Privacy Analysis of User Association Logs in a Large-scale Wireless LAN

Keren Tan   Guanhua Yan[†]   Jihwang Yeo   David Kotz
ISTS, Dartmouth College   [†]CCS-3, Los Alamos National Laboratory

Abstract—User association logs play an important role in wireless network research. One concern of sharing such logs with other researchers, however, is that they pose potential privacy risks for the network users. Today, the common practice in sanitizing these logs before releasing them to the public is to anonymize users' sensitive information, such as their devices' MAC addresses and their exact association locations. In this work, we aim to study whether such sanitization measures are sufficient to protect user privacy. By simulating an adversary's role, we propose a novel type of correlation attack in which the adversary uses the anonymized association log to build signatures against each user, and when combined with auxiliary information, such signatures can help to identify users within the anonymized log. Using a user association log that contains more than four thousand users and millions of association records, we demonstrate that this attack technique, under certain circumstances, is able to pinpoint the victim's identity exactly with a probability as high as 70%, or narrow it down to a set of 20 candidates with a probability close to 100%. We further evaluate the effectiveness of standard anonymization techniques, including generalization and perturbation, in mitigating correlation attacks; our experimental results reveal only limited success of these methods, suggesting that more thorough treatment is needed when anonymizing wireless user association logs before public release.

I. INTRODUCTION

To preserve users' privacy, a network trace publisher must *sanitize* the traces before sharing them with the public. Although many network sanitization techniques have been proposed and developed, recent research has shown that these techniques provide limited protection against user (or host) re-identification attacks. Existing sanitization techniques usually deal with explicit sensitive fields in the dataset, such as IP/MAC addresses, port number, and TCP/UDP payloads, but ignore implicit information that can be potentially extracted and used to identify an anonymized user (or host). For an enterprise-wide network with thousands of users, privacy analysis on *wired* network traces has been widely studied to understand the severity of some potential trace-sharing risks [1], [2]. However, similar research is scarce for enterprise-wide, large-scale *wireless* networks [3], [4]. As the edge of the Internet is increasingly becoming wireless, and because wireless networks have some unique characteristics, such as user mobility, it is important to evaluate privacy threats posed due to shared wireless network traces.

In this paper, we conduct privacy analysis on one of the simplest wireless network traces, a user association log collected from a large-scale WLAN. Such a log keeps a record of each association and disassociation event between users' wireless devices and the network's access points (APs). Compared to other semantically rich wireless-network traces, we would hope the simplicity of the user association log could make it more resistant to potential privacy risks. We consider the following two questions: 1) Using only the "insensitive" information in an anonymized user association log, is it possible to build a signature for each user, such that when these signatures are combined with auxiliary information, an adversary can distinguish users within the anonymized log. 2) If a privacy breach is possible, how effective are traditional mitigation approaches in preventing an adversary from building such signatures?

In a nutshell, we make three major contributions in this work. First, we simulate the role of an adversary and propose a "correlation attack" – a method based on Conditional Random Field (CRF) – that can be used to breach user privacy from a released WLAN user association log. Second, we use extensive experiments to demonstrate the effectiveness of the CRF-based correlation attack. Using an anonymized campus-wide WLAN user association log with more than four thousand users and millions of user association records, and a short-term observation of the victim's association activities, we show that the CRF-based correlation attack, under certain circumstances, can reveal the victim's identity in the released dataset with a probability as high as 70%, or narrow down the victim's identity among 20 candidates with a probability close to 100%. Third, we evaluate the effectiveness of standard sanitization techniques, including generalization and perturbation, in mitigating the proposed correlation attack; the results reveal only limited success of these methods, suggesting that more thorough treatment is needed when anonymizing wireless user association logs before the public release.

II. RELATED WORK

To share network traces while preserving privacy, data publishers usually define sanitization policies according to their specific privacy concerns. These policies determine which sanitization methods to apply and how [5]. Due to the intrinsic complexity of network trace sanitization, however, recent research has revealed that there are few, if any, available network-trace sanitization schemes that can provide a watertight guarantee under the worst-case analysis. These works often mimic the role of an adversary that tries to launch a de-sanitization attack against the sanitized trace [1], [2]. For a comprehensive survey of state-of-art network trace sanitization and de-sanitization research, we refer interested readers our previous work [6].

In the domain of wireless networks, many physical-device-fingerprinting techniques could potentially be used to launch de-sanitization attacks [7], [8]. Because most of these techniques work by monitoring unique variations in protocol behaviors, such as those seen across different vendors or device-driver implementations, they often require very-high-

resolution data or even special measurement equipment. Such requirements greatly limit their applicability for de-sanitization on most types of released traces. Pang's work [9] relies on much more abundant trace semantics than our work and has only been evaluated with much smaller wireless network traces than the one we used. Most close to this work, Kumar and Helmy have recently shown that it is possible to breach privacy from WLAN user association logs [4]. Their attack model assumes that the adversary can inject data into the wireless network during the trace collection or has some out-of-band information such as the victim's academic major and gender. The attack discussed in this paper, however, does not require these assumptions.

III. WLAN User Association Logs

At Dartmouth College, we have been monitoring the campus-wide WLAN network usage since 2001. As of January 2010, this WLAN network consists of over 1300 Aruba APs that provide 54Mbps coverage to the entire campus. These Aruba APs are connected with and controlled by a small set of Aruba Mobility Controllers. We poll every controller every 5 minutes using the SNMP protocol and receive replies that contain a list of users associated with each AP. After processing these replies, each row of the resulting user association log, which we call a *user association record*, has 4 comma-separated fields: the MAC address of the wireless card, the name of the AP that the wireless card has connected with, and the start and the end POSIX timestamp of this connection. The following is a snippet of the user association log that we extract from the SNMP information (it shows anonymized MAC addresses to protect user privacy):

```
001d4f3bc496,14.5.1, 1251690285,1251691544
0021e9082bfd,142.6.1,1251689384,1251691544
```

There are a few things worth noting. First, although it is possible that a wireless card may have been used in multiple devices or a device has been used by multiple people, we assume that such cases are rare in our dataset. Hence in this paper we use a "wireless card" and a "network user" (or a "user") interchangeably. Second, because the Aruba Mobility Controller only generates the start timestamp for each connection and we poll the controller every 5 minutes, the connection's end timestamp is only an estimated value, whose error is therefore bounded by 5 minutes. Third, we use a hierarchical naming scheme for APs in the dataset. For an AP named $x.y.z$, x is its building number, y is its floor number, and z is its serial number within the floor.

Sanitization. We use one-to-one mapping function to rename the MAC addresses in the original dataset. Hence, the anonymized MAC addresses in the sanitized dataset do not have any physical meaning and thus are only symbolic names; a similar sanitization scheme has been used in other work [4]. By leveraging the hierarchical naming scheme, we truncate an AP's name according to different sanitization levels. For example, if we want to only keep building and floor information, we truncate the AP's name from $x.y.z$ to $x.y$.

IV. Threat Model and Problem Formulation

Complying with Narayanan's definition of privacy breach [10], the threat we study here is whether the limited insensitive information left in a sanitized association log could still form implicit signatures for individual users. These implicit signatures, when combined with auxiliary information, may provide the adversary the knowledge that the sanitization process has aimed to protect, such as whether an anonymized ID in the released dataset corresponds to a specific user. We make the following three assumptions in our threat model.

Assumption 1: The adversary has access to a sanitized WLAN user association log $\mathcal{L}_s$, which is released to the public by a trace publisher. There are N_s users in this association log. All users' real MAC addresses are anonymized in $\mathcal{L}_s$ as follows: during the trace publisher's sanitization process, each real MAC address has been replaced with a new identifier ID_i ($1 \leq i \leq N_s$) according to some one-to-one one-way mapping function. Hence, given an anonymized MAC address ID_i, the adversary cannot find the real MAC address that is mapped to ID_i. The AP's name can be either preserved or truncated. The rest of the fields, such as the start and end timestamp of each connection, are preserved during the sanitization process.

Assumption 2: The adversary knows a sequence of association records about a victim user's device. This sequence of records, $\mathcal{Q}$, need not be collected during the same time period as $\mathcal{L}_s$ (otherwise the problem will be trivial). It is important to note that the information provided in $\mathcal{Q}$ can be rather coarse. For example, the adversary may only need to know which buildings the victim has visited rather than which exact APs the victim has associated with.

Assumption 3: The adversary knows that the sanitized dataset $\mathcal{L}_s$ must contain the victim's AP association records. In many cases, $\mathcal{L}_s$ is published at an organization level (e.g., by a university) and thus contains complete AP association logs of the organization's wireless users. Hence, if the adversary knows that the victim was a member of the organization when $\mathcal{L}_s$ was collected, it is easy for him to know that $\mathcal{L}_s$ should contain the victim's AP association records.

Given the three assumptions in the adversarial model, the *(exact) correlation attack problem* is then formulated as follows: *given $\mathcal{L}_s$ and $\mathcal{Q}$, which anonymized identity ID_i ($1 \leq i \leq N_s$) in $\mathcal{L}_s$ has also generated $\mathcal{Q}$?* In practice, however, due to incomplete data for training or inference, or some intra- and inter-user association activity variations, finding an algorithm to solve the exact correlation attack problem is difficult or even impossible. In this work, we consider a relaxed and more practical version of this problem. The *(relaxed) correlation attack problem* is formulated as follows: *given $\mathcal{L}_s$ and $\mathcal{Q}$, which subset of anonymized identities would contain the one that generated $\mathcal{Q}$ with high probability?*

V. Algorithm Description

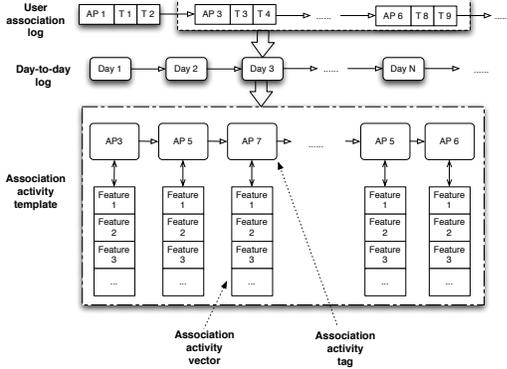In the previous section, we formulate correlation attack as a classification problem, in which the two key components

Fig. 1. Represent each user's association log as association activity template.

TABLE I
FEATURES OF AN ASSOCIATION ACTIVITY VECTOR

Feature name	Meaning	Value	Comments
duration	Adjusted connection duration	Integer	Normalized [11]
day of week	Day of the week of this record	Enum. type, from Monday to Sunday	To represent periodic patterns [12]
starting time	Time slot of a day of this record	Enum. type, from Midnight to Night	
previous AP	The AP in the previous record		To represent context informationg [13], [14]
next-to-previous AP	The AP in the next-to-previous record	String, AP's name	
next AP	The AP in the next record		

are feature representation and the learning algorithm. We use association activity templates to represent user association logs and employ CRF as the learning algorithm.

A. Data Representation

We propose a new approach that uses *association activity templates* to represent user association logs. In this method, we first split the user's association log into day-to-day pieces and then for each day build an individual association activity template, because human activities often exhibit regularities associated with days of the week. An association activity template is a collection of association activity tags and their corresponding association activity vectors. As shown in Figure 1, the association activity tag is the name of the visited AP. Each element in an association activity vector is called a *feature*. In the current implementation, we let an activity vector have six features: *duration*, *day of week*, *starting time*, *previous AP*, *next to previous AP*, and *next AP*. Table I explains these features.

B. Algorithm Procedure

As an overview of the correlation attack, we describe the attack algorithm in this section and defer the introduction to CRF to Section V-C.

Step 1. For each user in $\mathcal{L}_s$, split his/her association log into day-to-day pieces and represent each day's log using an association activity template as described in Section V-A.

Step 2. Feed each user's association activity templates into a linear-chain CRF to model this user's association behavior. As there are N_s users in $\mathcal{L}_s$, we build N_s CRF models. The input fed to a CRF model is a sequence of association activity vectors (Figure 1) and the output is a sequence of association activity tags, which are actually AP names. Let $CRF_i(\mathcal{V})$ denote the output from the i-th user's CRF model, where $1 \leq i \leq N_s$ and $\mathcal{V}$ denotes the sequence of association activity vectors fed to the CRF model.

Step 3. For the observed user association record $\mathcal{Q}$, we preprocess it as described above to obtain an association activity template $\mathcal{T}$. Let $\mathcal{V}_\mathcal{T}$ and $\mathcal{G}_\mathcal{T}$ denote the sequence of association activity vectors and the sequence of association activity tags in template $\mathcal{T}$, respectively.

Step 4. We feed $\mathcal{V}_\mathcal{T}$ to all CRF models trained in Step 2 and count the number of tags that overlap between $\mathcal{G}_\mathcal{T}$ and $CRF_i(\mathcal{V}_\mathcal{T})$ ($1 \leq i \leq N_s$), a score we denote w_i. The intuition applied here is that the victim's CRF model is more likely to produce correct activity association tags from her observed activity association vectors in $\mathcal{Q}$, and therefore score w_i is higher than the others if ID_i is the victim's identifier in the released user association log.

Step 5. We sort all users based on score w_i in non-increasing order and the algorithm outputs this sorted list.

Ideally the top identifier on the sorted list should be treated as the sole candidate that generated the observed user association sequence $\mathcal{Q}$. In practice, however, due to incomplete data for training or inference, or some intra- and inter-user association activity variations, the top identifier may not correspond to the victim who produced $\mathcal{Q}$. As mentioned earlier, we tackle the relaxed correlation attack problem instead and thus use a small number of top identifiers on the sorted list. Clearly, from the attacker's perspective, the smaller the number of top identifiers needed to include the victim's, the more successful his attack.

C. Conditional Random Field

Let $X = (X_1, X_2, ..., X_n)$ denote a random variable of an observed sequence, each element of which has k features. In our problem, a realization of X is a sequence of association activity vectors with the six features described in Table I. Let Y denote a random variable of a label sequence. A label here is actually an association activity tag that indicates an AP name. According to Figure 1, each association activity vector corresponds to an association activity tag. Hence, given an observed sequence of X (i.e., sequence $\mathcal{V}_\mathcal{T}$ in Step 3 of the algorithm shown in Section V-B), we need to produce a label sequence for it. It is thus a task of assigning label sequences to observation sequences, which is common to many applications in bioinformatics, computational linguistics and speed recognition [15], [16].

The Hidden Markov Model (HMM) is known to be a popular *generative* model that characterizes the joint distribution $p(X, Y)$ directly [16]. The challenge facing HMM is that it has to model the entire set of observation sequences $p(X)$ explicitly, which is intractable in our case (and many other domains) due to the limited amount of data to estimate a full-fledged $p(X)$ and the correlation between the features in X (the features in the association activity vector). The CRF method, in contrast, eliminates the necessity of knowing $p(X)$

by building models to predict label sequences Y conditional on observation sequences X. Hence, CRF is indifferent to the dependence among features in X because X is now treated as given (i.e., a condition). Because CRF models the conditional probability $p(Y|X)$ instead of the joint distribution $p(X,Y)$, it is a *discriminative* approach rather than a generative one. In this work, we used CRFsuite [17], a linear-chain CRF implementation for parameter estimation and inference. Due to limited space, we refer interested readers to the literature for more thorough treatment on the topic of CRF [18].

VI. Experimental Evaluation

We use 62-day user association log collected at Dartmouth College between January 4, 2010 and March 6, 2010, in this evaluation. We filter out transient users who were active in fewer than 45 days during this 62-day period, and the resulting dataset still contained 2,450,903 (79.67%) user association records with 4,285 distinct users and 1,364 distinct APs. The 62-day user association log is partitioned into 10 bins of approximately the same length for each user. In the j-th round ($1 \leq j \leq 10$), we use the j-th bin of each user's association records as the testing dataset ($\mathcal{L}_u$) and the remaining nine as the training dataset ($\mathcal{L}_s$) to build the CRF models. The results shown below are the 10-round averages.

The *Minimum Size of Candidate Identifier Set* (MSCIS) is our metric to measure the attack efficiency. Consider the relaxed correlation attack problem with a sanitized user association dataset $\mathcal{L}_s$ and an observed sequence of AP association records $\mathcal{Q}$. For each ID_i where $1 \leq i \leq N_s$ in $\mathcal{L}_s$, we compute score w_i according to Step 4 in the CRF-based method. Suppose that ID_j is the user ID of the victim who generated $\mathcal{Q}$. The MSCIS is defined as the number of user IDs whose scores are no smaller than w_j. MSCIS establishes an upper bound on how many candidate user IDs need be considered in order to contain the victim's user ID in the sanitized dataset. Note that if a user has the same score as the victim's (i.e., w_j), his ID should also be counted into MSCIS.

To set up a baseline case for comparison, we developed a simple distance-based method described as follows:

Step 1. For each user in $\mathcal{L}_s$, we build a time vector each day that contains how much time this user spent at each AP.

Step 2. Similarly, we compute a set of daily time vectors for each user in $\mathcal{L}_u$.

Step 3. For each user in $\mathcal{L}_u$, we compute the Euclidean distance between each of her time vectors and every user's time vectors in $\mathcal{L}_s$, to obtain an average score for every user in $\mathcal{L}_s$.

Step 4. For each user in $\mathcal{L}_u$, we sort the scores derived from Step 3 in non-decreasing order to obtain a sorted list of user IDs in $\mathcal{L}_s$, then compute the MSCIS for each user in $\mathcal{L}_u$.

Figure 2 compares the results of the CRF-based method and the distance-based method. The sanitization is done by anonymizing only the MAC addresses but leaving the other fields intact. When the length of $\mathcal{Q}$ is 5-6 days, the CRF-based method significantly outperforms the distance-based method in attack efficiency: 73.2% of the 4,285 users can
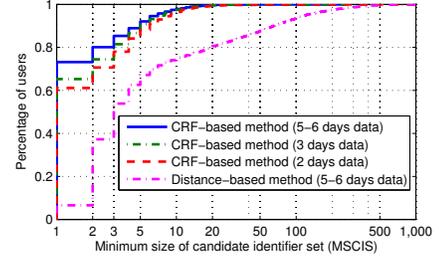

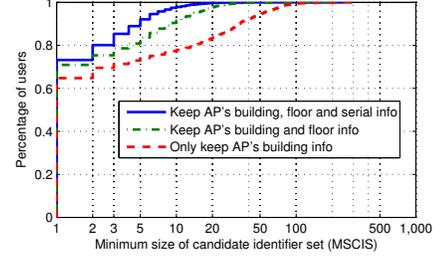Fig. 2. Relationship between the attack performance and the amount of auxiliary information.

Fig. 3. Effectiveness of generalization-based mitigation against the proposed correlation attack.

be pinpointed exactly from $\mathcal{L}_s$; for 99.7% of the users, their MSCIS is no more than 20. Hence, using the CRF-based method, the adversary could almost surely narrow down the victim's possible user ID into a set of 20 candidates from an anonymized dataset with more than 4,000 users. By tuning the length of $\mathcal{Q}$ to different values (from 5-6 days to 2 or 3 days), we show how the amount of auxiliary knowledge affects the attack efficiency. Clearly, reducing the auxiliary knowledge available to the attacker (shorter $\mathcal{Q}$) degrades the performance of the attack. However, even in the worst case here that the length of $\mathcal{Q}$ is only two days, the adversary still can pinpoint her identity exactly from $\mathcal{L}_s$ with probability 61.7%, and for 98.5% of the users, he can narrow down her identity in $\mathcal{L}_s$ to only 20 candidates. From the attacker's perspective, this is favorable because he needs to know a victim's association activities for only a short period to launch the correlation attack effectively.

VII. Mitigation Strategies

As a network trace publisher ourselves and as the host of the CRAWDAD [19], we are concerned about how effectively standard sanitization measures can prevent such privacy breaches.

A. Generalization

Recall that the AP-naming scheme in the user association logs uses a hierarchical structure: building ID, floor level, and AP serial number. We consider two generalization schemes here: one keeping only the building information of each AP, and the other keeping both the building ID and the floor level. The results on these two anonymized datasets, together with results from CRF without any generalization, are depicted in Figure 3. All the experiments in Section VII work on the same sanitized dataset $\mathcal{L}_s$ and unsanitized dataset $\mathcal{L}_u$ (with 5-6 days) as those in the previous section.
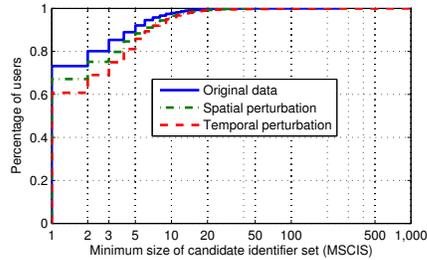
Fig. 4. Effectiveness of perturbation-based mitigation against the proposed correlation attack.

It is clear that applying generalization-based anonymization techniques helps mitigate correlation attacks. On the other hand, because keeping only the AP's building information is the best we can do to generalize AP names, we can see only limited effectiveness of generalization-based schemes in mitigating correlation attacks on user association logs.

B. Perturbation

Based on the characteristics of the user association logs, we consider two perturbation methods: spatial perturbation and temporal perturbation. The *spatial perturbation* method changes the AP information in the original dataset as follows. Let $\mathcal{S}_i$ denote the sequence of user ID_i's AP association records, sorted in increasing order of starting timestamps. For each record R_j in $\mathcal{S}_i$, we change the AP in R_j to the AP in R_{j-1} with probability 15%, change it to the AP in R_{j+1} with probability 15%, or keep it intact with probability 70%. The *temporal perturbation* method changes the start and end timestamps in the original dataset by adding Gaussian noise with mean 0 and standard deviation 3600 seconds to these two timestamps. The effectiveness of both methods in mitigating correlation attacks is illustrated in Figure 4.

Considering the results in Figures 3 and 4, we conclude that for all the mitigation techniques evaluated, their effectiveness in mitigating CRF-based correlation attacks is rather limited. Although adding more noise in the perturbation-based methods can further constrain the adversary's capability in launching correlation attacks, it may also damage the usability of the released user association datasets.

VIII. CONCLUSION

User association logs collected from real-world WLANs have played an important role in understanding these networks. Sharing them with the public, however, poses potential risks to the privacy of the users involved. In this work, we show that people's association behaviors form implicit signatures for individual users. When combined with auxiliary information, such signatures can help reveal the true identities of anonymized IDs in a sanitized WLAN user association log. On a pessimistic note, standard anonymization techniques, such as generalization and perturbation, are unable to mitigate such CRF-based correlation attack effectively. The results from this work call for a more thorough study of potential privacy risks when wireless user association logs are shared with the public. For a more complete presentation of the results in this paper,

see our technical report [20]. A preliminary, short version of this paper appeared in the ACM MobiCom S3 workshop [3].

REFERENCES

[1] T. Brekne, A. Årnes, and A. Øslebø, "Anonymization of IP traffic monitoring data: Attacks on two prefix-preserving anonymization schemes and some proposed remedies," in *Proceedings of PET*, May 2005.

[2] S. E. Coull, C. V. Wright, F. Monrose, M. P. Collins, and M. K. Reiter, "Playing Devil's advocate: Inferring sensitive information from anonymized network traces," in *Proceedings of NDSS*, Feb. 2007.

[3] K. Tan, G. Yan, J. Yeo, and D. Kotz, "A correlation attack against user mobility privacy in a large-scale WLAN network (extended abstract)," in *Proceedings of the ACM Mobicom S3 workshop*, Sep. 2010.

[4] U. Kumar and A. Helmy, "Human behavior and challenges of anonymizing WLAN traces," in *Proceedings of GLOBECOM*, Nov. 2009.

[5] R. Pang, M. Allman, V. Paxson, and J. Lee, "The devil and packet trace anonymization," *ACM SIGCOMM Computer Communication Review*, vol. 36, no. 1, pp. 29–38, 2006.

[6] K. Tan, J. Yeo, M. E. Locasto, and D. Kotz, "Catch, clean, and release: A survey of obstacles and opportunities for network trace sanitization," in *Privacy-Aware Knowledge Discovery: Novel Applications and New Techniques*. Chapman and Hall/CRC Press, Dec. 2010.

[7] J. Franklin, D. McCoy, P. Tabriz, V. Neagoe, J. V. Randwyk, and D. Sicker, "Passive data link layer 802.11 wireless device driver fingerprinting," in *Proceedings of USENIX Security*, Jul. 2006.

[8] K. Bauer, D. McCoy, B. Greenstein, D. Grunwald, and D. Sicker, "Using wireless physical layer information to construct implicit identifiers," in *Proceedings of HotPETS*, Jul. 2008.

[9] J. Pang, B. Greenstein, R. Gummadi, S. Seshan, and D. Wetherall, "802.11 user fingerprinting," in *Proceedings of ACM MobiCom*, Sep. 2007.

[10] A. Narayanan and V. Shmatikov, "Robust de-anonymization of large sparse datasets," in *Proceedings of IEEE S&P*, Dec. 2008.

[11] W. J. Hsu, D. Dutta, and A. Helmy, "Mining behavioral groups in large wireless LANs," in *Proceedings of ACM MobiCom*, Sep. 2007.

[12] M. Kim and D. Kotz, "Periodic properties of user mobility and access-point popularity," *Journal of Personal and Ubiquitous Computing*, vol. 11, no. 6, pp. 465–479, Aug. 2007.

[13] L. Song, D. Kotz, R. Jain, and X. He, "Evaluating next cell predictors with extensive Wi-Fi mobility data," *IEEE Transactions on Mobile Computing*, vol. 5, no. 12, pp. 1633–1649, Dec. 2006.

[14] T. Kudoh and Y. Matsumoto, "Use of support vector learning for chunk identification," in *Proceedings of CoNLL and LLL*, Sep. 2000.

[15] R. Durbin, S. Eddy, A. Krogh, and G. Mitchison, *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. Cambridge University Press, 1998.

[16] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–285, Feb. 1989.

[17] N. Okazaki, "CRFsuite: a fast implementation of Conditional Random Fields (CRFs)," http://www.chokkan.org/software/crfsuite/, 2007.

[18] C. Sutton and A. McCallum, *Introduction to Conditional Random Fields for Relational Learning*. MIT Press, 2006.

[19] "Community Resource for Archiving Wireless Data At Dartmouth (CRAWDAD)," http://www.crawdad.org/.

[20] K. Tan, G. Yan, J. Yeo, and D. Kotz, "Privacy analysis of user association logs in a large-scale wireless LAN," Dept. of Computer Science, Dartmouth College, Tech. Rep. TR2011-679, Jan. 2011.