AdSelector: A privacy-preserving advertisement selection mechanism for mobile devices

Yang Liu and Andrew Simpson

Department of Computer Science, University of Oxford
Wolfson Building, Parks Road, Oxford OX1 3QD
United Kingdom
Email: firstname.secondname@cs.ox.ac.uk

Targeted mobile advertising (TMA) enables organisations to tailor advertisements to specific consumers by analysing the personal information collected from consumers' mobile devices. Although TMA offers great benefits to advertisers, the privacy concerns associated with it may reduce the advertising effectiveness. It follows that there is a need for an advertisement selection mechanism that can support the existing TMA business model in a manner that takes into account consumers' privacy concerns. We present such an ad selection mechanism that has the potential to provide benefits to both consumers and advertisers. The mechanism is novel in its combination of a user subscription mechanism, a two-stage ad selection process, and the application of a trustworthy billing system. In particular: (1) the user subscription mechanism helps users to identify their interests and subscribe to desirable categories of ads; (2) the two-stage ad selection process ensures that ad servers can only obtain coarse-grained user profiles, with fine-grained user profiles stored and used only on the mobile devices; and (3) the trustworthy billing system helps to report ad-clicks without revealing users' identities and assists in detecting click-fraud attacks. The performance of the mechanism is evaluated in the context of a prototype privacy-preserving targeted mobile advertising framework.

Keywords: Mobile devices, Advertising, Privacy

Received 00 January 2009; revised 00 Month 2009

1. INTRODUCTION

Mobile advertising has seen significant developments in recent years. In 2011, the value of mobile ad spending in the USA was approximately US $1.1 billion per year [1]; this figure reached US $30.45 billion by the end of 2015, accounting for 51.9% of total digital ad spending in the USA [2] — meaning that spending associated with mobile ads in the USA has now surpassed spending associated with desktop ads. In [3] it was predicted that the value of mobile ad spending in the UK would overtake that of TV ad spending and reach £4.58 billion per year, accounting for 50.8% of all digital ad spending and 27% of total media spending in the UK by the end of 2016 — meaning that desktop ad spending in the UK will be overtaken by mobile ad spending by the end of 2016.

Ad-networks bill advertisers for delivering ads, and share the payment with developers who provide space for displaying ads. In the mobile app ecosystem, many enterprises or developers provide their mobile apps for free and rely on targeted advertising for revenue. For example, according to a study from 2012 [4], about 80% of the free apps in the Google Play Market rely on mobile advertising as their main source of revenue. Research undertaken by the present authors [5] found that at least one ad library was embedded in 46 out of 60 off-the-shelf apps selected randomly from the top free apps of the Google Play Store in September 2015.

However, it is well known that mobile users typically cannot use the 'free apps' freely. In exchange for the benefits of the received services, users, who perhaps do not fully comprehend the consequences of their actions, voluntarily give up some of their privacy. To improve advertising effectiveness, ad-networks select tailor-made ads targeted at individuals by analysing consumers' personal information, which is primarily collected from consumers' mobile devices. Users' privacy concerns, in turn, give rise to a degree of hostility towards advertisers, which might lead to the utilisation of ad-blocking tools. Hence, there is a need for a targeted mobile ad selection mechanism that can support the

existing targeted mobile advertising (TMA) business model, and can do so in a manner that helps to protect users' personal information rather than simply bypassing their privacy concerns.

To this end, we describe *AdSelector*, a profile-based advertisement selection mechanism for mobile advertising. The mechanism is novel in its combination of a user subscription mechanism, a two-stage ad selection process, and the application of a trustworthy billing system. In particular:

- The user subscription mechanism helps users to identify their interests and subscribe to desirable categories of ads.
- The two-stage ad selection process ensures that ad servers can only obtain coarse-grained user profiles, with fine-grained user profiles stored and used only on the mobile devices.
- The trustworthy billing system helps to report ad-views and ad-clicks without revealing users' identities and can assist in detecting advertising fraud attacks.

Previous work in this area has tended to take the side either of the consumers or of the advertisers. By contrast, our aim is to provide benefits to both consumers and advertisers in the context of targeted mobile advertising. The paper focuses on combining the above mechanisms to help users protect their personal information, and to help ad-networks perform advertising and billing in a way that respects privacy. We have previously prototyped a system called Privacy-Preserving Targeted Mobile Advertising (PPTMA) [5], which works in the background of mobile devices and can help mobile advertisers to deliver ads without compromising users' privacy. We validate our solution in the context of PPTMA.

The remainder of this paper is organised as follows. Section 2 provides the motivation for, and the background to, our contribution. Then, in Section 3 we present our design. We evaluate the performance of AdSelector in Section 4. Next, Section 5 places our contribution in the context of related work. Finally, in Section 6, we summarise the contribution, present conclusions, and identify potential areas of future work.

2. MOTIVATION AND BACKGROUND

The motivation behind AdSelector is the desire to complement current TMA systems with a view to providing benefits to all involved stakeholders. We first introduce the TMA ecosystem and then give consideration to the requirements of our privacy-preserving advertisement selection mechanism. Then we provide a brief overview of our previous work, *PPTMA* [5], a privacy-preserving TMA system that, in the context of this paper, is used to validate AdSelector. Finally, we discuss the typical ad interest categorisation approach used in most TMA systems.

2.1. The TMA ecosystem

The core concept of a TMA system is the mechanism that can automatically select and display relevant ads for mobile users. In general, the automatic process is accomplished by collecting the target users' profiles and filtering data from a pool of ads.

Most ad-networks maintain their own TMA system, which consists of numerous apps and ads. Mobile users' personal data is collected by the ad-plugins that are embedded in the apps and sent to the ad-networks' servers. An ad selection mechanism then selects ads from a large pool of of ads submitted by advertisers. Advertisers need to pay related ad-networks if their ads are viewed or clicked by the users. The ad-networks will then share the payment with app developers who display the ads within their apps. Figure 1 shows the interactions between these stakeholders.

2.2. Requirements for a privacy-preserving advertisement selection mechanism

From the point of view of the ad-networks, a crucial aspect of the ad selection mechanism is personalisation [6, 7]. Users' personal information, such as their location, income, interests and visit history, can help ad-networks reach potential customers more effectively [8, 9, 10]. To make more efficient use of the collected users' information, the ability to compute the most relevant ads for each individual user is also required [11]. In addition, a billing system that has the potential to defend against various types of attacks and provide trustworthy ad view/click reports must be secured [12, 13]. Hence, an ideal ad selection mechanism for ad-networks should meet the following requirements.

1. The mechanism must be able to collect mobile users' personal information, which can be used to create users' profiles and deduce their interests.
2. The mechanism must be able to select the most relevant ads for a particular user with a high degree of accuracy and efficiency.
3. The mechanism must be able to provide trustworthy ad view/click reports for the ad-networks to process billing and accounting with respect to advertisers and app developers.

In terms of mobile users, many users worry about the potential abuse of their personal information [14] so they might deliberately obfuscate their data [15] or block all ads [16]. On the other hand, although users' general attitudes towards TMA might be rather negative, some industrial reports and academic studies suggest that there are some price-conscious customers who do actively welcome mobile ads if they could be rewarded [17, 18, 19], particularly when they are provided with the ability to control their personal information [20]. In addition, some of these users would like to be able to optimise the targeting process

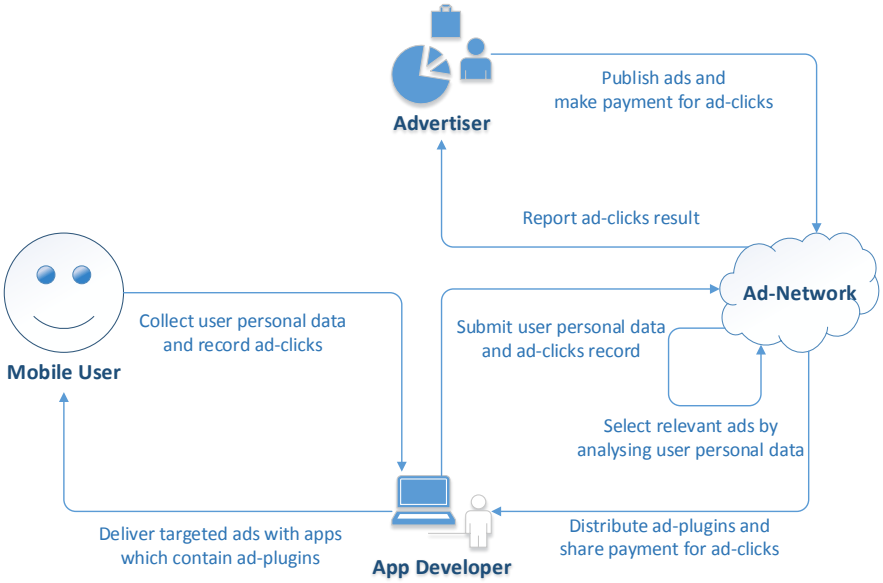

FIGURE 1. The TMA ecosystem

to improve the relevance of ads to get recognisable benefits [21]. Therefore, from the mobile users' standpoint, a satisfactory ad selection mechanism should meet the following requirements.

1. The mechanism must be able to provide useful ads without compromising users' privacy.
2. The mechanism must be able to provide the ability for users to opt-in and opt-out of the TMA process.
3. The mechanism must be able to provide the ability for users to engage in and optimise the TMA process.
4. The mechanism must be able to define clearly which parts of users' information are accessed, collected and shared by third parties (and which third parties do so).

Motivated by both sets of requirements, we have developed *AdSelector*, a profile-based, privacy-preserving advertisement selection mechanism. The mechanism is validated in the context of the PPTMA (Privacy-Preserving Targeted Mobile Advertising) system [5].

2.3. The PPTMA system

The PPTMA framework [5] is a privacy-preserving TMA system. The system serves privacy-friendly targeted ads in two ways: via either a 'cooperative mode' or a 'mandatory mode'. In the cooperative mode, ad-networks are aware of PPTMA, and tailor-made ads are selected with users' local information on the client. In the mandatory mode, the system hooks API calls that can expose users' personal information, and consults a local policy to decide how to proceed.

At a high level, PPTMA works as a service that runs on mobile platforms to perform permission management and sensitive data processing between the underlying database of the mobile system and untrusted third-party applications. The following features are supported in the initial prototype.

1. **Centralised management of users' privacy.** PPTMA enables users to examine and customise their personal information through a unified interface. Different copies of the user's profile can be created and edited manually, then shared with different third parties that have been approved by the user.
2. **Fine-grained access control.** This feature enables users to control permissions associated with accessing their personal information held by third parties, and decide which parts of their personal information can be collected by which ad-networks. In the initial prototype of PPTMA, which runs on the Android system, this feature is implemented by hooking sensitive Android APIs.
3. **Ad-plugins scanning and verifying.** PPTMA enables users to discover ad-plugins embedded in the installed mobile apps and distinguish between the related ad-networks. Apps import ad libraries provided by ad-networks to perform targeted advertising. By scanning external libraries contained in apps and comparing the feature codes with pre-collected ad libraries, the system can identify which ad-networks are associated with a particular app, then deduce the ad styles and potential behaviour of the app. In addition, a behaviour monitoring service is also provided to detect malicious third-party apps.
4. **Support for privacy-aware advertisement selection.** PPTMA enables ad-networks to implement advertisement selection on mobile clients with customised algorithms. This feature

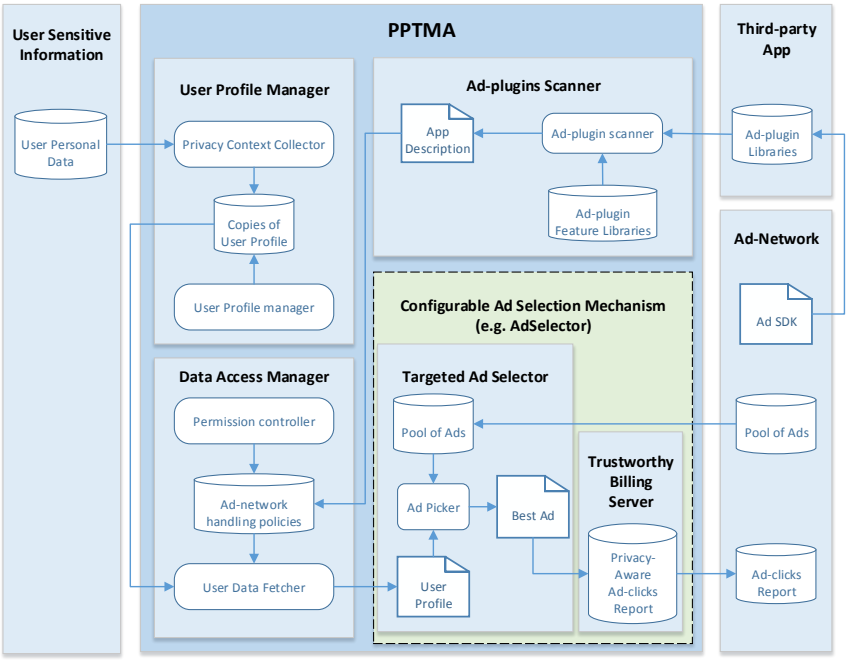

FIGURE 2. Architecture overview of PPTMA

enables ad-networks to create tailor-made ads without delivering users' personal information to the ad-networks' servers, which limits the use of users' personal information inside their mobile devices. To implement this feature, PPTMA makes use of the light version of ad-SDKs provided by cooperative ad-networks. The ad-SDKs enable PPTMA to perform some basic functions such as pre-downloading potential lists of ads and submitting trustworthy ad view/click reports to support local advertisement selection.

Figure 2 presents an overview of the architecture of PPTMA. As a general advertisement selection mechanism, AdSelector can be deployed in PPTMA to implement the 'Configurable Ad Selection Mechanism' module. The cooperative mode of PPTMA is used to examine the performance of AdSelector.

2.4. Ad interest categorisation

In most TMA systems, advertisers are required to specify particular ad interest categories when publishing ads. Users involved in these TMA systems are also associated with one or more of these ad interest categories — either deduced automatically by ad-networks or specified by users themselves — to indicate the kinds of ads and content they are interested in. The ad-networks then give consideration to some other factors (e.g. advertisers' budgets, users' ad preferences and location, etc.), and deliver particular ads to related users who are classified in the same categories.

In general, ad interest categorisations have a hierarchical structure. For example, at the time of writing (July 2016), the Google ad platform [22] uses a multi-level categorisation consisting of 25 top-level categories and up to 7 sub-levels. By way of comparison, the Yahoo ad platform [23] uses a categorisation consisting of 16 top-level categories and 5 sub-levels. Finally, the Facebook ad platform [24] uses a hierarchical ad interest categorisation consisting of only 9 top-levels. However, apart from the interest categorisation, Facebook also applies a hierarchical demographics categorisation (e.g. education background and income condition) and a hierarchical behaviours categorisation (e.g. purchase behaviour and trip frequency) to help profile users.

Despite the different classification criteria used in the above ad platforms, they all apply the same categorisation structure. The hierarchical structure makes a contribution to characterising users and ads in an effective way. A user or an ad can be assigned to different levels of the categorisation according to the particular requirement of accuracy or according to users' data access preferences. Additionally, the feature of assigning an item to more than one category provides flexibility.

As an example, we write A > B to represent the notion that B is a sub-level item of A. Thus, a typical category of the Google ad interest categorisation might be the three-level item Sports > Individual Sports > Golf. Any ads about golf, for example, pertaining to equipment, video games, or tickets can all be associated with this category. If the advertiser wants to narrow the scope to make their ads more precise, they can then specify an additional category Sports > Sporting Goods > Golf Equipment for ads for golf clubs and assign

Games > Computer & Video Games > Sports Games to ads for golf video games. Meanwhile, according to a user's data access control policy, they can be associated with different levels of the item — Sports > Individual Sports > Golf, Sports > Individual Sports, or, even more coarsely, with the top-level category Sports.

3. DESIGN OF ADSELECTOR

The fundamental driver of our system is the need to exchange coarse-grained information pertaining to ads from the ad-networks with very limited anonymous user data, and selecting the most useful ads with detailed user profiles on the mobile clients. To provide protection for both users' and ad-networks' benefits, several mechanisms are combined.

3.1. Mechanisms and workflow

The core concept of AdSelector involves selecting the most relevant ad with detailed user data being stored on users' devices, rather than on ad-networks' servers. Some previous approaches, such as Adnostic [12], Privad [25] and MobiAd [26], have similar goals. The novel feature of our system is the combination of several mechanisms that help to improve ad selection effectiveness, to increase user participation, to provide billing and accounting function in a privacy-preserving fashion, and to defend against click-fraud attacks. We consider each in turn.

1. **User subscription mechanism**. Users are able to identify their interests and subscribe to specific categories of ads. The mechanism has the potential to increase users' tolerance to targeted ads and increase their engagement in targeted mobile advertising.
2. **Two-stage ad selection process**. The most relevant ads for a particular user are selected via a two-stage process. Users' subscription profiles and related coarse-grained information are first delivered to ad-networks anonymously for pre-downloading lists of ads that the user may be interested in. Then the fine-grained user profile stored in the mobile device is used locally to select the best ads to display. The two-stage selection process helps to keep users' private data inside the local device.
3. **Privacy-friendly billing system**. Ad-networks need the view or click records of ads for billing advertisers and paying app developers. The view/click records, in turn, can be analysed to infer users' interests. The privacy-friendly billing system we provide is an instance of the Trustworthy Remote Entity (TRE) [27], a highly-specialised networked system with very minimal function for data processing. It helps to identify which ads have been displayed by which apps without telling ad-networks which users performed the views or clicks.
4. **Advertising frauds defence mechanism**. Advertising frauds, particularly view-fraud and click-fraud, cause serious damage to the TMA ecosystem. Click-fraud is a type of fraud that involves attackers simply clicking ads manually or by deploying automated scripts to earn money or consume advertisers' budgets [28]. Undetected click-fraud can cause large losses to advertisers — Pearce *et al.* [29] claim that advertising losses of about US $100,000 per day can be caused by a single click-fraud botnet (ZeroAccess). Further, Dave *et al.* [30] suggest that around one third of mobile ad-clicks are suspected to be click-spam. Hence, there is a need to detect mobile ad click-fraud in every ad-network system. The fact that ad-networks cannot identify users in our solution may reduce the ability to detect advertising frauds; we propose several approaches to resist such attacks as a compensation mechanism.

The system obtains the ability to perform privacy-preserving targeted mobile advertising by combining all of the above mechanisms in terms of the workflow of Figure 3. This workflow is described below.

1. A user subscribes to particular ad categories that he or she is interested in, and presets the data sharing levels of all installed apps (i.e. which parts of the user information can be obtained by which third-party app).
2. The information provided by the user, together with detailed user data that is collected by our system, is then used to generate user stereotypes in the form of coarse-grained anonymous user data.
3. A particular copy of the user stereotype is sent to a specific ad-network. The copy is then used to select ads that the user might be interested in and generate the potential list of ads in the server of the ad-network. The list of ads is sent back to the mobile device and stored in the local pool of ads.
4. On the mobile client, the most relevant ads are selected from the local pool of ads by computing the fine-grained user data. The selected ads are then displayed on the mobile device and the user can then view and click on them.
5. After the view/click of an ad, the view/click report with the real user ID will be sent to the billing system from the mobile client. In the billing system, a random single-use user ID is generated to replace the real one. The processed view/click report is then submitted to the server of the related ad-network, while the mapping of the real user ID to the single-use user ID is stored in the billing system for a certain period determined by the ad-network.
6. The ad-network can bill advertisers and share payment with app developers based on the view/click reports. When there is a possibility of view-fraud or click-fraud attacks, the ad-network

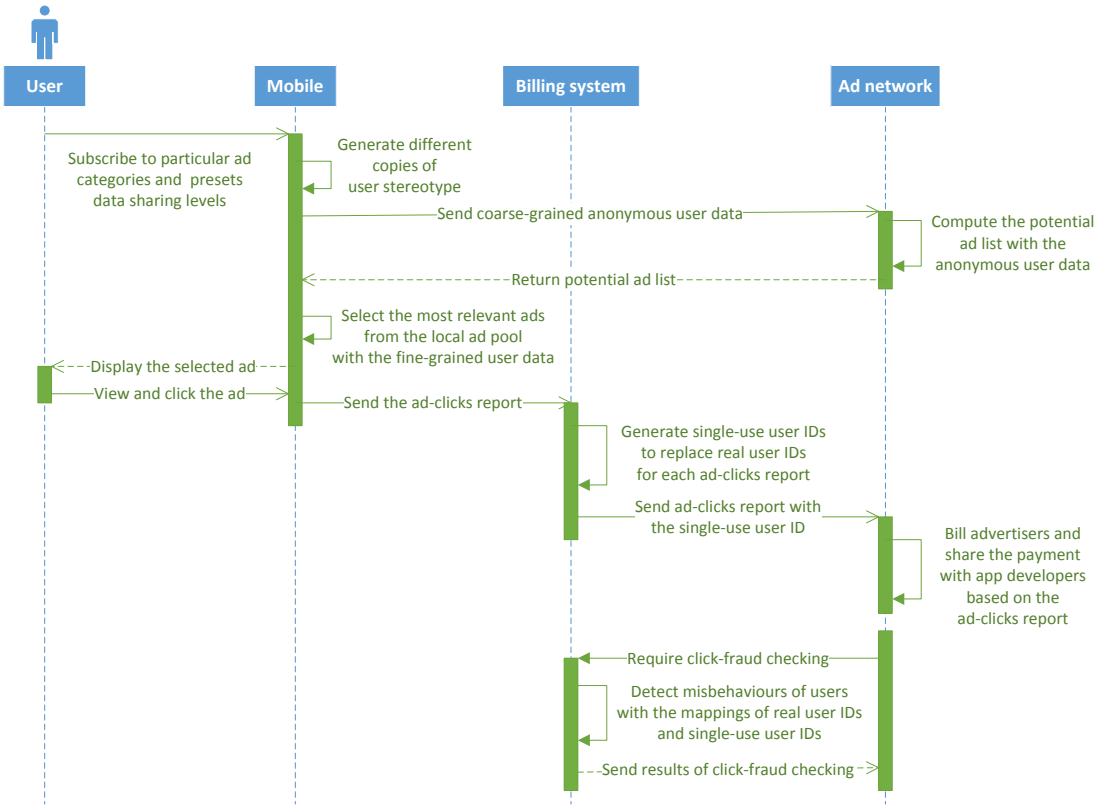

FIGURE 3. System core workflow

sends a request to the billing system to perform related detection.

7. The billing system detects misbehaviours of users with the ID mappings and view/click reports, then submits results of view-fraud or click-fraud checking to the related ad-network.

3.2. System architecture

There are two key interactions between our system and ad-networks:

1. AdSelector provides coarse-grained user information to ad-networks to exchange potential lists of ads.
2. Ad-networks obtain trustworthy but privacy-conscious ad view/click reports from AdSelector to support billing.

At a high level, our system can be implemented in two distinct ways: as a third-party app or as an external library. In the first mode, users install the system on their mobile devices as an app. Ad-networks need to provide only simple APIs to interact with the system instead of modifying their existing facilities. In the second mode, users are not required to install any extra apps. However, ad-networks need to import an external library into their ad SDK, and obtain coarse-grained users' data and view/click reports by calling related APIs provided by the external library.

There are some trade-offs associated with both modes. In the third-party app mode, ad-networks do not need to modify the way they serve ads. Following the installation, users' devices can detect related ad-networks automatically and call the light version APIs to accomplish the key interactions. If the app of AdSelector is not installed, ad-networks can serve ads via their existing processes. In the external library mode, ad-networks can serve ads in the privacy-friendly way to all mobile users. However, they would need to modify their existing codes and servers to apply the new features.

Figure 4 shows the system architecture of AdSelector for the first of these modes. There are three main components:

1. **User data manager**. This module maintains all kinds of user data (e.g. device ID, location information, installed apps, etc.) that are collected from the mobile device. User profiles are generated in terms of the collected data. In addition, together with the information of particular ad categories that users have subscribed to and the preset access control policies, different copies of user stereotypes — the coarse-grained anonymous user data that helps ad-networks to generate potential lists of ads — are also created by the user data manager.
2. **Ad data manager**. The ad data manager

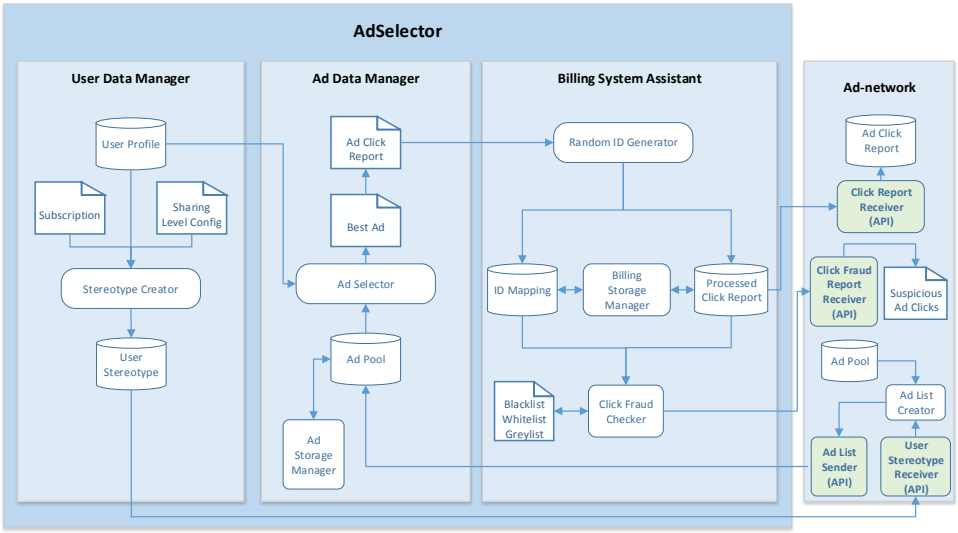

FIGURE 4. System architecture

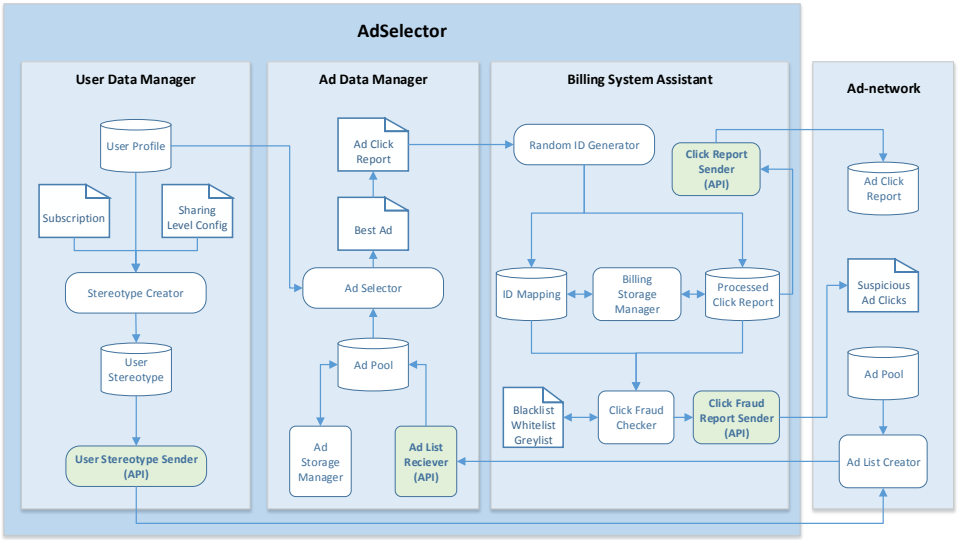

FIGURE 5. Alternative system architecture

maintains a local pool of ads. The lists of ads in the pool are obtained from different ad-networks with different copies of user stereotypes. A storage manager helps to optimise the pool of ads — removing, for example, the ads that meet certain conditions, e.g. an ad that has been displayed many times but never clicked. The ad selector filters ads from the pool of ads and selects the most relevant ad for the user based on their detailed and contextual user profile. The original data associated with ad view/click reports is also stored in this module.

3. **Billing system assistant**. This module helps to process ad view/click reports before they are delivered to ad-networks. A single-use user ID is generated by the random ID generator to replace the real user ID inside a new view/click report that is received only by this module. The mapping of the two kinds of user ID are stored in the system for detecting view-fraud and click-fraud attacks, while the view/click reports with single-use user IDs are sent to the related ad-network for billing advertisers and paying app developers. In addition, a billing storage manager is provided to control the retention and deletion periods of the mappings and ad view/click reports. There is also a blacklist–greylist–whitelist mechanism built into this module that is used to mark different levels of users' behaviours.

Third party apps connected with ad-networks can interact with the system in either of the two modes mentioned above. Figure 4 represents the first mode: users install the app of AdSelector on their mobile devices and ad-networks provide related APIs to

implement the ad list creator and the view/click report fetcher to perform the interactions. The second mode works similarly, but the related APIs are provided by the AdSelector library. The alternative system architecture is shown in Figure 5.

3.3. User subscription mechanism

User profiles generated by ad-networks might sometimes be questionable or untrustworthy because users might obfuscate their data for the sake of privacy-preservation [31]. Additionally, the algorithms used for deducing users' interests might not be sufficiently effective — users understand their own interests and needs better than ad-networks working with incomplete users' profiles [32].

In the privacy-friendly TMA environment supported by AdSelector, users can take advantage of mobile ads without worrying about their personal information being misused. Furthermore, users can actively make efforts to optimise the targeting process to increase the accuracy of selecting relevant ads for their own benefit. Findings from previous studies [19, 20] suggest that independent choice and mutual benefit can contribute to users' acceptance of mobile advertising and increase their engagement in the targeting process. Hence, a user subscription mechanism is provided to give users the required abilities.

The following attributes can be customised by users. These attributes are chosen because they are generally collected and used by the likes of Google and Facebook.

1. **Age & Gender**. Personal information like age and gender is collected by most advertisers as these attributes may affect consumer response to advertising appeals significantly. With the subscription mechanism, a user can indicate her age with a range rather than with an exact figure. In addition, the mechanism enables users to decline to state their gender.
2. **Interests.** The interests of a user will be deduced by the user data manager of AdSelector and automatically added to the user's profile as they interact with the mobile device. In addition, users can edit the deduced interests and manually add keywords as specific interests.
3. **Location.** Users can decide how accurately their location data should be shared with different ad-networks. Additionally, they can subscribe to ads related to a specific location. For example, users planing to travel to London can choose to receive ads and discount coupons for there.
4. **Types of ads.** There are different types of mobile ads such as introduction of new products, discount coupons, public service ads, etc. Users can indicate which particular types they are interested in.
5. **Categories.** Generally, ads are published in ad-networks and associated with one or more categories. Accordingly, users can subscribe to categories of interest.

We have chosen to leverage Google ad interest categories. As one of the largest ad-networks serving ads on both Android and iOS platforms, Google makes its ad interest categories openly available. Applying Google ad interest categories has the potential to make it easier to adapt our mechanism to current TMA systems.

Additionally, users can set weights for the attributes to which they have subscribed. The data of users' subscriptions is maintained by the user data manager of AdSelector. According to the requirements of advertisement selection algorithms applied by different ad-networks, the subscription data can be presented in different formats. For example, Food & Drink > Food > Meat & Seafood > Poultry can be converted to the set { Food, Drink, Meat, Seafood, Poultry } for fuzzy matching — which ignores the hierarchical information so that all factors in the set are considered with the same weight. Alternatively, the category can be converted to the sequence ⟨ Food & Drink, Food, Meat & Seafood, Poultry ⟩ for more sophisticated matching. Here, the hierarchical order of factors in the sequence is considered. Thus, the factors could be computed with different weights. Furthermore, if a related ad-network uses a fixed category list, the subscription information can be converted to a vector-based representation that enables simple comparison.

3.4. Two-stage ad selection process

Having generated the user profiles and collected the subscription information, further work can be done to complete the process of ad selection and display. To this end, we now provide details of the ad selection process applied in AdSelector.

The main idea of the ad selection process is to keep and use the fine-grained user profiles only on the mobile devices so as to reduce the possibility of users' information leaking. To accomplish this, we perform the ad selection process in two stages. The first stage decides which potential ads should be cached in the mobile clients; the second stage decides which particular ad should be displayed.

In the first stage, the coarse-grained version of a user's profile is delivered to the server anonymously. Based on the coarse-grained information, the ad-network can select a number of ads that the user may be interested in. These ads, together with some essential attributes, such as targeting location, gender and ad interest category, are then stored in an ordered list of ads. The list of ads is downloaded to the user's mobile client at the end of this stage.

In the second stage, the fine-grained version of the user's profile and the pre-downloaded ads are encoded in a common format to perform the comparison. All of the ads are scored and ranked according to how accurately they match the user's profile and the bid budget set on

the ads. The most relevant ad at the top of the list will be displayed.

The default ad selection approaches for the two stages are also presented below so as to describe the complete workflow of AdSelector. These approaches can be replaced with pre-existing ad selection algorithms in both stages.

3.4.1. Pre-download: get the potential list of ads from an ad-network

The best way to perform the pre-download process is to download all ads from all servers of the ad-networks to the user's mobile device. Thus, the user would not lose any potential ads and the ad-networks would not learn anything about the user.

Although the ideal approach cannot be implemented simply due to many hurdles (e.g. storage, bandwidth, and performance), we strive to achieve an appropriate approximation. A suitable pre-download approach should meet the following requirements.

1. Users should be able to download enough relevant ads to be selected and displayed in the second stage.
2. Ad-networks should not be able to identify the users nor learn their accurate interests.
3. The approach should be compatible with approaches adopted by, for example, Google, Facebook, and Twitter.
4. The representation used in the approach should enable simple comparison in both stages.

A keyword-matching approach is chosen to be the basic ad selection approach in the pre-download process. On the one hand, most of the existing ad-networks support keyword matching, as it is one of the main approaches to filter ads. On the other hand, user profiles generated in AdSelector can be converted easily to a set of keywords. Figure 6 shows an overview of the two-stage ad selection process based on the keyword-matching approach.

The information used to generate the user profiles are collected automatically and/or edited by the users manually via the subscription mechanism. Configured by the user, a copy of the user profile can be encoded into different keyword sets with different accuracies. An example copy of a user profile might contain the following information:

> Gender: Male.
> Age: 38 years old.
> Interests: Football and reading.[1]
> Location: Paddington, London, UK.
> Types of ads: Discount coupons.
> Categories: Food & Drink > Food > Meat & Seafood > Poultry.

[1]As interests can be edited manually by adding or removing keywords via the subscription mechanism, any words can be associated with this attribute.

Depending on the access control policies applied by the user, the keyword set could take one of several forms, such as { Male, 38 years old, Football, Reading, Paddington, Discount Coupons, Poultry } and { 20 to 40 years old, Team Sports, Books & Literature, London, Food }.

Note that in the second set, which is relatively coarse-grained, the user's interests are convert into keywords that are extracted from different levels of the ad interest categories we use: Football is extracted from Sports > Team Sports and Reading is extracted from Books & Literature; Poultry, the category manually subscribed to, is promoted to the higher level category Food to reduce the accuracy of the information to be submitted. In addition, the age and location are generalised, and other information, such as gender and the types of ads, are suppressed.

After receiving the keyword sets, ad-networks can execute their own algorithms to select potential ads. Although the keyword-matching mechanism is supported well in most existing ad-networks, many algorithms pertaining to keyword-matching or ad auctions are commercially sensitive and cannot be shared. In addition, it is also unrealistic for an approach such as ours to provide a universal ad ranking algorithm for all ad-networks.

Taking Google AdWords [33] as an example, some bids are set at the ad group level, rather than at the keyword level. Each ad group is associated with a set of keywords; for example, the keywords Solid-State Laser, Solid State Lasers and Gas Laser are all associated with the same ad group — Laser Systems. If a keyword from the set triggers an ad to appear, the price set on the ad group will be charged by the ad-network. Another example is negative keywords, which is also supported by AdWords. An advertiser who runs a pet food store but does not sell cat food can add *cat* as a negative keyword, so that the keywords *pet cat food* will not trigger the advertiser's ads to appear.

Since it is impossible to provide a unified algorithm that is compatible with the complicated bidding strategies and keyword-matching mechanisms behind all ad-networks, the approach only converts user profiles into keyword sets at the user end. The keyword-matching process can then be accomplished by individual ad-networks. Nevertheless, when generating the potential list of ads, the ads are required to be grouped by their trigger keywords with their original ranks maintained. Thus, in the local ads filter stage — the second stage of the ad selection process — AdSelector can decide which ads are displayed in what order without knowing the particular bidding strategy of the ad-network and the bidding budget of the involved advertisers.

Other essential attributes (e.g. target attributes, ad type and ad format) of each ad are also stored in the list of ads for filtering out no matching ads in the second stage. For instance, the target attributes such

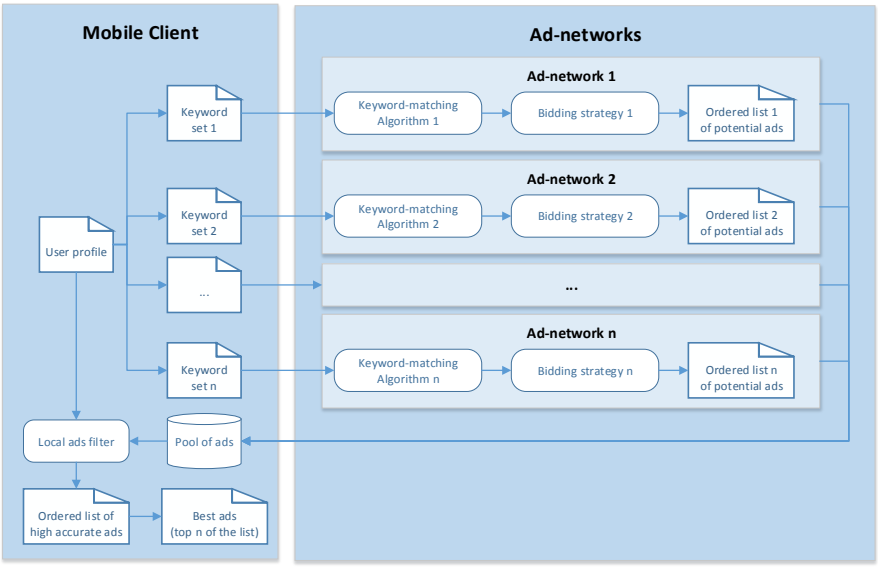

FIGURE 6. The two-stage ad selection process

as targeting location, gender and language help to filter out ads that do not match the fine-grained user profiles. The ad type and format help to select the most suitable ad view for the current screen to display (e.g. banner image ad and skippable video ads).

To reduce the consumption of storage and bandwidth in the pre-downloading process, users can set the number of ads to be downloaded each time and set the total number of ads to be cached in the mobile devices. In addition, only the essential attributes are downloaded to the list and stored in the cache of mobile devices. The long text descriptions and images of ads, which are expensive in terms of storage and bandwidth, are only retrieved from the ad servers when the relevant ad is selected and needed to be shown on the mobile device.

3.4.2. Local ads filter: select and display the most relevant ad on a mobile client

The process of the local ads filter stage is similar to the existing ad selection process utilised by ad-networks: the system holds a set of ads and tries to select the most relevant one by analysing the fine-grained user profile.

Realistically, ad-networks would not be prepared to disclose their ad selection strategies. As such, we describe a simple approach to show how the complete process of ad selection can still be accomplished.

A pool of ads is maintained on the mobile client to store a potential list of ads from different ad-networks. After an ad request is generated, AdSelector first picks up the list of ads of the involved ad-network from the pool. As the list is ordered and each ad preserves its own rank computed by the ad-network in the pre-download stage, AdSelector can simply filter out ads that do not match the fine-grained user profile, do not match the ad type and format of the ad view in the current page, and do not match the right cache duration. Thus, the ranking information is still maintained in the shorter but more precise list. The client can then show ads from the shortlist according to their ranks.

In addition, some other mechanisms can be applied at the same time to revise the result. For instance, a time-weighted mechanism generates a different result from the same keyword according to the time of day (e.g. the result of searching for the keyword Food in the early morning may be different from searching for the same keyword at midday). The weight of different factors that a user sets via the subscription mechanism can also alter the ad selection result. (We do not discuss this further here, as our the focus is the privacy-preserving mobile advertisement selection mechanism, rather than a specific ad selection algorithm.)

Apart from the local ads filter process, some strategies are also performed to maintain the pool of ads. For example, the ads view history is recorded to facilitate the removal of particular ads that are frequently displayed but rarely clicked.

3.5. Privacy-conscious billing

The key to enhancing the privacy of the billing system is to ensure that ad-networks can obtain the information of which ads are viewed or clicked in which apps, without identifying which users performed the operations. To achieve this, AdSelector utilises the Trustworthy Remote Entity (TRE) approach of [27].

A TRE is a computational and communication system that enhances privacy in communication exchanges. The TRE system is situated between two or more communicating parties to perform information processing. The TRE provides strong guarantees of its trustworthiness by using technologies and

approaches from the field of Trusted Computing — the communicating parties verify the state of the intermediary to confirm its trustworthiness, rather than relying on a trusted third party.

In our system, the operation data of ads is first processed by the TRE system. A privacy-aware version of the data is then submitted to related ad-networks. The billing process is briefly described as follows.

1. A user views or clicks on an advertisement.
2. The view/click report with the real user ID is submitted to a TRE.
3. The TRE generates a single-use random ID for the user, and stores the mapping of the real user ID and the generated single-use ID for the purpose of detecting advertising frauds or tracing malicious behaviours.
4. The TRE submits the view/click report with the generated single-use ID to the ad-network.
5. The ad-network uses the report as proof to bill advertisers and pay app developers.
6. The ad-network can verify the state of the TRE to confirm its trustworthiness in each stage of the process.

3.6. Advertising fraud defence

By importing the TRE system, AdSelector obtains the ability to record ad-views and ad-clicks without disclosing information about users. However, the existence of advertising fraud prevents the stakeholders of the TMA ecosystem from simply using these records as billing proofs.

It has been shown that advertising fraud, particularly view-fraud and click-fraud, are causing serious damage to the TMA ecosystem [34, 35, 36]. The main revenue models of TMA are threatened by advertising fraud in the following ways.

1. **Cost-Per-View** (CPV), also known as Cost-Per-Impression (CPI), involves an ad-network charging an advertiser when an advertisement occurs on a user's mobile screen, no matter if the advertisement is clicked or not. In general, this model is implemented on a Cost-Per-Mille/Thousand (CPM/CPT) basis, with the fee being charged for every 1,000 impressions of an advertisement. Frauds against the CPV model include: resizing an ad view to one pixel; covering an ad view with other content; positioning an ad view to a nonexistent coordinate; and listing dozens of ad views in the same page (some mobile ad-networks require that only one ad view can be shown on a single screen [37]). Thus, ad-impressions are recorded without the ads being shown to the users. Alternatively, users might be dazzled by countless ads shown on a single page.
2. **Cost-Per-Click** (CPC) involves the advertiser paying an ad-network when an end user clicks on an advertisement. Click-fraud is the main threat to the CPC model, whereby an attacker can simply click on an advertisement by hand or via automated scripts to consume the advertising budget of related advertisers.

While other revenue models exist — including Cost-Per-Action (CPA), Cost-Per-Sale (CPS), Cost-Per-Lead (CPL), Cost-Per-Install (CPI) and Cost-Per-Hour (CPH) — our focus in this paper is threats to the CPV and CPC models.

3.6.1. Defending against view-fraud

In the CPV model, an ad-network determines whether an impression of ads is reliable or not by considering several different factors. We abstract the related factors to form the content of an ad-view record with the following attributes.

1. **Ad ID and ad-network identifiers**. In the two-stage ad selection process of AdSelector, an advertisement is pre-downloaded before being displayed. The two identifiers of the displayed advertisement are used by the TRE to locate the source of the original copy and to identify the ad-network that needs to receive the related record.
2. **Ad type**. Ads are categorised into three types: raw text, image and video.
3. **Ad format**. Each advertisement can be associated with a particular format such as banner ads, interstitial ads, skippable video ads, and so on. Each format has its own valid range of size, location, and duration. For instance, a small banner of the ad-network AdSense [38] should be sized to 320x50 and always be vertically aligned at the top of the screen. An impression of a skippable video advertisement is only valid if its duration exceeds 30 seconds.
4. **Timestamps of ad pre-download, ad request, and ad display**. The three timestamps are used to attest the valid date and the corresponding information flow of the advertisement. They can be used to detect invalid displays and replay attacks.
5. **Duration**. The final duration of an advertisement is recorded when the ad view ends. In addition, AdSelector checks and updates the duration of a displayed advertisement every 3 seconds in case of abnormal shutdown of the app.
6. **Screen size and resolution**. This is used to compute valid size and position of the ads displayed on the particular mobile screen.
7. **Ad size and position**. This information is verified according to the ad format of the displayed advertisement and the configuration of the mobile screen.
8. **Device identifier**. The device identifier is the real user identifier in AdSelector. The TRE stores the mapping of the device identifier to the generated

single-use identifier for a short time to detect malicious behaviour.
9. **App identifier**.

To authenticate the ad-view record, AdSelector signs it with the private key of the user on the mobile client and verifies the signature on the TRE server. If the credibility of the ad-view record is confirmed, the TRE will then replace the Device identifier with a single-use random identifier and submit the verified record to the related ad-network. Thus, the ad-network obtains reliable ad-views for charging advertisers. Figure 7 illustrates this workflow.

3.6.2. Defending against click-fraud

The record of ad-clicks in the CPC model is generated in a similar way to the generation of the record of ad-views in the CPV model. The following attributes of the ad-clicks record differ.

1. The **ad click timestamp** is added to complete the cycle of processing ads.
2. The **duration** in the ad-clicks record is the difference between the ad click timestamp and the ad display timestamp.
3. The **ad click location** records the position where the user touches the screen. The location of the click should be in the area of the displayed ad view to be valid.

In addition, ad-networks can predefine some policies to exclude some real but invalid click operations. For example, a user clicking on an ad view only half a second after it is displayed may indicate a lack of any genuine interest in the advertisement.

3.6.3. Additional defence mechanisms

The above defence mechanisms help to ensure that the ad-views and ad-clicks are performed by real mobile users instead of being generated by scripts. However, there is still a possibility that the clicks are performed manually by attackers. Hence, we provide some additional approaches such as pattern matching, blacklisting, and bait ads as compensation mechanisms.

The TRE stores the mapping of real user IDs and single-use random IDs for a short time. Therefore, the TRE system has the ability of backtracking users' operation sequences. If the ad-networks predefine some suspicious patterns of actions, the TRE can then perform pattern matching to detect certain types of frauds.

For instance, a single user clicking on several ads in a row or different users clicking on the same advertisement (or different ads, but published by the same advertiser) in a short period of time are both suspicious patterns for some ad-networks. Hence, the records of ad-clicks in these patterns might be ignored immediately by the TRE before submitting them to the ad-networks. If a user is confirmed or identified as a malicious user, the TRE system can add the real user ID to a blacklist and simply block all future ad-clicks from this user. In addition, ad-networks can make use of bait ads [25] or bluff ads [13], which are both targeted ads with irrelevant ad content, to detect suspicious users that have the potential to be added to the blacklist.

4. SIMULATION AND EVALUATION

An initial prototype of AdSelector has been implemented in the context of the aforementioned PPTMA. The evaluation has been primarily concerned with establishing the feasibility of the approach, rather than, for example, concerning itself with the specifics of ad selection algorithms. The performance overheads of some important operations (e.g. local ad selection and click report obfuscating) have been given due consideration.

4.1. Simulation setup

The prototype consists of three modules: an ad-network server, a server for a billing system, and a mobile device.

1. **Ad-network server.**
   The main functions of the ad server are maintaining data pertaining to ads, selecting targeted ads, and reviewing ad-click reports.
   In a fashion similar to the approach taken by Guha *et al.*, who use a trace of Bing search ads to evaluate Privad [25], we manually chose 100 real ads from Google AdWords to comprise the test data. To enlarge the sample, we also generated 9900 dummy ads with random targeting attributes.
   The ad selection algorithm that runs on the server is based on five targeting attributes: gender, location, age, interests, and subscribed ad categories. Other factors, such as the remaining ad budget, the ad-publish date, and the bidding strategy are not taken into account.
   The ad server runs on a virtual machine, the configuration of which is: 2.2 GHz single-core processor, 2GB DDR3 RAM, Ubuntu 14.04.4 LTS.
2. **Billing system server.**
   The billing system is a TRE instance with very minimal functionality: it replaces the user ID of the click report with a random single-use ID and stores the mapping of the two IDs. The billing system also runs on a virtual machine, with the same configuration as that of the ad-network server.
3. **Mobile device.**
   As mentioned in Section 3.2, AdSelector can be implemented in two distinct ways: as a third-party app or as an external library. In the initial prototype, the system is implemented as a third-party app.
   An app named PPTMA is installed on the mobile device; the app allows the user to manually edit the user profile and subscribe to interested ad categories. In addition, the user can

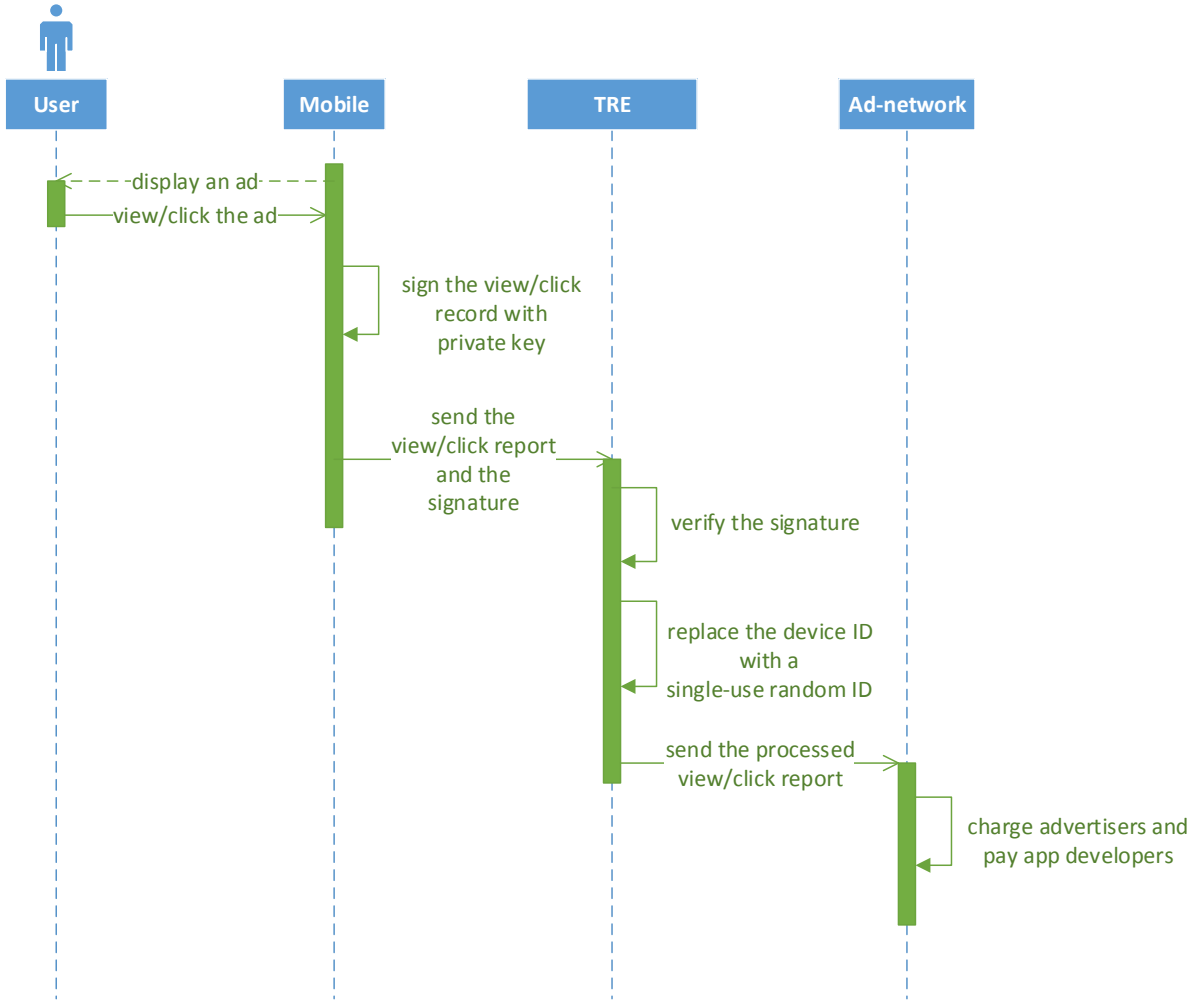

FIGURE 7. Flow of steps in processing the view/click reports

decide whether to activate the privacy-preserving mechanism or not.

A second app named App-with-ads is responsible for collecting user profiles and displaying ads, as per common apps with ad-plugins. App-with-ads interacts with the ad server and PPTMA by importing related SDKs provided by the systems.

The test mobile device is a Moto Nexus 6, the configuration of which is: 2.7 GHz quad-core Snapdragon 805 processor, 3GB LPDDR3 RAM, Android 6.0.1.

4.2. Simulation scenarios

We now present three illustrative scenarios.

4.2.1. Scenario 1

The aim of our first scenario is to observe which ad would be selected for the testing user profile without AdSelector, how much personal information would be collected by the ad-network, and how much time would be taken by these processes.

In this scenario: the device ID of the mobile device is collected as the user's ID by calling the related Android API; the user's gender and age are collected by using a questionnaire built into App-with-ads; the user's interests are deduced by analysing apps installed on the mobile device; and the user's location is obtained by calling the GPS sensor.

Users' information pertaining to gender, age and interests can be collected by applying different strategies, all of which have different time consumption. However, information pertaining to users' locations are collected by different ad-networks in the same way with relatively consistent time consumption. Therefore, we record the time cost in obtaining a location from a GPS sensor to represent the time cost of collecting a user's profile in this scenario.

The simulation was performed with 10 different user profiles. For the sake of brevity, we use only one of these profiles to illustrate the simulation:

ID: ZX1G52533F.
Gender: Male.

Location: Oxford, UK.
Age: 25.
Interests: Basketball and video games.

The following operations are performed in this scenario.

1. App-with-ads collects the testing user's fine-grained information and submits it to the ad server.
2. The ad server selects the best ad for the user and shows the ad on the mobile device.
3. The user clicks on the displayed ad and sends the ad-click report to ad server.

The results of the simulation are shown in Tables 1 and 2.

4.2.2. Scenario 2

In this scenario the privacy-preserving mode of PPTMA is turned on, meaning that the user can manually edit their fine-grained profile and generate coarse-grained copies. The aim of this scenario is to check, with the intervention of AdSelector, if the final selected ads are consistent with the displayed ad in Scenario 1. We also observe the differences of the disclosed personal information and the time consumption between this scenario and Scenario 1.

The user's fine-grained profile in this scenario is that of Scenario 1. The coarse-grained copy of the profile is edited as follows.

Gender: Decline to state.
Location: UK.
Age: 21-41.
Interests: Team sports and games.
Subscribed ad categories: Games > Computer & Video Games > Sports Games.
Number of ads that are downloaded to mobile per ad selection: 50.

In Scenario 2, App-with-ads collects both the user's fine-grained profile and the coarse-grained copy by calling the PPTMA API, as the information is already stored in the PPTMA app.

The operation flow of this scenario is as follows.

1. App-with-ads collects the user's coarse-grained information and submits it to the ad server.
2. The ad server selects the potential list of ads in the server, and sends the list to the mobile device.
3. Based on the fine-grained profile, App-with-ads selects the best three ads from the downloaded list.
4. The user clicks on the displayed ad (one of the best three) and the ad-click report is sent to the billing system.
5. The billing system replaces the user's ID from the ad-click report with a random ID and sends the obfuscated ad-click report to the ad server.

The results of the simulation are shown in Tables 3 and 4.

4.2.3. Scenario 3

The aim of the third scenario is to test the click-fraud defence mechanism. The following operations are simulated:

1. The user clicks the ad without reading the content (the user clicks in 0.5 second after the ad is displayed).
2. The user clicks a bait ad.
3. Several users click ads from the same advertiser in a short period of time.
4. The user clicks one ad several times in a short period of time.

The ad server maintains detailed information of ads and advertisers, while the billing system server holds the real user ID associated with the ad-click reports. Therefore, the detection of click-fraud attacks requires the cooperation of both servers.

In particular, the obfuscated user IDs involved in the first three operations can be detected by the ad server itself with simple SQL statements. The ad server can then send the IDs to the billing system and requires the billing system to add the original user IDs to a blacklist. In terms of the last operation, the process of the detection is reversed: the involved users are first detected by the billing system.

4.3. Evaluation and summary

The simulations show that the ads selected by AdSelector (Scenario 2) are consistent with the ad selected in the original TMA system (Scenario 1). In addition, the final selected ad 1 in Scenario 2 suggests that a more accurate ad can be delivered to the user due to the user's subscription mechanism — the ad pertaining to Basketball video game corresponds to the subscribed ad category Games > Computer & Video Games > Sports Games.

The differences in the user's disclosed personal information between Scenario 1 and Scenario 2 suggest that only the coarse-grained user profile can be obtained by the ad server. Furthermore, the ad sever cannot identify the real user involved in the coarse-grained profile because the user ID is obfuscated by the billing system.

The time cost of the operations involved in the first two scenarios suggests that, although the number of operations in Scenario 2 is twice that for Scenario 1, the performance overhead of the prototype is not significant. The time cost of obtaining a user profile with AdSelector is only 15% of doing that by calling the Android API and checking the GPS sensor. In addition, local ad selection is about 700 times faster than remote ad selection. A significant overhead is caused by ad-click report obfuscating. AdSelector divides the original ad-click report submitting process into two steps: submitting the report from the mobile device to the billing system, then submitting it again

TABLE 1. Time cost in Scenario 1

Term	Average time	Sample variance
Obtain location from GPS sensor	4.18 ms	2.46 (n=100)
Select the best ad in the ad-server	949 ms	284.83 (n=100)
Submit original click report to ad-server	965 ms	290.49 (n=100)

TABLE 2. Selected ad and disclosed personal information in Scenario 1

Term	Result
Final selected ad	Basketball shoe store in Oxford
User's information in the ad server	All information of the fine-grained profile

from the billing system to the ad server. Therefore, the cost of processing an ad-click report with AdSelector is about twice of that in the original TMA system.

The simulation result of Scenario 3 shows that the ad-click report obfuscating feature of the billing system does not affect its ability to detect click-fraud. Cooperating with the ad server, the billing system can support a variety of click-fraud defence mechanisms.

Another issue to be discussed is the consumption of storage and network bandwidth. Apparently, pre-downloading and caching a list of ads would cause additional consumption in our model. Compared to Adnostic [12], in which approach a list of n ads are fully downloaded, AdSelector only downloads the essential targeting attributes and the URL of each ad in the list. The rest of the ad data such as long text description, image or video, which is expensive in terms of storage and bandwidth, will be downloaded later when the relevant ad is finally selected for the user. Thus, the consumption of storage and network bandwidth, as well as the time cost, are reduced.

There are many popular formats of mobile ads, including raw text ads in minimalistic style, banner ads built with relatively small images, interstitial ads shown as full screen images, and video ads displayed with 10 to 30 seconds skippable/unskippable video stream. The consumption of storage and network bandwidth of an ad varies significantly due to its format. We randomly pick one ad of each format from our test ad-network server. The size of each ad is listed in Table 5.

Note that the number of ads pre-downloaded per ad selection is set to 50 in Scenario 2. This means that, irrespective of format, 50 pieces of targeting attributes and URL information will need to be downloaded from the server and stored in the local cache. Thus, an extra storage and bandwidth of 8KB (0.16KB x 50) would be consumed.

At first glance, it might appear that 8KB is significant, when compared to only 0.7KB required by a raw text ad. However, a few factors should be kept in mind.

1. Raw text is rarely used alone for in-app ads. When compared to the consumption of more popular ad formats — image ads or video ads — the consumption overhead only ranges from 1.4% to 28.5%. Note that, while we use compressed images and videos in the test ad-network sever, ads in a real ad-server may be more resource-intensive. For example, on Google AdWords, the allowed max size of ad image is 150KB, and a skippable video ad could be up to 3 minutes long. In such cases, the consumption overhead could be further decreased.
2. From a user's perspective, 8KB per ad selection is bearable in the current 3G/4G/WIFI network environment. The consumption is also not significant for most mobile devices with relatively large storage capacity.
3. We assume that there are only slight changes in people's interests during a short period of time. Therefore, in a certain time frame, which is configurable, there is no necessity to download and store another new 8KB data for a new ad selection. The previously downloaded data can be reused when retrieving ads from the same app, or from different apps associated with the same ad-network in the period of time.
4. As reported by ThinkWithGoogle [39] in 2015, the average app user has 36 apps installed on their mobile, but only 9 are used daily. Another report, by MobileMarketing [40] in 2016, suggests that the average UK adult has 27 apps on their smartphone, but uses only 6 daily. Based on the data above, we assume that a user runs all the 36 installed apps every day in the worst case, and all apps are associated with different ad-networks. If the time frame of caching the list of ads is set to 5 days, then the finally consumption of storage and bandwidth of pre-downloading ads would be 1728KB per month, which is also tolerable.

Hence, taking into account the factors above, we would argue that the overall consumption of storage and bandwidth should not cause significant concerns on the user side.

TABLE 3. Time cost in Scenario 2

Term	Average time	Sample variance
Obtain location from fine-grained profile	0.31 ms	0.24 (n=100)
Obtain location from coarse-grained profile	0.30 ms	0.29 (n=100)
Select and download list of ads in the ad-server	891 ms	247.03 (n=100)
Select the best ads in mobile	1.30 ms	1.58 (n=100)
Submit original click report to privacy-friendly billing system	867 ms	252.41 (n=100)
Obfuscate click report in billing system and send it to ad-server	931 ms	234.97 (n=100)

TABLE 4. Selected ads and disclosed personal information in Scenario 2

Term	Result
Final selected ad 1	NBA2K16 video game on game.co.uk
Final selected ad 2	Basketball shoe store in Oxford
Final selected ad 3	Basketball hoop at Amazon-UK
User's information in the ad server	Obfuscated ID and the coarse-grained copy of gender, location, age, and interests
User's information in the billing system	Original user ID

4.4. Limitations and challenges

In this paper, our focus has, somewhat inevitably, been on the users' perspective: that, after all, was the original motivation for the work. Many issues have yet to be explored from the perspectives of app developers and ad-networks. We outline initial thoughts in the regard now.

First, although AdSelector need not replace the existing infrastructure of ad-networks, slight modifications to business logic will be required. For example, new algorithms for selecting ads with coarse-grained user profiles need to be designed on the ad-networks' side, new workflow should be applied as the process of ad-selection is divided to two stages, and the app developers might need to update their apps with new Ad-SDKs that could support the changes. The modification work in the implementation phase might limit the deployment of AdSelector.

Second, while on the users' side the time cost and consumption of storage and bandwidth is acceptable, the additional 8KB consumption and the extra network request for every single user will be a considerable overhead for an ad-network with millions of users.

Third, AdSelector provide an additional infrastructure — the trustworthy billing system — to provide the feature of reporting ad click without revealing involved users. Planting trust in a third-party billing system would also be a factor that would need to be overcome in any real implementation.

Additionally, in the existing targeted mobile advertising ecosystem, ad-networks are able to obtain a huge number of fine-grained personal data. The user data is not only used in real-time targeting, but also used in other aspects such as designing marketing strategy or developing new advertising services. Applying privacy-preserving framework such as AdSelector would reduce their ability to collect users' personal information and consequently influence other business based on analysing fine-grained user data. Thus, provide ad-networks with enough incentives — for instance, lower users' hostility, higher response rates, and adapting to the trend of increasing privacy-conscious — for deploying our framework is also a challenge.

How to reduce the limitations and handle the challenges is an interesting focus for our future work.

5. RELATED WORK

In general, existing studies related to TMA can be classified into three categories: mechanisms for performing targeted advertising, mechanisms for preserving privacy, and mechanisms for compensating privacy leakage. Our framework leverages techniques from each: our aim is to combine different mechanisms to provide benefits to both consumers and advertisers in the context of TMA.

5.1. Mechanisms for performing targeted advertising

Location-based advertising (LBA) is an advertising approach that aims to deliver relevant ads of services or goods that can be accessed near to where a mobile user is geographically located. With LBA, ad-networks can display more accurate ads to mobile users when they are most likely to make a purchase. MobiAd [26] is an approach for performing localised and personalised advertising. A unique feature of MobiAd is that the downloading, displaying and click reporting of ads are all accomplished via a Delay Tolerant Networking (DTN) protocol, which provides anonymity for users when using wireless communication. MALCR [41] is another advertising approach based on a user's physical

TABLE 5. Consumption of storage and bandwidth of different ad formats

Term	Size
Targeting attributes information & URL for one ad in the pre-downloaded list of ads	0.16KB
Long text description for raw text ad	0.7KB
Image for standard banner ad	28KB
Image for full screen ad	74KB
Max size image allowed by Google AdWords	150KB
10 seconds video for video ad	191KB
30 seconds video for video ad	566KB

location. It makes use of two-level neural network learning to analyse users' profiles: the first neural network learns users' behaviours, and the second learns their interests. Apart from the particular technologies used in LBA, Zou *et al.* [42] conducted a randomised field experiment to analyse the LBA strategies crucial to advertisers.

Probabilistic reasoning advertising differs from LBA in that it aims to predict the best ads for a specific user by not only analysing the user's current location, but also by exploiting the user's behavioural patterns. For example, delivering an ad of a nearby pizza restaurant to a user by simply taking into account the user's current location and food preferences might not be appropriate, especially if the user has just visited another pizza restaurant. To this end, AdNext [11] provides a visit-pattern-aware mobile advertising approach. By analysing sequential visit patterns of users, AdNext can predict places that they are likely to visit in a city and deliver them relevant ads. The key mechanism of AdNext is a probabilistic prediction model that helps to make the prediction on the basis of users' visit histories.

User-profile-based advertising aims to exploit users' information to construct stereotypes. A user profile might indicate the user's age, gender, income, interests, pattern of daily routine, etc. MALCR [41] filters ads based on some relatively simple user attributes. Another approach developed by Bilchev and Marston [43] also delivers personalised advertising based on user profiles, with a key feature being that the approach is sensitive to the fact that a single user's profile might be distributed across devices and accounts.

5.2. Mechanisms for preserving privacy

Hybrid personalisation mechanisms are applied in a number of contributions to protect users' personal information while serving ads. The hybrid personalisation mechanism allows users to pre-download several ads from ad-networks with a generalised context, and then select the best one on the client on the basis of accurate user information. Adnostic [12], Privad [25] and MobiAd [26] are examples of this approach.

Data aggregation mechanisms are also widely applied in TMA as a privacy preserving data collection method. The data aggregation approach ensures that only statistics over the data contributed by multiple mobile users can be collected by advertisers, while individuals' personal information is preserved in their own devices. As an example, Zhang *et al.* [44] proposed such an approach that makes use of information hiding and homomorphic encryption to guarantee the data privacy of mobile users.

5.3. Mechanisms for compensating privacy leakage

Valuations of users' privacy are treated as a part of the users' utility by Nissim *et al.* [45]. This view suggests that users should be able to trade their privacy data for benefits according to their own wishes. To this end, several mechanisms are proposed in TMA for performing targeted advertising by compensating users' privacy leakage. For example, in the targeted advertising framework proposed by Wang *et al.* [46], the ad broker (ad-network) receives money from advertisers and pays compensation to users for clicking related ads. Thus, users can determine their clicking behaviours based on the amount of compensation and level of sensitivity of their exposed information. Although users are aware of their privacy leakage, the compensation mechanism encourages them to get involved in the privacy trading process.

6. CONCLUSIONS

In this paper we have built upon the contribution of [5], which describes PPTMA — a Privacy-Preserving Targeted Mobile Advertising framework. Specifically, we have described a profile-based mobile advertisement selection mechanism, AdSelector, which aims to enable mobile users to take advantage of useful advertisements without disclosing their personal information. AdSelector is an instance of the privacy-aware ad selection mechanism that can be supported by PPTMA. The main functionality of AdSelector utilises the centralised management of personal information and fine-grained access control provided by PPTMA.

AdSelector consists of several mechanisms. The user subscription mechanism helps to increase user engagement; the two-stage ad selection process enables

users to obtain accurate ads based on their fine-grained profiles without disclosing the profiles to advertisers; the trustworthy billing system provides reliable ad-view and ad-click reports without revealing the involved users, and also assists in detecting advertising frauds. We have implemented an initial prototype for Android to simulate and evaluate AdSelector.

In the initial prototype we have applied a simple approach — in terms of a filtering ordered list — to illustrate the local ad selection stage. The immediate focus of our future work will be the development of ad selection algorithms on the basis of a user's privacy preference, which can be applied to replace the filtering approach. In addition, more work will be done in order to validate our privacy-preserving framework. In this respect we plan to run a series of field experiments. We will also perform a user study to explore the direction that should be followed to foster adoption of our framework. Finally, we intend to develop formal models of both the TMA system and the PPTMA system to better understand the interactions of ad-networks and mobile users, to specify privacy-related operations and workflows, and to provide increased assurance.

In conclusion, we believe that, given the continued growth of targeted mobile advertising, mechanisms such as the one described in this paper will have an important role to play in terms of balancing the drivers of the economic model and the requirements of user in the emerging TMA ecosystem.

ACKNOWLEDGEMENTS

The authors would like to thank the anonymous reviewers for their helpful and constructive comments. Yang Liu is grateful for his scholarship from the China Scholarship Council (File No.201508060193).

REFERENCES

[1] Bender, R. (2011). Mobile-ad market still faces hurdles. `http://online.wsj.com/article/SB10001424052748704900004576152132113932442.html`. [Last accessed October 2015].

[2] eMarketer (2016). US digital display ad spending to surpass search ad spending in 2016. `http://www.emarketer.com/Article/US-Digital-Display-Ad-Spending-Surpass-Search-Ad-Spending-2016/1013442`. [Last accessed June 2016].

[3] eMarketer (2016). Mobile is driving UK ad spend growth. `http://www.emarketer.com/Article/Mobile-Driving-UK-Ad-Spend-Growth/1013685`. [Last accessed June 2016].

[4] Leontiadis, I., Efstratiou, C., Picone, M., and Mascolo, C. (2012) Don't kill my ads!: Balancing privacy in an ad-supported mobile application market. *Proceedings of the 13th Workshop on Mobile Computing Systems & Applications (HotMobile 2012)*, San Diego, California, pp. 2:1–2:6. ACM, New York.

[5] Liu, Y. and Simpson, A. C. (2016) Privacy-preserving targeted mobile advertising: requirements, design and a prototype implementation. *Software: Practice and Experience*, **46**, 1657–1684.

[6] Dhar, S. and Varshney, U. (2011) Challenges and business models for mobile location-based services and advertising. *Commun. ACM*, **54**, 121–128.

[7] Scharl, A., Dickinger, A., and Murphy, J. (2005) Diffusion and success factors of mobile marketing. *Electronic commerce research and applications*, **4**, 159–173.

[8] Robins, F. (2003) The marketing of 3g. *Marketing Intelligence & Planning*, **21**, 370–378.

[9] Yan, J., Liu, N., Wang, G., Zhang, W., Jiang, Y., and Chen, Z. (2009) How much can behavioral targeting help online advertising? *Proceedings of the 18th International Conference on World Wide Web (WWW 2009)*, Madrid, Spain, pp. 261–270. ACM, New York.

[10] Goldfarb, A. and Tucker, C. E. (2011) Privacy regulation and online advertising. *Management Science*, **57**, 57–71.

[11] Kim, B., Ha, J.-Y., Lee, S., Kang, S., Lee, Y., Rhee, Y., Nachman, L., and Song, J. (2011) Adnext: A visit-pattern-aware mobile advertising system for urban commercial complexes. *Proceedings of the 12th Workshop on Mobile Computing Systems and Applications (HotMobile 2011)*, Phoenix, Arizona, USA, pp. 7–12. ACM New York,.

[12] Toubiana, V., Narayanan, A., Boneh, D., Nissenbaum, H., and Barocas, S. (2010) Adnostic: Privacy preserving targeted advertising. *Proceedings of the 17th Annual Network and Distributed System Security Symposium (NDSS 2010)*, San Diego, CA, USA. Internet Society, Reston.

[13] Haddadi, H. (2010) Fighting online click-fraud using bluff ads. *SIGCOMM Comput. Commun. Rev.*, **40**, 21–25.

[14] Hardt, M. and Nath, S. (2012) Privacy-aware personalization for mobile advertising. *Proceedings of the 2012 ACM Conference on Computer and Communications Security (CCS 2012)*, Raleigh, North Carolina, USA, pp. 662–673. ACM, New York.

[15] Martin, K. (2015) Privacy notices as tabula rasa: An empirical investigation into how complying with a privacy notice is related to meeting privacy expectations online. *Journal of Public Policy & Marketing*, **34**, 210–227.

[16] Pujol, E., Hohlfeld, O., and Feldmann, A. (2015) Annoyed users: Ads and ad-block usage in the wild. *Proceedings of the 2015 ACM Conference on Internet Measurement Conference (IMC 2015)*, Tokyo, Japan, pp. 93–106. ACM, New York.

[17] Leppaniemi, M. and Karjaluoto, H. (2005) Factors influencing consumers' willingness to accept mobile advertising: a conceptual model. *International Journal of Mobile Communications*, **3**, 197–213.

[18] Barutçu, S. (2007) Attitudes towards mobile marketing tools: A study of Turkish consumers. *Journal of Targeting, Measurement and Analysis for Marketing*, **16**, 26–38.

[19] Wang, K., Chen, S.-H., and Chang, H.-L. (2008) The effects of forced ad exposure on the web. *Journal of Informatics & Electronics*, **13**, 27–38.

[20] Nokia. New Nokia research shows consumers ready for m-marketing via mobile handsets. http://company.nokia.com/en/news/press-releases/2002/01/30/new-nokia-research-shows-consumers-ready-for-m-marketing-via-mobile-handsets. [Last accessed September 2015].

[21] Saadeghvaziri, F. and Hosseini, H. K. (2011) Mobile advertising: An investigation of factors creating positive attitude in iranian customers. *African journal of business management*, **5**, 394–404.

[22] Google. Google ad interest categories. https://support.google.com/ads/answer/2842480?hl=en&ref_topic=2971788. [Last accessed January 2016].

[23] Yahoo. Yahoo ad interest manager. https://aim.yahoo.com/aim/us/en/optout/categories. [Last accessed January 2016].

[24] Facebook. How to target Facebook Adverts. https://www.facebook.com/business/a/online-sales/ad-targeting-details#Interests. [Last accessed January 2016].

[25] Guha, S., Cheng, B., and Francis, P. (2011) Privad: Practical privacy in online advertising. *Proceedings of the 8th USENIX Conference on Networked Systems Design and Implementation (NSDI 2011)*, Boston, MA, USA, pp. 169–182. ACM, New York.

[26] Haddadi, H., Hui, P., and Brown, I. (2010) Mobiad: Private and scalable mobile advertising. *Proceedings of the 5th ACM International Workshop on Mobility in the Evolving Internet Architecture (MobiArch 2010)*, Chicago, Illinois, USA, pp. 33–38. ACM, New York.

[27] Paverd, A., Martin, A., and Brown, I. (2014) Security and Privacy in Smart Grid Demand Response Systems. In Cuellar, J. (ed.), *Smart Grid Security: Second International Workshop, SmartGridSec 2014, Munich, Germany, February 26, 2014, Revised Selected Papers*. Springer International Publishing, Cham.

[28] Zhang, L. and Guan, Y. (2008) Detecting click fraud in pay-per-click streams of online advertising networks. *The 28th International Conference on Distributed Computing Systems (ICDCS 2008)*, Beijing, China, pp. 77–84. IEEE.

[29] Pearce, P., Dave, V., Grier, C., Levchenko, K., Guha, S., McCoy, D., Paxson, V., Savage, S., and Voelker, G. M. (2014) Characterizing large-scale click fraud in zeroaccess. *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security (CCS 2014)*, Scottsdale, Arizona, USA, pp. 141–152. ACM, New York.

[30] Dave, V., Guha, S., and Zhang, Y. (2012) Measuring and fingerprinting click-spam in ad networks. *SIGCOMM Comput. Commun. Rev.*, **42**, 175–186.

[31] Schiff, A. (2015). Location inaccuracy is a bigger problem than fraud. http://adexchanger.com/mobile/location-inaccuracy-is-a-bigger-problem-than-fraud/. [Last accessed January 2016].

[32] Speretta, M. and Gauch, S. (2005) Personalized search based on user search histories. *Proceedings of the 2005 IEEE/WIC/ACM International Conference on Web Intelligence (WI 2005)*, Compigne, France WI '05, pp. 622–628. IEEE Computer Society, Washington DC.

[33] Google. Google AdWords. http://www.google.co.uk/adwords/. [Last accessed January 2016].

[34] Cho, G., Cho, J., Song, Y., and Kim, H. (2015) An empirical study of click fraud in mobile advertising networks. *Proceedings of the 10th International Conference on Availability, Reliability and Security (ARES 2015)*, Toulouse, France, Aug, pp. 382–388. IEEE Computer Society, Washington DC.

[35] Alrwais, S. A., Gerber, A., Dunn, C. W., Spatscheck, O., Gupta, M., and Osterweil, E. (2012) Dissecting ghost clicks: Ad fraud via misdirected human clicks. *Proceedings of the 28th Annual Computer Security Applications Conference (ACSAC 2012)*, Orlando, Florida, USA, pp. 21–30. ACM, New York.

[36] Kitts, B., Zhang, J. Y., Wu, G., Brandi, W., Beasley, J., Morrill, K., Ettedgui, J., Siddhartha, S., Yuan, H., Gao, F., et al. (2015) Click fraud detection: Adversarial pattern recognition over 5 years at microsoft. *Real World Data Mining Applications*, pp. 181–201. Springer International Publishing.

[37] Google. AdMobi behavioural policies. https://support.google.com/admob/answer/2753860. [Last accessed January 2016].

[38] Google. AdSense ad sizes available for mobile ads. https://support.google.com/adsense/answer/68727?hl=en-GB. [Last accessed January 2016].

[39] ThinkWithGoogle (2015). Mobile app marketing insights: How consumers really find and use your apps. https://think.storage.googleapis.com/docs/mobile-app-marketing-insights.pdf. [Last accessed January 2017].

[40] MobileMarketing (2016). Average brit has 27 apps installed, uses just six daily. http://mobilemarketingmagazine.com/average-brit-27-apps-installed-uses-just-six-daily/. [Last accessed January 2017].

[41] Yuan, S.-T. and Tsao, Y. W. (2003) A recommendation mechanism for contextualized mobile advertising. *Expert Systems with Applications*, **24**, 399–414.

[42] Zou, P., Xu, Y., Fang, Z., and Li, W. (2016) The effectiveness of location-based advertising: when, where, and to whom. *International Journal of Mobile Communications*, **14**, 273–290.

[43] Bilchev, G. and Marston, D. (2003) Personalised advertising-exploiting the distributed user profile. *BT Technology Journal*, **21**, 84–90.

[44] Zhang, L., Wang, X., Lu, J., Li, P., and Cai, Z. (2016) An efficient privacy preserving data aggregation approach for mobile sensing. *Security and Communication Networks*, **9**, 3844–3853.

[45] Nissim, K., Orlandi, C., and Smorodinsky, R. (2012) Privacy-aware mechanism design. *Proceedings of the 13th ACM Conference on Electronic Commerce (EC 2012)*, Valencia, Spain, pp. 774–789. ACM, New York.

[46] Wang, W., Yang, L., Chen, Y., and Zhang, Q. (2015) A privacy-aware framework for targeted advertising. *Computer Networks*, **79**, 17–29.