Demographic Information Inference through Meta-data Analysis of Wi-Fi Traffic

Huaxin Li, Haojin Zhu, *Senior Member, IEEE* and Di Ma, *Member, IEEE*

Abstract—Privacy inference through *meta-data* (e.g., IP, Host) analysis of Wi-Fi traffic poses a potentially more serious threat to user privacy. Firstly, it provides a more efficient and scalable approach to infer users' sensitive information without checking the content of Wi-Fi traffic. Secondly, meta-data based demographics inference can work on both unencrypted and encrypted traffic (e.g., HTTPS traffic). In this study, we present a novel approach to infer user demographic information by exploiting the meta-data of Wi-Fi traffic. We develop an inference framework based on machine learning and evaluate its performance on a real-world dataset, which includes the Wi-Fi access of 28,158 users in 5 months. The framework extracts four kinds of features from real-world Wi-Fi traffic and applies a novel machine learning technique (XGBoost) to predict user demographics. Our analytical results show that, the overall accuracy of inferring gender and education level of users can be 82% and 78% respectively. It is surprising to show that, even for HTTPS traffic, user demographics can still be predicted at accuracy of 69% and 76% respectively, which well demonstrates the practicality of the proposed privacy inference scheme. Finally, we discuss and evaluate potential mitigation methods for such inference attacks.

Index Terms—Privacy leakage, traffic analysis, demographics inference.

1 Introduction

The wide deployment of public wireless access points and the prevalence of portable mobile devices allow people to have ubiquitous wireless access to the Internet. According to a study from Juniper Research, it is estimated that by 2018 there will be over 10.5 million Wi-Fi hotspots owned by mobile operators worldwide [1]. It is also expected that the amount of smartphone and tablet data traffic on Wi-Fi networks will increase to more than 115,000 petabytes by 2019 [1]. Compared with 3G/4G services, Wi-Fi access is one user-preferred connectivity option when using popular applications due to its superiority of cost and connectivity.

While public Wi-Fi provides convenience and free access, it may potentially pose a serious threat to the privacy of mobile users by leaving their computers and other electronic devices open to hacking. Existing works have demonstrated it is feasible to deploy malicious Wi-Fi hotspots along with traffic monitors in a public area to passively eavesdrop network traffic and infer more sensitive information of users [2], [3], [5]. For example, an iPhone can turn itself into a Wi-Fi hotspot. If the iPhone user sets the session ID as "Starbucks Free WiFi", other people may be misled that it is a free Wi-Fi hotspot from a nearby Starbucks store, thus connect their phones to the iPhone. Even though there exist a series of security solutions which provide link-to-link security (e.g., WPA2-AES) and end-to-end encryption (e.g., HTTPS), mobile users are still facing a big security challenge due to the lack of security protection, inappropriate implementation of security protocols, or untrusted/fake hotspot service providers. Potential privacy leakage in public hotspots can be addressed by examining user activities such as web browsing, search engine querying, and smartphone apps' usage [2]. Most existing studies are based on assumptions of unencrypted traffic or a full knowledge of user behaviors, and they cannot work in the case of incomplete information [2], [5], [8].

In this study, we raise the following question: *can an attacker infer sensitive information (e.g., gender, age, or education background) of targeted users by observing the meta-data of Wi-Fi traffic (e.g., IP, Host)?* The answer to this question is not straightforward. Firstly, mobile users usually stay at hotspots for short durations and thus public Wi-Fi traffic represents a partial view of its full traffic. Secondly, since most of common network traffic does not have content payloads [4], only meta-data can be leveraged for analysis in a large scale. However, the meta-data does not contain information that directly reveals user demographics. Thirdly, the problem is more challenging in the case that a certain percent of websites utilizing HTTPS protocol to encrypt the browsing traffic, which prevents any external observer from accessing the traffic contents. According to a recent report in 2015, HTTPS traffic reaches 46% for browser traffic (increased 7% in 12 months) and 61% for app traffic (increased 9%) [9]. Due to these reasons, only less than 10% gender information of mobile users can be directly obtained through the analysis of Wi-Fi traffic content [2].

To answer the question above, we study how to infer user demographic information from meta-data of Wi-Fi network traffic and propose a novel approach in this paper. The proposed approach is motivated by the observation that even for the encrypted traffic, it is still possible for an eavesdropper to obtain the meta-data of Wi-Fi traffic, which leaves a new attack interface for both insider attackers (e.g., fake/untrusted service providers) and external attackers (e.g., external hackers who break the password). Our insight is that users sharing similar attributes usually have similar network characteristics. To achieve this, we extract four kinds of features which can create distinct signatures for different demographics. Then, we propose a novel demographic information inference scheme and evaluate the successful rate with comprehensive experiments. Our study is based on a large real-world dataset which involves 98 Wi-Fi access spots and 28,158

- *Haojin Zhu is the corresponding author*
- *Huaxin Li and Haojin Zhu are with Shanghai Key Laboratory of Scalable Computing and Systems, Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, 200240, China (E-mail: lihuaxin003@sjtu.edu.cn, zhu-hj@cs.sjtu.edu.cn).*
- *Di Ma is with the College of Engineering and Computer Science, University of Michigan-Dearborn, Dearborn, MI, 48128, USA (E-mail: dmadma@umich.edu).*

users.

The contributions of this work are summarized as follow:

- Based on a large-scale real-world dataset, we demonstrate that the network traffic originated from users with different demographics has distinct signatures. We select four kinds of features which can create distinct signatures for different demographics.
- We propose a novel inference scheme, which can learn users' demographics by just passively monitoring users' traffic flows, based on supervised machine learning techniques. This scheme employs a novel XGBoost model [43] to infer users' demographics. We evaluate results using our real-world Wi-Fi traffic dataset and show that the overall accuracy of inferring gender and education level can be 82% and 78% respectively.
- We measure to what extent demographics can be inferred through encrypted network traffic, such as HTTPS traffic. We perform simulation via considering the lower bound of information leakage, i.e., assuming all HTTP traffic is encrypted as HTTPS traffic. Surprisingly, users' demographics can still be predicted at the accuracy of 69% and 76% respectively.
- We study potential countermeasures against such inference attack and evaluate the effectiveness of such methods through simulations. This study aims to call for attention of the society and sheds light on privacy protection techniques.

In the current version of the paper (which is an extended version of the work in [7]), we propose a new inference framework based on XGBoost to achieve better inference successful rates and time performance to reflect the severity of privacy leakage. We also add experiments and discussion (not shown in [7]) about implementing the demographics inference approach. In particular, we improve feature engineering by applying various feature selection techniques to select proper features. We also apply cross validation to reliably show that the new model outperforms the previous one (in [3]). Furthermore, we propose a novel countermeasure by exploiting the adversarial machine learning in traffic data publishing as well as a new dummy traffic approach based on Laplace noise. We also evaluate the trade-off between privacy and overhead via extensive simulations.

The remainder of the paper is structured as follows. Section II introduces the background about traffic leakage and traffic analysis. Section III describes the proposed methodology, including data processing, feature selection, and inference model. Section IV presents datasets, experiments, and evaluations. We discuss and investigate the potential mitigation methods in Section V. Finally, we discuss related works in Section VI and conclude the paper in Section VII.

2 Background

2.1 Traffic Leakage in Real-World

Previous researches demonstrate the insecurity of public Wi-Fi, which may potentially leak user privacy information from Wi-Fi traffic. In the following, we summarize various cases of Wi-Fi traffic leakage.

2.1.1 Public Open Wi-Fi or Rogue Hotspots

Although there are existing Wi-Fi security solutions such as 802.11i proposed in 2004, open public Wi-Fi networks without any protection are still popular due to free and simple wireless connections. In fact, a typical selling point of many restaurant chains nowadays is that they offer free Wi-Fi connections to customers. In an open public Wi-Fi environment, wireless connections are vulnerable to man-in-the-middle (MITM) attack, which allows the attacker to tap into wireless channels and obtain the Wi-Fi traffic.

Unauthorized 'rogue' hotspots allowing back-door access to the network, and honeypot access points that lure users to connect to unsecured external networks, represent two other types of threats to Wi-Fi traffic. Rogue Wi-Fi containment is not an easy job in practice due to difficulty of accurate rogue Wi-Fi detections. There has been a high interest in the industry on improving security against rogue hotspots, e.g., using certificates to authenticate the Wi-Fi [36]. However, it is far from being widely deployed in practice.

2.1.2 Security Enabled Wi-Fi without Proper Implementation

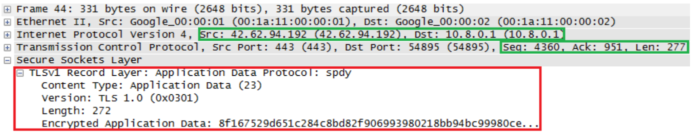
IEEE 802.11i provides important security features for Wi-Fi. However, without appropriate implementations, security vulnerabilities can still be exploited by the attackers. For example, using a pre-shared key (PSK) can be strong, but using a single passphrase limits security to its weakest link, the human factor. Further, protocol attacks ranging from key discovery to multi-layer Evil Twin impersonation are periodically being discovered [37], [38]. In the case of being hacked by the adversary, the Wi-Fi traffic will also be exposed to the attackers. Hotspot 2.0 [36] is expected to rely on a better technology and be able to overcome present vulnerabilities by encrypting every interaction and isolating all client sessions. Although technical security has been improved in comparison with the previous hotspot version, many issues still need addressing before its full deployment and usage in parallel with that previous version [39].

2.1.3 Untrusted Service Provider

For Wi-Fi service providers, an important business model is advertisement. According to use case reports from Cisco, new revenue can be earned by service providers through providing contextually-based advertisements to subscribers [40], or through deep packet inspection (DPI) [41]. Targeted advertisement is expected to be an important way for improving the CPM while targeted advertisement is based on users' locations and demographic information. Therefore, service providers have the incentive to collect users' traffic and infer corresponding information.

2.2 Meta-data Analysis

In this paper, the *meta-data analysis* is leveraged to infer users' demographic information. Meta-data is information about interactions through network connections, such as "Host", "IP address", "Port", "MAC address", "Seq", "Len" in traffic packets. We only consider meta-data rather than the payload of traffic packets here because of the following reasons. Firstly, according to [4], most of common network traces do not have content payloads. And content payloads do not necessarily contain information that reflects user's demographics [2]. Secondly, information in traffic payload can be highly sensitive, which may cause some legal



```
⊞ Frame 44: 331 bytes on wire (2648 bits), 331 bytes captured (2648 bits)
⊞ Ethernet II, Src: Google_00:00:01 (00:1a:11:00:00:01), Dst: Google_00:00:02 (00:1a:11:00:00:02)
⊞ Internet Protocol Version 4, Src: 42.62.94.192 (42.62.94.192), Dst: 10.8.0.1 (10.8.0.1)
⊞ Transmission Control Protocol, Src Port: 443 (443), Dst Port: 54895 (54895), Seq: 4360, Ack: 951, Len: 277
⊟ Secure Sockets Layer
  ⊟ TLSv1 Record Layer: Application Data Protocol: spdy
      Content Type: Application Data (23)
      Version: TLS 1.0 (0x0301)
      Length: 272
      Encrypted Application Data: 8f167529d651c284c8bd82f906993980218bb94bc99980ce...
```

Fig. 1: An illustration of traffic packet with HTTPS

issues. Finally, the meta-data analysis is easier to be performed automatically in a large scale, because the information in meta-data is usually well-formatted. So it is feasible to automatically process meta-data and apply learning techniques to investigate potential privacy leakage. Some existing works analyzed meta-data in network traffic from different perspectives [5], [12], [13].

Based on the discussion in this section, we propose a more scalable, larger coverage, and HTTPS-tolerant framework to reflect potential privacy leakage through demographic inference, which is presented in Section 3.

2.3 HTTPS Traffic

Utilizing SSL to encrypt traffic data is regarded as an important approach to enhance network security. With the popularity of HTTPS protocol, more and more websites employ the HTTPS protocol to secure the communication between servers and clients. HTTPS is the result of layering the HTTP on top of the SSL/TLS protocol, thus adding the security capabilities of SSL/TLS to standard HTTP communications. The main goal of HTTPS is to provide authentication of the visited websites and to protect the privacy and integrity of exchanged data.

With HTTPS, the content of packets, including the headers, request URL, query parameters, and cookies (which often contain identity information about users), are successfully masked via encryption, which is shown in the red box of Fig. 1. However, HTTPS cannot hide IP addresses, port numbers, and some statistics, such as Seq and Len, as shown in the green box of Fig. 1. In practice, this means that attackers can still acquire the IP address and port number of the Wi-Fi access point, or the web server that one is communicating with, as well as the duration of session and amount of data transferred during the communication.

3 METHODOLOGY

In this section, we first formulate the research problem, then discuss challenges, and present our approaches.

3.1 Goals of Study

We seek to raise and answer two key questions about user privacy in network traffic analyses:

1) According to the previous works, only less than 10% of users' demographic privacy is leaked through traffic content [2]. Since the meta-data are always available in the traffic packets, is it possible to infer the demographic information (e.g., gender, education level) of a large number of users by only leveraging the meta-data of Wi-Fi traffic (e.g., Host, statistic of traffic)?
2) Given raising awareness of network security and privacy protection, encrypted traffic (e.g., HTTPS) is regarded as an important approach to prevent privacy leakage. Can an attacker still infer the users' demographics through HTTPS traffic?

To answer these two questions, we need to first formulate the problem by proposing a traffic privacy model. Given a set of traffic data $\mathcal{P}$ generated by users $\mathcal{U}$ within a time duration $\mathcal{T}$, many meta-data fields $\mathcal{F}_i = \{f_1, f_2, ..., f_n\}$ in different layers' protocols, such as "Host", "User-agent" from HyperText Transfer Protocol (HTTP) in Application Layer, "Port", "Seq", "Len" from Transmission Control Protocol (TCP) in Transport Layer, "IP address" from Internet Protocol in Network Layer, 'MAC address" from Data Link Layer. These fields can be extracted from a sequence of traffic packets $\{p_1, p_2, ..., p_m\} \subset \mathcal{P}$. So a traffic profile of a specific user $u \in \mathcal{U}$ can be defined as a function extracting the meaningful fields $\mathcal{F}$ from traffic packets $\mathcal{P}$, i.e., $\alpha_u : \mathcal{P} \rightarrow \mathcal{F}$.

An attacker, under different scenarios, can obtain the traffic data $\mathcal{P}$ or capture a subset of all traffic packets, $\mathcal{P}_{cap} \subseteq \mathcal{P}$, from one or more sources of network traffic $\mathcal{L} = \{l_1, l_2, ..., l_q\}$. And meta-data fields $\mathcal{F}_{cap} \subseteq \mathcal{F}$ extracted from $\mathcal{P}_{cap}$ will be exposed to the attacker and leak privacy information directly or indirectly, from the perspective of an attacker. Under different conditions, $\mathcal{F}_{cap}$ contains different kinds of contents and different amount of information. For example, an attacker can obtain MAC address, IP address, Host, and User-agent in a traffic packet with HTTP, because these information can be extracted from Ethernet II frame, IP Protocol, and HTTP, from corresponding packet layers, respectively. Thus $\mathcal{F}_{cap}$ = {MAC, IP, Host, User-agent} given traffic packets under HTTP protocol. However, the attacker can't observe Host and User-agent in a traffic packet under HTTPS protocol because the Application Layer is encrypted in TLS/SSL. Nevertheless, the attacker can still observe the MAC address and IP address from other layers in a packet with HTTPS, i.e., $\mathcal{F}_{cap}$={MAC,IP}.

Using $\mathcal{F}_{cap}$, the goal of the attacker is to infer demographic information, which is considered as a kind of privacy leakage issues of mobile users in this work. Formally, it is a function β translating $f_i \in \mathcal{F}_{cap}$ into information which can be used to infer demographic information $\mathcal{DI}$: $\beta(f_i) \rightarrow \mathcal{DI}$.

3.2 The Approach Overview

In this section, we present the framework design, which extracts information from traffic and predicts users' demographics based on the meta-data of Wi-Fi traffic. Our framework aims to automatically extract information from traffic and generate profile signatures to predict users' demographics. Our idea is based on the fact that it is highly possible that users having similar demographics have similar network usages. Besides, mobility characteristics and network access behaviors will also share the similar properties of demographic information, which has been supported by the previous work on web browsing analysis [23]. So we apply supervised machine learning methods to infer users' demographics based on a trained predictor, as shown in Fig. 2. In machine learning, supervised learning aims to estimate a model (or a function) from labeled training data. It has a training phase and a testing phase. In the training phase, traffic with known users' labels is processed to train a supervised model as a predictor. In the testing phase, traffic with new users is given as the input of the predictor which then outputs the predicted labels of these users. The framework mainly includes data preprocessing, feature engineering, and demographics predicting.

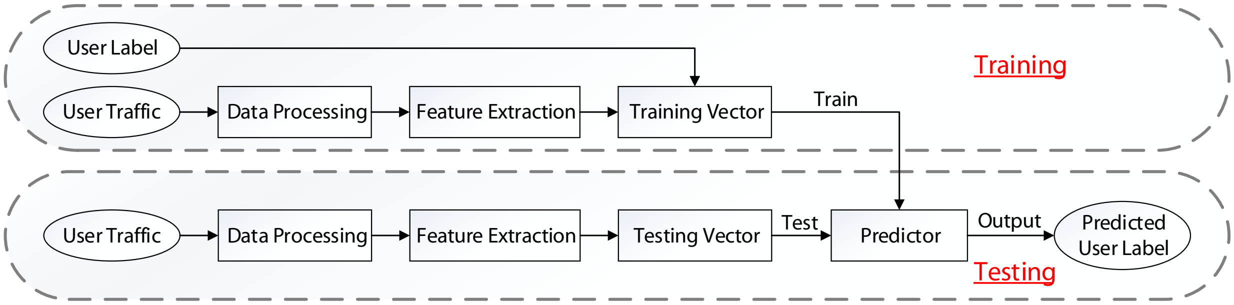


Fig. 2: Framework design

3.3 Data Preprocessing

In order to correctly and conveniently leverage and analyze the data, the first step is to process, clean, and transform raw data into formatted data. Given a series of traffic as input, traffic packets are parsed and targeted fields are extracted for the specific usage. For example, MAC addresses are used to identify devices and aggregate flows from the same devices or the same IP addresses. Then we extract targeted meta-data such as URL, Host, Port, and User-agent in HTTP protocol and preserve the statistics of the traffic packets. We also remove duplicate records or missing value of network connection, and handle some operations including aggregating domains addresses from the same service providers. For example, “a.domain.org” and “b.domain.com” are two addresses from a same application’s different servers, we aggregate them according to the text similarity. Our framework can also be deployed either in an ISP or a Wi-Fi hotspot to process traffic in a real-time manner.

3.4 Feature Engineering

Given the sanitized traffic data after preprocessing, we extract features for machine learning and classify them into four categories: application-based features, category-based features, location-based features, and statistical features. We describe these features, justify reasons that we select these features, and explain the feature selection in this section.

3.4.1 Application-based Features

Application-based features are extracted from hosts of the HTTP protocol. The hosts reflect application usage of users. It usually describes which websites users visited or which applications users ran. Preferred or frequently used applications show strong tendency towards different groups of users, which own different attributes of demographics, such as gender and education. To characterize the tendency between different attributes, *entropy* of each application is calculated as follows.

$$\varepsilon(A) = -\sum_{a\in A} \theta(a) log_2 \theta(a) \qquad (1)$$

where A is a kind of demographic attributes, e.g., gender $A = \{\text{male, female}\}$ or education $A = \{\text{bachelor, master, doctor}\}$, and θ is the user distribution of an attribute $a \in A$.

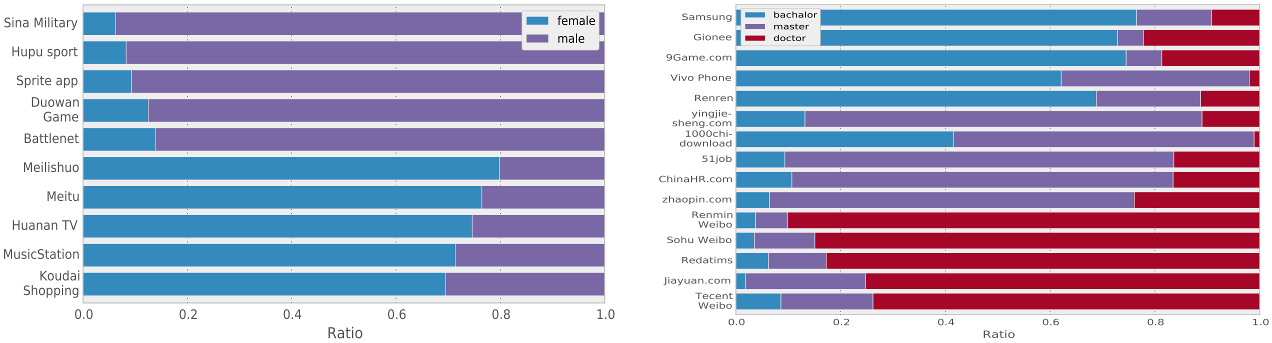
Entropy measures the amount of uncertainty of an attribute. Entropy has the minimum value when the probability of one tendency is dominant and has the maximum value when the probability of each tendency follows a uniform probability. So the lower entropy of an application indicates it is distinguishable with respect to an attribute. Fig. 3(a) shows the 5 applications with the lowest entropy for male users and female users respectively. In general, *Game* and *Sport* applications are more popular among male users while *Fashion* and *Shopping* applications are more popular among female users. Fig. 3(b) shows the 5 applications with the lowest entropy for the users with education levels of bachelors, masters, and doctors, respectively. It can be observed that bachelors are more interested in *Electronic Product*, while *Job* applications are the most popular among masters, and *Marriage* applications are the most popular among doctors.

Feature representation and selection: Since applications that users used are categorical variables (i.e., users might be observed to use different applications, and applications are independent to each other), we transform them into features for machine learning model using the widely-used *one-hot encoding* method. One-hot encoding encodes categorical variables as numerical variables in order to be used as features in any given model. For example, assuming applications that are taken into consideration include [Youtube, Netflix, Facebook, Twitter], user A used Youtube and Facebook, while user B used Netflix and Twitter. So the feature vector of A is [1, 0, 1, 0] and feature vector of B is [0, 1, 0, 1], where 1 indicates the presence of the corresponding application and 0 indicates the absence of an application.

However, since the number of applications that all users used can be very large, the generated feature set can be sparse and high-dimensional. As discussed above, some applications have a strong tendency towards one certain attribute, while some others not. Features with low tendency towards attributes will reduce inference accuracy and increase time complexity. So in order to reduce the dimension of feature vectors, we compute a score of *chi-squared* test for each one of the features [5], and select the best K features to train the predicting model.

3.4.2 Category-based Features

As application-based features describe detailed network usage of users, in a higher perspective, we classify applications in our dataset into 39 categories to discover whether different groups of users tend to prefer different categories of applications. Table 1 shows the list of categories in our study. To evaluate tendency of each category, we calculate *entropy* again. Fig. 4(a) shows 10 categories with the lowest entropy for the gender attribute. It shows that *Sports*, *Finance*, and *Real Estate* are the most popular ones in male users while *Women*, *Entertainment* are the most popular in female users. Similarly, Fig. 4(b) shows the 10 categories with the lowest entropy for education level attributes. It shows that *Social Networks*, *Job*, and *Finance* are the most



(a) High application tendencies in gender

(b) High application tendencies in education level

Fig. 3: High tendency application-based features

TABLE 1: Category-based features

Categories	Sports, Entertainment, Finance, Education, Job, Female, Estate, Automobiles, News, Forums, Social Networks, Technology, Music, Shopping, Family, Blogs, Portals, Communications, Downloads, Games, Mobile, Advertisement, Search, Email, Travel, Video, Reading, File-sharing, Fashion, CDN Services, Weather, Health, Politics, Environment, Public Welfare, Science, Others

popular among bachelors, masters, and doctors, respectively. The results show that the category-based features can also be used to distinguish different groups of users.

Feature representation and selection: We use one-hot encoding to encode category-based features. Given a smaller dimension of feature vectors (compared with application-based features), we can use all features to train the model or wrapper methods [46], which allows to detect the possible interactions between variables and obtain an optimal feature combination on training set, to evaluate subsets of variables .

3.4.3 Location-based Features

Locations usually have strong correlations with people's network usage [5], identities [47], or even hobbies [45]. If locations can be extracted from network traffic, the attacker can have more confidence to infer the demographics. In a Wi-Fi network, different access points are assigned different IP addresses, so IP addresses can be used to distinguish locations. Besides, traffic patterns also have high correlations with the semantic locations, which may indicate users' trajectories and location profiles [5], [16]. So locations extracted from the traffic are supposed to show strong correlations with demographics, thus we choose them as one of the features.

Feature representation and selection: Since location-based features are also categorical features, the simplest way to express them is to transform each location as one feature. If one user appears at a location, the feature value of this location is set as 1. Otherwise it will be set as 0. For the location-based features, how to perform feature selection depends on the scale of feature dimension. If the dimension is very high, we propose to use Filter method (e.g., chi-squared test) to filter out the redundant features. If the dimension is not very high, we can use wrapper methods [46] to select features according to their performance on training set.

TABLE 2: Statistical features

Mean	The arithmetic mean of the statistics at different connection
Standard Deviation	Standard deviation of the statistics
Skewness	Measure of asymmetry about mean
Kurtosis	Measure of the flatness or spikiness of a distribution
RMS	Square root of the arithmetic mean of the squares of the statistics at various connection
Max	Maximum statistics
Min	Minimum statistics

3.4.4 Statistical Features

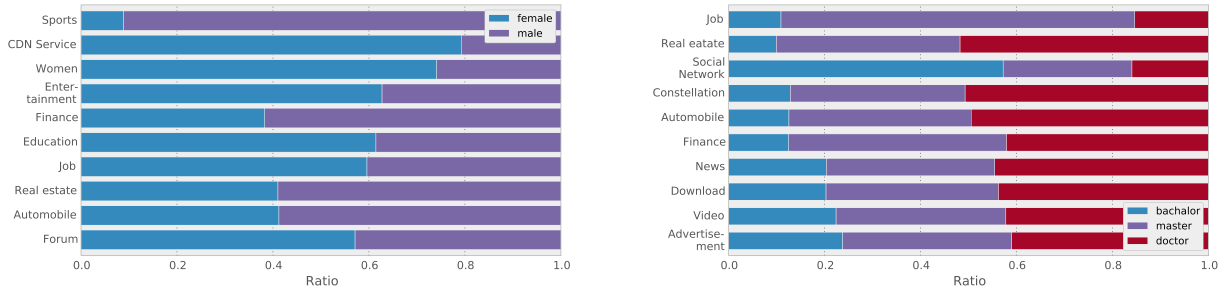
Besides the semantic features mentioned above, we can also characterize the traffic by statistics. Different applications have different network behaviors [48], thus generate different statistics such as the number of HTTP requests per flow or the size of HTTP requests per flow. And different groups of users can also have different network usage, as discussed above, so the statistics like connection time are diversified. As a result, statistical features may reveal distinct information that can distinguish users with different demographic information implicitly. Fig. 5 shows cumulative distribution function of different types of users' average time durations, HTTP number, and traffic packet size. For example, we can know the female users have larger network statistics than the male users, and masters and doctors usually have larger statistics than bachelors.

Feature representation and selection: Our previous work [7] adopted rounded statistics of each connection to one-hot encode features, which generates a very large feature space (i.e., dimension). In this paper, we calculate the characteristics of the network statistics for each user. These characteristics are listed in Table 2. Features are derived from each user's characteristics of connection duration, traffic packet size, and HTTP packet number.

3.5 Inference Model

As shown in Fig. 2, a set of training features from training data are used to train a predictor in the training phase, then the predictor predicts labels of testing data in the testing phase. Here, the goal of the predictor is to infer the users' demographics based on the supervised learning method, as mentioned in Section 3.1. Given a set of users $\mathcal{U}$, whose demographic information is known as prior knowledge by the adversary or third parties, the goal is to predict demographics of other users $\mathbf{u}$. So it can be formulated as a classifier Ψ which predicts demographic class labels $j \in$

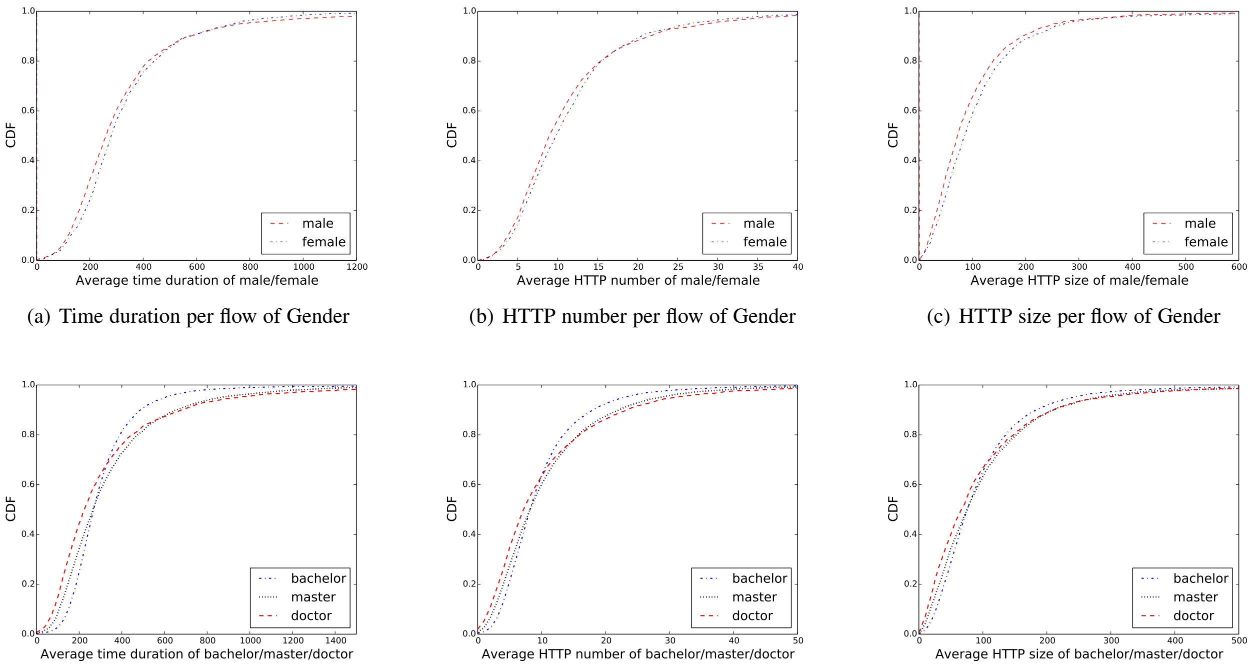
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2017.2753244, IEEE Transactions on Mobile Computing



(a) High category tendencies in gender

(b) High category tendencies in education level

Fig. 4: High tendency category-based features



(a) Time duration per flow of Gender

(b) HTTP number per flow of Gender

(c) HTTP size per flow of Gender

(d) Time duration per flow of Education

(e) HTTP number per flow of Education

(f) HTTP size per flow of Education

Fig. 5: Cumulative Distribution Functions of statistical features

$\mathcal{J} = \{1, ..., J\}$ at the input $\mathbf{u}$ over independent replicates of the learning set $\mathcal{U}$. The classifier is denoted as:

$$\Psi(\mathbf{u}, \mathcal{U}) = j \tag{2}$$

In the previous work [7], we adopted Random Forest model [42] as the Ψ. In this work, we enhance the inferring performance in both accuracy and time efficiency by applying the advanced XGBoost algorithm, which was proposed in [43]. XGBoost is an optimized distributed gradient boosting system designed to be highly efficient, flexible, and portable. Different from Random Forest which is based on the majority voting of Decision Trees, XGBoost is implemented on the ensemble and boosting of Classification and Regression Trees (CART) [50].

In the following, we first explain the reason we adopt the new algorithm, then introduce Classification and Regression Trees in XGBoost system, and describe the Gradient Tree Boosting process of XGBoost model.

3.5.1 The Rational of Choosing XGBoost Model

One of the major reasons of choosing XGBoost Model is to enhance the inference performance including both of accuracy and efficiency. On the one hand, as a classification model, the inference model is always expected to achieve better inference accuracy. On the other hand, time efficiency is also an important consideration for machine learning models, especially when the size of the dataset is large. If a model is time-efficient, i.e., costing less time for training and predicting, it can be more practical to conduct a large-scale demographic inference. Last but not least, the considered demographic inference question poses an unique challenge for the inference model, which has the sparse feature vectors due to the One-Hot Encoding. This makes XGBoost model quite suitable for the considered problem.

In particular, compared with the previous work in [7], the proposed new model has the following desirable merits:

- Improved Performance: Firstly, XGBoost is a sparse aware tree learning that can optimize for sparse data. Since we apply one-hot encoding on host fields to generate the application features, the features are usually very sparse because the number of applications that one user uses can be limited, thus there are frequent zero entries in the feature vector. In XGBoost model, a default direction is added in each tree node to handle sparsity of categorical encoding. Secondly, regularization function is added to the objective function [43]. This reduces complexity of the model and achieves a better bias-variance trade-off compared with traditional gradient tree boosting.

- Improved Efficiency: Different from Random Forest which is based on the majority voting of fully grown decision tree voting [42], XGBoost is based on Gradient-Boosted Trees (GBT), which generally outperforms Random Forest [44]. Compared with Random Forest, XGBoost tries to add new trees that compliment those already built ones. This normally gives you better accuracy with less trees. Besides, XGBoost also borrows brilliant ideas from Random Forest to make it a more powerful model. A column sub-sampling technique, which is commonly used in Random Forest model [42], is applied to tree boosting. The column sub-sampling not only greatly helps prevent overfitting, but also accelerates computations and speeds up the training and prediction process.

In Section. 4, we compare the XGBoost model with the Random Forest and some other popular models to prove the its superiority through experiments.

3.5.2 Classification and Regression Tree

Classification and Regression Tree (CART) is a model that employs simple if-then-else rules for both classification and regression. CART for regression in XGBoost model has same decision rules as in classification trees, but it contains one score that indicts the tendency to the labels in each leaf value. The tree model is a collection f of nodes n_i organized in a hierarchical tree structure. The node in the tree is either a split node (that provides a certain rule to split users with potentially different demographics in our problem) or a terminal leaf node (that outputs the predicted demographic information and its score). To construct a tree model, the split starts at the root node, then for each split node n_i, it finds a subset of all features that minimizes the sum of the node impurities in the two child nodes and chooses the split that gives the minimum overall impurities. Given a d-dimensional feature vector $\mathcal{M} = \{m_1, ..., m_d\}$, where m_i refers to the features we mentioned in Section 3.4, the splitting function $S(\mathcal{M}, \pi_i, \phi_i)$ can be represented as:

$$S(\mathcal{M}, \pi_i, \phi_i) = \begin{cases} 1 & \text{if } \mathcal{M}_{\pi_i} > \phi_i \\ 0 & \text{if } \mathcal{M}_{\pi_i} < \phi_i \end{cases} \tag{3}$$

where $\pi_i \in \{1, ..., d\}$ is the feature index, ϕ_i is the threshold to divide two classes, and 0 and 1 represent traversing to two child nodes respectively. In CART, Gini Index is computed for the splitting function. Gini Index measures how often a randomly chosen element from the set would be incorrectly labeled if it was randomly labeled according to the distribution of labels in the subset. Let $p(i)$ be the fraction of objects labeled with class $i \in \mathcal{J}$, Gini Index is computed as follows:

$$Gini = \sum_{i \neq j} p(i)p(j) \tag{4}$$

With training data, feature index π_i^* and threshold ϕ_i^* trained for this node are chosen as:

$$\pi_i^*, \phi_i^* = \operatorname*{argmax}_{\pi,\phi} S(\mathcal{M}_k, \pi, \phi) \tag{5}$$

Then, nodes will continue to split according to the rules until a stopping criterion is reached. In XGBoost model, each tree is defined as a vector of scores in leaves:

$$f_k(x) = w_{q(x)}, \quad w \in \mathbf{R}^T, q : \mathbf{R}^d \to \{1, 2, ..., T\} \tag{6}$$

where w represents the leaf score of the tree, q is the structure of each tree that maps an instance to the corresponding leaf index, and T is the number of leaves in the tree. For example, a score indicating a user's gender locates between [-1, 1]. If the score is closer to 1, the confidence that the user is a male user is higher; if the score is closer to -1, the confidence that the user is a female user is higher. When users' traffic data are put into the tree model, a user's output can be located at a leaf node that indicates the user is a male user with a score of 0.9, which shows strong confidence, while another user's predicting result reaches a leaf node whose label is female and its score is -0.3, but it has less confidence. The final demographics prediction of a user will be calculated by summing up scores in corresponding leaves (of all trees in the boosting).

3.5.3 Tree Boosting in XGBoost

Given a set of users and features extracted from their network traffic $\mathcal{U} = \{(\mathbf{x}_i, y_i)\}(\mathbf{x}_i \in \mathbb{R}^d, y_i \in \mathbb{R})$ with n samples and d features, the K-additive function is used to predict the user's label:

$$\hat{y_i} = \psi(x_i) = \sum_{k=1}^{K} f_k(x_i), \quad f_k \in \mathcal{F} \tag{7}$$

where $\mathcal{F} = \{f_k(x)\}$ is the space of CART. Different from Decision Tree, each CART contains a continuous score on every leaf, as mentioned above. And a user's final prediction result is summed by the leaf scores from all trees in the boosting, as shown in Equ. 7. For example, a user gets scores of 0.9 and -0.1 from two different CART trees, then this user's final score $0.9 + (-0.1) = 0.8 > 0$, so the user is predicted as a male user.

To learn the set of functions used in the model, the goal is to minimize the following regularized objective.

$$\mathcal{L} = \sum_i l(\hat{y_i}, y_i) + \sum_k \Omega(f_k) \tag{8}$$

where l is a training loss function that measures the difference between the prediction $\hat{y_i}$ and the target y_i, and Ω is a regularization function that penalizes complexity of the model (i.e., the regression tree functions) and helps avoid overfitting. Given the number of leaves T and the L2 norm of leaf scores $sum_{j=1}^{T} w_j^2$, the Ω is:

$$\Omega(f_t) = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2 \tag{9}$$

By minimizing l, training data are fitted well to build a more predictive model. By minimizing the Ω, the model is simplified, leading to a smaller variance of future prediction. So Equ. 8 balances the trade-off between bias and variance [53]. The detailed calculation of Equ. 7 can be referred in [43].

4 Experiments and Evaluations

In this section, we first introduce the details of our Wi-Fi network traffic dataset collected from 28,158 users, then present experiments and evaluation results for demographics inference.

4.1 Dataset

The real-world Wi-Fi traffic dataset is collected from 98 Wi-Fi hotspots from a campus. Traffic from 28,158 users (accounts) within 5 months (2014.09-2015.01) is recorded. In total, it contains more than 12.7 million Wi-Fi connection *sessions*. A *session* here is defined as a continuous time duration in which a user connects to the Wi-Fi hotspot before the timeout. If timing out for more that 5 minutes, the next connection is considered as a new session. To preserve the privacy of the users, the traffic data are sanitized. Users' IDs are anonymized and personal identity related information or sensitive information is removed. After sanitization, each session contains meta-data including connection start time, duration time, IP address, server host, and some statistics such as packets size, HTTP flow number, and so on.

Meanwhile, the dataset also contains anonymized user attribute labels including gender and education level, which are recorded by the network center and represented numerically (e.g., 0 represents male users, 1 represents female users, etc.). These labels serve as ground truth results to evaluate the performance of demographics inference. Since a person may have multiple accounts in the system, we integrate accounts that belong to the same user and remove duplicate accounts. As a result, 22,843 users are included in our dataset for experiments, and the distribution of the users' demographic attributes is shown in Table 3.

TABLE 3: User demographic attributes distribution

Gender		Education		
Male	Female	Bachelor	Master	Doctor
11509	11334	11509	7896	3438
50.4%	49.6%	50.4%	34.6%	15.0%

4.2 Experiments

In our experiment, we represent each user as a feature vector using the features mentioned in Section 3.4, and train the XGBoost model mentioned in Section 3.5. In order to avoid overfitting and bias on sampling, classifiers used in our experiment are validated with 5-fold cross validation. Specifically, the 22,843 data are randomly divided into 5 different folds. For each fold, the other 4 folds of data are used as the training set to train the model. Then the trained model is used to validate the remaining fold that is taken as the testing set. We also tune parameters of models by performing grid searching and cross validation on the training set.

To quantitatively evaluate risks from inference attack, *Accuracy* is considered as an important metric. However, under the condition that classes of data are not balanced, accuracy might not provide a comprehensive view of the results. So we also use *Precision, Recall*, and *F1-score*, which are broadly used metrics in classification problem, to evaluate results.

- *Accuracy* is defined as the number of accurate predictions divided by number of all predictions.
- *Precision* represents the fraction of categories in inferred demographic attributes that match the users' real demographic attributes (the number of true positive results divided by the number of all positive results). It measures how precisely the demographics can be inferred.
- *Recall* is the fraction of categories in real user demographic attributes that are presented in inferred demographic attributes (the number of true positive results divided by the number of positive results that should have been returned). It represents the inference's coverage of the users' real demographics.
- *F1-score* is defined as $F_1 = 2 \cdot \frac{precision \cdot recall}{precision + recall}$. It can be interpreted as a weighted average of the precision and recall, where an F1-score reaches its best value at 1 and worst score at 0.

4.3 Evaluation Results

We first provide an overall result of the inference, and compare the model we used with other popular models. Then we analyze the performance by considering different time, locations, and whether the data are encrypted.

4.3.1 Overall Results

Our overall results for inferring demographics with all features and data are shown in Table 4. To compare our method, a dummy classifier that predicts by randomly guessing is used as the baseline classifier for comparison. For inferring gender, which is a binary classification, the dummy classifier achieves the accuracy of 50%. For inferring education level, which is a multi-class classification, accuracy of dummy classifier is divided by the number of possible classes, i.e., 33%.

As shown in Table 4, our model's accuracy for inferring user gender is 82%, which outperforms the dummy classifier by 32%. And our model is able to accurately predict the education level of more than 70% of users, in contrast with the 33% accuracy of the dummy classifier.

TABLE 4: Overall Results

Demographics	Accuracy	Precision	Recall	F1-score
Gender	0.82	0.82	0.82	0.82
Education	0.76	0.76	0.77	0.76

TABLE 5: Comparisons of different classifiers

Model	Attr.	Acc.	Prec.	Reca.	F1.	Avg. Time(s)
Decision Tree	Gender	0.70	0.72	0.72	0.70	1.29
	Edu.	0.66	0.69	0.66	0.66	
LR	Gender	0.79	0.79	0.78	0.78	2.35
	Edu.	0.73	0.71	0.73	0.71	
SVM	Gender	0.80	0.80	0.80	0.79	3063.06
	Edu.	0.70	0.72	0.69	0.70	
Bayes	Gender	0.77	0.77	0.77	0.77	0.22
	Edu.	0.62	0.61	0.62	0.60	
KNN	Gender	0.68	0.68	0.68	0.68	9.98
	Edu.	0.70	0.68	0.70	0.67	
Random Forest	Gender	0.78	0.77	0.78	0.76	60.63
	Edu.	0.72	0.74	0.72	0.71	
XGBoost	Gender	0.82	0.82	0.82	0.82	9.87
	Edu.	0.78	0.79	0.78	0.79	

To further demonstrate benefits of choosing the XGBoost model in our approach, we also compare results of different prediction models with the result of XGBoost model. We implement a set of popular classification algorithms (Decision Tree, Perception, Support Vector Machine, Naive Bayes, K-Nearest Neighbors).

These algorithms are implemented by *scikit-learn* package[1] in Python. Table 5 shows metrics and running time for executing training and testing on a server with Intel Xeon 2.4GHz 14-core CPU and 64GB memory (the time of data process and feature engineering has not been counted in). We evaluate and compare our model from the following aspects.

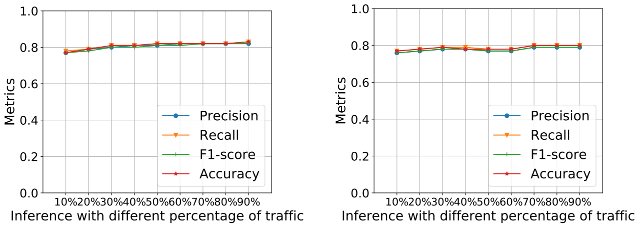
Inference Metrics: The XGBoost model achieves the best prediction performance among all classification models. Compared with the Random Forest model we used in the conference version [7], our new model increases accuracy to 82% and 78% respectively. The increase of inference metrics arises from the improvements in the tree ensemble process as mentioned in Section 3.5.1. XGBoost not only employs Gradient-Boost Tree to achieve low variance, but also employs regularization to reduce bias and avoid overfitting, thus it can perform well on testing sets or more general data. The bias-variance trade-off is well addressed in XGBoost [43], which is an important reason why XGBoost achieves the best inference metrics.

Time Efficiency: Considering the time consuming for training and inferring, XGBoost only consumes a considerable execution time. Compared with Random Forest, XGBoost only costs one sixth of running time, which shows the optimization in the implementation of XGBoost. Though some models achieve better time efficiency, they do not achieve as good inference metrics as XGBoost. And it is worth to mention that XGBoost can be executed in parallel, so it is expected to be more time efficient on large datasets. The better time efficiency of the new model makes the inference feasible to be extended to large-scale inference (e.g., in metropolitan area). These results support our choice of XGBoost model as the classifier (i.e., predictor) in our framework. More detailed analyses in different scenarios will be given in the following sub-sections.

Trade-off: The above comparisons have shown that XGBoost outperforms Random Forest in our problem from different aspects. Depending on the different design goals, XGBoost can achieve the trade-off between inference performance and computation complexity. Random Forest usually has only one hyper-parameter. Thus the number of the features to randomly select at each node needs to be tuned. Usually, the square root of the number of total features is chose as the parameter and it works very well in most cases [52]. For XGBoost, it has several hyper-parameters that include the number of trees, the depth (or number of leaves), and the shrinkage (or learning rate) to tune. So it is more time consuming to tune the algorithm to the maximum for each of the many datasets. The better results come from the well tuned parameters in XGBoost. Also, the more complicated model in XGBoost brings more application scenarios such as ranking and Poisson regression, in which Random Forest is harder to achieve [51].

4.3.2 Varying Time Durations for traffic leakage

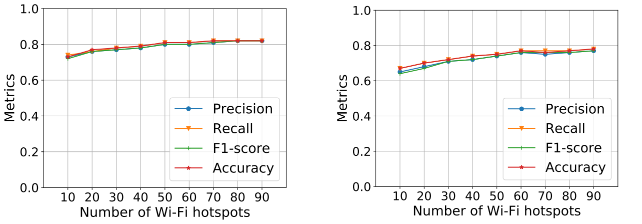
Firstly, we consider the leaking time of traffic as a variable. In practice, network traffic leakage may last for different time durations. For example, a victim can connect to a compromised Wi-Fi for different durations of time depending on his or her network usage or mobility patterns. And an attacker is able to choose to sniff Wi-Fi traffic for a long time in some cases while in other cases he or she can only sniff Wi-Fi for a while. So we consider different time durations of sniffing the traffic to mimic different traffic leakage scenarios. To simulate different time durations, we randomly select different percentages of traffic of each user from 10% to 100%. Results of predicting gender and education level are shown in Fig. 6, respectively.



(a) Gender

(b) Education

Fig. 6: Inference with different percentage of traffic



(a) Gender

(b) Education

Fig. 7: Inference with traffic from different number of Wi-Fi

The results show a slight increase of metrics including accuracy, precision, recall, and F1-score with the increase of leaking time duration. However, even a small percentage of Wi-Fi traffic achieves a high inference successful rate. For gender attributes, accuracy, precision, recall, and F1-score with 10% of traffic are 0.77, 0.78, 0.77, and 0.77 respectively, and only 30% traffic in our datasets reaches the metrics that exceed 0.80. For education attributes, the metrics with 10% of traffic are 0.76, 0.77, 0.76, and 0.77, respectively. Compared with our previous work [7], it shows a reasonable increase of inference performance when leaking time is increasing. It also shows that a small part of traffic is enough to achieve a considerable accuracy of demographics prediction. In other words, the traffic leakage during a short period is able to pose a serious threat to users privacy as well.

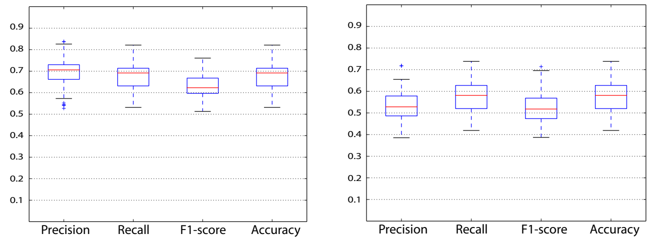
4.3.3 Varying Traffic Leaking Sources

Number of traffic leaking sources (e.g., compromised Wi-Fi hotspots) can vary in different scenarios and traffic leaking models. For example, an attacker can sniff several Wi-Fi hotspots or hack into a local network to obtain more traffic depending on the attacker's capability. Also, a victim can connect to one or more Wi-Fi hotspots during some time. More compromised Wi-Fi hotspots may provide more information to infer the demographics. To verify it, we classify network traffic according to the sources of Wi-Fi hotspots and perform demographics inference using traffic from different number of Wi-Fi hotspots. As shown in Fig. 7(a) and Fig. 7(b), metrics increase when the number of sniffed Wi-Fi hotspots increases.

In order to evaluate a lower bound on the privacy breach, we consider the scenario that an attacker only gets access to one Wi-Fi hotspot to sniff the traffic. So we use traffic from each Wi-Fi hotspot to infer users' demographics information respectively.

[1] scikit-learn: Machine learning in python. http://scikit-learn.org/stable/.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2017.2753244, IEEE Transactions on Mobile Computing



(a) Gender

(b) Education

Fig. 8: Inference with traffic from single Wi-Fi

Results are illustrated in box figures as shown in Fig. 8(a) and Fig. 8(b), where the minimum, first quartile (bottom edge of the box), median, second quartile (top edge of the box), and the maximum of metrics of inference using network traffic from the single Wi-Fi hotspot are indicated. For gender inference, the median precision exceeds 70% and the maximum precision and accuracy both exceed 80%. For education level inference, the median accuracy exceeds 55% and the maximum accuracy exceeds 70%. The results show that the attacker still has a high chance of breaching user privacy even if only one Wi-Fi hotspot is compromised.

TABLE 6: Results of prediction in HTTPS traffic

Demographics	Features	Pre.	Rec.	F1	Acc.
Gender	location-based	0.67	0.69	0.67	0.69
	statistics	0.61	0.66	0.62	0.66
Edu.	location-based	0.75	0.76	0.75	0.76
	statistics	0.53	0.57	0.54	0.57

4.3.4 HTTPS Traffic

To further analyze the extent of privacy leakage through network traffic, we consider a scenario that all HTTP traffic is encrypted as HTTPS traffic. As discussed in Section 2, not only plain text data but also some semantic fields are encrypted in HTTPS, which prevents the direct privacy leakage due to content analysis on network traffic. Whereas, there is still information that can be extracted from a HTTPS-enabled network traffic packet, such as MAC address, IP address, and statistics (e.g., packet length, connection time, etc.). So in HTTPS network traffic, it is possible to infer a user's demographics by observing the meta-data. We use IP addresses to generate location features and summarize statistics to get statistical features, as introduced in Section 3.4. We assume that all HTTP packets are encrypted as HTTPS packets, which is a lower bound of privacy leakage for HTTP traffic and no semantic information is considered. Results are shown in Table 6.

The results show that even in encrypted traffic, demographics can still be predicted to a considerable extent. Location-based features reach high accuracy, i.e., 69% for gender and 76% for education level. Obviously, the location-based features perform better than statistical features. It is reasonable because mobility directly relates to people' demographics while statistics of traffic packets are implicit reflection of network activities and not so distinguishable for demographics. But if we only consider predicting results of statistics features, the accuracy still achieves 66% for gender attributes and 57% of education attributes, which are obviously higher than baseline by 16% and 23%. It shows that relying on encryption cannot address all of the problems and the attacker can still infer the users' demographics by observing the encrypted data.

4.4 Implication

A major concern or limitation about the dataset is that the source of data is from a campus, which may result in the bias of our dataset. However, we argue that it does not invalidate our approach or privacy inference through network traffic analyses. Our study is based on the observation that users sharing similar demographics usually have similar network usage, which has been partially validated by previous works under different contexts such as web browsing, smartphone apps, and mobile social networks [23], [24]. Therefore, our proposed approach can be applied to other datasets though the considered features may have some differences. Our study confirms that the threat of leaking users' sensitive demographic information through the traffic analysis is realistic. Because of the large scale of our dataset, we also believe that bias can be limited for the large coverage of people. As one of our future works, we will consider a more resourceful adversary which can collect a large scale public traffic to have a better understanding on the impact of the different users on privacy leakage arising from network traffic.

5 Mitigation

In this section, we aim to propose some mitigation methods to limit the attacking capability of the attacker. We hope that the following discussions and experimental results would raise privacy awareness of the network traffic usage and would also inspire other researchers to find more advanced protection techniques.

5.1 Applying VPN or Tor

A potential strategy to mitigate the demographics inference attack in Wi-Fi traffic is to prevent the attackers from obtaining the meta-data of network traffic. So virtual private network (VPN) or anonymity network Tor can be used to prevent the attackers from tracking the routing information or collecting the traffic characteristics. However, such solutions may incur significant network overhead and suffer from reduced network performance. Typically, VPN speeds are much slower than those experienced with a traditional connection. The imperfect implementation and vulnerable software in current VPN and Tor products make them susceptible to a variety of practical user de-anonymization attacks [58].

5.2 MAC Address Randomization

Another strategy of thwarting the demographics inference attack is to avoid long-time tracking via MAC address Randomization, which prevents the attackers from linking a specific user with his or her Wi-Fi traffic. MAC randomization has been applied on various operating system including iOS 8, Android, Windows 10, and Linux with kernel version 3.18 [59]. With MAC randomization, a user's device obtains a randomized MAC address whenever establishing a new connection to a Wi-Fi access point. We perform an experiment by simulating this strategy. For a user's traffic recorded in our dataset, traffic from two consecutive different Wi-Fi hotspots are assigned to two different identifiers, which mimics the randomly assigned MAC addresses. From the attacker's perspective, more 'fake' users appear because of the

MAC address randomization, thus each user's captured traffic is limited. Then the proposed demographics inference is performed on the new traffic dataset. Our experiments show the accuracy is reduced by 11% for gender and 12% for education respectively, which shows preventing long-time tracking reduces the chance of privacy leaking.

However, MAC address randomization has been proved to be defeated under some circumstances according to the latest study [59]. For example, probe requests can be exploited to fingerprint devices and perform identifier-free tracking. New and more advanced techniques are still required for thwarting the tracking attack.

5.3 Adding Dummy Traffic

Encryption on traffic, such as HTTPS traffic, is a kind of methods to preserve privacy in network connection. Encryption can help hide semantic information, but statistical analysis can be exploited to breach the privacy [7], [48]. So we consider a natural method which is possible to mitigate the inference based on statistics of traffic. Dummy traffic is a technique to defend against statistical analysis attacks on network traffic [54], [55]. In this work, we continue investigating the effect of adding dummy traffic.

5.3.1 Dummy Traffic

Different from the previous version [7], we extend the dummy traffic countermeasure and consider two types of noise, uniform noise and Laplace noise, added by dummy traffic. For uniform noise, the statistics of added dummy traffic follow a uniform distribution $U[min(statistics\ of\ traffic), max(statistics\ of\ traffic)]$. For Laplace noise, we adopted an approach similar to differential privacy where we randomly selected statistics of added dummy traffic from a Laplace distribution [56]. According to the definition of differential privacy, a randomized function K gives ϵ-differential privacy if for datasets D_1 and D_2 differing on at most one element:

$$Pr[K(D_1) \in S] \leq e^{\epsilon} Pr[K(D_2) \in S] \tag{10}$$

where S is the range of the function K. By changing ϵ, we can control to what extent two statistics distributions are alike. For each kind of statistics, the noise is selected from the following Laplace distribution: $Lap(\mu; \beta)$ where $\mu = mean(statistics\ of\ traffic)$ and $\beta = (max(statistics\ of\ traffic) - min(statistics\ of\ traffic))/\epsilon$.

5.3.2 Evaluation

To evaluate the effectiveness of adding dummy traffic, we generate dummy traffic based on the two distributions (i.e., uniform noise and Laplace noise) respectively. Fig. 9 illustrates the effect of adding uniform and Laplace dummy traffic regarding of accuracy of the inferring education levels using statistical data, because the result of inferring education level is obviously higher than the baseline. By adding uniform traffic, the accuracy for inferring users' education levels can be reduced to 48%, while adding Laplace noise can achieve a better result as the accuracy is reduced to around 46%. So the Laplace noise achieves better results than the uniform noise when adding the same amount of dummy traffic. These results show that different distributions of added noise can achieve different effects. More sophisticated designs toward more effective inference resistance are desirable and deserve a separate work.

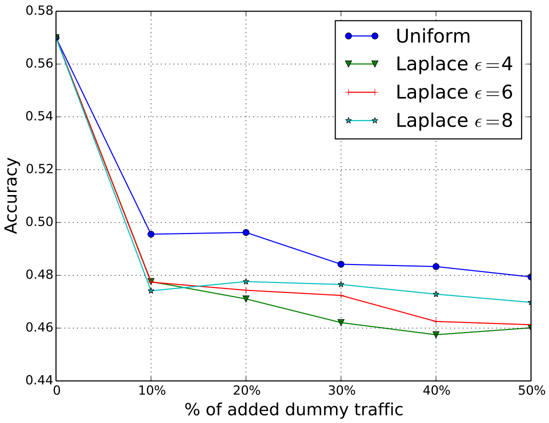


Fig. 9: Accuracy of inferring education level after adding dummy traffic

5.3.3 Discussion

Adding dummy traffic will inevitably increase the network workload and reduce its operational efficiency. In this work, we vary different percentages of added dummy traffic to mimic the extra network overhead. The 10% added dummy traffic might represent the slight extra workload onto the network's operations, while more added dummy traffic cause more overhead to the network. According to Fig. 9, adding 10% Laplace noise dummy traffic can reduce the inference accuracy to around 48%, which shows the effectiveness without adding too much overhead. And adding 40%-50% Laplace noise dummy traffic can further reduce the inference accuracy but it will generate more extra overhead and affect the operational efficiency. There is always a trade-off between privacy preserving and network utility, as some previous works tried to balance the trade-off and minimize the overhead [57]. Since our approach only adds some extra dummy traffic instead of modifying the statistics of existing traffic packets, it cannot largely affect the average statistics of each user.

5.4 The Countermeasure based on Adversarial Machine Learning

The above mentioned countermeasures can be applied in real-time and online scenarios. In this section, we propose a novel approach that can fool the adversary to have misleading inference results. This approach can work when the Wi-Fi network traffic is published as a dataset for study. In this case, the adversary is able to leverage the published dataset to train classifiers, and use these trained classifiers to infer demographics through other traffic sources. So before publishing the traffic data, countermeasures should be applied to reduce the possibility of demographics inference. Here, we consider to achieve this goal using adversarial machine learning concepts. The initial goal of adversarial machine learning is to ensure the effective machine learning techniques against adversarial examples from the adversary [60]. The adversary's goal is to fool the learning algorithm. For example, exploiting adversarial machine learning to produce the spam email that can avoid being detected. The adversarial examples are inputs to machine learning models that an attacker has intentionally designed to cause the model to make a mistake. For example, by adding a small perturbation that has been calculated to an image, an image of a panda will be recognized as a gibbon with high confidence [60].

5.4.1 Methodology

In our problem, the traffic data publisher acts as the role of adversary in adversarial machine learning and tries to build the dataset as an "adversarial example". In other words, the data traffic publisher processes traffic data so that attackers aiming at inferring demographics cannot leverage them to infer demographics accurately. Various approaches have been proposed to generate adversarial examples. For example, fast gradient sign method is the most popular method to generate adversarial examples [60]. However, this method requires modifying the data in dataset, which will reduce utility of network traffic if being applied in our problem. To avoid modifying traffic data and reducing the utility of network traffic dataset, we can choose to publish the datasets that will intentionally generate camouflage classifiers (i.e., classifiers that cannot achieve good inference results), which is inspired by poisoning attacks [61].

Specifically, we first train a linear SVM using whole dataset in our experiments. Then we select part of samples (depending on how many data need to be published) which are farthest from the classification hyperplanes. For example, those samples are most likely to be male users or female users, respectively. Formally, given the hyperplane $w \cdot x + b = 0$ and data $D = \{..., (x^{(i)}, y^{(i)}), ...\}$, we choose:

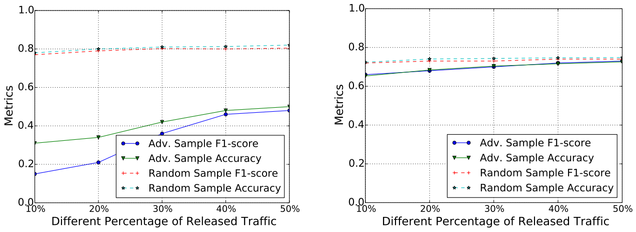
$$\underset{i=1,...,m}{\operatorname{argmax}} y^{(i)}((\frac{w}{||w||})^T x^{(i)} + \frac{b}{||w||}) \tag{11}$$

as $D' = \{(x'^{(1)}, y'^{(1)}), ..., (x'^{(m)}, y'^{(m)})\}$, where m is the number of chose samples. These samples are assumed to be released as the published dataset so that attackers cannot use them to train a high-accuracy inference model. The insight here is the hyperplane $w' \cdot x' + b' = 0$ for new classifiers cannot be accurately estimated given the intentionally selected dataset D', because it is biased from the optimal one, i.e., $w \cdot x + b = 0$. Note that attackers can select various supervised learning models other than the linear SVM. However, adversarial examples are able to generalize across models trained to perform the same task, even if those models have different architectures and were trained on a different training set [60], [62].

5.4.2 Evaluation

We implement the method mentioned above and simulate experiments. Different percentages of samples (i.e., m in Equ. 11) are intentionally selected through linear SVM models, and randomly sampled, respectively. These selected data are assumed as the published data. Then we use these selected data to train the XGBoost model we mentioned in this paper and infer on our testing dataset. The results are shown in Fig. 10(a) and Fig. 10(b). Compared with randomly sampling, F1-score and accuracy of inferring genders on intentionally selected samples are apparently lower. For example, the accuracy is reduced to around 50% when intentionally selecting 50% of data from our dataset, while the accuracy is around 80% when randomly selecting the same amount of data. The metrics of inferring education levels are slightly reduced compared with randomly sampling because it is a multi-class classification, so the hyperplanes are vulnerable to the selection of data. These results reflect the idea of adversarial machine learning can help resist inference attacks to some extents without modifying the published data.

In practice, users can adopt multiple defense strategies to enhance their privacy under the demographic analysis attack. It is noted that there is always a trade-off between privacy and utility. A practical security defense strategy should strike a balance among multiple factors, including users' convenience, privacy requirement, and network performance.



(a) Gender

(b) Education

Fig. 10: Inference experiments with sampled traffic

6 Related Works

Relevant to our work includes privacy analysis and quantization in network traffic, information inference via network traffic analysis targeting mobile devices, demographics inference through various techniques, and defenses against traffic analysis attack.

Privacy Analysis in Network Traffic: Privacy issues in network traffic are receiving increasing attention in recent years. From network traffic, information can be extracted to breach the privacy of network users. For example, Cheng et al. discovered direct information leakage through public Wi-Fi hotspots [2]. They performed experiments on Wi-Fi network traffic captured from 20 airport hotspots. Their analysis reveals that two thirds of travelers leaked privacy sensitive data by DNS queries, web browsing, or querying search engine. Our work further discovers the large scale demographics leakage through meta-data analysis and privacy inference. Das et al. [5] presented a system called PCAL (Privacy-Aware Contextual Localizer) which can learn users' contextual locations (e.g., residence, cafe) just by passively monitoring users' network traffic. They also used supervised machine learning techniques to predict the localizations based on a training set. Our work uses an advanced supervised machine learning technique (XGBoost) to predict demographics.

Besides personal information, many kinds of information are threatened through traffic analyses. Sensitive contents accessed by multi-device users were characterized and analyzed by the latest study [6]. Xia et al. [8] presented a framework that correlates the user identity extracted from the social network traffic to its online behavior by associating the browsing traces to OSN's ID of a user. Konings et al. [11] monitor the mDNS announcements from a semi-public Wi-Fi network deployed in a university. Their study reveals the fact that 59% of 2,957 unique device names contained both real names of users, and 17.6% of these device names contained first and last name of the user. Dai et al. [10] presented a novel technique that can automatically identify Android applications through generating network profiles using the HTTP traffic. Even though network traffic is encrypted, privacy violation [12], [13] and fingerprinting [14] are still possible. Different from previous works, our work takes a new approach to infer user demographic information by exploiting the meta-data of Wi-Fi traffic.

Information Inference via Network Traffic Analysis: Except direct information leakage and analysis from network traffic mentioned above, information inference through traffic analysis is emerging in recent years. For examples, Barbera et al. found that the vendor ID can be exploited to reflect the sociological aspects of the people like nationality, age, and socioeconomic status. So they performed language detection on the broadcast SSIDs to reflect the problem [15]. Musa et al. demonstrated that tracking unmodified smartphones using Wi-Fi monitors can be practical, economical, and accurate. In their work, second-by-second detections of a moving device are measured by Wi-Fi monitors, and a trajectory estimation method is proposed to produce the most likely spatio-temporal path [16]. Similarly, another work showed that, through a statistical analysis of a user's encrypted network traffic, the position of the user can be estimated by an adversary with accuracy of almost 90% [17].

The fingerprinting is usually an attack where the adversary attempts to recognize patterns of specific objects (e.g., a user, an app, a device, etc.) without knowing its ID. Supervised learning based approaches were proposed to fingerprint mobile applications using unencrypted network traffic [49] and even encrypted network traffic [48]. N. V. Verde et al. showed that it is possible to fingerprint NAT'd individuals when only NetFlow records are available [18]. T. Stöber et al. extracted side-channel features from network traffic generated from applications to fingerprint smartphone devices [19]. More surprisingly, by eavesdropping the network traffic of a device, the specific actions that a user is performing on his or her mobile applications can be accurately identified by an attacker. The experiment showed that the accuracy is higher than 95% [20], [21]. Our work also tries to address the potential and indirect privacy concern from the demographics inference through Wi-Fi network traffic.

Demographics Inference: Inference on demographic information has been discussed using various signatures. Meng et al. inferred demographic information by personalized ads used by the hosting applications. This is based on the observation that there are significant correlations between observed advertisements and the user's profile [22]. Hu et al. [23] extracted content-based features and category-based features from webpage click-through logs to infer users' gender and age. Seneviratne et al. [24] employed Naive Bayes model and Support Vector Machine to reveal users' gender from their installed apps. Schwartz et al. [25] applied differential language analysis on Facebook status update messages to predict user demographics. Bi et al. [26] showed that user demographic attributes such as age, gender, political views, and religious views can be inferred based on *Facebook likes* efficiently and accurately. Online social networks contain abundant information that can be used to infer demographics [27], [28], [29]. For example, Chaabane et al. [29] inferred OSN users' undisclosed (private) attributes (e.g., gender, relationship, age, and country) by exploiting public attributes (e.g., hobby) of other users who share similar interests. Different from previous works, our work selects Wi-Fi traffic meta-data, which can be sniffed passively, as features to reveal demographics leakage.

Defending against traffic analysis attack Previous works proposed methods to defend various traffic analysis attacks. Intersection attacks are possible when not all users of a service are active all the time and part of the transferred messages are linkable. Different strategies and techniques were proposed to resist intersection attacks [31], [32]. Besides, a novel privacy-preserving scheme in network coding, which applies homomorphic encryption operation on Global Encoding Vectors (GEVs), was proposed to resist traffic analysis attacks [30]. Mathewson et al. defended anonymous message systems by resisting passive long-term end-to-end traffic analysis attacks [33]. Luo et al. defended against traffic analysis attacks by preventing locations of critical sensor nodes from being tracked by attackers [34]. And countermeasures of active traffic analysis attacks such as probing traffic injection are studied by a previous work [35]. Our work investigates potential mitigation of demographics inference through traffic analysis, and tries to attract more attention on finding more advanced protection techniques.

7 Conclusion and Future Work

In this work, we investigate the potential privacy leakage of large-scale demographics inference. To achieve this goal, we present a framework to extract four kinds of features from real-world Wi-Fi traffic and leverage supervised machine learning techniques to infer users' demographics. The study is based on a Wi-Fi traffic dataset from 28,158 users in 5 months, and experiments are performed under the simulations of various scenarios with different time durations, traffic sources, and whether data are encrypted or not. The experimental results show that the best accuracy of predicting gender and education level achieve 82% and 78% respectively. Even in encrypted traffic, i.e., HTTPS traffic, users' demographics can be predicted at precision of 69% and 76%. Finally, we consider some potential countermeasures and evaluate the performance by experiments. Our results suggest the successful rate of inference attacks could be very serious, and potential privacy leakage through Wi-Fi network traffic should become a more serious concern.

Our future effort will be focused on the two following aspects. Firstly, the inference attack may be further evaluated on more general datasets with more demographic attributes considered. Secondly, we plan to design more sophisticated countermeasures to target the resisting of inference attacks.

Acknowledgments

This work is supported by National Science Foundation of China (no. U1401253, 61672350 and U1405251).

References

[1] Pablo Valerio. "WiFi Offloading To Skyrocket," (2015. June). [Online] Available at: http://www.networkcomputing.com/wireless/wifi-offloading-skyrocket/1733513641

[2] N. Cheng, X. O. Wang, W. Cheng, P. Mohapatra, and A. Seneviratne, "Characterizing privacy leakage of public WiFi networks for users on travel," in *Proc. of IEEE INFOCOM*, 2013, pp. 2769-2777.

[3] C. Hoffman. "Why Using a Public Wi-Fi Network Can Be Dangerous, Even When Accessing Encrypted Websites," (2014. Jan.). [Online] Available at: https://www.howtogeek.com/178696/why-using-a-public-wi-fi-network-can-be-dangerous-even-when-accessing-encrypted-websites/

[4] T. Wang, F. Wang, R. Sailer and D. Schales, "Kaleido: Network Traffic Attribution using Multifaceted Footprinting," in *Proc. of SIAM International Conference on Data Mining*, 2014, pp. 695-703.

[5] A. K. Das, P. H. Pathak, C. N. Chuah and P. Mohapatra, "Contextual localization through network traffic analysis," in *Proc. of IEEE INFOCOM*, 2014, pp. 925-933.

[6] A. K. Das, P. H. Pathak, C. N. Chuah, P. Mohapatra, "Characterization of wireless multi-device users," in *ACM Transactions on Internet Technology - Special Issue on Internet of Things*, vol. 16, issue 4, no. 29, 2016.

[7] H. Li, Z. Xu, H. Zhu, D. Ma, S. Li, and K. Xing, "Demographics inference through Wi-Fi network traffic analysis," in *Proc. of IEEE INFOCOM*, 2016, pp. 1-9.

[8] N. Xia, H. H. Song, Y. Liao, M. Iliofotou, A. Nucci, Z. L. Zhang and A. Kuzmanovic "Mosaic: Quantifying privacy leakage in mobile networks," in *ACM SIGCOMM Computer Communication Review*, vol. 43, No. 4, pp. 279-290, 2013.

[9] Wandera, "Wandera reveals Q2 2015 mobile security and data usage figures," (2015. July). [Online] Available at: http://www.realwire.com/releases/Wandera-reveals-Q2-2015-mobile-security-and-data-usage-figures

[10] S. Dai, A. Tongaonkar, X. Wang, A. Nucci, and D. Song, "Networkprofiler: Towards automatic fingerprinting of android apps," in *Proc. of IEEE INFOCOM*, 2013, pp. 809-817.

[11] B. Konings, C. Bachmaier, F. Schaub, and M. Weber, "Device names in the wild: Investigating privacy risks of zero configuration networking," In *Proc. of IEEE Mobile Data Management*, 2013, vol. 2, pp. 51-56.

[12] M. U. Ilyas, M. Z. Shafiq, A. X. Liu, and H. Radha, "Who are you talking to? Breaching privacy in encrypted IM networks," in *Proc. of IEEE ICNP*, 2013, pp. 1-10.

[13] M. Korczyński, and A. Duda, "Markov chain fingerprinting to classify encrypted traffic," in *Proc. of IEEE INFOCOM*, 2014, pp. 781-789.

[14] A. K. Das, P. H. Pathak, C. N. Chuah, and P. Mohapatra. "Uncovering privacy leakage in ble network traffic of wearable fitness trackers," In *Proc. of ACM International Workshop on Mobile Computing Systems and Applications*, 2016, pp. 99-104.

[15] M. V. Barbera, A. Epasto, A. Mei, V. C. Perta, and J. Stefa, "Signals from the crowd: uncovering social relationships through smartphone probes". in *Proc. of ACM Conference on Internet Measurement Conference*, 2013, pp. 265-276.

[16] A. B. M. Musa, and J. Eriksson, "Tracking unmodified smartphones using wi-fi monitors". in *Proc. of ACM Conference on Embedded Network Sensor Systems*, 2012, pp. 281-294.

[17] G. Ateniese, B. Hitaj, L. V. Mancini, N. V. Verde, and A. Villani, "No place to hide that bytes won't reveal: Sniffing location-based encrypted traffic to track a user's position". in *Springer International Conference on Network and System Security*, 2015, pp. 46-59.

[18] N. V. Verde, G. Ateniese, E. Gabrielli, L. V. Mancini, and A. Spognardi, "No nat'd user left behind: Fingerprinting users behind nat from netflow records alone". in *Proc. of IEEE International Conference on Distributed Computing Systems*, 2015, pp. 218-227.

[19] T. Stöber, M. Frank, J. Schmitt, and I. Martinovic, "Who do you sync you are?: smartphone fingerprinting via application behaviour", in *Proc. of ACM conference on Security and privacy in wireless and mobile networks*, 2013, pp. 7-12.

[20] M. Conti, L. V. Mancini, R. Spolaor, and N. V. Verde, "Can't you hear me knocking: Identification of user actions on android apps via traffic analysis". in *Proc. of ACM Conference on Data and Application Security and Privacy*, 2015, pp. 297-304.

[21] M. Conti, L. V. Mancini, R. Spolaor, and N. V. Verde, "Analyzing android encrypted network traffic to identify user actions",*IEEE Transactions on Information Forensics and Security*, vol. 11, no. 1, pp. 114-125, 2016.

[22] W. Meng, R. Ding, S. P. Chung, S. Han, W. Lee, "The Price of Free: Privacy Leakage in Personalized Mobile In-App Ads," in *Proc. of NDSS*, 2016.

[23] J. Hu, H. J. Zeng, H. Li, C. Niu and Z. Chen, "Demographic prediction based on user's browsing behavior," in *Proc. of ACM Conference on World Wide Web*, 2007, pp. 151-160.

[24] S. Seneviratne, A. Seneviratne, P. Mohapatra, and A. Mahanti, "Your installed apps reveal your gender and more!". In *Mobile Computer Communication Review*, vol. 18, no. 3, pp. 55-61, 2015.

[25] H. A. Schwartz, J. C. Eichstaedt, M. L. Kern, L. Dziurzynski, S. M. Ramones, M. Agrawal, ... and L. H. Ungar, "Personality, gender, and age in the language of social media: The open-vocabulary approach," *PloS one*, vol. 8, no. 9, e73791, 2013.

[26] B. Bi, M. Shokouhi, M. Kosinski, T. Graepel, "Inferring the demographics of search users: Social data meets search queries," in *Proc. of ACM Conference on World Wide Web*, 2013, pp. 131-140.

[27] X. Chen, Y. Wang, E. Agichtein, F. Wang, "A comparative study of demographic attribute inference in twitter," *Proc. of ICWSM*, 2015, pp. 590-593.

[28] Y. Dong, Y. Yang, J. Tang, Y. Yang, N. V. Chawla, "Inferring user demographics and social strategies in mobile social networks," in *Proc. of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2014, pp. 15-24.

[29] A. Chaabane, G. Acs, M. A. Kaafar, "You Are What You Like! Information Leakage Through Users' Interests," in *Proc. of NDSS*, 2012.

[30] Y. Fan, Y. Jiang, H. Zhu, and X. Shen, "An efficient privacy-preserving scheme against traffic analysis attacks in network coding," in *Proc. of IEEE INFOCOM*, 2009, pp. 2213-2221.

[31] O. Berthold, and H. Langos, "Dummy traffic against long term intersection attacks," in *Proc. of Springer International Workshop on Privacy Enhancing Technologies*, 2002, pp. 110-128.

[32] D. I. Wolinsky, E. Syta, B. Ford, "Hang with your buddies to resist intersection attacks," in *Proc. of ACM SIGSAC conference on Computer & communications security*, 2013, pp. 1153-1166.

[33] N. Mathewson, R. Dingledine, "Practical traffic analysis: Extending and resisting statistical disclosure," in *Privacy Enhancing Technologies*, vol. 3423, pp. 17-34, 2004.

[34] X. Luo, X. Ji, M. S. Park, "Location privacy against traffic analysis attacks in wireless sensor networks," in *Proc. of IEEE International Conference on Information Science and Applications*, 2010, pp. 1-6.

[35] X. Fu, B. Graham, R. Bettati, W. Zhao, "Active traffic analysis attacks and countermeasures," in *Proc. IEEE International Conference on Computer Networks and Mobile Computing*, 2003, pp. 31-39.

[36] Wi-Fi Alliance, "Hotspot 2.0 (Release 2) Technical Specification Package v1.2," (2017). [Online]. Available at: https://www.wi-fi.org/downloads-registered-guest/Hotspot_2-0_%2528R2%2529_Technical_Specification_Package_v1-3_0.zip/29728

[37] A. Bittau, M. Handley and J. Lackey, "The Final Nail in WEP's Coffin," in *Proc. of IEEE Symposium on Security and Privacy*, 2006, pp. 15-pp.

[38] E. Tews and M. Beck, "Practical attacks against WEP and WPA," in *Proc. of ACM conference on Wireless network security*, 2009, pp. 79-86.

[39] A. Ferreira, J. L. Huynen, V. Koenig, and G. Lenzini, "Socio-technical security analysis of wireless hotspots". in *Proc. of International Conference on Human Aspects of Information Security, Privacy, and Trust*, 2014, pp. 306-317.

[40] Cisco. "Use Case: Location-Based Advertising over Wi-Fi," (2014. July). [Online]. Available at: http://www.cisco.com/c/dam/en/us/solutions/collateral/service-provider/mobile-internet-applications-services/at-a-glance-c45-731324.pdf

[41] Cisco. "Use Case: Mobile Targeted Advertising," (2014. July). [Online]. Available: http://www.cisco.com/c/dam/en/us/solutions/collateral/service-provider/mobile-internet-applications-services/at-a-glance-c45-731335.pdf

[42] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5-32, 2001.

[43] T. Chen, C. Guestrin, "Xgboost: A scalable tree boosting system," In *Proc. of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 785-794.

[44] R. Caruana, A. Niculescu-Mizil, "An empirical comparison of supervised learning algorithms," In *Proc. of ACM International Conference on Machine Learning*, 2006, pp. 161-168.

[45] M. Srivatsa and M. Hicks. "Deanonymizing mobility traces: Using social network as a side-channel," in *Proc. of ACM SIGSAC conference on Computer & communications security*, 2012, pp. 628-637.

[46] R. Kohavi, and D. Sommerfield, "Feature Subset Selection Using the Wrapper Method: Overfitting and Dynamic Search Space Topology," in *Proc. of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1995, pp. 192-197.

[47] H. Li, H. Zhu, S. Du, X. Liang, X. Shen, "Privacy Leakage of Location Sharing in Mobile Social Networks: Attacks and Defense," IEEE *IEEE Trans. on Dependable and Secure Computing*, vol. PP, no. 99, pp. 1-1, 2016.

[48] V. F. Taylor, R. Spolaor, M. Conti, and I. Martinovic, "Appscanner: Automatic fingerprinting of smartphone apps from encrypted network traffic," in *IEEE European Symposium on Security and Privacy* 2016, pp. 439-454.

[49] H. Yao, G. Ranjan, A. Tongaonkar, Y. Liao, and Z. M. Mao, "Samples: Self adaptive mining of persistent lexical snippets for classifying mobile application traffic," in *Proc. of ACM International Conference on Mobile Computing and Networking*, 2015, pp. 439-451.

[50] W. Y. Loh, "Classification and regression trees," in *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 1, no. 1, pp. 14-23, 2011.

[51] T. Chen. "Introduction to Boosted Trees", (2014 Oct.). [Online]. Available at: https://homes.cs.washington.edu/∼tqchen/pdf/BoostedTree.pdf

[52] S. Bernard, L. Heutte, and S. Adam, "Influence of hyperparameters on random forest accuracy," in *Proc. of Springer International Workshop on Multiple Classifier Systems*, 2009, pp. 171-180.

[53] F. Cucker, and S. Smale, "Best choices for regularization parameters in learning theory: on the bias-variance problem," *Foundations of Computational Mathematics*, vol. 2, no. 4, pp. 413-428, 2002.

[54] W. M. Shbair, A. R. Bashandy, and S. I. Shaheen. "A New Security Mechanism to Perform Traffic Anonymity with Dummy Traffic Synthesis," in *IEEE Computational Science and Engineering*, vol. 1, pp. 405-411, 2009.

[55] S. Oya, C. Troncoso, and F. Pérez-González, "Do Dummies Pay Off? Limits of Dummy Traffic Protection in Anonymous Communications," in *Proc. of International Symposium on Privacy Enhancing Technologies Symposium*, 2014, pp. 204-223.

[56] A. Das, N. Borisov, and M. Caesar, "Tracking mobile web users through motion sensors: Attacks and defenses," in *Proc. of NDSS*, 2016.

[57] Y. Yang, M. Shao, S. Zhu, B. Urgaonkar, and G. Cao, "Towards event source unobservability with minimum network traffic in sensor networks," in *Proc. of ACM conference on Wireless network security*, 2008, pp. 77-88.

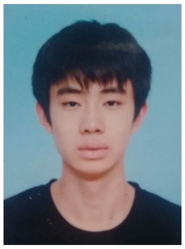
[58] J. Appelbaum, M. Ray, K. Koscher, and I. Finder. "vpwns: Virtual pwned networks," in *USENIX Workshop on Free and Open Communications on the Internet*, 2012.

[59] M. Vanhoef, C. Matte, M. Cunche, L. S. Cardoso, F. Piessens, "Why MAC Address Randomization is not Enough: An Analysis of Wi-Fi Network Discovery Mechanisms," in *Proc. of ACM on Asia Conference on Computer and Communications Security*, 2016, pp. 413-424.

[60] I. J. Goodfellow, J. Shlens, C. Szegedy, "Explaining and harnessing adversarial examples," arXiv preprint arXiv:1412.6572. 2014.

[61] B. Biggio, B. Nelson, and P. Laskov. "Support vector machines under adversarial label noise," in *Proc. of Asian Conference on Machine Learning*, 2011, pp. 97-112.

[62] N. Papernot, P. McDaniel, I. Goodfellow, "Transferability in machine learning: from phenomena to black-box attacks using adversarial samples," arXiv preprint arXiv:1605.07277, 2016.



Huaxin Li is a graduate student working towards his M.Sc. degree in Department of Computer Science and Engineering, Shanghai Jiao Tong University. He received the B.Sc. degree in Department of Computer Science and Engineering, Shanghai Jiao Tong University, China, in 2011. His research interests include social networks privacy, smartphone security, network security and privacy, and machine learning.

Haojin Zhu (IEEE M'09-SM'16) received his B.Sc. degree (2002) from Wuhan University (China), his M.Sc.(2005) degree from Shanghai Jiao Tong University (China), both in computer science and the Ph.D. in Electrical and Computer Engineering from the University of Waterloo (Canada), in 2009. Since 2017, he has been a full professor with Computer Science department in Shanghai Jiao Tong University. His current research interests include network security and privacy enhancing technologies. He published 35 international journal papers, including JSAC, TDSC, TPDS, TMC, TWC, TVT, and 60 international conference papers, including ACM CCS, ACM MOBICOM, ACM MOBIHOC, IEEE INFOCOM, IEEE ICDCS. He received a number of awards including: IEEE ComSoc Asia-Pacific Outstanding Young Researcher Award (2014), Top 100 Most Cited Chinese Papers Published in International Journals (2014), Supervisor of Shanghai Excellent Master Thesis Award (2014), Distinguished Member of the IEEE INFOCOM Technical Program Committee (2015), Outstanding Youth Post Expert Award for Shanghai Jiao Tong University (2014), SMC Young Research Award of Shanghai Jiao Tong University (2011). He was a co-recipient of best paper awards of IEEE ICC (2007) and Chinacom (2008) as well as IEEE GLOBECOM Best Paper Nomination (2014). He received Young Scholar Award of Changjiang Scholar Program by Ministry of Education of P.R. China in 2016.

Di Ma is an Associate Professor in the Computer and Information Science Department at the University of Michigan-Dearborn, where she leads the Security and Forensics Research Lab (SAFE). She is broadly interested in the general area of security, privacy, and applied cryptography. Her research spans a wide range of topics, including smartphone and mobile device security, RFID and sensor security, vehicular network and vehicle security, computation over authenticated/encrypted data, fine-grained access control, secure storage systems, and so on. Her research is supported by NSF, NHTSA, AFOSR, Intel, Ford, and Research in Motion. She received the PhD degree from the University of California, Irvine, in 2009. She was with IBM Almaden Research Center in 2008 and the Institute for Infocomm Research, Singapore in 2000-2005. She won the Tan Kah Kee Young Inventor Award in 2004.