Title

Privacy-aware contextual localization using network traffic analysis

Permalink

https://escholarship.org/uc/item/5217k37s

Authors

Das, Aveek K

Pathak, Parth H

Chuah, Chen-Nee

et al.

Publication Date

2017-05-01

DOI

10.1016/j.comnet.2017.02.011

Computer Networks 118 (2017) 24–36

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

Privacy-aware contextual localization using network traffic analysis[☆]

Aveek K. Das[a,*], Parth H. Pathak[b], Chen-Nee Chuah[c], Prasant Mohapatra[a]

[a] *Computer Science Department, University of California, Davis, CA, USA*
[b] *Computer Science Department, George Mason University, Fairfax, VA, USA*
[c] *Electrical and Computer Engineering Department, University of California, Davis, CA, USA*

ARTICLE INFO

Article history:
Received 20 August 2015
Revised 20 December 2016
Accepted 16 February 2017
Available online 24 February 2017

Keywords:
Localization
Contextual location
Wireless traffic monitoring
Internet measurement

ABSTRACT

The rise of location-based services has enabled many opportunities for content service providers to optimize the content delivery to user's wireless devices based on her location. Since the sharing precise location remains a major privacy concern among the users, certain location-based services rely on *contextual location* (e.g. residence, work, etc.) as opposed to acquiring user's exact physical location. In this paper, we present PACL (Privacy-Aware Contextual Localizer) model, which can learn user's contextual location just by passively monitoring user's network traffic. PACL can discern a set of vital attributes (statistical and application-based) from user's network traffic, and predict user's contextual location with a very high accuracy. We design and evaluate PACL using real-world network traces of over 1700 users with over 100GB of total data. Our results show that PACL, when built using the Bayesian Network machine learning algorithm, can predict user's contextual location with the accuracy of around 89%.

1. Introduction

A tremendous growth has been observed in location-based services, in the last few years. On a broad scale, current location-based services can be classified into two categories. Users navigate to specific locations, search for restaurants and businesses near a certain location, check-in on social networks, etc. using these location-based services. The first category requires *precise* user location to provide its services. One example for such a service is smartphone navigation system where exact latitude and longitude information is essential. The other type of services only need contextual information about the users' location. For example, knowing that a user is at an airport or a shopping mall is sufficient (and necessary) to provide certain services specific to that location category. *Contextual location* information is also important for content providers and Content Distribution Networks (CDNs) which can use this knowledge to optimize the content delivery and provide useful recommendations based on user's location type. Third party services, also, can provide targeted advertisements related to the contextual location of the user. Most users believe that contextual location based services are based on precise user location, which they are not comfortable to share, in most occasions, to receive contextual location based services. If these services can be provided to users without compromising their privacy (about precise location), we believe users would be benefited by such services. In this paper, we present a privacy-preserving system that can determine user's location category (or contextual location) just by passively monitoring and learning from aggregate network traffic from different categories of location.

Existing services such as FourSquare [1] can be used by content providers to map a user's precise location to her contextual location category but this requires the user to share their precise physical location. Increasing concerns about location privacy, have prompted more and more users to be unwilling about provide their location information, especially for contextual location-based services. This insecurity among users have led to the *Do Not Track Me Online Act of 2011* [2] which provides users with an option to disable tracking of its location by content providers or websites. As an example of privacy preferences, we can say that users are willing to share their GPS location for Google Maps Navigation but when services such as YouTube ask for user's location, users often deny the request even though content delivery could have been optimized by YouTube if the location was available.

In this paper, we propose a network traffic analysis technique whereby an ISP or any third-party entity capable of passively monitoring network traffic can determine user's contextual location (without knowing user's exact physical location). The ISP can

[☆] An earlier version of this work was submitted and accepted for publication at IEEE INFOCOM 2014. The work was titled "Contextual Localization through Network Traffic Analysis".

[*] Corresponding author.
E-mail addresses: akdas@ucdavis.edu (A.K. Das), phpathak@gmu.edu (P.H. Pathak), chuah@ucdavis.edu (C.-N. Chuah), pmohapatra@ucdavis.edu (P. Mohapatra).

http://dx.doi.org/10.1016/j.comnet.2017.02.011
1389-1286/© 2017 Elsevier B.V. All rights reserved.

use the traffic analysis technique to determine the users' location category. Once the contextual location has been identified, CDNs can probe to obtain this information from the ISPs using the proposed ISP-CDN collaboration model [3,4]. This information can then be utilized by the CDNs to provide contextual location based services to users, like targeted advertisements. For example, at work, a person would prefer to get an advertisement of a word-processing software on sale rather than get an advertisement for a movie ticket. Thus, one of the major applications of the proposed technique is to provide location context based advertisements to users.

Our method can also work without an ISP accessing the contents of the packets (such as website being accessed or payload). Protocol identification and relevant statistical features are sufficient for location categorization. As we see later in the paper, statistical features of flow, packets and protocols in the user created network data can be used to achieve an accuracy of location prediction which is as high as 83%. This is accomplished without looking at the content of the packets. This kind of inspection is often carried out by the ISP for traffic engineering and security purposes. Hence, we believe that ISPs can assist in location categorization using our technique while adhering to the privacy acts. After determining the location category, the ISPs can also fine-tune their security policies, as public locations (like cafeteria/restaurants) needs different policies as compared to private locations (like apartments). For example, certain ports and flows in a public location context can be blocked to provide more security to users from attackers.

In this work, first, we show that network traffic originating from different types of locations (such as cafe, university campus, residence etc.) have built-in distinct signatures based on the location category. Second, we propose a traffic analysis engine that can leverage information collected by existing passive traffic monitoring systems to discern the contextual location signature. The signature is composed of different attributes that may differ depending on the type of location (e.g., applications users access at different locations, flow length, packet size distributions etc.) These location signatures can be used to identify the contextual location of any IP address.

The contributions of our work are as follows:

1. First, we show that traffic originating from different *types* of locations have distinct signatures embedded in them. To establish this, we have collected nearly a 100GB of real-world network traffic traces for over 1700 users at different types of locations. We identify a number of attributes which when used together can create a distinct contextual location signature.
2. Next, we present a system (named PACL - Privacy-Aware Contextual Localizer) that can learn user's contextual location only by passively monitoring user's traffic flows. The core of PACL is a supervised machine learning engine that can predict user's contextual location efficiently and accurately. We evaluate our PACL model using our network traces, based on six machine learning algorithms. The best prediction accuracy is observed using the Bayesian Network classification algorithm which show that PACL can predict contextual location with an overall accuracy of 89%. This model not only gives overall good accuracy, the accuracy for the individual classes are also very similar and equally efficient.

This paper is structured as follows. We start out with discussion of related research works in Section 2. In Section 3, we introduce the PACL system and describe its functioning in details. Section 4 includes details about the dataset used for analysis. The features which differentiate each contextual location are discussed in Section 5. In Section 6, we present the methods used for feature selection. The prediction model and the prediction results observed using our proposed model are in Section 7, followed by conclusions in Section 8.

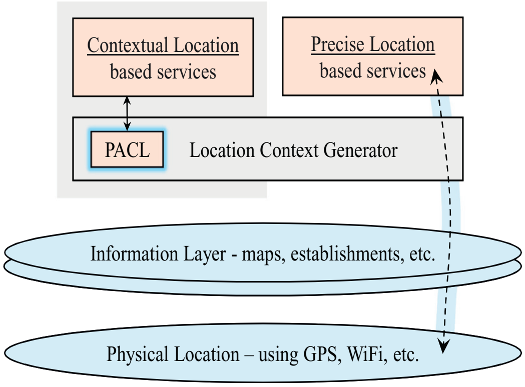

Contextual Location based services
Precise Location based services
PACL
Location Context Generator
Information Layer - maps, establishments, etc.
Physical Location – using GPS, WiFi, etc.

Fig. 1. PACL as compared to regular localization using precise location.

2. Background and related work

Traditional location-based services are built on top of positioning systems (e.g. GPS) and information layer (e.g. maps, database of establishments etc.). This is depicted in Fig. 1. Here, location-based services that require exact physical location typically use data from user's positioning system combined with details of information layer. This opens up many entry points for privacy invasion of users. On the other hand, certain services (such as targeted advertising, content delivery optimization etc.) do not require user's exact physical location. Also, users are less likely to provide their location for such services. Our solution, PACL, can address this challenge by eliminating the need of user's physical location in the case of contextual location-based services (see Fig. 1). Instead of querying users for precise location, PACL passively learns user's contextual location by monitoring users' network traffic.

Determining Location and Preserving Privacy: Significant amount of past research has mostly focused on two topics: (i) accurate and energy-efficient determination of user's physical location and, (ii) preserving user's privacy when sharing user's location information. In the first category of research, a variety of location determination mechanisms have been proposed like in [5,6]. The central focus of these studies is to reduce the energy consumption of determining the location while increasing the accuracy. Also, other techniques such as map matching [7] are used to improve the accuracy. Location privacy preserving techniques have attracted a lot research starting from initial studies such as [8]. Methods such as cloaking [9] and obfuscation [10] are proposed as ways to prevent privacy leakage of users using location-based services. PACL is different from these studies as it does not require actual physical location and other privacy preserving methods for protecting the physical location.

Traffic Classification: Another thread of research that is relevant to PACL is known as Internet traffic classification. The purpose of traffic classification is to monitor and analyze network traffic for determining applications and protocols being used. It is a well-established method ([11] and references therein) of profiling network traffic, anomaly detection and detecting file sharing of copyrighted content. Such traffic classification techniques and PACL share a few common characteristics. They both utilized traffic monitoring and are built using machine learning algorithms. Nevertheless, we believe that PACL takes a step forward by learning and predicting contextual location purely through network traffic analysis.

Another research work relevant to ours is [12] in which Trestian et al. provide a detailed study on applications accessed by users at

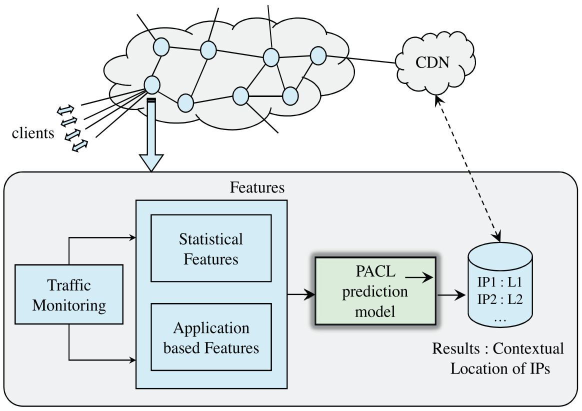


Fig. 2. Architecture of the PACL system: network traffic is monitored for a number of features, which when used in the PACL model gives contextual location prediction of an IP.

different locations and show that they tend to be different at work and home, irrespective of the time of the day. Our model not only profiles the usage of applications and services by users at different locations but also combines them with other statistical features to predict their contextual location.

There are many online third-party software tools which claim to predict the geographical location of an IP address [13]. However these services only provide city-level information of the IP address but neither the exact location or the contextual location is available. Some of these tools provide geographical coordinates, but those mostly refer to the coordinates of the ISP the IP address is registered to.

3. Privacy Aware Contextual Localizer (PACL) system

In this work, we design Privacy Aware Contextual Localizer (PACL) system, which can determine the *category* of user's location. PACL is built on a simple fundamental idea that user's network activity is highly dependent on user's contextual location. If one is able to identify the attributes of network traffic that are sufficiently different across different contextual location, ISP or any third party entity capable of passively monitoring traffic, can use the same set of attributes to determine user's location context. This location context can then be shared with content service providers who can optimize the content deliver accordingly. The foremost advantage of the PACL system is that users are not required to share their precise location with anyone, and at the same time, they can be served using the content that is optimized based on their location context. The components of the PACL system are shown in Fig. 2.

Traffic Monitoring: PACL can be deployed within traffic monitoring systems of an ISP or an AS (Autonomous System). Flows originating from user IPs can be monitored for a fixed amount of time after which PACL determines its contextual location. Note that PACL is similar to traditional Internet traffic classification methods as it performs better when complete bi-directional network traffic of end-user IPs can be monitored. Since this is the first attempt towards determining type of location purely using network traffic, we restrict our study to the case where PACL is deployed on traffic monitors with complete bi-directional network flows.

In our measured dataset, we collect network traffic over the edge at WiFi hotspots deployed at different types of locations (details in Section 4). We build and verify PACL using the traces of over a 100GB collected at different location over the period of 20 days.

Identifying Location Signature: In the PACL, we first identify specific attributes of IPs which are likely to be correlated to IP's location. In the training phase, we use the available ground-truth of location to find the correlation between the attributes with the location. The attributes (or features) we use can be classified in two categories - statistical features and application-based features. Examples of statistical features include number of flows originated by an IP, packet length distribution of all packets of an IP etc. On the other hand, in the application-based features, we classify user's network flows in different categories of applications (such as emails, games, social-networks etc.). To understand what kind of content users are interested in (independent of which application they use to access it) when at a specific location, we also classify flows into different interest categories. We show that both statistical and application-based features can generate a distinct signature for different locations.

Applying Location Signatures to Determine Location Context: Once the location signature has been identified, PACL prediction model predicts the contextual location of a user based on location signature mentioned above and the observed statistical and application-based features associated with the particular user (or IP address). As shown in Fig. 2, the results are stored in a repository, which can be accessed by the content providers to optimize content delivery and provide location-specific services. However, even after prediction of contextual location of an IP address, PACL continues to predict contextual location as dynamic reallocation of IPs might change IP's location category.

The prediction model is built using a machine learning prediction algorithm. Out of the six algorithms, the Bayesian Network algorithm gives the best prediction accuracy. It is observed that the combination of both the statistical features and application based features give better prediction of location context than using each set individually. Application of this model on our dataset of over 1700 users yields a prediction accuracy of over 89%.

In our dataset, we collect data from WiFi hotspots and hence are aware of the location category. For the PACL model, knowledge of the location category for some user devices is necessary - this provides the ground-truth for the initial model building phase. For this purpose, the PACL during traffic monitoring can anonymously probe the users in a network for their location category information. As we know, some users, who are willing to share location intermittently, will reply to such queries. As a result, we will be able to collect the location category information for the initial model building phase.

Before describing PACL in details, we discuss the application scope and limitations of PACL. First and foremost, PACL cannot be used for location-based services where user's precise location is essential. In other words, it cannot be used for applications where precise location is more important than preservation of privacy. Second, PACL is capable of predicting most common "location types" but its current form cannot characterize traffic from short-term gatherings (such as a sports event). Thirdly, the PACL model does not need to be deployed in the network where the traffic is from one location context only. It has the capability to sort out different IP addresses and determine their location context. That way multiple deployments at different locations are not required - deployment at data aggregation points serves the purpose.

4. Network traffic collection and datasets

One major challenge we faced in developing the PACL system is to acquire network traffic traces which precisely originate at specific locations. If network traces from ISP or AS are used, they might not always have the ground-truth location for different IPs. To address this challenge, we capture the network traffic at the edge at different WiFi hotspots deployed at different locations. The details of the datasets are presented in Table 1.

Table 1
Dataset used For location signature analysis.

Location Type	Traces	No. of IPs	Total IPs.	Total flows	Packet Count (Million)	Duration (Hours:Minutes)	Trace Size
Residential	Apartment-1	91	315	16,695	16.47	7:40	7.2GB
	Apartment-2	78		20,505	31.15	10:40	14.9GB
	Apartment-3	72		14,396	17.45	3:22	7.9GB
	Apartment-4	52		6465	14.82	2:44	6.8GB
	Apartment-5	22		12,469	8.38	3:16	3.1GB
University Campus	Department hall	114	529	14,887	27.34	5:12	5.9GB
	Library-1	313		20,153	83.62	7:55	21.9GB
	Library-2	102		26,861	65.29	8:19	19.2GB
Cafeteria/Restaurant	Starbucks-1	234	450	39,532	12.89	8:03	5.6GB
	Starbucks-2	216		44,720	12.73	8:48	4.9GB
Airport/Travel	Washington-1	88	458	10,682	2.01	0:18	682MB
	Sydney-1	80		8586	4.05	1:24	1.4GB
	Orlando	63		2280	1.35	0:20	499MB
	Washington-2	55		3201	1.00	0:13	209MB
	Denver	53		7264	2.02	0:21	515MB
	Washington-3	40		1338	1.37	0:20	340MB
	Los Angeles	39		2691	1.01	0:15	411MB
	Sydney-2	23		872	0.84	0:25	190MB
	San Francisco	17		2024	1.17	0:15	624MB

4.1. WiFi packet captures

The data is collected by passively sniffing WiFi packets from the air near the WiFi hotspot. We chose four different categories of locations - residential, university campus, cafeteria/restaurants and airport/travel (see Table 1). The four location categories that we consider are representative of locations where users have some sort of distinct internet usage pattern. For example, the access of videos and games at residential locations create sessions with large amount of data transfer and longer durations, whereas the access of travel websites at airports will create smaller sessions with very low byte count. There can be other location categories, but we consider these four for our experiments and our prediction model.

For each category, we collected traces at multiple different locations of that category to extract/learn the category-specific characteristics. The traces were collected using TP-Link WN722N WiFi USB adapters [14] connected to a laptop running Linux. The WiFi adapters run in monitor mode of ath9k driver [15] and Wireshark is used to capture the packets. We connect three different adapters to each laptop in order to simultaneously capture on 3 different channels (channels 1, 6 and 11 of 2.4 GHz IEEE 802.11 b/g/n). The traces account for a total of over 100GB of data captured over 20 different days. The airport traces were captured in 2012 as described in [16].

The dataset and the subsequent analysis is based on classification of contextual location into four classes. However, the PACL model can be extended to incorporate other location categories, provided the model is trained beforehand based on the features from those locations. The analysis done here is based on wireless network traces, but the analysis is applicable for wired network traffic. We use WiFi traces as they can be collected easily in public settings, and in any case, most of the devices that are used at these locations are wireless devices.

4.2. Data sanitization

Before processing the data as input to the PACL learning model, we sanitize the network traces. The process of the sanitization phase is divided into two steps. First, the collected dataset is anonymized to remove any personal identity related information. The second step involves removing all the packets from the network traces which will not be forwarded to the ISP. In this step, all the MAC layer frames (such as WiFi beacons etc.) as well as MAC layer headers are removed from all IP packets as these information is not forwarded beyond WLAN.

5. Finding location signature

We propose a traffic analysis system, which can passively monitor network traffic and extract the **statistical features** and **application and service based features**, on a per-IP basis, to be used for learning and prediction.

5.1. Statistical features

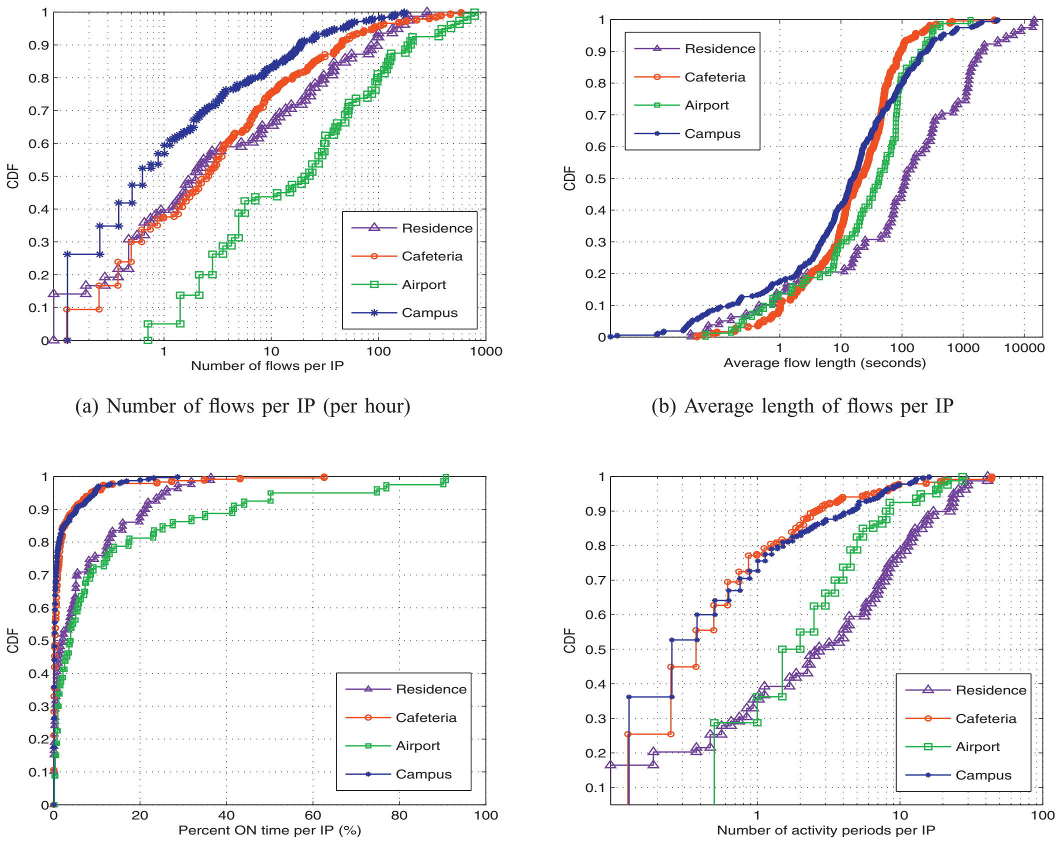
For each IP address in the trace, we calculated the statistical features listed below. They are divided into 4 subsets as shown below. Type I and II attributes hold single numerical values, while the attributes of Type III and IV are distributions, which are represented using $<$ min, max, average, median, standard deviation, skewness, kurtosis$>$. While extracting the features from the traffic, we have no prior idea about the shape of the distributions (Gaussian or not). We are primarily concerned with accurate representation and description of distributions obtained from the data. Thus, similar to the network features used in [11] we consider the first four moments (mean, variance, skewness and kurtosis) in addition to maxima, median and minima. Note that, a flow is identified using a 5-tuple $<$ source IP, source port, destination IP, destination port, protocol$>$.

Type I - Coarse-grain statistics:

1) Total number of flows
2) Average number of concurrent sessions
3) Percentage ON time - ratio of number of 10 second blocks when IP was active (had at least one flow) to the total time of the trace
4) Number of activity periods (one activity period = a period of time when the IP was continually active, i.e. had at least one flow active)
5) Number of bytes transferred
6) Number of packets transferred
7) Average application data rate

Type II - Protocol level statistics:

8) Number of HTTP flows
9) Number of HTTPS flows
10) Number of TCP (non-HTTP/HTTPS) flows



(a) Number of flows per IP (per hour)

(b) Average length of flows per IP

(c) Percentage ON time per IP

(d) Activity period per IP (per hour)

Fig. 3. Figures represent variation of four key attributes across four different location classes.

11) Number of UDP flows

Type III - Flow level statistics:

12) Flow length
13) Application data rate of the flows
14) Bytes transferred per flow
15) Packets transferred per flow

Type IV - Packet level statistics:

16) Packet inter-arrival time
17) Packet size

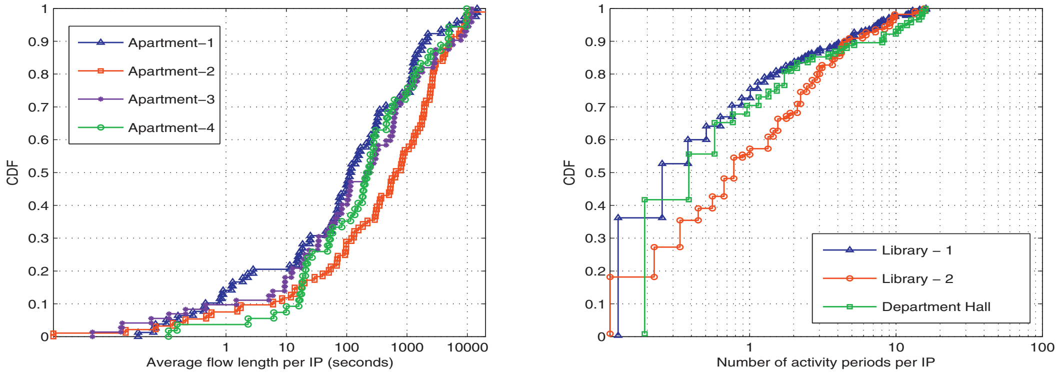
The total number of statistical features are 53 (1 feature each for Type I and II and 7 features for each distribution for the statistics of Type III and IV).

During the entire time of the trace, the DHCP lease to a particular device does not expire and thus for all calculations, we assume one IP address is assigned to one device (we also verify this by checking the MAC addresses corresponding to each IP address). For the calculation of activity period, percentage ON time and concurrent flows per IP address, the entire trace duration was divided into bins of 10 s intervals each and the analysis was done based on the whether an IP address created any flow during each of these time bins. The statistical attributes which are directly dependent on the total time of the trace (e.g., total flows per IP, total number of HTTP flows, etc.) were normalized on a per hour basis, to eliminate any biases due to difference in the duration of different traces.

Analysis of Statistical Features: The statistical attributes reveal distinct information that can serve as location signature and in turn, used to predict contextual location. Some of these characteristics are shown in Fig. 3. As we can see, airport trace has the highest number of flows per IP per hour as compared to the other locations, whereas Campus has the lowest, as seen in Fig. 3a. Airport and cafeteria traces have mostly smartphone based network traffic and thus each device generates a large number of flows (due to background applications and ads). On the other hand, campus traces have a large number of IP addresses with very low flow count - as there are users who pass by the WiFi hotspot and their devices, which are connected to the campus network, by default, may generate traffic for that transient period of time.

Fig. 3b and d shows the length of flows and the number of activity periods per IP are the largest in case of residence as compared to others. This is expected, as in residential buildings users tend to keep their devices on for longer duration, even though the usage can be in on-off manner and not continuously. From Fig. 3b we can observe that more than 50% of the IP addresses in the residential traces have flow lengths greater than top 10% IP flow-lengths in cafeteria trace. This is because most users tend to stay for a very short time in cafeterias. This proportion of users is smaller in campus as many users prefer to sit at once place. However there are several IP addresses with very small flow-lengths in campus trace, generated due to users who happen to pass by, as mentioned above.

Activity Period: One of the most distinct attributes among different location categories is activity period, as we will later see in Section 7. We calculate activity period count as the number of times an IP was continuously generating at least one flow in each of the 10 s time intervals, the whole trace was divided into. Fig. 3d indicates the higher number of activity periods in apartments, but questions may arise as to why such a trend is observed in airports too. This is because the activity period is normalized on a per-hour basis and the activity periods actually calculated are for approxi-



(a) Average flow length in residential traces

(b) Activity period in campus traces(per hour)

Fig. 4. Figures represent the variation of a particular attribute across the different traces of the same location class.

mately 15–30 min traces. Hence we see higher number of activity periods in airport trace. Around 90% of IP addresses at campus and cafeteria have activity period count less than five. This is mainly as a result of passer-by user devices in campus traces and users in cafeteria traces who connect to the network for a few specific purposes.

Percentage ON Time: The percentage ON time of each IP address represents the total time an IP was active, as a percentage of the entire time of the trace. As seen in Fig. 3c, apartment and airport traces have the highest ON time percentage of all the four locations as most user devices are usually on for almost the entire time of the trace (note that airport traces are very short in duration). ON time percentages in cafeteria is smaller than those in campus, but there are some devices with very high percentage ON time in the cafeteria dataset. This is most likely to be due to the employees of the establishment who were present at that location during the entire data collection time.

Variation across datasets for the same location category: Fig. 4a and b shows the variation of two specific attributes across more than one trace of a particular location. These two figures help us to show that the variation of a particular attribute across multiple traces at the same category of location behaves similarly, inspite of the fact that the trace was collected in a different date and at a different location (but same contextual location). Similar trend across different traces at same location category is seen for almost all of the above mentioned features, which help us to assign a specific signature for each type of location.

To detect the interest of users in various kinds of applications at different locations, we use a keyword based search on the content of the captured packets, a method similar to the one used in [12]. Packets include the HTTP objects like GET, POST and URLs as well as DNS queries and answers. For the keyword based search, we created a keyword list, currently around 50 keywords for each category - generated using the common words of the *Keyword Tool* from Google Adwords [17] collected over one week, for each of the categories. Based on this search, we used the percentage of packets for a particular IP that had a keyword-match in any category as the score of the IP for that category. Apart from the 21 categories, we also did the above analysis on 12 commonly used services and used the scores as attributes. The 33 attributes in this category, combined with 53 statistical features, result in 86 attributes, in total.

5.2. Application based categorization

The keyword search on the trace showed that in general, around 60–70% of the IP addresses could be profiled on the basis of interest category. A particular IP address is considered to be interested in a specific application category if there is at least one packet that gives a keyword-match for that category. However, we observed that when a particular IP address was profiled to be belonging to a certain application category there were substantially large count of packets for which there was a keyword match in the same category. Table 2 shows the list of categories and services used for as the features in this category and Table 3 shows a few keywords of some of the categories. Fig. 5 represents the percentages of IP addresses that were profiled to be interested in one specific category.

Table 2
Application categories and services.

Categories	Entertainment, Games, News-Reading, Finance, Social network, Sports, Education-Career, Email, Family, File-sharing, Technology, Food-Culture, Health, Fashion, Politics, Shopping, Automobiles, Weather, Portals, Travel, Science
Services	YouTube, Netflix, Pandora, Amazon, Craigslist, Twitter, Facebook, Instagram, ESPN, Gmail, CNN, Dropbox

Table 3
Categories and keywords.

Interest Category	Keywords
Entertainment	youtube, netflix, itunes, mp3, video
Games	zynga, xbox, games, trivia, aws
News and Reading	nytimes, bbc, cnn, blogspot, news
Sports	espn, mlb, soccer, fifa, ncaa, nba
Social Networks	facebook, twitter, friends, plus.google
Travel	maps, expedia, tripadvisor, yelp
Technology	endgadget, cnet, bestbuy, techcrunch
Education and Career	.edu, stackoverflow, github, courseera
Shopping	craigslist, amazon, ebay, groupon
Email	gmail, pop3, imap, smtp, hotmail

Interpretation of Application based Categorization: The residential traces have the highest interest percentage in entertainment. Apart from that, food, family, shopping, politics, fashion and automobiles have higher percentage with lower interest in mails and portals as compared to the other locations. Mail and portals are not accessed by users at their own homes as compared to outside, like at work or when on the go. Also access to file-sharing websites are mostly seen in apartment traces. Traces collected in a campus WiFi hotspot have a very high percentage of IPs interested in education related websites, portals and emails, as can be expected. Music, video and games are accessed much less in a campus environment as compared to the others. Results in Fig. 5 verify this claim.

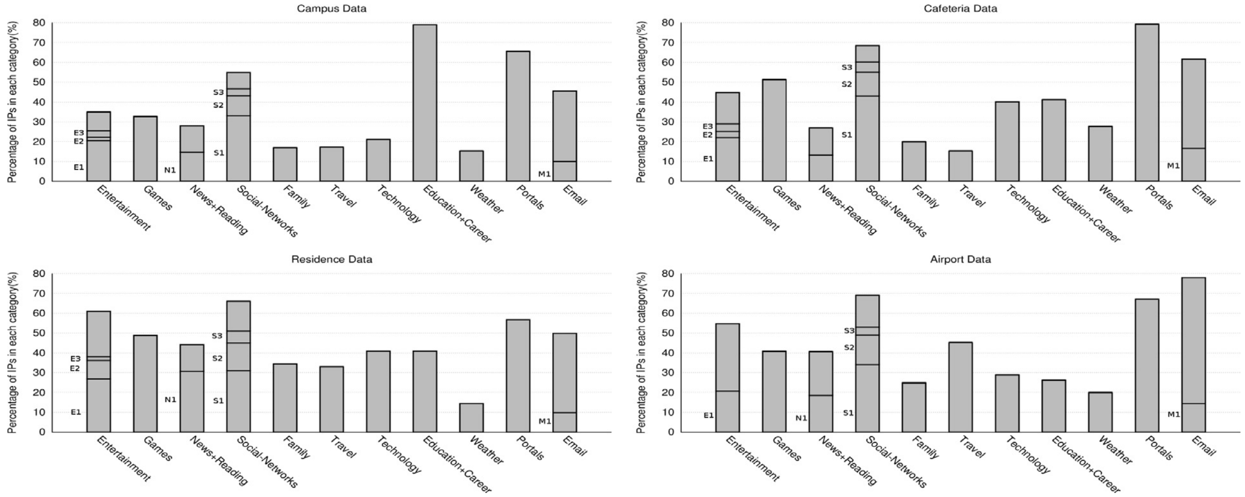


Fig. 5. Representation of interest categorization (E1: Youtube, E2: Netflix, E3: Pandora, N1: CNN, S1: Facebook, S2: Twitter, S3: Instagram, M1: Gmail).

Cafeteria and airport traces have very high number of IPs with interest in social-networks, portals and email. Outdoor locations are expected to have high percentage of users checking weather, as is observed in cafeteria and airport traces. There is a high number of IP addresses accessing travel related websites in the airport, as compared to other traces, which is an expected trend. Users interested in entertainment are much higher in apartment and cafeteria. Gaming websites or applications are found to be very high in the cafeteria trace (due to smart-phone games) and in apartments (due to dedicated gaming services, such as, xbox).

6. Feature selection

Before creating the model for prediction, we need to identify the specific features that contribute towards differentiating between location categories. For this purpose, Chi-squared statistic evaluation [18] and CFS Subset evaluation [19] is applied to the 86 attributes and some of the features, which do not contribute to the classification, are removed.

Chi-Squared Statistic: This statistic is used to evaluate the “distance” between the distribution of each class for an attribute. Initially, the values of an attribute are divided into separate intervals. Based on this division, the frequency of instances in each interval and class is calculated. Then the *Chi2* value is calculated based on Eq. (1) (with n = 2) for each pair of sorted adjacent intervals to ascertain if the relative frequencies of the classes are similar enough to justify their merging. If the *Chi2* distance is smaller than a certain threshold for the pair, the intervals are merged. Merging continues till all adjacent pairs have a *Chi2* value greater than the threshold (20 in our case).

$$\chi^2 = \sum_{i=1}^{n}\sum_{j=1}^{k}\frac{(A_{ij}-E_{ij})^2}{E_{ij}} \tag{1}$$

- ○ A_{ij} = frequency of ith interval and jth class.
- ○ E_{ij} = expected frequency of $A_{ij} = \frac{R_i * C_j}{N}$
- ○ R_i = number of values in ith interval $= \sum_{i=1}^{n} A_{ij}$
- ○ C_j = number of values in jth class $= \sum_{j=1}^{k} A_{ij}$
- ○ k = number of classes
- ○ n = number of intervals
- ○ N = total number of values

At the end of this step, if an attribute has been merged into one interval then the attribute is considered irrelevant in representing the original data and hence has a *Chi2* value of 0. Otherwise, the score is calculated as per Eq. (1). Fig. 6 represents the normalized Chi-squared statistic score of the statistical attributes based on (a) coarse-grain features (b) protocol-based features (c) packet-based features and (d) flow-based features.

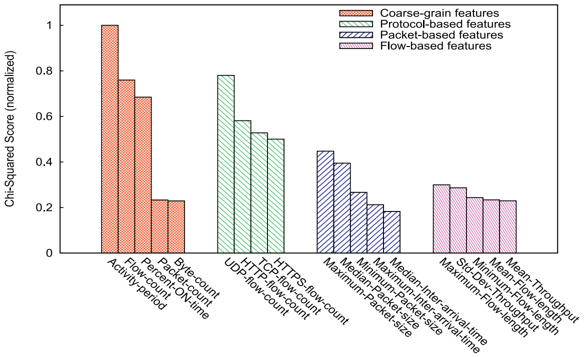


Fig. 6. Chi-square statistic score for the highest-correlated features for each subset of statistical attributes.

CFS Subset Selection: Correlation Feature Subset (CFS) employs a simple correlation based heuristic to rank different subsets formed out of the entire feature set. The objective of the heuristic is to find subsets that contains features that are highly correlated to the class and loosely correlated with each other. The CFS subset evaluation function which determines the “merit” of a feature subset is:

$$M_s = \frac{k.\overline{r_{cf}}}{\sqrt{k + k(k-1).\overline{r_{ff}}}} \tag{2}$$

where, M_s is the heuristic CFS Subset merit of a feature subset S containing k features, $\overline{r_{cf}}$ is the mean correlation value between the features and the class where ($f \in S$) and, $\overline{r_{ff}}$ is the average correlation value between two features in the subset. The numerator of Eq. (2) can be interpreted as providing an indication of how good the feature subset is, with high value of feature-class correlation. The denominator represents how redundant the features are among themselves, indicated by the value of the feature-feature correlation.

Application of the CFS Subset feature selection algorithm on our dataset of 86 features returns 10 features, which includes activity period, percentage ON time for an IP, flow count, UDP flow count and packet size per ip (mean), among others. The M_s value for the final selected feature subset is 0.482. This value tells us that the features have some level of redundancy and are not entirely non-correlated.

On the basis of the feature selection results, we choose to use CFS subset feature selection method. We remove 76 attributes from our data-set and build our model for prediction based on the remaining 10 features. In addition we also provide an analysis of how different subsets of features, based on how the features are calculated and their computational complexity, can predict the different location classes, in Section 7. All prediction results shown in Section 7 are based on a model built using the CFS subset attributes (unless otherwise stated).

7. PACL prediction model and results

In this section we describe the PACL model, created on the basis of the aforementioned features to efficiently predict users contextual location.

7.1. Model : machine learning prediction algorithm

Predicting the location category from the statistical and application based features is non-trivial as many of the statistical features are dependent on each other and their inter-relationship is non-linear. Different machine learning algorithms that are commonly used for traffic classification purposes have different computational complexity and perform differently based on the dataset properties. Kim, et. al, have used a number of machine learning algorithms for traffic classification [11]. Similarly, we use a number of machine learning classifiers to create the model involving these individual features. In this section, we give a short description of the algorithms.

7.1.1. Decision tree based algorithms

Reduced Error Pruning Tree: The algorithm implements a decision tree with Reduced Error Pruning. Due to the non-linear nature of the attributes the most prevalent algorithm used is decision trees. Decision tree models employ simple if-then-else statements which predict classes efficiently and are also human readable. Another very important advantage is that they do not require the features to be independent among themselves. The algorithm implements a C4.5 decision tree using the information gain ratio of different features. The information gain of an attribute is the expected reduction in entropy because of knowing the value of the attribute [20]. Attributes with higher information gain are likely to be more distinct among the classes, hence they are chosen first while building the decision tree from root to the leaves. The next step is the pruning of the tree. Reduced error pruning starts at the leaves and each node is replaced by the most popular class. If the accuracy of the prediction of the class is not altered then the change is kept and steps are repeated. Using the decision tree with pruning enables our model to run faster as the tree size reduces. Therefore, this algorithm has the capability to deal with noisy datasets containing features that do not contribute towards the ultimate classification model in a substantial way. Due to these reasons, decision trees are widely used in traffic classification [21].

Random Subspace: Decision Tree with Meta-learning: The meta-learning classifier consists of multiple trees constructed systematically by pseudo-randomly selecting subsets of the feature vector. Decision trees are constructed using random subsets of the feature set. Thereafter, the decision of each tree on the data used for prediction is combined together by averaging the conditional probability of each class at the leaves [22]. Decision tree algorithm overfits very easily. Meta-learning classifier helps to avoid overfitting as, at each stage, only a subset of features are used for the model.

7.1.2. Bayesian algorithms

Naive Bayes: This algorithm, which is based on the Bayes theorem, analyzes the inter-relationship between each attribute of the training dataset and the class for each prediction instance (feature vector). The algorithm assigns a conditional probability value to the relationship between the values of the attributes and the classes into which the entire data is classified [11,23,24]. Unlike decision trees, this algorithm cannot remove features that do not contribute towards the classification, and thus requires a thorough feature selection pre-processing stage. Naive Bayes simply relies on each attribute and its relationship with the class. It assumes each of the features to be independent of the others. Due to these properties, it is often used in network traffic analysis [21], even though it is known to perform poorly [23] as it cannot exploit the interdependencies among the features

Bayesian Network: This is a probabilistic graphical model that represents a set of features and classes and their probabilistic relationship via a directed acyclic graph (DAG) [11,24]. Nodes represent features or classes, while links between nodes represent the relationship between them. Conditional probability tables determine the strength of the links. Unlike Naive Bayes, this algorithm does not treat the attributes as independent to each other. This algorithm can find hidden inter-dependencies between the features where they are interrelated. Our dataset has features which are inter-dependent to a certain extent. A case in point is the number of bytes per flow and the number of packets per flow, which have a direct proportional correlation. We use this algorithm as it can maintain the simplicity of Naive Bayes while exploiting the relations between the features that are possible in our feature set.

7.1.3. Artificial neural network based algorithms

Multilayer Perceptron: A MultiLayer Perceptron (MLP) is a feedforward artificial neural network model that maps sets of feature vectors onto a set of appropriate classes [25]. A MLP consists of multiple layers of nodes in a directed graph, with each layer fully connected to the next one. Except for the input nodes, each node is a neuron (or processing element) with a nonlinear activation function. MLP utilizes a supervised learning technique called backpropagation for training the network. MLP is a modification of the standard linear perceptron and can distinguish data that are not linearly separable. In our dataset the attribute values do not vary linearly with the four classes and hence MLP is considered a valid candidate for machine learning algorithm.

7.1.4. k-nearest neighbor

If each feature vector is considered a point in a n-dimensional space, where n is the number of features, this algorithm computes Euclidean distances from each test instance to the k nearest neighbors in that n-dimensional feature space [11]. An instance is classified by a majority vote of its neighbors, with the instance being assigned to the class most common among its k nearest neighbors (k is a positive integer, typically small). We include this algorithm in our list of classifiers as it is shown to converge much faster than the other classifiers especially in the case of network traffic analysis with training flows less than 5000 [11].

7.2. PACL prediction accuracy

For the prediction of location category, the representative features are extracted from an IP address. These features are then used as an input (test data) in the aforementioned model and a location category is predicted.

To check the prediction accuracy of our model we divide our entire data set into n-folds and use $n-1$ folds for training and use the remaining one fold as test data to predict the location class. We repeat this step for the remaining $n-1$ sets of data. Here, we

Table 4
PACL prediction accuracy for all machine learning algorithms.

Machine Learning Algorithm	Correct Instances (%)	Time taken to build the model	Area under ROC Curve
Naive Bayes	963 (54.97)	30 ms	0.808
Multilayer Perceptron	1186 (67.69)	2.75 s	0.870
k-Nearest Neighbor	1224 (69.86)	5 ms	0.807
REP Decision Tree	1433 (81.79)	80 ms	0.923
Random Subspace	1541 (87.95)	110 ms	0.977
Bayesian Network	1570 (89.61)	40 ms	0.986

consider $n = 10$. We use this 10-fold cross validation method on the entire dataset of 1752 devices(or IPs), where 17.9% of instances belong to residential context, 30.2% to university campus, 25.7% to cafeteria context and the remaining 26.2 instances belong to the airport contextual location.

We measure the efficiency of prediction of the location classes on the basis of the following characteristics:

1. **True Positive Rate:** The fraction of instances correctly classified as class A, among all instances actually belonging to class A = $\frac{|TP|}{|TP|+|FN|}$, where TP = number of true positives and FN = number of false negatives.
2. **False Positive Rate:** The fraction of instances which were wrongly classified as class A, among all instances not belonging to class A = $\frac{|FP|}{|FP|+|TN|}$, where FP = number of false positives and TN = number of true negatives.
3. **Area under ROC Curve:** The Receiver Operating Characteristics curve (ROC) plots the variation of false positive rate vs. true positive rate for all the instances of the test data and for each class. The ideal ROC curve approaches the top left corner for 1 true positive rate and 0 false positive rate. The area under the ROC curve ($\in [0,1]$) gives an estimate of the effectiveness of the prediction model. A perfect model has a ROC area of one.
4. **Precision:** The fraction of instances which actually belong to class A, among all classified as class A = $\frac{|TP|}{|TP|+|FP|}$.

The results of our model and its behavior under different machine learning algorithms is presented in Table 4. The table represents the number and percentage of correctly classified instances, the time taken to build each model and the overall area under the ROC curve. As mentioned before, a perfect model, has a ROC area equal to 1.

We observe that while the Naive Bayes, Multilayer Perceptron and k-Nearest Neighbor algorithms do not perform very well, the results of Decision Tree and Random Subspace are acceptable. The Bayesian Network model gives the best prediction accuracy, correctly predicting 1570 out of the 1752 instances giving a prediction rate of 89.61%.

The performance of the different algorithms is as we expected:

- Naive Bayes treats all the attributes as independent which is not the case for our dataset. Hence poor performance is achieved when using this algorithm.
- Multilayer Perceptron handles data that is not linearly separable and thus results in moderate performance. A major disadvantage of this algorithm is that the time taken to build the prediction model is much higher than all the other algorithms that we have dealt with.
- k-Nearest Neighbor being a non-parametric learning algorithm, does not make any assumptions on the dataset (e.g. linearly separable). Thus with our real world dataset, the prediction accuracy is moderate.
- Decision Tree handles both non-linearity and non-independence. Since our dataset is nonlinear and inter-correlated, the results are relatively good. This algorithm also works very well when there is a lot of noisy features. CFS Subset removes all noisy features from the dataset. However, when we use Chi-Squared feature selection, the ultimate dataset is sufficiently noisy. In that case this algorithm gives the best performance.
- The CFS subset feature selection removes features that are very redundant. The features that are left are slightly correlated and these hidden inter-dependencies can be well identified by Bayesian Network, leading to a good prediction accuracy. When the dataset is noisy (as in the case of Chi Square filtered data), the high number of inter-dependencies cannot be identified in a very thorough way, resulting in a prediction model that performs moderately well. Another advantage of this model is that the PACL model can be built much faster than the decision tree algorithms (almost half the time).

One of the major purposes of proposing PACL is to deliver contextual location based services such as third party advertisements or content suggestions to users. In the event of an error in the classification, the content delivered to users will not be optimized based on her location. Most targeted content delivery systems (specially advertisements) do not have access to users' context (as users block the sharing of private information). As a result, the content delivered is not optimized under most circumstances. The error rate of PACL signifies that a user will be misclassified once out of every nine instances, which should not create any significant inconvenience to the network usage experience.

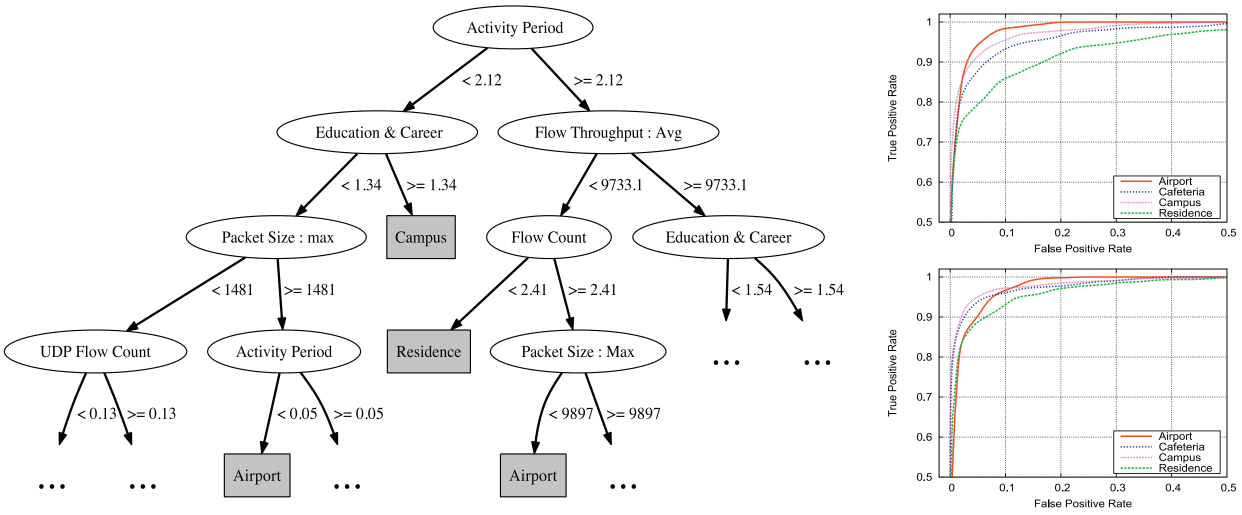
To see how the algorithms perform for each location class, we represent the location category-wise prediction results in Table 5. The prediction is weakest for residential location category across all the different algorithms. The major reason behind this is the lower number of data points (or feature vectors) representing the location category residence as compared to the other 3 locations. The location category residence has around 300 feature vectors whereas all other locations have in excess of 450. However, using Bayesian Network algorithm we see that the prediction accuracy of residential location is not so different from the others and that all the locations have a TP rate which falls within 0.051 of each other (from 0.911 to 0.860). Overall, we observe that airport location category has the best prediction accuracy, whereas cafeteria and campus dataset show similar prediction efficiency.

As the Bayesian Network and the Random Subspace algorithms give us the best accuracy, we look at some of the results for these in more details. We present the confusion matrix for prediction using both the algorithms in Table 6. Each element in the table is represented as (x,y) where x is row number representing the number of IPs actually belonging to that class, and y is column number representing the number of IPs predicted in the corresponding class.

The ROC curves for each algorithm for the 4 location categories are shown in Fig. 7b. The figure as well as Table 5 reconfirm that the prediction is most effective for airport traces whereas residence traces show least effectiveness. However, the ROC curves for the Bayesian Network algorithm are more close together, which confirms our observation above that Bayesian Network gives similar prediction accuracy. Hence the results are very good. for all the location categories.

Table 5
PACL location-wise prediction results : TP and FP rates are calculated for one class against all the other three classes in our dataset.

Algorithm	Location Class	TP Rate	FP Rate	Precision	ROC Area
MultiLayer Perceptron	Airport	0.817	0.075	0.794	0.925
	Cafeteria	0.733	0.165	0.606	0.857
	Campus	0.614	0.113	0.702	0.856
	Residence	0.498	0.081	0.575	0.832
	Combined Results	**0.677**	**0.111**	**0.678**	**0.870**
k-Nearest Neighbor	Airport	0.751	0.065	0.804	0.844
	Cafeteria	0.711	0.117	0.678	0.811
	Campus	0.684	0.129	0.696	0.789
	Residence	0.629	0.093	0.596	0.779
	Combined Results	**0.699**	**0.103**	**0.702**	**0.807**
REP Decision Tree	Airport	0.873	0.072	0.811	0.945
	Cafeteria	0.836	0.065	0.817	0.913
	Campus	0.839	0.072	0.835	0.938
	Residence	0.676	0.038	0.798	0.882
	Combined Results	**0.818**	**0.064**	**0.817**	**0.923**
Random Subspace	Airport	0.950	0.057	0.855	0.989
	Cafeteria	0.882	0.045	0.871	0.975
	Campus	0.902	0.043	0.902	0.971
	Residence	0.737	0.018	0.899	0.952
	Combined Results	**0.880**	**0.043**	**0.881**	**0.977**
Bayesian Network	Airport	0.910	0.056	0.851	0.985
	Cafeteria	0.911	0.028	0.917	0.987
	Campus	0.892	0.020	0.952	0.989
	Residence	0.860	0.033	0.850	0.978
	Combined Results	**0.896**	**0.034**	**0.898**	**0.986**



(a) Decision tree model based on training data of 1752 instances using Random Subspace

(b) ROC curve for individual contextual location classes - using Random Subspace (top) and Bayesian Network (bottom)

Fig. 7. Decision tree and ROC curves for PACL prediction model.

Table 6
Confusion matrix for PACL prediction.

(a) Random Subspace

Classified Class	Airport	Cafeteria	Campus	Residence
Airport	**435**	5	8	10
Cafeteria	24	**397**	23	6
Campus	17	25	**477**	10
Residence	33	29	21	**232**

(b) Bayesian Network

Classified Class	Airport	Cafeteria	Campus	Residence
Airport	**417**	6	15	20
Cafeteria	26	**410**	4	10
Campus	20	19	**472**	18
Residence	27	12	5	**271**

In Fig. 7a, we plot a pruned version of our decision tree model (built using all the CFS subset features). The model shows that the attribute “activity period” has the highest information gain. Fig. 3d shows that the variation of activity period across different location classes is very distinct and hence activity period is most effective in distinguishing the location categories. Fig. 6 shows that this attribute has the highest Chi-squared statistic score. Among all the application based features “the percentage of flows destined to education & career websites” has the highest information gain. The dataset we collect is in a university town (Davis,CA) where the access of school websites is prevalent in almost all location categories. But the amount of usage varies very distinctly at the campus location context, as compared to other locations, as seen in Fig. 5 - hence contributing to high information gain. The nodes

Table 7
PACL model accuracy using different feature-vector subsets for different machine learning models : for each feature subset, CFS subset feature selection is applied and then the model is built. Number of features in each subset after feature selection is shown in Table 8. All results are represented in the form of the percentage of correctly classified instances.

Machine Learning Algorithm	Coarse Grain	Protocol Based	Flow Level	Packet Level	Application Based	All features
Naive Bayes	41.27	33.39	45.32	34.98	35.44	54.97
Multilayer Perceptron	46.63	40.72	47.58	41.78	43.1	69.69
k-Nearest Neighbor	67.35	64.56	51.59	52.97	38.81	69.86
REP Decision Tree	72.83	75.17	58.22	67.01	43.04	81.79
Random Subspace	80.02	83.39	62.38	70.21	42.47	87.95
Bayesian Network	73.12	81.84	58.79	71.86	43.55	89.61

Table 8
PACL prediction model accuracy using different feature-vector subsets : all the models are built after applying CFS subset feature selection and then using the Bayesian Network prediction algorithm.

Feature subset	No. of Features (original set)	No. of Features (CFS Subset)	Correctly Classified Instances (%)	TP Rate	ROC Area	Attributes with highest information gain
Coarse-Grain	7	2	1281 (73.12)	0.731	0.909	Activity period, Flow count
Protocol Based	4	4	1434 (81.85)	0.818	0.953	UDP flow count, HTTP flow count
Flow Level	26	9	1030 (58.79)	0.588	0.824	Application data rate per flow: std. devn., Flow length : max, min Bytes per flow: mean
Packet Level	14	5	1259 (71.86)	0.719	0.903	Packet size:max, Packet size:median, Packet inter-arrival time: max Education and Career
Application Based	33	4	763 (43.55)	0.436	0.693	Emails, Netflix, Games
All Features	**86**	**10**	**1570 (89.62)**	**0.896**	**0.986**	Activity Period, Flow throughput:avg, Education and Career Flow count, Packet Size:max

near the root of the tree includes attributes that belong to all the different subset of features, which shows that the combination of the features are required for efficient prediction. It is also observed in Fig. 7a that the top portion of the tree has no class of cafeteria. We have observed that at least 6 attributes are required to determine the location to be a cafeteria in the best case, whereas that count is 2 for campus, 3 for residence and 4 for airport.

7.3. Prediction accuracy with feature subsets

We predict contextual location based on a number of features which are indicative of network usage patterns of various users. Combination of all features give a good prediction accuracy. But a question may arise as to how a certain subsets of features, calculated on the basis of a particular aspect of an IP address, contribute towards to the accuracy. Performance of the individual subsets of features using the same model and under the same experimental conditions is evaluated. The percentage prediction accuracy using the 4 sets of statistical features and the application based attributes mentioned in Section 5 and comparison with the overall results is shown in Table 7 for all of the machine learning algorithms used. In addition, Table 8 lists the prediction accuracy for the PACL model built using Bayesian Network in details. The table also lists the number of attributes, before and after CFS subset feature selection, TP rate, ROC area and the features that have the highest information gain in each of the attribute subsets.

In our analysis, the statistical features are calculated based on high-level statistics and header information. Payload information is used only in the categorization of application interest among users at various locations. Certain commercial tools [26] are available for extracting application based information systematically from the packet payload [27], more commonly known as Deep Packet Inspection (DPI). There are multiple issues with using DPI. First, most flows in modern day internet traffic are encrypted and hence cannot be decoded. Secondly, looking into the payload leads to privacy leakage issues from users' point of view. Thirdly, this procedure is resource and time intensive. Even though we have looked into payload for the application-based features, we have applied a keyword based search and did not look into the specific content accessed by users. An efficient tool to look into the content accessed by users might help us to distinguish between the applications better and in turn improve the result.

Extracting some of the features from the network traffic by an ISP is computationally simpler and faster for some attributes compared to others. In our feature subset, coarse-grain statistics, like flow count, number of flows belonging to different protocols, packet count, activity period count, etc., are easier to calculate as they are count-based statistics. The other feature values either depend on a particular distribution (packet level and flow level statistics) or require us to look into the payload (application level categorization).

It is observed from Table 7 that only the coarse-grain and protocol based statistical feature subsets individually give highest prediction accuracy in all the models as compared to the other subsets. As a result, we can say, these features are most efficient contributors in our prediction model among all the subsets. The low computational complexity involved in calculating these features for each user is specifically important for real-time prediction. In situations when the prediction has to be done without much delay, the PACL model can use these feature sets and get a prediction accuracy upto 83%.

8. Conclusions

In this paper, we present a model for prediction of users' contextual location by network traffic analysis. Using real world traces we train our model on the basis of statistical and application-based features, to classify users' into four representative contextual locations. The PACL prediction model, in our test case, gives an accuracy upto 89%. Decision tree with metalearning and Bayesian Network algorithms give the best prediction accuracy. However, the preferred algorithm is Bayesian Network as it gives similar efficiency of prediction among all the location classes and the model is built faster.

There are multiple directions of future work. First, looking into the payload of packets is computationally expensive and as a result, we believe that the application based categorization has a scope for improvement. Next, the application of PACL to predict flash-mobs or events (short term gathering) is another scope of the work. If the PACL classification has more than four classes, there would be an overlap of characteristics between the different location classes and machine learning algorithms might not be efficient to identify which distinguishing characters are there in the dataset. In that case, clustering of users based on their application usage would help us identifying the different location categories and give better accuracy than the machine learning algorithms.

Acknowledgments

This work was supported in part by the NSF CNS-1251029 grant and the Intel Science and Technology Center for Secure Computing. The authors would also like to thank Yunze Zeng, Ningning Cheng, Chao Xu and Hao Fu for helping in data collection.

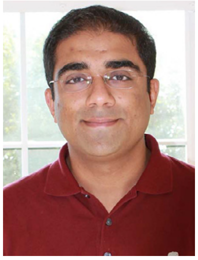
Supplementary material

Supplementary material associated with this article can be found, in the online version, at 10.1016/j.comnet.2017.02.011

References

[1] Foursquare, (https://foursquare.com).
[2] Do Not Track Me Online Act of 2011, (http://www.gpo.gov/fdsys/pkg/BILLS-112hr654ih/pdf/BILLS-112hr654ih.pdf).
[3] I. Poese, B. Frank, G. Smaragdakis, S. Uhlig, A. Feldmann, B. Maggs, Enabling content-aware traffic engineering, SIGCOMM Comput. Commun. Rev. 42 (5) (2012a) 21–28.
[4] I. Poese, B. Frank, S. Knight, N. Semmler, G. Smaragdakis, Padis emulator: an emulator to evaluate cdn-isp collaboration, in: ACM SIGCOMM 2012, in: ACM SIGCOMM '12, 2012b, pp. 81–82.
[5] I. Constandache, S. Gaonkar, M. Sayler, R. Choudhury, L. Cox, Enloc: energy-efficient localization for mobile phones, in: IEEE INFOCOMM, 2009, pp. 2716–2720, doi:10.1109/INFCOM.2009.5062218.
[6] K. Lin, A. Kansal, D. Lymberopoulos, F. Zhao, Energy-accuracy trade-off for continuous mobile device location, in: ACM MobiSys, in: MobiSys '10, ACM, New York, NY, USA, 2010, pp. 285–298, doi:10.1145/1814433.1814462.
[7] P. Newson, J. Krumm, Hidden Markov map matching through noise and sparseness, in: ACM GIS, in: GIS '09, ACM, New York, NY, USA, 2009, pp. 336–343, doi:10.1145/1653771.1653818.
[8] M. Gruteser, D. Grunwald, Anonymous usage of location-based services through spatial and temporal cloaking, in: ACM MobiSys, in: MobiSys '03, ACM, New York, NY, USA, 2003, pp. 31–42, doi:10.1145/1066116.1189037.
[9] M. Damiani, C. Silvestri, E. Bertino, Fine-grained cloaking of sensitive positions in location-sharing applications, Pervasive Comput. IEEE 10 (4) (2011) 64–72, doi:10.1109/MPRV.2011.18.
[10] M. Duckham, L. Kulik, A formal model of obfuscation and negotiation for location privacy, in: Proceedings of the Third international Conference on Pervasive Computing, in: PERVASIVE'05, Springer-Verlag, Berlin, Heidelberg, 2005, pp. 152–170, doi:10.1007/11428572_10.
[11] H. Kim, K. Claffy, M. Fomenkov, D. Barman, M. Faloutsos, K. Lee, Internet traffic classification demystified: myths, caveats, and the best practices, in: ACM CoNEXT, in: CoNEXT '08, ACM, New York, NY, USA, 2008, pp. 11:1–11:12, doi:10.1145/1544012.1544023.
[12] I. Trestian, S. Ranjan, A. Kuzmanovic, A. Nucci, Measuring serendipity: connecting people, locations and interests in a mobile 3g network, in: ACM SIGCOMM, in: IMC '09, ACM, New York, NY, USA, 2009, pp. 267–279, doi:10.1145/1644893.1644926.
[13] Maxmind GeoIP, (http://www.maxmind.com/en/home).
[14] TP-Link TL-WN722N high gain wireless USB adapter, (http://www.tp-link.com/en/products/details/?model=TL-WN722N).
[15] Ath9k - Atheros WLAN driver, (http://wireless.kernel.org/en/users/Drivers/ath9k).
[16] N. Cheng, X. Wang, P. Mohapatra, S. Aruna, Characterizing privacy leakage of public wifi networks for users on travel, IEEE INFOCOM, 2013.
[17] Google adwords: online keyword tool, (https://adwords.google.com/o/Targeting/Explorer?_c=1000000000&_u=1000000000&ideaRequestType=KEYWORD_IDEAS).
[18] H. Liu, R. Setiono, Chi2: feature selection and discretization of numeric attributes, in: International Conference on Tools with Artificial Intelligence, 1995, pp. 388–391, doi:10.1109/TAI.1995.479783.
[19] M.A. Hall, Correlation-Based Feature Subset Selection for Machine Learning, University of Waikato, Hamilton, New Zealand, 1998 Ph.D. thesis.
[20] I.H. Witten, E. Frank, M.A. Hall, Data Mining: Practical Machine Learning Tools and Techniques, Elsevier, 2011.
[21] N. Williams, S. Zander, G. Armitage, A preliminary performance comparison of five machine learning algorithms for practical ip traffic flow classification, SIGCOMM Comput. Commun. Rev. 36 (5) (2006) 5–16, doi:10.1145/1163593.1163596.
[22] T.K. Ho, The random subspace method for constructing decision forests, Pattern Anal. Mach. Intell. IEEE Trans. 20 (8) (1998) 832–844, doi:10.1109/34.709601.
[23] A.W. Moore, D. Zuev, Internet traffic classification using Bayesian analysis techniques, in: ACM SIGMETRICS, in: SIGMETRICS '05, ACM, New York, NY, USA, 2005, pp. 50–60, doi:10.1145/1064212.1064220.
[24] N. Williams, S. Zander, G. Armitage, A preliminary performance comparison of five machine learning algorithms for practical ip traffic flow classification, SIGCOMM Comput. Commun. Rev. 36 (5) (2006) 5–16, doi:10.1145/1163593.1163596.
[25] MultiLayer Perceptron, (http://en.wikipedia.org/wiki/Multilayer_perceptron).
[26] Packeteer, (http://www.packeteer.com).
[27] T. Choi, C. Kim, S. Yoon, J. Park, B. Lee, H. Kim, H. Chung, T. Jeong, Content-aware internet application traffic measurement and analysis, in: IEEE/IFIP NOMS, 1, 2004, pp. 511–524Vol.1, doi:10.1109/NOMS.2004.1317737.

Aveek K. Das is currently a Ph.D. candidate in the Computer Science Department at University of California, Davis. He received his B.E degree in Electronics and Tele-Communication Engineering from Jadavpur University, India in 2012. His current research interests include network analytics, cyber-physical systems, wearable and mobile computing and data mining. He is a recipient of the Best Paper Award at ACM Bodynets 2013.



Parth H. Pathak is an assistant professor in the Computer Science Department at George Mason University. Before joining George Mason University, he was a postdoctoral scholar at University of California, Davis, and before that he received his PhD in Computer Science from North Carolina State University in 2012. He received his M.S. and PhD degrees in Computer Science from North Carolina State University. His research interests include mobile and ubiquitous computing, energy-efficient sensing, Internet-of-Things systems, wireless networking, high-speed millimeter wave wireless networks, and network analytics. He is a recipient of the Award for Excellence in Postdoctoral Research at University of California, Davis in 2015. He has also received the Best Paper Award at IFIP Networking 2014 conference.

Chen-Nee Chuah is a Professor in Electrical and Computer Engineering at the University of California, Davis. She received her Ph.D. in Electrical Engineering and Computer Sciences from the University of California, Berkeley. Her research interests include Internet measurements, network management, data analytics applied to online social networks, security detection, healthcare, and intelligent transportation systems. Chuah is a Fellow of the IEEE and an ACM Distinguished Scientist. She received the NSF CAREER Award in 2003, and the Outstanding Junior Faculty Award from the UC Davis College of Engineering in 2004. In 2008, she was named a Chancellor's Fellow of UC Davis. She has served on the executive/technical program committee of several ACM and IEEE conferences. She has served as an Associate Editor for IEEE/ACM Transactions on Networking and IEEE Transactions on Mobile Computing.

Dr. Prasant Mohapatra is a Professor in the Department of Computer Science and is serving as the Dean and Vice-Provost of Graduate Studies at University of California, Davis. In the past, he has been on the faculty at Iowa State University and Michigan State University. He has also held Visiting Scientist positions at Intel Corporation, Panasonic Technologies, Institute of Infocomm Research (I2R), Singapore, and National ICT Australia (NICTA). He has been a Visiting Professor at the University of Padova, Italy and Yonsei University, and KAIST, South Korea.

Dr. Mohapatra was the Editor-in-Chief of the IEEE Transactions on Mobile Computing. He has served on the editorial board of the IEEE Transactions on Computers, IEEE Transactions on Mobile Computing, IEEE Transaction on Parallel and Distributed Systems, ACM WINET, and Ad Hoc Networks. He has served as the Program Chair and the General Chair and has been on the program/organizational committees of several international conferences. Dr. Mohapatra received his doctoral degree from Penn State University in 1993, and received an Outstanding Engineering Alumni Award in 2008. He is also the recipient of Distinguished Alumnus Award from the National Institute of Technology, Rourkela, India. He received an Outstanding Research Faculty Award from the College of Engineering at the University of California, Davis. He received the HP Labs Innovation awards in 2011, 2012, and 2013. He is a Fellow of the IEEE and a Fellow of AAAS.

Dr. Mohapatra's research interests are in the areas of wireless networks, mobile communications, cybersecurity, and Internet protocols. He has published more than 300 papers in reputed conferences and journals on these topics. Dr. Mohapatra's research has been funded through grants from the National Science Foundation, US Department of Defense, Army Research Labs, Intel Corporation, Siemens, Panasonic Technologies, Hewlett Packard, Raytheon, Huawei Technologies, and EMC Corporation.