sample of the apps in our set to gather background information and find illustrative examples for our paper. Using this first option, we examined each app store page, including the app store descriptions and screenshots, and recorded each discrete feature we could identify. This method of relying on app store pages captures all the features that developers think are important enough to entice users to download their apps. By limiting our study to the features that were described or pictured in the app store pages, we only rarely needed to exclude any features that seemed as though they were not aimed in any way at preventing rape. That is, developers who are advertising an app to address a specific problem tend to highlight the features that they think users will find appealing for solving that problem.

The second option, downloading and doing a "walkthrough" (Light, Burgess, and Duguay, 2018) of each app, would have only been practical for us with a smaller set of apps or more resources. This method involves recording all the features of each app, and requires using each app with one or more user profiles, including setting up the app, exploring all its options and features, and then deleting or discontinuing it—researchers might find dozens of features in each app. The advantages of the "walkthrough" method are that it captures all the features in the set of apps and that researchers could also then compare which features are publicized to users in the app stores and which features are only presented within the app itself.

After recording the features of each app in a database, researchers can create a classification system that captures and names all the relevant features in the set. Using inductive content analysis techniques, we began with an initial random sample, and analyzed all the apps one at a time, recording each new feature we discovered. Then, we returned to the entire set to analyze it again based on this classification system. We found new ways to classify features throughout the analysis and so we returned to the set a number of times until all apps had been classified according to the same comprehensive list of features. In the end our classification system consisted of 47 individually named features. For example, in our set of 215 apps, 8% offered a "loud alarm" feature and 43% offered a "GPS tracking" feature.

3 PHASE THREE: CATEGORIZE HOW THE FEATURES ADDRESS THE PROBLEM

In the third phase, researchers consider apps as tools, using the concept of "affordances" to assess each feature's mechanisms, conditions, and expected outcomes. Techniques from content analysis are basically sufficient to complete the identification and classification of features in the previous step (Phase Two). However, apps' capacity to act or to be used as tools for action sets them apart from other cultural objects like newspaper articles or TV shows. As Dieter and coauthors (2019) propose, "App research necessitates a renewed interest in the role of practices … as opposed to the study of content …[because] apps are first and foremost operational media; they are applications, things for doing" (p. 12). This is important because many apps are designed to solve a problem or accomplish a task, from managing a health condition, to creating to-do-lists, finding work, or tracking calories. In this way, like most software, apps are both tool and text, though of course the distinctions between these two categories are already fuzzy. Still, apps are more responsive, interactive, and non-linear than most other texts, such as TV shows or even most websites. Apps also change more dynamically than most other non-digital tools. Contrast, for example, the way a pencil changes as it is used, sharpened, broken, or chewed with the way an app's interface is programmed to respond to each tap of the user's finger.

Apps are also distinct from software: they are more mobile, ubiquitous, and integrated into everyday life, and are often in more intimate proximity to users' bodies (Morris and Murray, 2018). Apps' interactivity means that users' every action in an app can be potentially recorded and sold to advertisers. Though a well-loved novel might bear traces of readers' favorite passages or marginal notes, apps can harvest and distribute an unprecedentedly detailed record of each user's actions. Apps can also collect intimate and personal data. As Lupton (2014) points out, "sensor technologies in smartphones or wearable devices ... promote detailed and continuous monitoring of bodily functions and behaviours" (p. 611). All of this data collection raises important questions about security and privacy. Feature analysis—especially if researchers use the walkthrough method to collect all features—can help uncover and catalogue the ways that a set of apps could collect behavioral data about users.

We turn to Davis' model of "affordances" for its precise theoretical framework for assessing what features can do, for whom, and under what conditions. As Davis (2020) explains: "Affordances are how objects shape action for socially situated subjects" (p. 6). Using the model of affordances helps researchers avoid technological determinism while still appreciating

the complex interactions between people and objects. The model stresses that "technologies don't *make* people do things but instead, push, pull, enable, and constrain" (Davis, 2020, p. 6). In other words, the outcomes that developers expect their app to produce are not simply or directly determined by an app's features. Instead, as Davis (2020) explains: "affordances mediate between a technology's features and its outcomes" (p. 6). This mediating role of affordances is vital for feature analysis. Specifically, in Davis' (2020) model: "The mechanisms of affordance specify *how* technologies afford, while the conditions of affordance situate technologies in context" (p. 13). This approach of thinking about the mechanisms and conditions of affordances can help illuminate how the outcomes of features might vary for different people under different conditions.

The concept of "mechanisms" describes the different ways a feature and a user might interact with one another: "artifacts *request, demand, allow, encourage, discourage,* and *refuse*" (Davis and Chouinard, 2017, p. 242) actions from the user. For example, a speed bump *requests* that drivers reduce their speed while a speed-limiting device installed in a car *demands* it by restricting the engine's power. Davis and Chouinard (2017) explain that *requests* and *demands* are "bids that the artifact places upon the subject;" generally a *demand* is required while a *request* is optional (p. 242). The model also captures how features respond to a user's actions: features might *discourage* or *refuse* what a user wants. For example, one app might *discourage* users from deleting their account by burying the option deep in the settings menu while another app might *refuse* this desire by providing no such option at all.

For feature analysis, it is especially crucial to consider that the way affordances work also depends on the user and their contexts. In other words, affordances are relational, and "emerge in the mutuality between those using technologies, the material features of those technologies, and the situated nature of use" (Evans et al., 2017, p. 36). Davis' model captures this relational nature with a framework for assessing the "conditions" of affordances. Davis and Chouinard (2017) explain: "the conditions of affordances vary with subjects' awareness of the function (perception), their skill and ability to execute the function (dexterity), and social support in executing the function (cultural and institutional legitimacy)" (p. 245). The analysis in Phase Three uses these concepts to allow researchers to assess the nature and size of the gap between designer's imagined outcomes for the features they have included in their apps and the likely outcomes for a range of different users in different social and personal contexts.

3.1 Steps for Phase Three

In this phase, researchers use Davis' model of affordances to develop classification systems to organize and describe what features offer, to whom, and under what conditions. If they have chosen the walkthrough method described above, researchers will likely want to begin by classifying features based on whether they address the problem or whether they are instead merely suggested or required for using the app itself (e.g. "change the background" or "create a profile"). Recall that in contrast, the app store method generates a smaller set of features that are all (or almost all) designed to address the problem simply because those are typically the features developers think are worth advertising in app stores.

Researchers' task in feature analysis is not only to classify the outcomes that apps promise their features will provide, but to then analyze and assess them in relation to the literature on the nature and causes of the problem in question. Using Davis' framework for analyzing the mechanisms and conditions of affordances, researchers can consider how the features in their set of apps might lead to different outcomes for different users in different contexts. This step of the analysis is vital, as researchers will likely find that some claims about what features do or who they are for may be implausible or unlikely based on the literature about the characteristics of the problem the feature is claiming to address.

Consider an example from our study: a feature to call the police in an app that claims that it can help prevent date rape. Applying Davis' framework, we could say that a "call police" button is a feature that uses the *mechanism* of *encouraging* users to call police for help during or after an assault. However, consider the *conditions*: even if a user is aware of the option to call police (perception) and is physically capable of reaching for a phone and tapping a "call police" button during an attempted date rape (dexterity), the personal and cultural context—including emotional manipulation, abuses of power, fear of further violence from the person threatening to commit harm, or an expectation that police would disregard their report or make the situation worse (legitimacy)—means that such a feature likely will not lead to the purported outcome of preventing date rape. Indeed, we found no evidence in the extensive literature on the prevention of date rape that increasing ease of access to 911 calls (in this case presumably by providing a button within an app rather than dialing the number via the phone's keypad) would be an appropriate or useful solution to the problem of date rape. Through this kind of analysis of each feature, our study found that most of the features in our set would only likely lead to the imagined outcome (rape prevention) in the rarest cases (stranger perpetrators).

Analyzing how the mechanisms and conditions of affordances mediate the relationship between features and their supposed outcomes can help researchers consider the following questions: the exact type or version of the problem each feature is designed to address; which social and cultural contexts each feature would be useful for; the kinds of action each feature uses to address the problem; the intended user of each feature; and the implicit causes of the problem or theories of prevention that underlie each feature. Each of these lines of inquiry could yield distinct, and potentially useful, classification systems, which researchers can then compare to each other.

In our study, we created a total of four classification systems that were useful for our analysis. First, we classified (a) all 47 unique features in our set, as described in Phase Two. Then, we created three more classification systems to describe the affordances of these features and their mechanisms and conditions: (b) seven types of action; (c) two contexts of use (in relation to an incident or not); and (d) three types of intended users (victim, bystander, or perpetrator). While feature analysis asks researchers to examine features individually, sometimes researchers may also need to look at each app as a whole to classify a feature. For example, a feature such as "recording video" could be used or intended for use by any type of user (a potential victim, bystander, or perpetrator), so we classified its intended use based on the description on the app store page.

Table 1. Feature types and examples

Type of action	**Examples from our study**
Send information	A button to call a sexual assault hotline
	Share sexual harassment incidents on a map
Get information	Take a quiz about sexual assault
	View map of registered sex offenders
Monitor or track the user and/or their immediate environment	Automatic geofence check-in
	Keep a diary of abusive incidents
Use the device as a tool in the immediate physical environment	Sound a loud alarm
	Get a fake diversion call

One classification system we used in our study described features' specific types of action. While we ultimately used seven categories, we simplify them here into four (see Table 1) in hopes that it will be more broadly applicable as a starting point for other researchers. We encourage researchers who use this classification system to modify it to reflect the specific themes in their data sets. In our study, for example, we divided "get information" and "send information" into a total of five distinct categories.

Though we used the literature on sexual violence as a starting point to develop our classification systems, we modified them to create the best possible representation of our data set. For example, one classification system we started with based on the literature was intended to capture "prevention strategy" with three items: "incident intervention," "precautionary measures," and "education and awareness." However, we found too much overlap between the latter two categories in our set of features. We could not determine, for example, if things like maps of registered sex offenders or quizzes about safety strategies should be considered "precautionary" or were better classified as "education." In the end, we decided to collapse these two categories, creating a classification system we labeled "context of use" instead. This classification system used a binary distinction that was relatively uncomplicated to apply: whether the feature was intended for use in relation to a specific incident or not.

4 PHASE FOUR: USE SPECULATIVE DESIGN TO IMAGINE ALTERNATIVE APPS

The fourth and final phase of feature analysis uses speculative design to help researchers gain a better understanding of how social norms, assumptions, and taken-for-granted ideas are translated into technologies. Speculative design methods generally involve developing and analyzing fictional future scenarios (DiSalvo, 2012; Dunne and Raby, 2013; Forlano and Mathew, 2014). As DiSalvo (2012) explains, "Speculative design works by isolating facets of culture and recasting those facets in ways that alter their meaning in order to produce new images—new imaginative instantiations—of what might be" (p. 111). The purpose of speculative design is not to predict the future or design better apps, but "to better understand the present and to discuss the kind of future people want, and, of course, [futures] people do not want" (Dunne and Raby, 2013, p. 3). This approach can help researchers consider how ideologies are entangled with technological development.

4.1 Steps for Phase Four

Feature analysis applies speculative design by imagining fictional apps that use the existing components of mobile phones in different ways. This exercise can help clarify the particular role of technological constraints in the set of features. After all, if a feature is common in other app genres but is not used in the set of apps addressing a particular social problem, that provides some evidence that the choice to not deploy those features is likely not the result of a technological constraint. For example, in our study we found that social connectivity features in anti-rape apps position the users'

friends primarily as emergency responders, not as sources for increasing motivation or engagement (such as in fitness apps, games, and goal-setting apps).

In our study, we imagined and described some fictional apps that used features that were already available in mobile phones and were in use in other kinds of apps. These fictional apps were based on the rape-prevention literature and offered features designed to prevent rape that were absent from our set of apps. For example, we imagined a fictional app for potential victims that could provide users with evidence-based training in overcoming the social and emotional barriers to resisting unwanted sexual advances from people they know. We also imagined an app that could complement comprehensive rape-prevention programs that reduce the likelihood that a person will commit acts of sexual violence. And we imagined an app for potential bystanders that focused on dismantling rape myths. Imagining these fictional apps allowed us to bolster our conclusion that the limited set of features and imagined outcomes we found in our set of anti-rape apps were likely not the result of technical limitations and thus could be evidence of persistent rape myths instead.

To help researchers design fictional apps, we offer the following list of "actions" (Table 2) which describe the possible inputs and outputs of the components of mobile phones.[4] Features, as we define them above, generally use two or more actions in combination. For example, some safety apps offer a geofence check-in feature that automatically sends a message to another user (e.g. a spouse or parent) when the device arrives at a pre-set location (e.g. "home"). This feature operates through a number of device actions: detecting location, sending data to another device, and displaying a notification on the screen, among others.

We recommend that researchers review the literature on how best to address the social problem in question and then consider how this list of the possible "actions" of mobile phones could help users access those kinds of approaches and solutions. This will then allow researchers to discuss whether their set of apps has used all the available features effectively. Then, researchers can explain whether and how imaginary apps might use the possible features more effectively. Researchers may well determine that the existing features of mobile phones are entirely ill-suited to addressing the social problem in question.

[4] We created this list of the currently possible actions of most (though not all) smartphones by extrapolating from our own study and then reviewing Apple's and Android's documentation for mobile app developers.

Table 2. Some mobile phone components and their actions

Device component	Actions	Examples
buttons	press a button (power, volume up/down, home, etc.)	make selections; initiate actions
camera	capture images	record video; take a photograph
communication chips and antennas	receive GPS data from satellites	make a phone call; send an email or text message; communicate with other devices and networks (eg. airdrop, contactless pay, smartwatch connections)
	communicate with cell towers	
	connect to the internet via wifi	
	communicate with bluetooth devices	
	employ near field communication	
light	emit light	light as an alarm; illuminate the environment
microphone	capture sound	record a voice memo or other sound; monitor ambient noise level
screen display	display (including text, images, video)	display an image or web page; display an alert or pop-up message (e.g. a "toast" or "snackbar" that automatically appears and disappears without user action)
sensors	measure device motion	measure acceleration or speed
	measure environment around the device	measure humidity, temperature, or ambient light level
	assess the position of the device in space or time	measure proximity to another object; measure device orientation in space (e.g. face down initiates "do not disturb" mode); set a timer
	detect fingerprint	read and store biometric information; identify a user
	measure device conditions	measure battery level or internal temperature
speaker or headphone jack	emit sound	play music; emit phone call ringtone; play sound as alarm
storage	store information	record data in persistent memory; install apps
touchscreen	select from options on the screen	select a setting; use a keyboard to select letters to enter text; choose a quiz question answer
	gesture on a touchscreen	swipe left or right to make a choice; scroll through text; draw a letter to enter text
vibration motor	vibrate	notify the user; provide touch feedback

5 CONSIDER ADDING COMPLEMENTARY METHODS

Feature analysis can offer one type of evidence for the ideologies underlying design, but researchers could also consider adding complementary methods to further examine how ideology manifests in design, particularly by studying developers or users more directly.

5.1 Study developers

One way researchers have studied designers' choices is by observing their work directly or speaking with them (e.g. Cohn, 2017; Suchman, 2011). Asking developers questions about their views on the nature and causes of the social problem in question, and where they gained this knowledge, could also provide insights into their design processes. For example, for our study it could have been valuable to find out which developers referred to scholarly research or consulted rape-prevention practitioners and whether and how those practices were related to the set of features in their apps. Further, researchers could consider complementing feature analysis by directly studying the institutional, technological, and economic contexts of developers' design processes. For example, in-depth interviews with developers could provide answers to questions about how they chose to address the problem and what constraints they faced, including: how they imagined and investigated the problem; whether and how they tested the app with users; what approaches they attempted and abandoned; whether they lacked financial or technical resources; whether they were limited by app store policies, operating systems, and other hardware or software constraints; and how funders and/or granting agencies influenced their design choices.

Researchers could also complement feature analysis by pursuing other methods to investigate the precise nature of the infrastructural constraints developers navigate. Getting a better understanding of those constraints could provide further evidence of the relative influence of ideology on the features in the set as compared to technological, financial, legal, or other factors. For example, researchers could build on Dieter and coauthors (2019), who explain that infrastructural approaches to understanding how apps "operate within different sites and involve a diversity of often obscured stakeholders" are "unified by a commitment to unpacking the infrastructural embeddedness of apps and with an eye on political economy" (p. 13). For example, they suggest using app repositories to open app packages and investigate their component parts and creating logs of an app's network connections. Further, researchers could examine how political economic issues influence app design, including investigating

their monetization strategies and how incubators and investors have influenced app development (e.g. Murray and Ankerson, 2016).

The findings feature analysis can offer are limited to the specific sample of apps selected for a study. Indeed, feature analysis is not a mechanism for measuring public opinion about appropriate or desirable solutions for a social problem, as it does not offer a representative sample. Developers, and indeed the particular developers whose apps are included in the data set, may hold views about social problems that are uncommon or unconventional. However, researchers can compare the results of public opinion studies with the apparent assumptions among the developers who designed their set of apps. For example, in our study, we noted that it was not surprising that our set of apps reflected common rape myths, because a range of studies demonstrate that such beliefs are prevalent in the general population.

5.2 Study users

Feature analysis examines what developers assume about users through the features they create. As such, researchers might consider if complementing feature analysis with another method that studies users more directly could help answer their research questions. Researchers interested in studying users can draw on the rich methodological traditions in media studies and mobile media studies that examine how users interpret and use media. For this type of analysis, researchers could conduct surveys, observational studies, automated activity tracking, or participant-directed app use research (Leurs, 2017). For example, researchers could examine user comments in app store reviews to gain information about whether they perceive features as useful and how they use them in both intended and unintended ways. Researchers could also use surveys to compare apps' promises, features, and policies with how people use and understand them. Fiesler, Lampe, and Bruckman (2016), for example, analyze the copyright terms of a few dozen websites where users post content and then find, through a survey, that user expectations differ significantly from the terms they have agreed to. Another option to study users is to use critical theory to examine technological artifacts, user discourses, and the relationships between them by employing Brock's (2018) critical technoculture discourse analysis. With this method, "the goal is to sustain a subtle analysis of both the domestic context of use and the semiotic richness of the online world people engage in" (Brock, 2018, p. 1023). Because feature analysis focuses on uncovering ideology embedded in design choices and only examines what apps offer to users, researchers who also want to know how users perceive or engage with those apps and their features will need to turn to complementary methods.

6 CONCLUSION

We present the method of feature analysis without any normative claim that it offers a greater truth value than other qualitative or quantitative approaches to studying apps. Nonetheless, we conclude here by highlighting the advantages of feature analysis. First, we share our anecdotal observation that some audiences thought that our method's analysis of a large set of apps offered particularly persuasive findings. Second, we summarize the key methodological contributions that feature analysis offers.

Within a year of the publication of our article on anti-rape apps, Bivens was invited to Vienna to participate in a United Nations committee meeting on technology and gendered violence. At the meeting it became apparent that for some participants, the number of apps we included in our study ("over 200!") was especially impressive. Likewise, later that year at a conference in San Francisco, an audience of anti-violence advocates, legal professionals, and law enforcement personnel reacted in similar ways. Though some attendees had conducted their own critical analysis of a handful of apps (National Network to End Domestic Violence), many people remarked on the total number of apps we included in our study.

Yet despite our anecdotes, many studies have not found significant or consistent differences in how qualitative versus quantitative research is used in policy development. In particular, one researcher explains: "Policy decisions in the real world are not deduced from empirical-analytical models, but from politics and practical judgement … what counts as 'evidence' is diverse and contestable" (Head, 2010, p. 83). The extensive literature on how policymakers use research demonstrates that the process is complex, including factors such as personal relationships (e.g. Almeida and Báscolo, 2006; Head, 2010). While it may not be possible to establish direct policy effects, we have observed that at least some civil society stakeholders perceive that an analysis of a larger data set offers more persuasive findings than a qualitative analysis of one or a few apps.

Feature analysis also offers some methodological advantages. First, it allows researchers to analyze data that is collected without the involvement of the people being studied. The benefit of this observational data is that it represents the actual choices developers made by examining a set of design outcomes. While surveys and interviews with developers could provide valuable data on their intentions, beliefs, and perceptions, their answers would likely also be influenced by social desirability biases. The contribution feature analysis offers in this respect is that it allows researchers to analyze a rich data set of designers' actual choices within the financial, technological, and policy constraints of funding, creating, and releasing an app, which all developers face to varying degrees. Developers

do not make use of every device component or possible action, but instead create specific features they feel will be most useful to achieve their goals. Developers do not rely solely (or at all) on scientific evidence to develop appealing products aimed to solve problems; instead, they rely on their instincts and assumptions about what they think people will find appealing as "common sense" easy-to-understand solutions. Feature analysis can provide a systemic analysis of the nature of these assumptions by investigating how they show up in the final product by examining a set of apps created by a group of developers tackling the same problem.

The second advantage feature analysis offers is that combining techniques from inductive content analysis with the concept of affordances and techniques of speculative design illuminates how developers materialize social norms, common sense assumptions, and ideologies in the form of app features. While existing techniques of content analysis are sufficient for the initial phases of feature analysis, because apps are both texts and tools, we apply the concept of affordances to help categorize how the features in the set implicitly conceptualize the problem the app is trying to solve. In particular, using the analytic framework of the mechanisms and conditions of affordances can offer researchers crucial insights on the limitations and assumptions of each type of feature. Moreover, feature analysis offers a way to apply the concept of affordances to a precise analysis of a large set of apps. And, our method's use of speculative design in the final phase offers researchers a creative yet practical way to see which potential features might be missing from the set of apps. This phase of feature analysis can help reveal which ideologies and assumptions could have been embedded in the set of apps but were not.

New apps designed to solve social problems are often presented as revolutionary tools that offer progress, convenience, and control over complex challenges and uncertain conditions. For example, Balsamo argues that the US was founded upon and generally remains committed to a belief in "the capability of information technologies to produce desired social changes, to reinvigorate an ideal of human community and to overcome misuses of power and political advantage" (2011, p. 346; see also: Carey, 1989). Feature analysis' contribution to these observations is that it helps answer questions about how this kind of technological solutionism (Morozov, 2014) works in specific cases. Feature analysis is the first method, to our knowledge, that allows researchers to systematically analyze a large set of apps for evidence of how ideology manifests in design.

The potential political value of feature analysis is that it can support efforts at intervention. Technological solutions to complex social problems—such as anti-rape apps—may be enticing and profitable, but they are rarely effective and are typically simplistic. Such solutions are also

often laden with the same dominant values that led to the problem in the first place. As Davis explains:

> [T]echnologies themselves embody human values and politics in their design, implementation, and use. The bad news is that this means technologies will, by default, reflect and reinforce existing inequalities. The good news is that the default is neither necessary nor inevitable. A sharp analytic tool, like the mechanisms and conditions framework, renders politics visible and pliable. (Davis, 2020, p. 15)

By examining the dangerous fruits of technological solutionism, feature analysis is one method that offers researchers a robust and concrete way to "render politics visible and pliable."

FUNDING STATEMENT AND ACKNOWLEDGMENTS

We thank the American Council of Learned Societies, the Hanse-Wissenschaftskolleg, and the Social Sciences and Humanities Research Council for their contributions to this project.

REFERENCES

Almeida, C., and Báscolo, E. (2006). Use of research results in policy decision-making, formulation, and implementation: A review of the literature. *Cadernos de Saúde Pública, 22*(Suppl.), S7-S19. https://doi.org/10.1590/S0102-311X2006001300002

Balsamo, A. M. (2011). *Designing culture: The technological imagination at work*. Duke University Press.

Bivens, R., and Haimson, O. L. (2016). Baking gender into social media design: How platforms shape categories for users and advertisers. *Social Media + Society, 2*(4), 1-12. https://doi.org/10.1177/2056305116672486

Bivens, R., and Hasinoff, A. A. (2018). Rape: Is there an app for that? An empirical analysis of the features of anti-rape apps. *Information, Communication & Society, 21*(8), 1050-1067. https://doi.org/10.1080/1369118X.2017.1309444

Brock, A. (2018). Critical technocultural discourse analysis. *New Media & Society, 20*(3), 1012-1030. https://doi.org/10.1177/1461444816677532

Buolamwini, J., and Gebru, T. (2018). Gender shades: Intersectional accuracy disparities in commercial gender classification. *Proceedings of Machine Learning Research: Conference on fairness, accountability and transparency, USA, 81*, 1–15.

Carey, J. W. (1989). *Communication as culture: Essays on media and society*. Unwin Hyman.

Cohn, M. (2017). "Lifetime issues:" Temporal relations of design and maintenance. *continent, 6*(1), 4-12. http://www.continentcontinent.cc/index.php/continent/article/view/272

Costanza-Chock, S. (2018). Design justice, AI, and escape from the matrix of domination. *Journal of Design and Science, 3.5*. https://doi.org/10.21428/96c8d426

Davis, J. L. (2020). *How artifacts afford: The power and politics of everyday things*. MIT Press.

Davis, J. L., and Chouinard, J. B. (2017). Theorizing affordances: From request to refuse. *Bulletin of Science, Technology & Society, 36*(4), 241-248. https://doi.org/10.1177/0270467617714944

Dieter, M., Gerlitz, C., Helmond, A., Tkacz, N., van der Vlist, F. N., and Weltevrede, E. (2019). Multi-situated app studies: Methods and propositions. *Social Media + Society, 5*(2), 1-15. https://doi.org/10.1177/2056305119846486

DiSalvo, C. (2012). Spectacles and tropes: Speculative design and contemporary food cultures. *The Fibreculture Journal* (20), 109-122.

Dunne, A., and Raby, F. (2013). *Speculative everything: Design, fiction, and social dreaming*. MIT Press.

Entman, R. M. (1993). Framing: Toward clarification of a fractured paradigm. *Journal of Communication, 43*(4), 51-58.

Eubanks, V. (2018). *Automating inequality: How high-tech tools profile, police, and punish the poor*. St. Martin's Press.

Evans, S. K., Pearce, K. E., Vitak, J., and Treem, J. W. (2017). Explicating affordances: A conceptual framework for understanding affordances in communication research. *Journal of Computer-Mediated Communication, 22*(1), 35-52. https://doi.org/10.1111/jcc4.12180

Fiesler, C., Lampe, C., and Bruckman, A. S. (2016). Reality and perception of copyright terms of service for online content creation, *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing*, USA, 1450–1461. https://doi.org/10.1145/2818048.2819931

Forlano, L., and Mathew, A. (2014). From design fiction to design friction: Speculative and participatory design of values-embedded urban technology. *Journal of Urban Technology, 21*(4), 7-24. https://doi.org/10.1080/10630732.2014.971525

Gerlitz, C., Helmond, A., van der Vlist, F. N., and Weltevrede, E. (2019). Regramming the platform: Infrastructural relations between apps and social media. *Computational Culture* (7). http://computationalculture.net/regramming-the-platform/

Grundy, Q. H., Wang, Z., and Bero, L. A. (2016). Challenges in assessing mobile health app quality: A systematic review of prevalent and

innovative methods. *American Journal of Preventive Medicine, 51*(6), 1051-1059. https://doi.org/https://doi.org/10.1016/j.amepre.2016.07.009

Hall, S., Critcher, C., Jefferson, T., Clarke, J., and Roberts, B. (1978). *Policing the crisis: Mugging, the state, and law and order*. Holmes & Meier.

Hasinoff, A. A. (2017). Where are you? Location tracking and the promise of child safety. *Television & New Media, 18*(6), 496-512. https://doi.org/10.1177/1527476416680450

Head, B. W. (2010). Reconsidering evidence-based policy: Key issues and challenges. *Policy and Society, 29*(2), 77-94. https://doi.org/10.1016/j.polsoc.2010.03.001

Krippendorff, K. (2012). *Content analysis: An introduction to its methodology*. Sage.

Lapowsky, I. (2014, Sept 5). This app is a 21st century rape whistle. *Wired*. https://www.wired.com/2014/09/circle-of-6/

Latour, B. (2005). *Reassembling the social: An introduction to actor-network-theory*. Oxford University Press.

Leurs, K. (2017). Feminist data studies: Using digital methods for ethical, reflexive and situated socio-cultural research. *Feminist Review, 115*(1), 130-154. https://doi.org/10.1057/s41305-017-0043-1

Light, B., Burgess, J., and Duguay, S. (2018). The walkthrough method: An approach to the study of apps. *New Media & Society, 20*(3), 881-900. https://doi.org/10.1177/1461444816675438

Lupton, D. (2014). Apps as Artefacts: Towards a critical perspective on mobile health and medical apps. *Societies, 4*(4), 606-622. http://www.mdpi.com/2075-4698/4/4/606

Lupton, D. (2015). Quantified sex: A critical analysis of sexual and reproductive self-tracking using apps *Culture, Health & Sexuality, 17*(4), 440-453. https://doi.org/10.1080/13691058.2014.920528

Magnet, S., and Rodgers, T. (2012). Stripping for the state: Whole body imaging technologies and the surveillance of othered bodies. *Feminist Media Studies, 12*(1), 101-118.

Martínez-Pérez, B., de la Torre-Díez, I., and López-Coronado, M. (2013). Mobile health applications for the most prevalent conditions by the World Health Organization: Review and analysis. *Journal of medical Internet research, 15*(6), e120. https://doi.org/10.2196/jmir.2600

Morozov, E. (2014). *To save everything, click here: The folly of technological solutionism*. PublicAffairs.

Morris, J. W., and Murray, S. (Eds.). (2018). *Appified: Culture in the age of apps*. University of Michigan Press.

Murray, S., and Ankerson, M. S. (2016). Lez takes time: Designing lesbian contact in geosocial networking apps. *Critical Studies in Media*

Communication, 33(1), 53-69. https://doi.org/10.1080/15295036.2015.1133921

National Network to End Domestic Violence. (no date). *App safety center*. Retrieved February 18, 2021 from https://www.techsafety.org/appsafetycenter

Neuendorf, K. A. (2017). *The content analysis guidebook* (2nd ed.). Sage.

Noble, S. U. (2018). *Algorithms of oppression: How search engines reinforce racism*. NYU Press. https://doi.org/10.2307/j.ctt1pwt9w5

Oliver, M. B. (1994). Portrayals of crime, race, and aggression in "reality-based" police shows: A content analysis. *Journal of Broadcasting & Electronic Media, 38*(2), 179-192. https://doi.org/10.1080/08838159409364255

Pasulka, N. (2012, April 6). Want to stop rape? There's an app for that. *Mother Jones*. https://www.motherjones.com/politics/2012/04/way-to-prevent-rape-app-iphone-apps-against-abuse/

Rogers, R. (2013). *Digital methods*. MIT press.

Schüll, N. (2018). LoseIt! Calorie tracking and the discipline of consumption. In J. W. Morris & S. Murray (Eds.), *Appified: Culture in the Age of Apps* (pp. 103-114). University of Michigan Press.

Slack, J. D., and Wise, J. M. (2005). *Culture + technology: A primer*. Peter Lang. http://www.loc.gov/catdir/toc/ecip0513/2005015547.html

Smock, A. D., Ellison, N. B., Lampe, C., and Wohn, D. Y. (2011). Facebook as a toolkit: A uses and gratification approach to unbundling feature use. *Computers in Human Behavior, 27*(6), 2322-2329. https://doi.org/https://doi.org/10.1016/j.chb.2011.07.011

Stanfill, M. (2015). The interface as discourse: The production of norms through web design. *New Media & Society, 17*(7), 1059-1074. https://doi.org/10.1177/1461444814520873

Suchman, L. (2011). Anthropological relocations and the limits of design. *Annual Review of Anthropology, 40*(1), 1-18. https://doi.org/10.1146/annurev.anthro.041608.105640