Advertising-based Measurement: A Platform of 7 Billion Mobile Devices

Mark D. Corner, Brian N. Levine, Omar Ismail, and Angela Upreti
College of Information and Computer Sciences
University of Massachusetts Amherst, MA, USA
mcorner,levine,oismail,aupreti@cs.umass.edu

ABSTRACT

The most important step in an empirical computer scientist's research is gathering sufficient real-world data to validate a system. Unfortunately, it is also one of the most time-consuming and expensive tasks: placing measurement tools in remote networks or end-clients requires one to marshal resources from different administrative domains, devices, populations, and countries. Often such efforts culminate in a trace that is deficient in multiple ways: a small set of test subjects, a short time frame, missing ground truth for device IDs, networking environments lacking in diversity and geographic spread, or highly biased sampling.

We present a method of addressing these challenges by leveraging the most open and globally accessible test and measurement platform: digital advertising. Digital advertising instantly provides a window into 7 billion devices spanning every county for an extremely low cost. We propose *Advertising as a Platform* (AaaP), an ad-based system to perform massive-scale mobile measurement studies. In contrast with measurements made by large media companies who own platforms, ad networks, and apps, we concentrate on the opportunities and challenges for researchers that are end-users of advertising systems. We evaluate a prototype system, discuss ethical guidelines, and demonstrate its use in four scenarios: IP2Geo databases, bandwidth measurement, energy management, and the identifiability of mobile users. We show the efficacy and ease-of-use of AaaP, and illuminate key challenges and the great promise of using AaaP to study a wide variety of mobile phenomena.

CCS CONCEPTS

• **Networks** → **Network measurement**; **Mobile networks**;

KEYWORDS

Mobile measurement, Mobile advertising

1 INTRODUCTION

Empirical research on mobile systems is a process that ideally involves the placement of measurement tools in varied domains, on a variety of devices, for users with diverse demographics, and in many countries. The reality for many studies is that some or all of these ideals fail. In this paper, we show that this challenge can be met by leveraging the most open and globally accessible test and measurement platform: digital advertising.

Digital advertising provides a window into virtually every Internet-connected device on the planet, instantly, at an extremely low cost. U.S. users are exposed to an estimated 100 to 1,300 ads *per day* [31, 58]. Advertisements not restricted demographically; for instance black and white American adults were equally likely to own a smartphone [65]. Ads are cheap, often less than $0.0005 per displayed ad (i.e., per *impression*). And ads possess the resources of a full browser. Accordingly, advertisements have access to some native sensors, such as the accelerometer and the state of the device's battery, without special privilege. Ads can fetch images and run javascript. Ads have access to a unique identifier and device location. And all of these capabilities are available without user interaction, such as clicking.

Such data is already available, collected, and analyzed by a variety of actors. Handset and platform creators have access to all data. A large media company, such as Google or Twitter, that distributes an SDK to thousands of publisher's has access to similar data. These parties all collect measurements as a matter of course. Outside of these organizations, researchers sit in a less privileged position. This paper concentrates on the opportunities for designing and executing experiments for researchers that are end-user of advertising systems.

Consider a researcher who requires bandwidth measurements at many geographic locations through multiple providers. It is highly unlikely that ISPs would provide that information, so there is no choice but to directly measure it. The researcher could travel and measure bandwidth using phones from each provider; e.g., [67]. Or the researcher could recruit users to install a bandwidth measurement application; e.g., [4, 32]. Both options are time consuming and expensive.

In contrast, by purchasing advertisements a researcher can gather data about bandwidth, with GPS information, by measuring how long a client takes to download an image. One million data points can be generated from $100 without recruitment, without leaving their desks, and without deployment. The infrastructure is already in place on more than 7 billion mobile devices spanning the globe [17] and permits experimentation and measurement at unprecedented scale and at diminutive costs.

To leverage the opportunities of **advertising-based measurement**, we have developed a system called *Advertising as a Platform (AaaP)*, which places ads on mobile devices and can measure a wide variety of phenomena. In contrast to measurements made by larger

MobiCom'17, , October 16–20, 2017, Snowbird, UT, USA.

ACM ISBN 978-1-4503-4916-1/17/10…$15.00
https://doi.org/.1145/3117811.3117844

media companies, we introduce and quantify the power of this paradigm from the perspective of a research scientist. We evaluate our approach along many dimensions, including ethics, scalability, cost, and efficiency. All told, for this paper, after a careful IRB process, we took measurements from more than 553,043 distinct devices on 276,214 IP addresses, with 991,485 ads — yet the total cost was less than $1000 and we set up minimal infrastructure. All countries are within AaaP's scope, but it can also target specific geographic areas. We gathered data from as much as 6% of the mobile devices in a small town in 20 days. AaaP can also gather repeated measurements from specific devices. In a week-long experiment that targeted ads to a specific set of 8,358 devices, AaaP gathered measurements from 16% of the devices at least once a day for the entire week. When an experiment requires even more granular results, AaaP could gather more than 10 data points in one day for 20% of devices.

While these results show that AaaP has many advantages in deployability and cost over *direct measurement*, it is also clear that there are many challenges. The first challenge is to examine the ethics of conducting ad-based experiments. Hence we prioritize this concern as Section 2, where we examine human subjects concerns and draw a boundary around acceptable experiments.

The remaining challenges are technical, and methods for confronting them are best demonstrated through a series of example experiments:

- **Ads provide inaccurate location information from devices.** Can AaaP measure and utilize location data of sufficient fidelity and reliability to be useful in experiments? We demonstrate how to discriminate reliable location information from estimated location and use it to examine the accuracy of a popular IP2Geo database. Further we show how data from advertisements can reduce error in geolocation by a factor of $53x$.
- **Advertising networks have access to exact data from native calls in mobile devices, but often return less specific information to advertisers.** Can AaaP overcome limited specificity in measurements, such as model information, when comparing performance of systems in situ? We show the results of an experiment to measure if the two chipsets available for the iPhone 7 affect bandwidth.
- **Ads provide only samples of phenomena.** As AaaP can only take measurements when the device is displaying an advertisement, it is inherently a sampling-based system. We demonstrate techniques for prediction and interpolation through an experiment that gathers battery and charge state information from devices.
- **Ground truth identities are sometimes unavailable and may change over time.** Advertising systems return a unique device identifier, but it is sometimes unavailable, such as when iOS is set to turn on "limit ad tracking". Fingerprinting techniques can overcome this mechanism, and we replicate previous work quantifying identifiability with very little effort.

We selected these examples because they span a broad range of mobile systems research, and the last two are reproductions of results that were originally conducted using methods other than AaaP (e.g., [18, 28]).

2 HUMAN SUBJECTS CONCERNS

Strict adherence to a set of ethical standards was of paramount importance to us during this work. Because the ethical issues raised by AaaP are not unprecedented, we consulted heavily with existing ethical guidelines. In particular, we followed the 2012 Menlo report [25], which is an update of the 1979 Belmont Report [62] that focuses on technology. Here we summarize the most relevant aspects of the Menlo report and related work.

The first concern is, does it qualify as human subjects research? As advertisements do not contain any personally identifiable information, they cannot be linked to a real-world identity, and such research is technically not subject to IRB approval. Other IRBs have reached the same conclusion for similar studies [18, 33, 39, 52].

Nonetheless, we asked for guidance from our board, which was very helpful in crafting protocols, especially those dealing with location information, under numbers 2016-3112 and 2016-3141. Obtaining certification of these protocols required a written proposal, several deeply technical discussions with members of the board, as well as an in-depth presentation to the full board to explain how advertisements work, what data was being collected, and the expected scientific gain. The board's purpose is to weigh these risk factors against the scientific gain to approve, modify, or reject a protocol; see [33]. The interactive, multi-round, in-person IRB process, allows for a deeper discussion of ethics than might be had during a non-interactive blind review process; see [19, 52].

The second concern is, do such experiments require explicit consent? U.S. federal guidelines permit studies that lack explicit consent under circumstances where the research carries no more than minimal risk for the participants, it is impractical to obtain written consent, and the probability and magnitude of harm or discomfort is not greater than that encountered in their daily life. These were the guidelines we followed in designing experiments for which we received IRB approval. (See [52] for a discussion on Internet-scale consent.)

For instance, we are not increasing the number of advertisements users see (another ad would be shown in its place) thus we are not creating additional discomfort. Another example is testing bandwidth with multi-megabyte files (see Section 6.2). We have studied mobile advertisements beyond this work and it is extremely common for ads to request high definition imagery, complex javascript, streaming video, and even fully playable mini-games. Ads can use as much as 79% of a user's bandwidth [16]. Users may not welcome ads using bandwidth, but AaaP ads do not create greater harm than typically encountered. We explored these and other issues with our board before receiving approval.

Even with community guidelines and IRB approval, some experiments can prove controversial during peer review. For example, though our IRB approved an AaaP-based study of censorship, we do not include one here. The community is split over the studies that we could replicate on a much larger scale [18, 19, 39, 52].

Another question is whether such advertisements fall outside of the terms of service of advertising platforms. The terms of service that we have found do not explicitly preclude the kinds of advertisements AaaP uses. Advertising networks typically prohibit *misrepresentation*, but the definition of misrepresentation is in a business

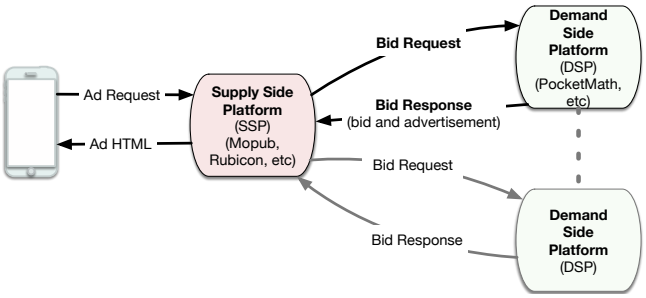

Figure 1: Auctions: Device requests ad from SSP. Each DSP bids, a winner is selected, and the ad HTML is shown.

sense (fake offers for products, impersonating user interface elements, etc.). Many ad tracking platforms will measure bandwidth, battery levels, etc. for the purposes of delivering different ads or digital fingerprinting, thus such uses are not unprecedented.

3 BACKGROUND

The two primary entities involved in digital advertising are the *publishers*, who show ads in their own web page or mobile app; and the *advertisers*, who purchase advertising space. For instance, a car company (an advertiser) buys advertising space from an online newspaper (a publisher) and the newspaper will show ads for the car company.

A large fraction of digital advertisements are bought through a process called *Real Time Bidding* (RTB). In RTB, both the publisher and the advertiser are intimately involved in showing ads. We show this process in Figure 1. When a device starts the publisher's app or website, a request is made to an RTB *Supply Side Platform* (SSP). The SSP then forwards all of the information it has about that ad opportunity to a large set of *Demand Side Platforms* (DSPs) that are acting on behalf of advertisers. Each of the DSPs bid for that one advertising impression and the SSP selects the highest bid and shows that DSP's ad. This process all takes place in less than 200ms and typical DSPs may receive several million opportunities per second. Note that very large publishers, such as Facebook or Twitter, typically do not use RTB to sell their own ad inventory.

To alleviate an advertiser having to act as a DSP, there are relatively easy to use *self-service* RTB DSP platforms. These self-service platforms allow advertisers to setup advertisements with targeting parameters and then collect data about their advertisements, in real-time, through two mechanisms: *macros* that are filled in with data available in the bid requests from the SSP; and data collected via an *ad-tag* using HTML and javascript of the advertiser's choosing. These two sources of data are complimentary in the context of our work.

Macros. Self-service platforms allow an advertiser to set up ad targeting and upload imagery for the ad (known as the *creative*) and configure the click destination. The self-service platform then formats a small HTML document containing the image with the click destination. This document is displayed in an embedded *WebView* in a mobile app or an *iframe* on a website.

As shown in Figure 2, self-service DSPs also allow an advertiser to configure an impression beacon that will be inserted into the advertising document. This tracking pixel or *beacon* is a URL for a

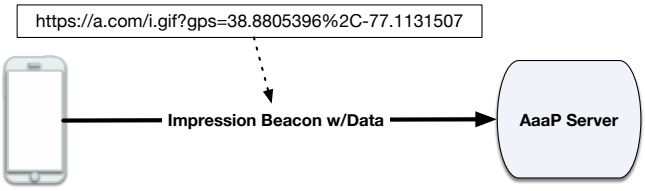

Figure 2: Impression Beacons: The HTML contains an impression beacon that requests additional javascript and HTML from the AAAP server, adding to the Ad DOM.

`page_url`	The URL of the page
`device_identifier`	Value of device identifier, may be SHA1-hashed
`device_isp`	Name of ISP
`device_model`	Device model
`device_os`	Device operating system (i.e. "ios", "android")
`os_version`	Device operating system version number.
`ip_add`	IP address of the ad viewer
`source_id`	Plain text name for the site or app
`timestamp`	UNIX timestamp of RTB auction
`exchange`	Name of RTB exchange where impression is won
`gps`	GPS coordinates, up to 6 decimals
`user_agent`	User-agent string for the user device
`campaign_id`	Campaign identification number
`gender`	Gender of the ad viewer "male" and "female"

Figure 3: Sample macros from publisher platforms that can recover information via advertisements.

1×1 invisible image served by the advertiser's servers, which will be requested by the device when the ad is shown. The impression beacon may also contain *macros* that the DSP substitutes with real values when the ad is shown using the data contained in the bid request from the publisher's app. For instance, if the bid request contains a GPS location determined by the publisher's app and sent to the SSP and then the DSP, then the DSP can substitute this data into the ad markup. A beacon URL such as https://advertiser.com/i.gif?gps={GPS} will cause the device's browser or app to request https://advertiser.com/i.gif?gps=38.8805396%2C-77.1131507 when the ad is shown. Through impression beacons, an advertiser can gather a multitude of information. Figure 3 shows a sample list of macros available on the DSP platform we have been using.

Ad Tags. A more complex form of interaction between the DSP and the advertiser uses *ad tags*. This process is shown in Figure 4. Ad tags are a javascript HTML tag that the advertiser gives to the DSP who inserts it into the markup causing the mobile device to retrieve and run the advertiser's javascript. Thus, instead of adding a single beacon URL, the advertiser can control the entire ad markup. Anything that is available through javascript is available to the advertiser. For instance, through a javascript callback on the image load, one can collect the amount of time the client took to download the image, thus computing bandwidth. Javascript and HTML5 provide a multitude of other information; e.g., the Android Browser provides the Battery Status API.

Note that the impression alone invokes javascript and gathers data; no clicking is needed.

Ad Targeting. Through RTB platforms, one can target ads on a variety of criteria: country, publisher showing the ad, IP addresses and ranges, and *advertising identifiers*. Both iOS and Android use an advertising identifier that is a random UUID shared across all

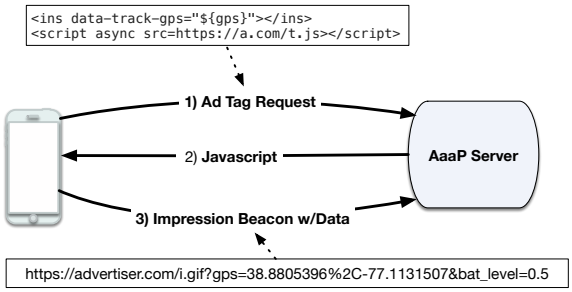

Figure 4: Ad Tags: HTML contains a *tag* that requests additional javascript and HTML from the AAAP server. Any data gathered (e.g. battery level) by the javascript is added to the impression beacon URL for our AAAP server.

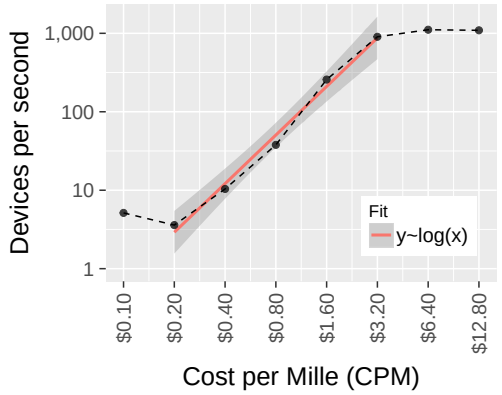

Figure 5: The relationship between the bid price (CPM=dollars/1000 impressions) and the number of new devices per second an experiment can measure. (Log-log scale. Points represent observed data. Fit includes only CPMs greater than $0.10 and less than $6.40.) Total cost = $95.58; Impressions = 19,764; Efficiency = 100%

apps on the device (though not available in the native browsers MobileSafari and Chrome). Targeting specific advertising identifiers enables our system to collect longitudinal data by targeting ads to specific devices. For instance, if we are interested in the behavior of devices assigned a certain IP address or are located in a certain geographic area, we can collect research data even when they change IP addresses or move from that location. We can answer questions such as, "Among mobile devices that are observed in a particular location, in which other locations where they also observed?".

4 IMPLEMENTATION

We have constructed a functioning AaaP service in Ruby on Rails, running on Amazon Web Services (AWS). Our prototype places ads, serves ad tags, gathers data via macros and javascript, and analyzes the results. Our AaaP service runs experiments based on a script, including a cron system for making regular changes to the experiment, such as increasing a bid or adding advertising identifiers to the set of targeted devices.

AaaP controls experiments programmatically through APIs provided by the self-service DSP. AaaP adjusts the targeting of advertisements, controls the starting and stopping times, and changes the amount the experiment is bidding for an advertisement. The system also uploads lists of advertising identifiers to target.

AaaP works through the API of a self-service Real-Time Bidding platform to place ads. As the self-service platform handles the bidding, the AaaP server only interacts with advertisements when an ad impression occurs (i.e., only when the ad is displayed on a device). Using a self-service platform avoids a great deal of work and expense that would be involved in our building of an RTB bidder capable of handling over 500k bid requests per second.

Our AaaP service uses macros and ad tags as explained in Section 3 and our custom javascript fetches and displays the creative and sets up the DOM for the advertisement and all measurements. We have developed a number of javascript functions that gather data and post it back to our server to be stored in a database and later analyzed. The system is scalable to large amounts of traffic and permits easy post-experiment analysis.

Because our system is based on a self-service platform, our server's work increases with the number of impressions per second, rather than the number of bids per second. Therefore, we can support many experiments with a handful of lightweight machines.

The current system uses an AWS Elastic Load Balancer, one to four *t2.micro* EC2 servers in an autoscale group, one *t1.small* RDS Postgresql database, and the AWS Cloudfront CDN, backed by S3. The entire backend system costs less than $100 per month to run, well within the reach of most researchers.

5 CHARACTERIZATION

In this section, we characterize the efficacy, cost, and granularity of data that is measurable using AaaP. For instance, if an experiment requires single measurements from a multitude of unique devices, how fast can those devices be discovered and at what cost? Or if an experiment requires not just one, but many samples from particular devices, typically how long is it before they are seen again? The driving factors for all the questions we pose in this section are the *cost* and *time* incurred.

We have taken a simple approach in this section, bidding only on the smallest ad size available (320×50 pixels) and only on one of the self-service platform's Supply Side Platform (SSP) partners, Mopub, though there are more than a dozen additional exchanges such as Rubicon, Nexage, and Smaato.

5.1 Device Discovery

Question 1. At what rate can AaaP obtain at least one measurement from unique mobile devices, given an offered spend? Without other constraints, we have found that a massive amount of opportunity is available, even at very low bids. The minimum RTB bid is approximately $0.10 Cost Per Mille (CPM) — that is, cost per one thousand impressions — which is $0.0001 per ad shown. To provide a picture of the number of unique devices (as defined by unique advertising identifiers), we bid different amounts and measured the number of unseen and unique devices we could discover. Unfortunately, even at low bids, the scale is so massive that we used a sampling approach to estimate the actual scale. Note, our goal isn't to measure the exact amount of traffic available at different costs as it is highly dependent on all of the targeting factors, including: ad size, ad exchange, country, IP address targeting, time of day or day of week, etc.

We sampled one out of every 10,000 IP addresses in the US by targeting a random selection of 629 class C subnets. These subnets may not have any mobile devices, but we sampled uniformly at random, and should therefore be reasonably representative of activity on US IP addresses. We bid from a set of CPMs at random for 10 minutes at a time throughout the day to eliminate diurnal effects and measured the number of unique mobile advertising identifiers. We then scaled the results by 10,000 to determine the expected rate at which we could gather data. The results are shown in Figure 5 and covers two weeks of experimentation and approximately 20K impressions. As the figure shows, without constraints on where to place the ad, we were able to discover about 5 new devices *per second* at a $0.10 CPM. The results for $0.10 are slightly lower than those at $0.20. We do not know the exact reason, but we suspect it is either due to the higher variance at lower traffic levels or due to market effects where lower priced traffic is actually less desirable due to quality.

In this and all subsequent figures we report the number of impressions, the cost, the number of unique devices, and efficiency (percentage of impressions used to produce the plot). Note that sometimes the data is reused across plots. For example, we measured battery charge and bandwidth with the same impressions. In sum, we spent approximately $400 on more than 500k ads to produce this paper.

5.2 Geographic Fencing

Question 2. What is the cardinality of available devices in a limited geographic area? Many experiments can be conducted when accurate location information is available, such as studies of context-based behavior and local bandwidth conditions. Location information is often passed from mobile applications to SSPs then to bidders. Our self-service provider also allows us to target geographic areas (called *fences*). The fencing is often based on an IP2Geo database which can lead to errors; fortunately, we can often get highly accurate locations in a macro field so we can filter all results by that location. Additionally, in the U.S. there is a mismatch between the name of a city in a postal address, which is used in targeting, and the administrative boundaries of a town. For instance, many postal address are labeled "Buffalo, NY" that are not within the administrative boundaries of Buffalo.

We conducted two experiments: *(a)* 70 days in Amherst, MA; and *(b)* 20 days in Amherst, MA, Charlottesville, VA, and New City, NY starting on a later date. Amherst, MA is a town of approximately 60k residents and 30k students; Charlottesville, VA is a town of 40k residents and 20k students; and New City, NY has a population of 30k residents. We limited the advertisement to exactly one impression for each unique device. We then filtered the results to only permit data that: *(i)* included accurate location information, which is detectable on the SSP we used by the macro field used, and *(ii)* truly within each city using *PostGIS* and *OpenStreetMap's* administrative boundaries for the town. The experiment used a CPM bid of $2 and the resulting cost of the experiment was $169.73, including impressions outside of the cities. The results of the experiment are shown in Figure 6. Due to the mismatch in addresses and boundaries, and missing GPS information, the overall efficiency was 14%.

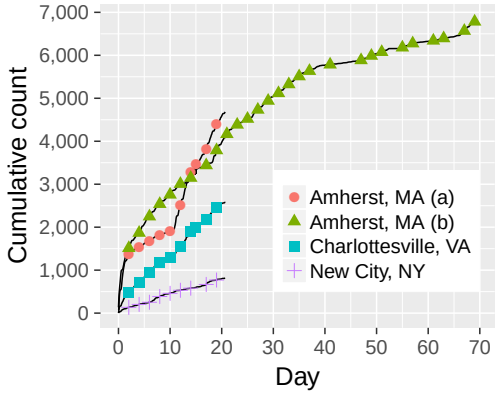

Figure 6: The rate at which AaaP can discover distinct devices in a limited geographic area. Total cost = $169.73; Impressions = 82,860; Efficiency = 14%. Efficiency is low as targeting is based on city name, not administrative boundaries and impressions lacking accurate location information are thrown out. (Points are a visual guide for legend; lines represent all data.)

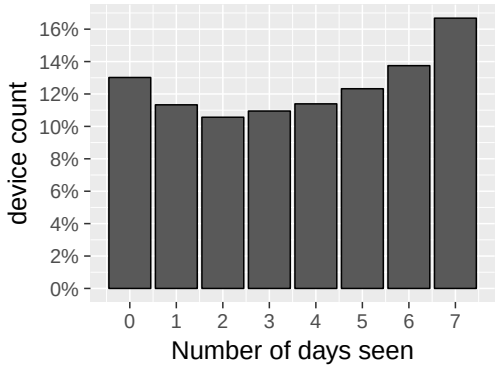

Figure 7: How often AaaP sees the same devices. Over a period of 7 days we added devices we saw to a target list. This graph shows the number of days following the device's first day that we saw them again. For instance we saw 50% of devices on at least 4 of the days. Total cost = $74.49; Impressions = 37,118; Efficiency = 94%. Imperfect impression cap per device leads to lowered efficiency.

The experiment demonstrates that, as expected, after an initial burst of finding new devices, new devices are discovered at a steady rate with returns diminishing after two months. Increases in this rate are possible by targeting more RTB exchanges (provided that we can find exchanges that provide honest location data) as well as targeting more ad sizes (factoring in that larger ad sizes are more expensive). There is also a burst of new devices found starting at day 11 in experiment (b). This occurred after US "Black Friday", a large shopping day and it is likely that students returned to town with new devices. This also occurred shortly after the release of a new iPhone.

As a very rough estimate, if we assume that Amherst, MA has about as many devices as people (60k), then we reached approximately 6% of the devices after 20 days and 12% of the devices after 60 days. The result is impressive considering how little effort is involved compared to any other methodology.

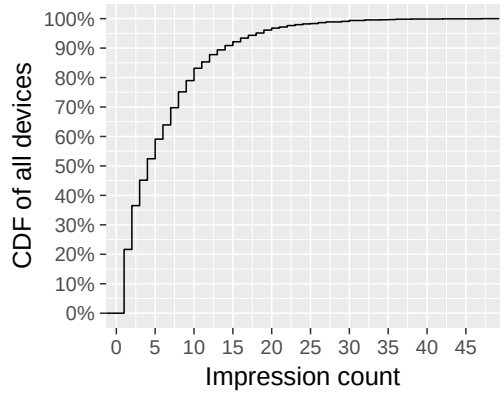

Figure 8: The count of impressions per device, as a CDF, when place a cap of 40 impressions per day, per device. Some devices see many, many ads in a short period, permitting high fidelity traces. Total cost = $12.88; Impressions = 9,921; Efficiency = 100%.

5.3 Repeat Observations

Question 3. With what frequency are devices observed multiple times? Experimenters may want to take multiple measurements from particular devices to see a pattern of behavior. Some devices will be seen frequently and others very infrequently. We measured how often devices in our Amherst, MA study were seen in a week. We limited it to one impression per day, per device, and targeted all of the devices that appeared in Amherst, MA, which was 8,358 devices . From that data we extracted the fraction of days we saw each device. The results are shown in Figure 7.

The experiment shows that more than half of the devices were seen on at least 4 of the days and 16% of the devices were seen all 7 days. This means that a large fraction of devices can provide measurements on a large fraction of days.

Experimenters may also be interested in even higher fidelity data from devices that appear very frequently. For instance if a device is contained in a traveling vehicle, would it be possible to get a continuous stream of locations? We conducted a similar experiment on approximately 10k Android devices (we reuse these results as part of section 6.3), but allowed 10 impressions per device during each of 4 six-hour periods in a day. The cumulative distribution function of the impressions seen is shown in Figure 8. While most devices only saw a few ads, many devices saw large numbers of ads, with 20% of devices seeing at least 10 ads and several devices seeing the maximum of 40. To collect high fidelity data, it may be necessary to target larger numbers of devices and filter the results accordingly for devices that present that opportunity. How such filtering influences sampling bias depends on what is measured. Optimizing the cost of discovery we leave as future work.

6 CHALLENGES AND TECHNIQUES

While AaaP exhibits superior scale, deployability, and cost when compared to direct measurement, it introduces new challenges, which we detail in this section. In this section, we demonstrate techniques for mitigating four primary deficiencies related to measuring location, bandwidth, batteries, and device identifiability. Through four studies, we address these issues and show how AaaP is relevant to the broad mobile systems topics of geography, networking, energy management, and security and privacy. Our intention is not to fully explore each, but to demonstrate how these challenges can be addressed.

6.1 Unreliable Location Information

Challenge 1: Location information provided by advertisements may be inaccurate. When an app is permitted access to a device's location APIs, advertisers learn a device's precise location. Otherwise, coarse estimates of the device location are sourced from IP-to-location databases (IP2Geo). Through a measurement study, we show that such databases are inaccurate for mobile device, as past work has found in small scale studies [10, 57, 70]. And we show that with careful filtering, ads with precise location information can be leveraged by experimenters to greatly improve existing geolocation services.

RTB standards define fields that pass the source of location information to the advertiser, so that the quality may be considered. Unfortunately, the self-service advertising system we used provided no such labeling. Through trial and error using ads targeted at our own devices on a single SSP (Mopub) with location access permitted or denied, we determined which apps return high-quality location information. The simple rule we discovered and verified is that when a devices permits an app to access location, the *lat* and *lon* impression macros receive the correct location (and the *gps* macro). Without permission, those two macros are null, and a IP2Geo-based location is supplied by the *gps* macro only. On other exchanges, the mapping can be different, resulting in some manual work to determine the origin of the location.

The most reliable apps appear in Table 1, along with the fraction of ads that were returned in our study with non-null *lat/lon* macros. Not surprisingly, apps that require a device's location to work, like casual dating and weather apps, are among the list. The Weather Channel app and TuneIn self-report having 50 million [3] and 60 million [9] monthly active users, respectively. Whereas Grindr, Growlr, and MeetMe self-report relatively smaller audiences of 5 million [38], 4 million [36], and 2 million [2] monthly active users, respectively.

By targeting the publishers shown in Figure 1, and restricting advertisements to non-cellular US-based IPs, we constructed our own geolocation lookup service for each IP we observed with ads. We omitted cellular IPs because they are shared across many devices and wide geographic areas [70]. In total we measured device-native locations for 112k distinct IP addresses and 350k devices. We used these locations as ground-truth as they came directly from the device; in truth, they are only as accurate as the method used by device's location API (e.g., GPS, Cellular).

For each IP address k, each of D devices sees n ads that generate a set of n locations $L_{i,k} = (x_1, y_1), \ldots, (x_n, y_n)$. We calculate the centroid for the dth device seen at k as

$$C_{k,d} = (\frac{x_1 + \cdots + x_n}{n}, \frac{y_1 + \cdots + y_n}{n}) = (x_d, y_d). \quad (1)$$

We then estimate the location of k as C_k, the centroid of all of the D centroids as

$$C_k = (\frac{x_1 + \cdots + x_D}{n}, \frac{y_1 + \cdots + y_D}{n}) = (x_k, y_k). \quad (2)$$

App Name	Perc.	Notes
GROWLr Android	99%	Required
GROWLr iOS	99	Required
Grindr iOS	98	Required
Weather Mood	97	For weather
Grindr Android	96	Required
The Weather Channel iOS	84	For weather
My Clock Android	81	For weather
The Weather Channel Android	78	For weather
MeetMe: Chat Android	76	Encouraged
TextNow Android	71	Asks at install
My Clock Free Android	67	For weather
TuneIn Radio iOS	61	Local stations
TuneIn Android	61	Local stations
MB3: Mixer Box	61	Asks at install

Table 1: Apps that provide accurate GPS information based on non-null *lat/lon* macros. The percentage of devices (and thus users who gave the app permission) with access to device-native location information is shown.

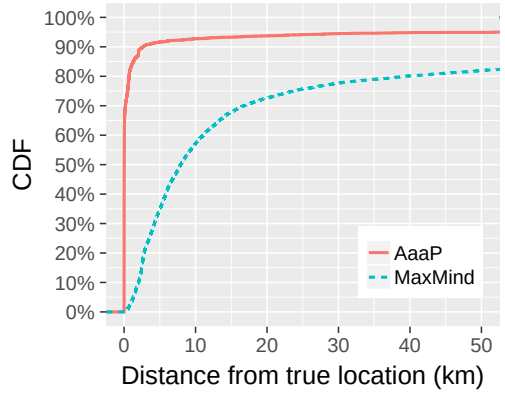

Figure 9: AaaP provides significantly lower error from true geographic location than using a commercial IP2Geo database. However, not all IPs can be geolocated with AaaP. Total cost = $40; Impressions = 396,382 ; Efficiency = 68%.

For each trial we left one device l out, and computed the Haversine distance between $C_{k\setminus l}$ and $C_{k,l}$; and also the distance between MaxMind's geographic location and $C_{k,l}$. We did not include any IP addresses where there weren't at least two devices as this test requires one to leave out and one to test.

As Figure 9 shows, using ads is tremendously more accurate than MaxMind: our AaaP approach is within 0.15 km 50% of the time, and within 1.0 km 85% of the time. In contrast, MaxMind is off by at least 8.0 km 50% of the time, and within only 40 km 85% of the time. The median error for AaaP is a 53x improvement. We investigated the places where AaaP made large errors and found devices from drastically different locations using the same IP address. We believe this is caused by devices using proxy/VPN servers at those IPs.

The union of the tens of millions of users of weather and dating apps (as well as the many, many other publishers we did not target), provides high levels of coverage of IPs, but ads will not be available at every IP. The best of use of such data is to augment, not replace, databases such as MaxMind or to provide high resolution location information to other experiments.

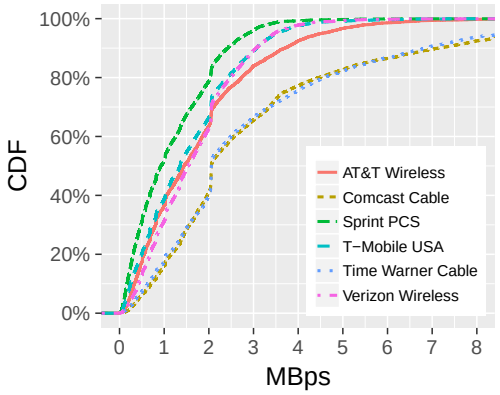

Figure 10: The bandwidth we observed for US-based carriers. Total cost = $26.72; Impressions = 45,581; Unique IPs: 21,193; Efficiency = 46%.

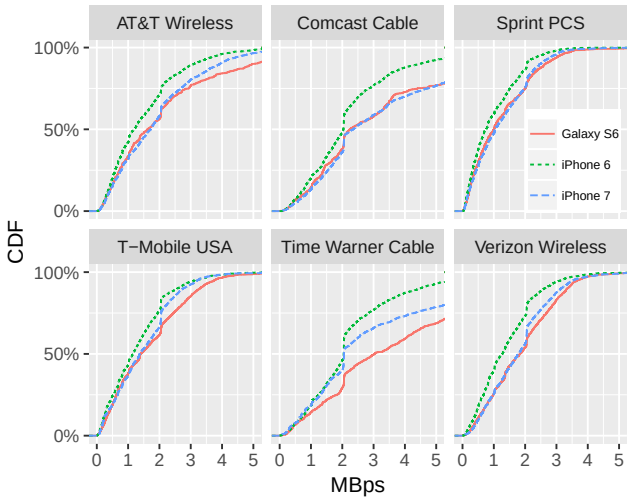

Figure 11: The bandwidth we observed for US carriers, per device model. (Uses same data as Figure 10.)

6.2 Abstracted Information and Bandwidth

Challenge 2: Information in advertisements is abstracted and aggregated. Similarly to location information, other, detailed information about a device may be abstracted by different actors between the experimenter and the end device's native components. For instance, in-app advertisements have access to the model of the device via native components and ad macros provided by the native components of the ad library. However, the model supplied via ad-macros for iOS yields the model of the phone at the level of iPhone 7 or iPhone 5s, but does not reveal the exact Apple model number, such as A1453 versus A1457, which are both iPhone 5s and would be available through direct measurement.

Even with these challenges we sought to measure mobile bandwidth provided to mobile devices, both over WiFi and cellular by timing how long it takes to fetch data from a server. In contrast with other techniques, AaaP measurements are precisely geolocated and attributed to mobile carriers and can scale quickly to millions of end-hosts.

There are a great number of questions that can be answered about bandwidth, but we chose to study the Apple iPhone 7. Popular press has highlighted the fact that Apple chose to produce two different versions of the iPhone 7 for the US market, one with an Intel modem and one with a Qualcomm modem. The Intel version did not support CDMA and was sold primarily through US carriers AT&T and T-Mobile and the Qualcomm version that supports GSM and CDMA was sold through Verizon and Sprint. Independent testing has shown that there are performance differences between these two models [1], but these are limited trials performed in a laboratory-like setting. We wanted to see what the practical effects of these differences are by studying the bandwidth achieved by in situ phones. Due to the abstracted information supplied to advertisements we do not know which model an ad is shown on, but we can detect which carrier they are using. Using this we can assume that the majority of devices on AT&T are of the Intel variant.

Our bandwidth test consisted of having the advertisement serially download three images sized 0.5,1, and 4 MB. These files are typical of the larger files downloaded on phones and are not dissimilar in size to some video advertisements. We found that 1MB and 4MB files resulted in about the same download speeds, suggesting both overcame startup effects. The files: *(i)* consisted of compressed image data to eliminate the effects of compressing proxies, *(ii)* were placed on a well-distributed CDN (AWS Cloudfront), and *(iii)* used a cache-busting query parameter to eliminate the effects of caching proxies. We targeted ads to iPhone7s in 7 major US cities to give a more consistent environment to the measurements. We also measured the bandwidth achieved by phones on two major US cable Internet ISPs, Comcast and Time Warner, which demonstrate an upper bound on the performance of downloading files of these sizes. The results of this test are shown in Figure 10.

The results show that AT&T consistently provides better performance to iPhone 7 devices than does Verizon. (We noted the performance bump at 2MB for all carriers and model and believe it was caused by the CDN.) If anything the Intel modem could be faster than the Qualcomm one, however given the limitations of targeting we cannot completely separate the effects of the ISP from the effects of the phone. If we were to bring the Qualcomm version to AT&T, would we achieve even higher bandwidths than the average because AT&T can achieve higher speeds than Verizon? To help answer this question, we repeated the test with the iPhone 6, which does not have the same bifurcation in chipsets and the Galaxy S6 to compare with Android. The results are shown in Figure 11 and demonstrate the same ordering of carriers. This provides strong evidence that the performance gap shown in the iPhone 7 experiment is largely due to differences in the network itself and not the device — consumers should feel comfortable purchasing either.

6.3 Sampled Data and Batteries

Challenge 3: AaaP can only collect data when an advertisement is shown, leading to varying numbers and irregular intervals of samples for each device. One of the advantages to AaaP is that it can target a subset of devices based on their advertising identifiers and bid relatively high to capture more data per user. However, even if AaaP bids very high, it can collect data only when the ad is being shown. AaaP experiments can interpolate missing data and predict values based on past measurements.

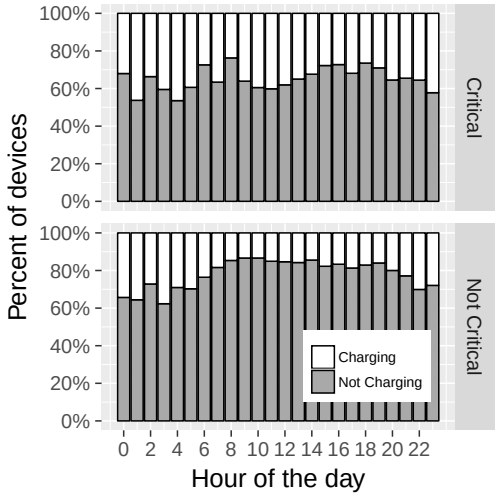

Figure 12: Battery level data we collected from Android devices by hour of the day, localized to their time zone. Critical battery level is defined as <20%. Impressions=142,706 ads; Unique devices=38,265; Total cost=Approximately $75 (reused data from other experiments). Efficiency=100%.

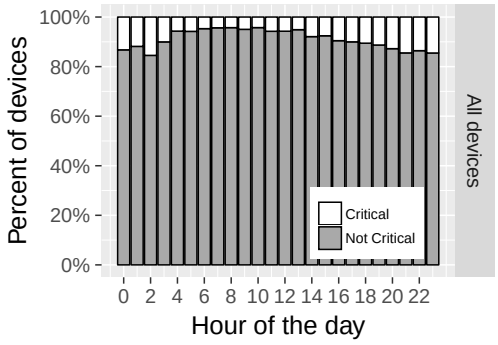

Figure 13: Most mobile phone batteries are not allowed to become critical, but the most frequent times are in late evening, among devices shown ads. (Same data as Fig.12.)

To demonstrate this technique we deliberately picked a difficult, yet important example: battery state and the time at which the device was plugged in or unplugged. Using javascript and HTML5 it is possible to query the state of a mobile device's battery using the Battery Status API. This API is currently supported on Android devices and some laptop browsers. We can query *(i)* the fraction of the battery remaining, *(ii)* whether the device is charging, and the *(iii)* discharging and *(iv)* charging rates. We used only the battery remaining and charging state — the two rates *(iii)* and *(iv)* are often incorrect or unavailable. Results from this data are shown in Figures 12 and 13.

The results show a clear diurnal pattern with the majority of mobile device use occurring in the afternoon and evening. One can also see that it is much more common for devices to be plugged in when the battery is above 20% and a small proportion of devices are used when the battery is below 20%, though this is most likely in the

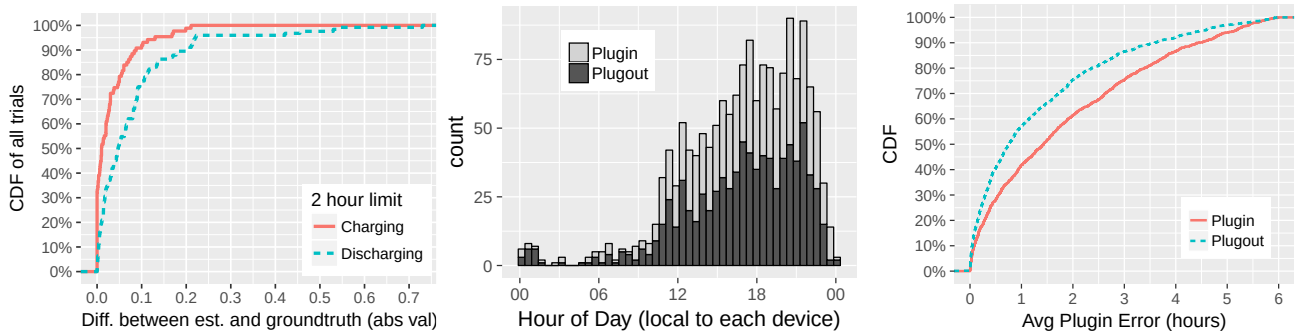

Figure 14: (Left) Error predicting battery level using measured ground truth. (Middle) Estimation of plug-in/unplug. (Right) Estimated error from using sampled data. Impressions=86,389; Unique devices=7,613; Cost=$80; Efficiency=48% (some devices show no plug in or out).

evening. However, this experiment brings up one of the unavoidable biases in our methodology. The data is solely derived from devices when they are being used and we have no ability to gather data while the device is idle (for instance while the user is asleep or the device is in a pocket.). In exchange for massive experimental scale, we incur some bias in the data and our conclusions must be couched in those biases.

Our goal is to infer information about what happened with the devices in between, or after measurements. To that end, we can use the pattern of a device's measurement to predict the state of the battery into the future and to estimate when the device may have been plugged in or unplugged. To do that we conducted an experiment using AaaP where we targeted devices for two days to gather battery information including the fraction the battery is charged ([0,1.0]) and the charging state (true or false). We targeted ads to 38k devices over two days. While we cannot ask the RTB platform to pace impressions, we did create four advertising campaigns per day, each one 6-hours long, and limited each to show up to 10 advertisements to each device.

Level Prediction. We can accurately predict the future state of the battery a short time into the future using a simple linear model of battery behavior. For each device, we have a sequence of observations; the ith observation $b_i = (t_i, c_i, s_i)$ has timestamp t_i, battery charge c_i, and charging state s_i. We then segment the sequence into observations that are of the same charging state. We then compute the linear regression over the segment (i.e., the values b_0 and b_1 such that $c_k = b_0 + b_1 t_k$ has the least sum square error for all observations, $k = 0 \ldots$, in the segment).

We measure accuracy by removing the last point in each segment (for segments with at least 3 points) and measure the error as the fraction of the battery the prediction is off by. This last, removed, point is the ground truth for measuring the prediction error. If the linear model intersects 100% or 0% battery, then we assume the battery stays in that state. We limit results to predictions less than two hours into the future, as accuracy degrades rapidly otherwise.

The results, shown in Figure 14(Left), demonstrate that within two hours, 90% of the time a linear model can predict the state of the battery to within 10% of the battery capacity when charging, and within 20% when discharging. The errors result from our use of a simple model; in practice, batteries behavior is more complicated. In our measurements we observed battery levels decreasing even while charing (perhaps due to high CPU activity), and devices that discharge much more rapidly than typical (perhaps due to faulty batteries).

Plug-in times. These same measurements also reveal when devices are plugged in or are operating on battery. While ads are not active long enough to usefully measure the exact time when the device was plugged in, AaaP can book-end such times by looking for points where the device was plugged in and then later not-plugged in, and vice versa. During this period, the state of charging must have changed *at least* once — we assume it was the midpoint. By excluding periods longer than 2 hours, we can estimate the histogram of when devices around the world were plugged in or unplugged, shown as Figure 14(Middle) and localized to the users' timezones.

Figure 14(Right) shows the error for this estimate. The maximum error would be the full duration of each period; since we use the midpoint as the plug-in (or -out) time, we assume the average error of half the period. Under this assumption, about 50% of our plugin periods are within 60 minutes, and 50% of our unplugged periods are within 90 minutes. (Doubling the minutes gives the maximum error.)

Prediction and interpolation are powerful tools for compensating for AaaP's sampling. More is possible: e.g., AaaP can measure location or the context of a user, such as using Wi-Fi or a particular application, take later measurements, and interpolate behaviors and movements that occurred between those points. All sampling-based systems make such inferences, and AaaP can leverage any such algorithms.

6.4 Longitudinal Data and Identifiability

Challenge 4: Solid, longitudinal, identifiers are not always available in advertisements. We heavily use advertising identifiers to target devices to focus data collection and to establish which measurements came from the same device, for instance when taking a trace of battery levels. However *(i)* some publishers apply a one-way hash to advertising identifiers before forwarding them to the SSP, *(ii)* advertising identifiers can be rotated through the operating systems settings, and *(iii)* for advertisements on the web we can apply cookies and read them, but the cookies are only available when the impression is shown, not when targeting. In AaaP we

can deal with *(i)* by "unhashing" the identifiers and *(ii)* and *(iii)* by re-identifying devices based on *digital fingerprinting* techniques.

Approximately 17k data points (7% of data) that we received carried a hashed device identifier. If overlooked, we might attribute data to two different devices when there is actually only one. To disambiguate devices, AaaP unhashes the identifiers by creating a dictionary of all advertising identifiers it sees, and hashes them with both MD5 and SHA-1. Using the dictionary we unhashed 8.4k devices (49% of the 17k) and use this identity in our other experiments. The remaining 51% remain hashed; and while none of them are the same as unhashed device identifiers, it is possible that some are different hashes of the same identifier, though this is less likely.

Other sources of ambiguity arise because devices can be configured to block their advertising identifier from being used or it can be rotated. When "limit ad tracking" is set in iOS, the advertising identifier is all zeros and the do not track (DNT) flag is set; we observed that no more than 7% of iOS devices had enabled this feature.[1]

However, when an identifier is reset or masked, devices can be re-identified based on persistent characteristics of their device or its connection to the network, e.g., device model and IP address. Many studies have been performed to measure identifiability from device metadata. For example, the Javascript AudioContext API can be leveraged to return a signature (without actually playing sound) [72]. To measure the entropy of this technique, the authors of a recent study created a web page and recruited visitors with 18,500 distinct cookies, resulting in 713 distinct fingerprints. All told, they measured 5.4 bits of entropy in the audio context. Other privacy researchers have also built javascript libraries [68, 77] and websites [15, 26] to expose as many fingerprinting techniques as possible. And most new capabilities added to browsers often engender a discussion of privacy [42].

To demonstrate the efficiency of our platform we set up a experiment — using ads rather than a web page or new app — that measured several features available to advertisers. Several features are already returned by ad impressions: IP address, timezone, GPS and/or IP-based geo-location, model, and OS version. As a simple method of aggregating GPS information, we used Zip Code. In our ads, we used the same javascript as the previous study [28]. Our goal was to repeat the experiment and quantify the identifiability of these features, using the advertising identifier as ground truth.

Building and deploying the ad code took only hours. We placed 142,592 ads on 40,001 distinct devices, and measured the entropy of features alone and in combination. For example, we gathered 728 distinct audio signatures, with a measured entropy of 5.5 bits, which is similar to the previous study [28]. In comparison, the entropy of the OS version is 2.8 bits; device model is 7.5 bits; and IP address is 16.7 bits of entropy. Combinations are more powerful. For example, an IP and audio signature is 20.4 bits; and the combination of ISP, Model, TZ, IP, Audio, OS version, Zip code was the highest at 22.8 bits. Obviously, some features are overlapping in terms of information.

To gain additional clarity, we ran a series of experiments that quantified device identifiability given these features and a decision tree classifier.[2] We performed 66 trials with a 80%/20% train and test split for each feature/combination. The advertising identifier forms the ground-truth for the experiment. Accuracy results for a variety of non-overlapping features are shown in Figure 15 as boxplots. There exists some controls over IP address (by using Tor) and over geographic information (by turning of GPS and by using Tor to thwart geo-location). Therefore, the results are show in groupings: neither geographic nor IP address features available; geographic available but not IP address; both geographic and IP address information available. In this study, without IP and geographic information, the most accurate identification was fairly low at 15% from a combination of audio signature, device model, ISP, and timezone. With the addition of geographic information (making timezone obvious), the combination of audio signature, device model, ISP, and zip code is 34% accurate. Finally, with IP information and without zip code, we see that the combination of the IP and model is the most identifying; adding audio as a third component actually lowers accuracy. This shows that in many cases AaaP can collect longitudinal data even without the use of platform supplied identifiers.

7 LIMITATIONS

AaaP is not appropriate for all experiments. Particularly, when compared to native applications, AaaP suffers from two limitations: *(i)* the uncontrolled nature of the platform, and *(ii)* the substrate is a web browser without direct access to native APIs.

AaaP cannot sample with a precise rate nor interact with users who never use applications that show advertisements—by its nature AaaP can only gather data when advertisements can be purchased for a device. While we have provided some guidance in the paper, the individual experiment will dictate the statistical conclusions. What AaaP does provide is a population of devices orders of magnitude larger than any research experiment based on native apps, with a very low level of selection bias, which we believe can often counteract the lack of sampling control.

AaaP is also limited to the capabilities provided by browser implementations, outlined in Figure 3. There is always a tension between the availability of data and the scale of an experiment, and researchers will naturally pick the substrate that provides the required data at the largest scale. AaaP exists in a *hierarchy* of environments, starting from lab-bench experiments that have direct access to hardware; to rooted devices that remove sandboxing restrictions and yield access private APIs; to native applications that have access to a subset of that information; to advertising that has access to a combination of native and browser information; and finally a pure browser environment. For the same level of effort, each method provides less access, but perhaps greater scale. And all methods are complimentary, with AaaP providing some of the broadest and most diverse measurements that are easiest to gather. We suggest that AaaP be an experimenter's first step; and if needed, code deployed natively on the device as a second step.

[1] Specifically, we placed ads on 43,620 iOS 10 *devices* that *did not* have the DNT flag set, and we placed on 3,095 *ads* on devices that *did* have it set; i.e., as an upper bound, only 7% of iOS devices limit tracking.

[2] See http://scikit-learn.org/stable/modules/tree.html.

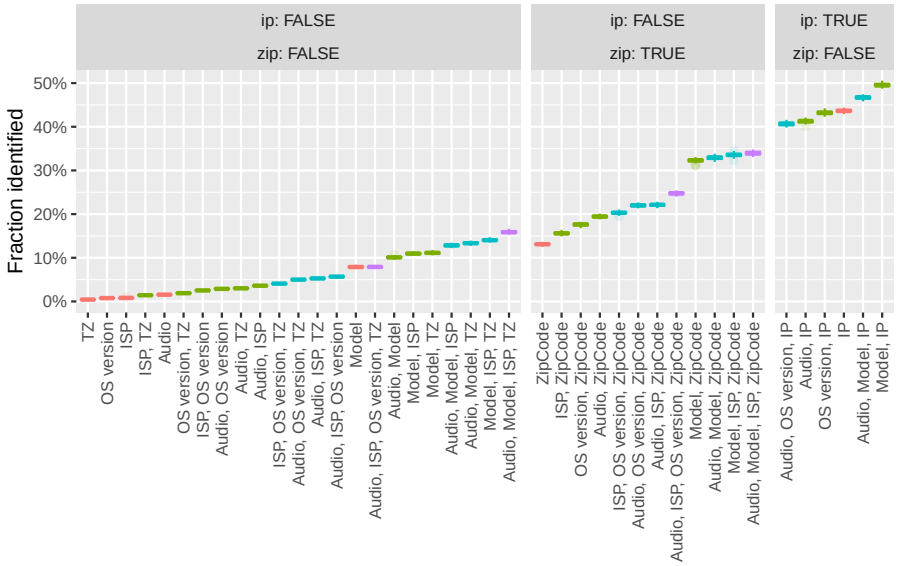

Figure 15: Identifying devices based on one or more features available from data returned from impressions. Each boxplot is the result of 66 trials, i.e., 2,706 trials total. Total cost = Approximately $75 (we reused data from a variety of experiments), Impressions = 142,592; Unique Devices: 40,010; Efficiency = 100%.

8 RELATED WORK

Measurement of mobile systems is a rich field. To our knowledge, AaaP is the first to leverage advertisements as a platform for measurement and experimentation. Below we provide comparisons of AaaP to the most relevant related work. See also Section 2 [25, 33, 39, 52].

Testbeds and their characterization (Section 5). AaaP can be viewed as the largest-ever mobile measurement testbed. Past testbed efforts include indoor wireless testbeds, WinLab [46], outdoor mobile testbeds [66], and related efforts such as LiveLab [64], PhoneLab [51], and PhantomNet [12]. Unlike these past approaches, AaaP requires no installation, maintenance, or major infrastructure. Code on devices is maintained by third parties who have a deep interest in ensuring ads are displayable and richly featured.

Measuring Location (Section 6.1). There are two primary methods to determine the location of a mobile device: *(i)* use the device-native capabilities or W3C Geolocation API in the browser, or *(ii)* approximate the location based on the IP address of the device using a commercial IP2Geo database [5–8]. The accuracy of such databases vary, and the sources of their data is mostly opaque. Several studies have attempted to quantify the accuracy using native apps [10, 70] or estimates of ground truth from POP locations [57]. AaaP can be used to both quantify the accuracy of such databases and greatly enhance their accuracy in a focused, inexpensive, and highly scalable manner.

Bandwidth Measurement (Section 6.2). Bandwidth measurement is a core task of networking research, and it is impractical to list all related work. In contrast to selected works, our study requires no carrier assistance [55] and is not limited to a single geographic area or country [71]. On the other hand, AaaP would be a challenging platform to gather statistics from base stations or to evaluate gains from delayed scheduling of requests [43], or from using two carriers at once [11]. An alternative is to recruit users to install a mobile app [4, 32]; however, such recruitment takes time and will inevitably be biased by geography and other factors.

Device Battery Behavior (Section 6.3). Many studies have looked at measuring battery power patterns in mobile devices, all using native measurements. Less recent studies were limited to less than 100 users [13, 21, 40, 40, 59, 60, 75]. These studies look to correlate battery usage with the processes running on the device, including background tasks [21, 75], which is not something that AaaP could measure. The OpenBattery project proposed a piecewise linear model for battery charging and discharging patterns [40]. We do the same, but we also contribute predictions about battery levels. Falaki et. al. [29] makes similar observations about approximating discharge and charge patterns as linear in a study of 255 users. Context aware battery management [60] attempts to control battery based on the user's context—AaaP can measure location, which could provide a greatly expanded study.

Carat is a recent native measurement tool, used by more than 300k users [37, 54]; see also Device Analyzer [78]. Carat was able to access the list of processes running and the battery level and identify energy intensive apps. AaaP cannot list processes on a device; but unfortunately, as of iOS 9.3.3, neither can native applications like Carat. The larger point is that recruiting tens or hundreds of thousands of users to a study relies on a massive effort or perhaps publicity in the popular press [23]. We were able to take battery measurements from 38k distinct users without recruitment. Further, we believe that an AaaP-base study can be far less biased in sampling than a self-selecting participant pool.

Identifiability (Section 6.4). Numerous papers have sought to evaluate distinguishing features of web browsers [27, 48–50, 74]. These studies are primarily restricted to the web, whereas mobile advertising libraries, and thus DSPs and advertisers, have access to many additional fingerprinting features, such as location, device names, storage sizes, etc. Such data has not been considered in other fingerprinting work, to our knowledge, and many studies are small.

Advertising. Past work has concentrated on measurements of ads themselves, including: consumer privacy [20, 34, 45, 56, 61, 63, 69]; algorithms to optimizing ad placement and delivery [22, 30, 41, 44, 47, 80]; and measurement of what ads users are shown [14, 24, 35, 53, 73, 76, 79].

9 CONCLUSIONS

We believe that AaaP is just the beginning in using ads to measure phenomena across mobile systems by researchers. While we have developed a number of techniques that solidify AaaP's capabilities, there will be more efficient methodologies — we reiterate that ethics discussions must keep pace with such innovations. Given the incredibly low effort needed to build new collection methods, the negligible costs, and the time required for deployment, this paper only scratches the surface of what is possible. Here we have made very little use of a wealth of location information, the access to the accelerometer, and the upcoming APIs that will be built to access numerous sensors on mobile devices. We expect that all of these will find their way into advertising-based measurement systems.

ACKNOWLEDGEMENTS

We are thankful to the anonymous reviewers for their insightful comments and suggestions, especially our shepherd Matt Welsh. We also thank Simson Garfinkel for insightful discussions about the IRB process. Our deployment was supported by the Amazon Web Services (AWS) Cloud Credits for Research program as well as high performance computing equipment obtained under a grant from the Collaborative R&D Fund managed by the Massachusetts Technology Collaborative.

REFERENCES

[1] iPhone 7 Plus: A Tale of Two Personalities. http://cellularinsights.com/iphone7/. (Oct 2016).
[2] MeetMe Reports U.S. Mobile CPMs Increased 14for November; Reiterates Fourth Quarter and Full Year 2016 Guidance. http://www.snl.com/irweblinkx/file.aspx?IID=4392695&FID=36970855. (December 05 2016).
[3] The Weather Company Fact Sheet. http://www.theweathercompany.com/weather-company-fact-sheet. (Oct 24 2016).
[4] http://www.eecs.umich.edu/3gtest (dead link). (2017).
[5] https://www.maxmind.com/. (2017).
[6] https://www.neustar.biz/risk/compliance-solutions/ip-intelligence. (2017).
[7] http://www.ip2location.com/. (2017).
[8] http://www.ipligence.com/. (2017).
[9] TuneIn Set to Deliver Coverage of Super Bowl LI to Global Audience. http://tunein.com/press-releases/TuneIn-Set-to-Deliver-Coverage-of-Super-Bowl-LI-to-Global-Audience/. (Jan 30 2017).
[10] Mahesh Balakrishnan, Iqbal Mohomed, and Venugopalan Ramasubramanian. 2009. Where's that phone?: geolocating IP addresses on 3G networks. In *Proc. ACM Internet measurement conference*. 294–300.
[11] Dziugas Baltrunas, Ahmed Elmokashfi, and Amund Kvalbein. 2014. Measuring the Reliability of Mobile Broadband Networks. In *Proc. ACM Internet Measurement Conference*. 45–58. http://doi.acm.org/10.1145/2663716.2663725
[12] Arijit Banerjee, Junguk Cho, Eric Eide, Jonathon Duerig, Binh Nguyen, Robert Ricci, Jacobus Van der Merwe, Kirk Webb, and Gary Wong. 2015. Phantomnet: Research infrastructure for mobile networking, cloud computing and software-defined networking. *GetMobile: Mobile Computing and Communications* 19, 2 (2015), 28–33.
[13] Nilanjan Banerjee, Ahmad Rahmati, Mark D Corner, Sami Rollins, and Lin Zhong. 2007. Users and Batteries: Interactions and Adaptive Energy Management in Mobile Systems. In *International Conference on Ubiquitous Computing (UbiComp)*. Springer, Innsbruck, Austria, 217–234.
[14] Paul Barford, Igor Canadi, Darja Krushevskaja, Qiang Ma, and S Muthukrishnan. 2014. Adscape: Harvesting and analyzing online display ads. In *Proceedings of the 23rd international conference on World wide web*. ACM, 597–608.
[15] Benoit Baudry. Am I unique? https://amiunique.org/. (2017).
[16] Karl Bode. Why Are People Using Ad Blockers? Ads Can Eat Up To 79% Of Mobile Data Allotments. https://www.techdirt.com/articles/20160317/09274333934/why-are-people-using-ad-blockers-ads-can-eat-up-to-79-mobile-data-allotments.shtml. (March 2016).
[17] Zachary Davies Boren. There are officially more mobile devices than people in the world. http://www.independent.co.uk/life-style/gadgets-and-tech/news/there-are-officially-more-mobile-devices-than-people-in-the-world-9780518.html. (Oct 7 2014).
[18] Sam Burnett and Nick Feamster. 2015. Encore: Lightweight measurement of web censorship with cross-origin requests. *ACM SIGCOMM Computer Communication Review* 45, 4 (2015), 653–667.
[19] John W. Byers. 2015. *Encore: Lightweight Measurement of Web Censorship with Cross-Origin Requests – Public Review,*. Technical Report http://conferences.sigcomm.org/sigcomm/2015/pdf/reviews/226pr.pdf. Department of Computer Science, Boston University.
[20] Claude Castelluccia, Mohamed-Ali Kaafar, and Minh-Dung Tran. 2012. Betrayed by your ads!. In *International Symposium on Privacy Enhancing Technologies Symposium*. Springer, 1–17.
[21] Hui Chen. 2016. *User-Centric Power Management For Mobile Operating Systems*. Ph.D. Dissertation. Wayne State University.
[22] Ye Chen, Pavel Berkhin, Bo Anderson, and Nikhil R Devanur. 2011. Real-time bidding algorithms for performance-based display ad allocation. In *Proc. ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 1307–1315.
[23] Josh Constantine. Carat: The Brilliant App That Increases Your Battery Life By Showing What Other Apps To Kill. https://techcrunch.com/2012/06/14/carat-battery/. (June 14 2012).
[24] Amit Datta, Michael Carl Tschantz, and Anupam Datta. 2015. Automated experiments on ad privacy settings. *Proceedings on Privacy Enhancing Technologies* 2015, 1 (2015), 92–112.
[25] David Dittrich, Erin Kenneally, et al. 2012. *The Menlo Report: Ethical Principles Guiding Information and Communication Technology Research*. Technical Report https://www.caida.org/publications/papers/2012/menlo_report_actual_formatted/menlo_report_actual_formatted.pdf. U.S. Department of Homeland Security Science and Technology Directorate, Cyber Security Division.
[26] Peter Eckersley. 2010. How unique is your web browser?. In *Privacy Enhancing Technologies (https://panopticlick.eff.org/)*. Springer, 1–18.
[27] Peter Eckersley. 2010. How unique is your web browser?. In *International Symposium on Privacy Enhancing Technologies Symposium*. Springer, 1–18.
[28] Steven Englehardt and Arvind Narayanan. 2016. Online tracking: A 1-million-site measurement and analysis. In *Proc. ACM Conference on Computer and Communications Security*.
[29] Hossein Falaki, Ratul Mahajan, Srikanth Kandula, Dimitrios Lymberopoulos, Ramesh Govindan, and Deborah Estrin. 2010. Diversity in smartphone usage. In *Proc. ACM Mobisys*. 179–194.
[30] Ayman Farahat and Michael C Bailey. 2012. How effective is targeted advertising?. In *Proc. international conference on World Wide Web*. ACM, 111–120.
[31] Jay Friedman. How do I estimate the total number of online ad impressions that might be seen by internet users over a 30-day period for a given U.S. DMA? https://www.quora.com/How-do-I-estimate-the-total-number-of-online-ad-impressions-that-might-be-seen-by-internet-users-over-a-30-day-period-for-a-given-U-S-DMA. (February 2015).
[32] Zhaoyu Gao, Arun Venkataramani, James F. Kurose, and Simon Heimlicher. 2014. Towards a Quantitative Comparison of Location-independent Network Architectures. In *Proc. ACM Sigcomm*. 259–270.
[33] Simson L. Garfinkel. 2008. *IRBs and Security Research: Myths, Facts and Mission Creep*. Technical Report https://calhoun.nps.edu/bitstream/handle/10945/40330/garfinkel_IRBs_and_Security_Research.pdf?sequence=1. Naval Postgraduate School.
[34] Michael C Grace, Wu Zhou, Xuxian Jiang, and Ahmad-Reza Sadeghi. 2012. Unsafe exposure analysis of mobile in-app advertisements. In *Proceedings of the fifth ACM conference on Security and Privacy in Wireless and Mobile Networks*. ACM, 101–112.
[35] Saikat Guha, Bin Cheng, and Paul Francis. 2010. Challenges in measuring online advertising systems. In *Proc. ACM Internet measurement conference*. ACM, 81–87.
[36] Stuart Haggas. Grindr & Scruff & Hornet… Oh My! Using Gay Social Networking Apps When Traveling. http://passportmagazine.com/grindr-scruff-hornet-oh-my-using-gay-social-networking-apps-when-traveling/. (retrieved March 14 2017).
[37] Mohammad A. Hoque and Sasu Tarkoma. 2016. Characterizing Smartphone Power Management in the Wild. In *International Conference on Ubiquitous Computing (UbiComp)*. 1279–1286.
[38] Gustavo Inciarte. 2014. Grindr Turns Five! http://queermeup.com/lgbt-community-2/grindr-turns-five (March 2014).
[39] Ben Jones, Roya Ensafi, Nick Feamster, Vern Paxson, and Nick Weaver. 2015. Ethical concerns for censorship measurement. In *Proc. ACM SIGCOMM Workshop on Ethics in Networked Systems Research*. ACM, 17–19.

[40] Gareth L. Jones and Peter G. Harrison. 2012. Collecting battery data with Open Battery. In *2012 Imperial College Computing Student Workshop (OpenAccess Series in Informatics (OASIcs))*, Andrew V. Jones (Ed.), Vol. 28. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 75–80.

[41] Azeem J Khan, Kasthuri Jayarajah, Dongsu Han, Archan Misra, Rajesh Balan, and Srinivasan Seshan. 2013. CAMEO: A middleware for mobile advertisement delivery. In *Proc. ACM Mobisys*. ACM, 125–138.

[42] Anssi Kostiainen and Mounir Lamouri. 2016. *Battery Status API W3C Candidate Recommendation*. Technical Report.

[43] Kyunghan Lee, Joohyun Lee, Yung Yi, Injong Rhee, and Song Chong. 2013. Mobile Data Offloading: How Much Can WiFi Deliver? *IEEE/ACM Trans. Netw.* 21, 2 (April 2013), 536–550.

[44] Kuang-chih Lee, Burkay Orten, Ali Dasdan, and Wentong Li. 2012. Estimating conversion rate in display advertising from past performance data. In *Proc. ACM SIGKDD international conference on Knowledge discovery and data mining*. 768–776.

[45] Ilias Leontiadis, Christos Efstratiou, Marco Picone, and Cecilia Mascolo. 2012. Don't kill my ads!: balancing privacy in an ad-supported mobile application market. In *Proc. Workshop on Mobile Computing Systems & Applications*. ACM, 2.

[46] Zoran Miljanic, Ivan Seskar, Khanh Le, and Dipankar Raychaudhuri. 2007. The WINLAB network centric cognitive radio hardware platform-WiNC2R. In *Proc. International Conference on Cognitive Radio Oriented Wireless Networks and Communications*. IEEE, 155–160.

[47] Prashanth Mohan, Suman Nath, and Oriana Riva. 2013. Prefetching mobile ads: Can advertising systems afford it?. In *Proc. ACM European Conference on Computer Systems*. ACM, 267–280.

[48] Keaton Mowery, Dillon Bogenreif, Scott Yilek, and Hovav Shacham. 2011. Fingerprinting information in JavaScript implementations. *Proceedings of W2SP* 2 (2011), 180–193.

[49] Keaton Mowery and Hovav Shacham. 2012. Pixel perfect: Fingerprinting canvas in HTML5. *Proceedings of W2SP* (2012).

[50] Martin Mulazzani, Philipp Reschl, Markus Huber, Manuel Leithner, Sebastian Schrittwieser, Edgar Weippl, and FC Wien. 2013. Fast and reliable browser identification with javascript engine fingerprinting. In *Web 2.0 Workshop on Security and Privacy (W2SP)*, Vol. 5.

[51] Anandatirtha Nandugudi, Anudipa Maiti, Taeyeon Ki, Fatih Bulut, Murat Demirbas, Tevfik Kosar, Chunming Qiao, Steven Y Ko, and Geoffrey Challen. 2013. Phonelab: A large programmable smartphone testbed. In *Proc. International Workshop on Sensing and Big Data Mining*. ACM, 1–6.

[52] Arvind Narayanan and Bendert Zevenbergen. 2015. *No Encore for Encore? Ethical Questions for Web-Based Censorship Measurement*. Technical Report https://ssrn.com/abstract=2665148. SSRN.

[53] Suman Nath. 2015. Madscope: Characterizing mobile in-app targeted ads. In *Proc. ACM Mobisys*. ACM, 59–73.

[54] Adam J Oliner, Anand P Iyer, Ion Stoica, Eemil Lagerspetz, and Sasu Tarkoma. 2013. Carat: Collaborative energy diagnosis for mobile devices. In *Proc. ACM Conference on Embedded Networked Sensor Systems*. 10.

[55] U. Paul, A. P. Subramanian, M. M. Buddhikot, and S. R. Das. 2011. Understanding traffic dynamics in cellular data networks. In *2011 Proceedings IEEE INFOCOM*. 882–890. https://doi.org/10.1109/INFCOM.2011.5935313

[56] Paul Pearce, Adrienne Porter Felt, Gabriel Nunez, and David Wagner. 2012. Addroid: Privilege separation for applications and advertisers in android. In *Proc. ACM Symposium on Information, Computer and Communications Security*. 71–72.

[57] Ingmar Poese, Steve Uhlig, Mohamed Ali Kaafar, Benoit Donnet, and Bamba Gueye. 2011. IP geolocation databases: Unreliable? *ACM SIGCOMM Computer Communication Review* 41, 2 (2011), 53–56.

[58] David Raab. How Many Ads Do You See Each Day? Fewer Than It Seems (I Think). http://customerexperiencematrix.blogspot.com/2015/09/how-many-ads-per-day-do-you-see-fewer.html. (September 2015).

[59] Ahmad Rahmati, Angela Qian, and Lin Zhong. 2007. Understanding Human-battery Interaction on Mobile Phones. In *Proc. International Conference on Human Computer Interaction with Mobile Devices and Services*. ACM, 265–272.

[60] Nishkam Ravi, James Scott, Lu Han, and Liviu Iftode. 2008. Context-aware battery management for mobile phones. In *IEEE International Conference on Pervasive Computing and Communications*. 224–233.

[61] Franziska Roesner, Tadayoshi Kohno, and David Wetherall. 2012. Detecting and defending against third-party tracking on the web. In *Proc. USENIX conference on Networked Systems Design and Implementation*. 12.

[62] Kenneth John Ryan et al. 1979. *The Belmont Report*. Technical Report http://www.hhs.gov/ohrp/humansubjects/guidance/belmont.html. U.S. Department of Health and Human Services.

[63] Shashi Shekhar, Michael Dietz, and Dan S Wallach. 2012. Adsplit: Separating smartphone advertising from applications. In *Proc. USENIX Security Symposium*. 553–567.

[64] Clayton Shepard, Ahmad Rahmati, Chad Tossell, Lin Zhong, and Phillip Kortum. 2011. LiveLab: measuring wireless networks and smartphone users in the field. *ACM SIGMETRICS Performance Evaluation Review* 38, 3 (2011), 15–20.

[65] Aaron Smith. African Americans and Technology Use. http://www.pewinternet.org/2014/01/06/african-americans-and-technology-use/. (January 2014).

[66] Hamed Soroush, Nilanjan Banerjee, Mark Corner, Brian Levine, and Brian Lynn. 2011. A retrospective look at the UMass DOME mobile testbed. *ACM SIGMOBILE Mobile Computing and Communications Review* 15, 4 (October 2011), 2–15.

[67] Hamed Soroush, Keen Sung, Erik Learned-Miller, Brian Neil Levine, and Marc Liberatore. 2013. Turning Off GPS is Not Enough: Cellular location leaks over the Internet. In *Proc. Privacy Enhancing Technologies Symposium (PETS)*. 103–122.

[68] Jack Spirou. Clientjs. https://github.com/jackspirou/clientjs. (2017).

[69] Ryan Stevens, Clint Gibler, Jon Crussell, Jeremy Erickson, and Hao Chen. 2012. Investigating user privacy in android ad libraries. In *Workshop on Mobile Security Technologies (MoST)*. 10.

[70] Sipat Triukose, Sebastien Ardon, Anirban Mahanti, and Aaditeshwar Seth. 2012. Geolocating IP addresses in cellular data networks. In *International Conference on Passive and Active Network Measurement*. Springer, 158–167.

[71] Fung Po Tso, Jin Teng, Weijia Jia, and Dong Xuan. 2010. Mobility: A Double-edged Sword for HSPA Networks: A Large-scale Test on Hong Kong Mobile HSPA Networks. In *Proc. ACM MobiHoc*. 81–90.

[72] Liam Tung. Think you're not being tracked? Now websites turn to audio fingerprinting to follow you. http://www.zdnet.com/article/think-youre-not-being-tracked-now-websites-turn-to-audio-fingerprinting-to-follow-you/. (May 20 2016).

[73] Imdad Ullah, Roksana Boreli, Mohamed Ali Kaafar, and Salil S Kanhere. 2014. Characterising user targeting for in-app mobile ads. In *IEEE Infocom Workshops*. IEEE, 547–552.

[74] Thomas Unger, Martin Mulazzani, Dominik Frühwirt, Markus Huber, Sebastian Schrittwieser, and Edgar Weippl. 2013. Shpf: Enhancing http(s) session security with browser fingerprinting. In *Proc. International Conference on Availability, Reliability and Security (ARES)*. IEEE, 255–261.

[75] Narseo Vallina-Rodriguez, Pan Hui, Jon Crowcroft, and Andrew Rice. 2010. Exhausting Battery Statistics: Understanding the Energy Demands on Mobile Handsets. In *Proc. ACM Workshop on Networking, Systems, and Applications on Mobile Handhelds*. 9–14.

[76] Narseo Vallina-Rodriguez, Jay Shah, Alessandro Finamore, Yan Grunenberger, Konstantina Papagiannaki, Hamed Haddadi, and Jon Crowcroft. 2012. Breaking for commercials: characterizing mobile advertising. In *Proc. ACM Internet Measurement Conference*. ACM, 343–356.

[77] Valentin Vasilyev. fingerprintjs2. https://github.com/Valve/fingerprintjs2. (Retreived June 2017).

[78] Daniel T Wagner, Andrew Rice, and Alastair R Beresford. 2013. Device analyzer: Understanding smartphone usage. In *International Conference on Mobile and Ubiquitous Systems*. Springer, 195–208.

[79] Jun Yan, Ning Liu, Gang Wang, Wen Zhang, Yun Jiang, and Zheng Chen. 2009. How much can behavioral targeting help online advertising?. In *Proc. International conference on World Wide Web*. ACM, 261–270.

[80] Weinan Zhang, Shuai Yuan, and Jun Wang. 2014. Optimal real-time bidding for display advertising. In *Proc. ACM SIGKDD international conference on Knowledge discovery and data mining*. 1077–1086.