# Towards Accountable Mobility Model:
## A Language Approach on User Behavior Modeling in Office WLAN

Jiang Zhu, Ying Zhang
Carnegie Mellon University
Moffett Field, CA, USA
{jiang.zhu, joy.zhang}@sv.cmu.edu

***Abstract*—Modeling mobile users' behavior can lead to crucial applications in accountable mobile computing such as casual authentication and anomaly detection. We introduced a language approach to model mobile users' behavior from heterogeneous sensor data. By converting temporal and spacial features generated from WiFi RSS trace into symbols and fusing them into a 1-dimension "language" representation, we were able to leverage algorithms developed for statistical NLP to build accountable user mobility models in an office WLAN environment. We explored the continuous $n$-gram and skipped $n$-gram models to detect anomaly of mobile user's behavior, such as device theft. We have collected data from a corporate office environment over 5 days. The proposed model only needs to observe the users for 8 hours to build a reliable behavior model which can detect 86% of simulated anomalies. We have also evaluated the effectiveness of using the $n$-gram models to predict the future location of the user.**

***Index Terms*—Computer-Communication Networks, User Behavior Modeling, Mobile Security.**

## I. Introduction

Mobile applications and devices are becoming ubiquitous and will increasingly interact with different sensors, services and other mobile users. It is crucial for mobile users to privately and securely interact with their environment and data and for mobile services to trust the identity of the user. While mobile devices such as smartphones make our lives convenient in ways that were unimaginable before, broad adoption of mobile applications such as email, messaging, online banking and mobile finance subject our identities and privacy to greater risks when sensitive information can be accessed from these devices. Moreover, with the proliferation of mobile devices in corporate environments, keeping the device and its data secure become a critical task for both individuals and corporations.

Recently, a new survey [1] has revealed that 36 percent of consumers in the United States have either lost their mobile phone or had it stolen. Another survey [2] has also revealed that 329 organizations polled had collectively lost more than 86,000 devices with average cost of *lost data* at $49,246 per device, worth $2.1 billion or $6.4 million per organization. Given the high loss rate and high cost associated with them, accountable schemes are needed to promptly and accurately detect these losses or thefts. Such detection will facilitate subsequent notification, mitigation and recovery process to control or even avoid the damages.

Today, several applications [17] [5] have been manually constructed to leverage sensed contextual information such as users's location and time of the day to provide secure and private access to services in a mobile computing environment. However, users must explicitly interact with the security aspect of the system, and developers have to manually and laboriously develop context models and supporting code afresh for each application. The heterogeneity of the context makes it even harder to fuse them in order to make security decisions.

Therefore, we envision an *accountable mobile computing* system which constantly monitors the user's behavior and enforce security policies when abnormal behavior is observed. Such casual behavior-driven scheme does not require explicit interactions from the user and is harder to replicate.

In this paper, we propose an accountable mobility model based on user behavior modeling. Mobile users' behaviors are modeled via a symbolic representation which we refer to as "behavior text". By converting time-series sensory information into symbols and fusing heterogeneous sensor "strings" into a single-dimensional behavioral text string, many well-studied text processing algorithms can be applied on the sensor data for interesting applications such as activity recognition, classification, prediction, retrieval, and summarization. In particular, we explored the case of using $n$-gram models to learn user's mobility behavior from the WiFi traces combined with temporal information for casual authentication, anomaly detection and future activity prediction.

## II. Related Work

The use of probabilistic methods for modeling user mobility behavior in wireless network including WiFi and Cellular environment has been attempted in various previous work [8], [10], [13], [14], [16] with predominate focus on using probabilistic approaches such as Bayesian and Markov models.

Work in [7] by Kim et al. is one of the first attempts to construct a WLAN mobility model from real-world wireless user traces. Using *syslog* approach, they collected traces containing sequences of WLAN association records. They explored several methods to extract mobility tracks, including *triangle centroid*, *time-based centroid* and *Kalman filter*. They also developed a heuristic to extract pause time from mobility tracks. They validated these methods through controlled walks

[1] Strategy One survey conducted among a U.S. sample of 3,017 adults age 18 years or older on September 21-28, 2010, with an oversample in the top 20 cities (based on population)

[2] "The Billion Dollar Lost-Laptop Study" conducted by Intel Corporation and the Ponemon Institute, analyzed the scope and circumstances of missing laptop PCs

where the tracks (via GPS) and pause-time were also recorded. Using the similar data set, Lee and Hou presented a semi-markov model [8] for characterizing user mobility both in the temporal and spatial domains. With the semi-markov model, they were able to capture and analyze both the steady-state and transient behaviors of user mobility. From the steady-state analysis, they were able to identify the long-term mobility characteristics, such as the steady-state user distribution over APs. Among other contributions, the effect of ping-pong phenomena in user mobility was analyzed and a solution to identify and remove ping-pong transitions was proposed.

As suggested by the close ties between human activity and language, predictive models used in language modeling can also be applied to mobility behavior modeling. Aipperspach et al in [1] showed that smoothed $n$-grams can provide a fast and accurate method for making single-step predictions on binary sensor output and user's location in a smart-home environment. The results also showed that the mapping between language and human behavior, evidenced through the Zipf distribution of movement sequences and the "local structure" of those sequences, supports the application of language models to user behavior modeling. The simplicity of their approach inspire us to explore similar language approach to model user mobility behavior with fine-grain sensory data with heterogenous nature. We have demonstrated that such simple language approaches can be used to model user activities in our lifelogger system [4] and to perform anomaly detection in our geo-tracing system [3]

Our work differs from the aforementioned efforts in that 1) we collected Received Signal Strength (RSS) readings of the beacons emitted from mobile devices and aggregate the data from multiple Wireless Access Points (WAPs) to form sequences of fingerprints, which can be used to easily infer the actual locations of the devices. This approach reduces the overhead to estimate mobility tracks and pause time. 2) We converted heterogenous sensory data into behaviorial text and adopt $n$-gram model to efficiently construct sufficient statistics of user mobility behavior, capturing both the steady-state (uni-gram) and transient behavior (bi-gram and $n$-gram) simply through counting. 3) We used *skipped* $n$-gram model to capture the long-term dependency in user mobility behavior. Ping-pong phenomena can also be partially mitigated through this method if ping-pong transitions are skipped.

## III. User Behavior Modeling by Text Representation

Mobile devices are equipped with various types of sensor which could constantly monitor certain aspects of a user's behavior. However, these sensors are heterogeneous in data sampling rate, data format and meaning of sensor readings. Other approaches such as [6], [11], [15] build classifiers on each sensor readings and combine the classification results to model user's behavior.

Based on the principle of "language as action", natural language and human behavior share common characteristics. They are both "meditational means" or tools by which we achieve our ends, their meanings both depend on the structure of the observed sequence of the composing elements, and they both have grammars that help to explain the underlying "syntactic structure" of an observed sequence. By representing the sensor readings as symbols through quantization or clustering and applying statistical natural language processing algorithms on these symbols sequences, we can then picturing mobile sensors constantly "writing" a lifelogger in text about our lives. Applying statistical natural language processing algorithms on this *lifelog* will allow us to build behavior models for a user.

In this section, we describe using continuous $n$-gram and skipped $n$-grams to model users' mobility behavior once the input is converted into behavior text.

### A. Continuous $n$-gram model

$n$-gram models have proven to be very robust in modeling sequences of data. In our study, we modeled WiFi traces using the $n$-gram model. Similar to natural language, we assume that the sequence of mobile device's pseudo locations can also be predicted by $n$ consecutive locations in the past. We consider a pseudo location label as a "word" in the language and train an $n$-gram language model on the WiFi trace data as text by converting each WiFi RSS vector into a symbolic label.

The model estimates the next pseduo-location label $l_i$ given the previous $n-1$ pseudo locations from the WiFi trace as $P(l_i|l_{i-n+1}, l_{i-n+2}, \ldots, l_{i-1})$ or $P(l_i|l_{i-n+1}^{i-1})$ in short. We can also estimate the probability of a sequence of pseudo location labels $L = l_1, l_2, \ldots, l_N$ as

$$P(L) = P(l_1, l_2, \ldots, l_N) = \prod_{i=1}^{N} P(l_i|l_{i-n+1}^{i-1}) \qquad (1)$$

or average log probability as

$$\frac{1}{N}\sum_{i=1}^{N} \log P(l_i|l_{i-n+1}^{i-1}) \qquad (2)$$

The model probabilities $P(l_i|l_{i-n+1}^{i-1})$ can be estimated through the Maximum Likelihood Estimation (MLE) from the training data by counting the occurrences of pseudo location labels: $P_{\text{MLE}}(l_i|l_{i-n+1}^{i-1}) = \frac{C(l_{i-n+1},\ldots,l_{i-1},l_i)}{C(l_{i-n+1},\ldots,l_{i-1})}$

MLE assigns probability zero to any unseen $n$-grams if a data set contains $n$-grams that have never occurred in the training data. To address this issue, we apply Good-Turing discounting and Katz backoff smoothing [18] to discount the MLE probability for each observed $n$-grams in the training data and reserve some probability mass for unseen events.

### B. Skipped $n$-gram model

In natural language, words in a sentence may have *long-distance dependencies*. For example, the sentence "I hit the tennis ball" has three word level tri-grams: "I hit the", "hit the tennis" and "the tennis ball". However, it is clear that an equally important tri-gram implied by this sentence, "hit the ball", is not normally captured because of the separator "tennis". If we skip the word "tennis", we can form this important trigram.

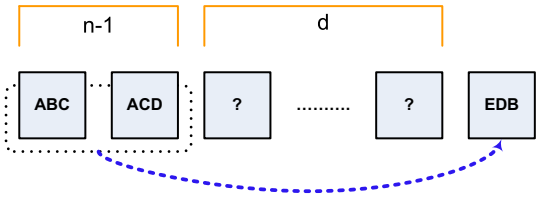


Fig. 1. Skipped $n$-gram to model long-term dependent mobility behaviors

Similarly, our continuous $n$-gram model assumes that a user's next action is dependent solely on his/her previous $n-1$ actions. However, in many cases one's future mobility behavior depends on behaviors that happened a while ago, while the intermediate behaviors have little relevance or influence on the present and future behaviors. For example, knowing that a user is leaving the break room and entering the hallway which leads to his office, we can predict that he will be in his office soon. In other words, his intermediate actions along the hallway and his actions right before entering the office are not that important once we know that he is leaving the break room.

To model such long-distance dependencies, we trained a skipped $n$-gram model. A skipped $n$-gram is a pair $(L, l)$ extracted from the behavior text. $L$ is a label sequence of $n-1$ labels and $l$ is the label that after skipping $d$ labels after $L$ (Figure. 1). Usually $L$ and $l$ have strong correlations. In other words, the occurrence of $L$ *triggers* the occurrence of $l$ in the future.

By skipping location entries that are detractors or non-contributors, the *skipped* $n$-gram model becomes more robust to noise in data which is caused by either interference or "ping-pong" phenomenon as described in [8]. Furthermore, skipping detracting n-grams reduces the size of the model in terms of computational time and storage size (as the model has comparable performance for a lower value of n than when the model takes detracting grams into account).

Another benefit of using $n$-gram model is that we do not need to store mobility traces of the users. Instead, we only construct sufficient statistics ($n$-gram models) of user mobility behaviors. The behavioral text representation, which is generated from mobility traces, contains sufficient information to model user's mobility behavior. In the mean time, given a behavioral text string, it is not possible to reconstruct this user's mobility trace, therefore well protecting user's privacy.

## IV. Data Collection and Preprocessing

We adopted *syslog* [7] approach to collect data for our experiments. Unlike [7], in which only the associated WAPs is recorded, we collect RSS of the beacons for mobile devices on multiple WAPs. Because all these WAPs' internal clocks are synchronized, it allows us to aggregate traces from different WAPs and serialize them on a single time line. This approach allows us to infer mobile devices' location in finer granularity, in addition to their associated WAP. In our study, we constructed a series of RSS vector $\vec{s_t} = (s_{1,t}, s_{2,t}, \ldots, s_{N,t})$, where $s_{k,t}$ is the RSS of mobile device's beacon detected by the $k$th WAP at time $t$. We also used two types of information to model a user's mobility behavior: (pseudo) location and time. We adopted the fingerprinting approach similar to [12] to assign a pseudo location label for each segment of RSS readings to represent device's current location. Compared with [7], our approach removes the overheads of estimating the actual mobility path from the associated WAPs path and the pause time at each location.

### A. Pseudo location label generation

We assume that the infrastructure in the building is relatively stable, i.e., positions of WAPs, devices used as WAPs and radio environment are not changed over time, which usually holds true for office buildings. Therefore, RSS vectors with high similarities will indicate the device is within proximity. We used k-means clustering algorithm with a distance function inspired by Redpin [2] and WASP system [9] to convert the trace for each mobile user into a series of pseudo location labels, where each label represents a class from the clustering resultd. Our experiments showed that choosing $3N$ as the number of clusters, where $N$ is the RSS vector dimension, produces the best quality/performance trade-offs.

Using the algorithm described above, we convert the trace for each mobile user $c$ into a series of pseudo location labels $L_c = \{l_0, l_1, l_2, ...l_t\}$.

### B. Collapsing recurring pseudo location labels

Due to the recurring nature of the data collection process, if a user stays at a pseudo location for a long period of time, there will be sequences of repeating labels in the data. $n$-gram model would fail to model the transition of actions and the probability mass will be dominated by such most visited places.

To mitigate this effect, we collapsed repeating pseudo location labels into a set of new pseudo location labels. Time spent at a location can be user-dependent. For example, a faculty member may stay in his office for a long time, while a student visiting him may spend less time at the same location. The process described above may result in the loss of this information. To overcome this, we constructed another *Duration* feature by counting the repeating pseudo location labels before collapsing.

### C. Sequence Segmentation and Time-of-day label

Users' behavior follows certain regularities due to regular meetings and routines associated with their responsibilities and roles as shown in Figure 2. For example, a group of students who work on the same course project may get together in a conference room or a common area after the class; faculty members may have quick group meetings at certain time of the day to get status reports from the students.

In our raw data set, every RSS vector is marked with a time stamp (epoch). It provides extra information to assist the modeling of user behaviors especially with routine and fixed schedules. We partitioned the duration of 24 hours into multiple non-equal-size sections and used the tags $NA, AM, PM, EV$ as the time-of-day (TOD) labels. While not the focus of this

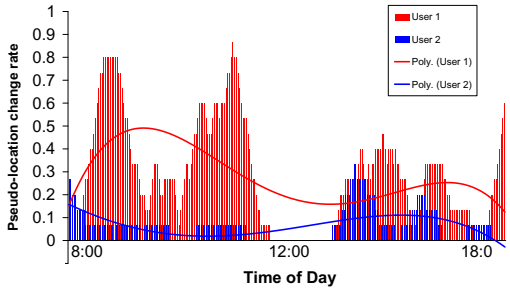


Fig. 2. Regular patterns from user's daily mobility behaviors

study, we found that the universal division of the duration may cause problems if the users have shifted working hours by which a series of associated mobility behaviors are right across the section boundary. We proposed Activity-Level Adjusted Segmentation(ALAS) scheme where we adjusted the section boundary based on a given user's mobility behavior. The rationale is to cut the section boundary so that the continuity of user mobility behavior is maximized. This approach will be discussed in details in future work.

*1) Multi-label fusion based on Pair-Wise Mutual Information :* As described in Section IV-B and IV-C, we have multi label streams in addition to pseudo location labels from the RSS trace. Blindly concatenating the labels for different features together will either inadequately reduce data resolution or necessarily increase the model complexity if the sampling rates are very different and the granularity of the data source is not comparable.

Therefore, we explored heterogenous sensor fusion techniques to combine these different streams of labels into a single stream of behavior text before fit it to our $n$-gram model. We use Mutual Information, a simple metric, to determine whether time-at-location features should be combined with pseudo location feature. This approach is to include duration label only if it is highly correlated with the location label, which in turn to capture the high level activities associated with this location/time pair.

The change rate of the Time-of-Day label is small compared to the other two features. Therefore, we decided to concatenate them with the fused label in this study.

This multi-label fusion process allows us to add more features, such as the type of applications running on the mobile device, their network resource consumption and battery information, into behavior text streams without losing their importance in the $n$-gram modeling. We will address this problem in full details in a separate work.

## V. Experiments and Results

### A. Data Set

We obtained series of pseudo location labels of 40 users, who were selected from the traces of 587 users collected from 87 WAPs over 5 days. If two users' mobility areas are entirely separable, it is relatively easy to identify the users by their pseudo location label histogram. As shown in Figure 3, user 2 and user 4 are completely partitioned into two separate areas, while user 1 and user 3 have some overlap. We calculated the cross entropy of those labels among all these users. To test our $n$-gram model's strong capability in anomaly detection, we choose 10 users that have the least cross entropy. This ensures that users' moving paths strongly overlap and the location label histogram can not be used to efficiently identify users.

### B. Anomaly Detection

*1) Simulated testing data through splicing:* We used 3/4 of the samples as the training set and the remaining 1/4 data to create the testing data. Since there were no real device stolen events recorded in the data we collected, to make the experiments more generalizable, we created simulated device stolen events by *splicing* two users' trace segments at their intersection point. This simulated device-stolen trace has the real moving patterns from both the original owner and the "unauthorized user" and those patterns intersect at the *event time* as show in Figure 4. We combined these simulated device-stolen samples and the normal samples to create a testing set.

*2) Anomaly detection and threshold:* In the testing phase, we adopted a sliding window scheme. As the sliding window advances, we continuously calculate the likelihood (average log probability as shown in Equation 2) of the behavior text in the sliding window that is generated by the $n$-gram model of the original user. We will trigger an anomaly detection alert when the log probability drops below a predefined threshold. When such situation is detected, we record the location of the sliding window and the "response time". In other words, this estimates how much time it takes for the system to detect the mobile device has been stolen.

As shown in Figure 5, at first the log probability of the testing segment is high because the data matches the user's behavior model. As the sliding window moves closer to the event point $A$, the log probability drops and eventually goes completely below the threshold 2, where an alarm is triggered. We performed these experiments for all 10 users with different training data sizes (by means of hours $L$), different history lengths ($n$) and different detection thresholds. Among the users with high cross entropy in pseudo location labels, $n$-gram model correctly detected $86\%$ of simulated device-stolen events with $n = 10$ and just 12 hours of training data.

About $10\%$ of the non-stolen testing samples trigged the

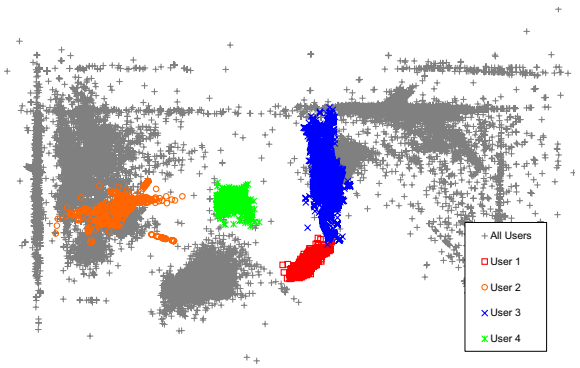


Fig. 3. Pseduo-location density from 4 selected users vs all users

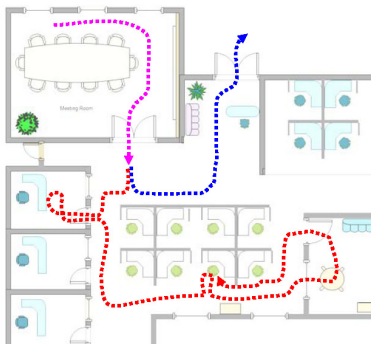

Fig. 4. Two users' mobility patterns are concatenated to create a sample for a simulated stolen event. Originally pink and red are user A's trace. blue is user B's trace. We splice pink with the blue to simulate a device stolen event.

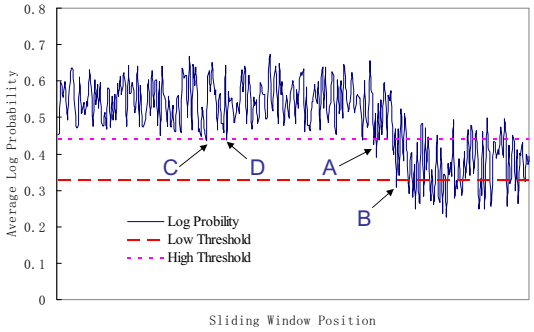


Fig. 5. Anomaly detection experiment results for a user with different thresholds. The stolen event happens at $A$. $C$ and $D$ are false positives for threshold 0.42. $B$ is the true positive for threshold 0.37. The response time is the distance between $B$ and $A$.

alerts as false positive cases. These results were obtained using 0.40 (average log probability) as the detection threshold and 10 labels as maximum allowable response time. If an event is undetected at 10th label after the event happens, we consider this testing sample undetected, and classify it as false negative.

Figure 6 shows Receiver Operating Characteristic (ROC) curves for anomaly detection results. Each points on the ROC curve corresponds to a certain detection threshold. The upper left corner represents perfect detection, while diagonal represents the model that is performing no better than a coin-flip. Figure 6 also illustrates the trade-offs on detection threshold and data size.

*3) Order of $n$-gram and training data size:* We studied the impact of history length $n$ on the detection accuracy. A larger $n$ captures more context in the $n$-gram model which increases the accuracy. However, it also increases the size of the model. Figure 7 shows that anomaly detection performance would saturate when $n > 5$. This suggests that in majority of the cases, user's mobility behavior only depends on the most recent 5 pseudo locations. Figure 7 also shows the effect on the training size. If the training size is as small as 4 hours, it may not capture user's mobility behavior, yielding poor accuracy. The closeness of the two curves with training size 8 and 12 hours also suggests that the system only needs to monitor the

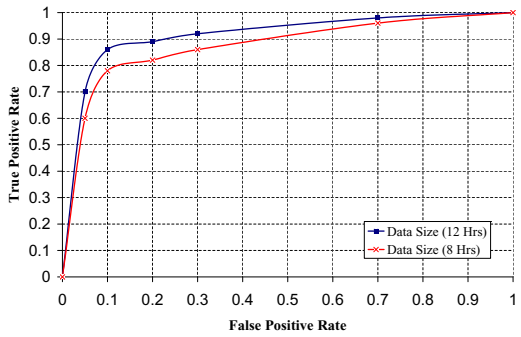


Fig. 6. Receiver Operating Characteristic (ROC) curves of the $n$-gram model for training size of 4 and 12 hours.

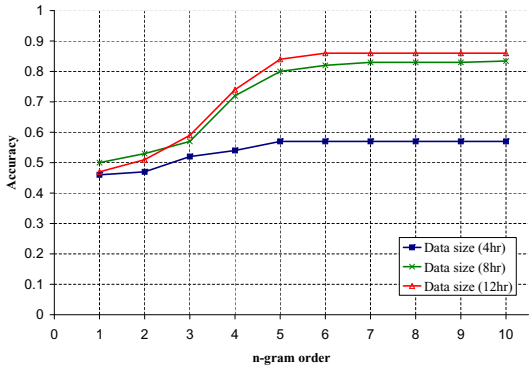


Fig. 7. Variation of anomaly detection accuracy vs $n$-gram order. Threshold is set to .59

user's behavior for less than one day in order to build up a reliable mobility behavior model for anomaly detection.

### C. Future Location Prediction

In addition to accountability, mobile behavior model can also be used to predict the future activity of the user based on his/her past and current behaviors. Predicting future activity can be used to optimize resource allocations, such as spectrum and bandwidth, to improve quality of experience of multimedia services including VoIP and video streaming, and to preserve user privacy and maintain network security.

To demonstrate $n$-gram model's predictive capabilities, we divided each user's trace into 4 equal-length consecutive segments. We used the 1st two segments to train the $n$-gram model and tried to predict the user's behavior for the remaining 2 segments. In our study, we empirically set $n = 3, d = 7$. The results are shown in Table I.

We found that the prediction accuracy dropped significantly for some users, specifically, user 4 and user 9 in Table I. Further investigation revealed that our segmentation scheme to divide the training and testing data might have cut behavior changing boundary. As a result, their mobility patterns are different between the training and testing set. We swapped 1

| UID | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| (%) | 74.1 | 60.5 | 63.3 | 38.4 | 72.3 | 64.9 | 61.7 | 65.0 | 32.8 | 70.2 |

TABLE I
PREDICTION ACCURACY FOR A SELECTED SET OF USERS.

segments between testing and training. With this adjustment, we archived 63.1% and 65.7% accuracy for user 4 and user 9 respectively similar to other users.

## VI. Conclusion and Future work

In this paper, we present a novel method for modeling user mobility behaviors in WLAN networks using $n$-gram models. We also present a set of methodologies to construct and select features for analysis using NLP tools and techniques.

We collected RSS readings of the beacons emitted from mobile devices. We aggregated the data from multiple WAPs to form sequences of fingerprints. These sequences were used to infer the actual locations of the devices in a fine-grain scale. This approach reduces the overhead to estimate mobility tracks and pause time using more complex models or ad-hoc heuristics. We converted these heterogenous sensory data into behaviorial text and adopted a *skipped* $n$-gram model to capture the long-term dependency in user mobility behavior. In addition to single pseudo location information, we leveraged simple heterogenous data fusion techniques to incorporate other context information, such as time-of-day, duration to mobility behavior text representation.

We applied our model to mobile anomaly detection and future behavior prediction. Through experimenting with data set captured from production networks, we have shown that this simple $n$-gram approach can capture the personal mobility patterns. Since our data set did not contain any samples from a device-stolen event, we tested our model in anomaly detection by including a series of simulated device-stolen scenarios. The results showed that the model can detect such an anomaly by examining the log probability calculated with $n$-gram model. We also analyzed how certain parameters would affect the performance of the algorithm in term of scale of the model and accuracy of detections.

Inspired by the success of application of our $n$-gram language model to mobility behavior modeling, in the next step, we plan to investigate how Factored Language Model(FLM) would improve our user behavior modeling. By using heterogeneous sensor information, a behavior text could be treated as a collection of K parallel factors. We will choose the factors based on various data sources, including traffic pattern, GPS, accelerometers and gyroscope readings, device memory and battery status, application status and user interactions, to improve the performance of the user behavior modeling. We also hope this would simplify the process so that it's easier to aggregate multiple users' behavior and to discover routines shared by a group people, such as meetings, parties and other group activities.

## VII. Acknowledgements

This research was supported in part by the CyLab Mobility Research Center at Carnegie Mellon under grants DAAD19-02-1-0389 and W911NF0910273 from the Army Research Office and by the Cisco Research Award. The authors would also like to thank Ivy Liu and Santosh Pandey from Cisco Systems Inc. for their valuable comments and suggestions.

## References

[1] R. Aipperspach, E. Cohen, and J. Canny. Modeling human behavior from simple sensors in the home. In *Proceedings of IEEE Conf. on Pervasive Computing*, pages 337–348, Dublin, Ireland, April 2006.

[2] P. Bolliger. Redpin - adaptive, zero-configuration indoor localization through user collaboration. In *MELT '08: Proceedings of the first ACM international workshop on Mobile entity localization and tracking in GPS-less environments*, pages 55–60, New York, NY, USA, 2008. ACM.

[3] S. Buthpitiya, Y. Zhang, A. Dey, and M. Griss. $n$-gram geo-trace modeling. In *Proceedings of Ninth International Conference on Pervasive Computing*, San Francisco, CA, June 12-15 2011.

[4] S. Chennuru, P.-w. Chen, J. Zhu, and Y. Zhang. Mobile Lifelogger - recording , indexing , and understanding a mobile user s life. *MobiCase*, (September 2009), 2010.

[5] L. Francis, K. Mayes, G. Hancke, and K. Markantonakis. A location based security framework for authenticating mobile phones. In *Proceedings of the 2nd International Workshop on Middleware for Pervasive Mobile and Embedded Computing*, M-MPAC '10, pages 5:1–5:8, New York, NY, USA, 2010.

[6] H. W. Gellersen, A. Schmidt, and M. Beigl. Multi-sensor context-awareness in mobile devices and smart artifacts. *Mob. Netw. Appl.*, 7:341–351, October 2002.

[7] M. Kim, D. Kotz, and S. Kim. Extracting a mobility model from real user traces. In *Proceedings of the 25th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, Barcelona, Spain, April 2006.

[8] J.-K. Lee and J. C. Hou. Modeling steady-state and transient behaviors of user mobility: formulation, analysis, and application. In *Proceedings of the 7th ACM international symposium on Mobile ad hoc networking and computing*, MobiHoc '06, pages 85–96, New York, NY, USA, 2006. ACM.

[9] H. Lin, Y. Zhang, M. Griss, and I. Landa. Wasp: an enhanced indoor locationing algorithm for a congested wi-fi environment. In *Proceedings of the 2nd international conference on Mobile entity localization and tracking in GPS-less environments*, MELT'09, pages 183–196, Berlin, Heidelberg, 2009.

[10] G. Liu and G. Maguire, Jr. A class of mobile motion prediction algorithms for wireless mobile computing and communication. *Mobile Networks and Applications*, 1(2):113–121, 1996.

[11] U. Maurer, A. Smailagic, D. P. Siewiorek, and M. Deisher. Activity recognition and monitoring using multiple sensors on different body positions. In *Proceedings of the International Workshop on Wearable and Implantable Body Sensor Networks*, pages 113–116, Washington, DC, USA, 2006.

[12] E. Mok and G. Retscher. Location determination using wifi fingerprinting versus wifi trilateration. *J. Locat. Based Serv.*, 1:145–159, June 2007.

[13] M. Shin, A. Mishra, and W. A. Arbaugh. Improving the latency of 802.11 hand-offs using neighbor graphs. In *Proceedings of the 2nd international conference on Mobile systems, applications, and services*, MobiSys '04, pages 70–83, New York, NY, USA, 2004. ACM.

[14] L. Song, U. Deshpande, U. C. Kozat, D. Kotz, and R. Jain. Predictability of wlan mobility and its effects on bandwidth provisioning. In *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, pages 1 –13, April 2006.

[15] E. M. Tapia, S. S. Intille, W. Haskell, K. Larson, J. Wright, A. King, and R. Friedman. Real-time recognition of physical activities and their intensities using wireless accelerometers and a heart rate monitor. In *Proceedings of the 2007 11th IEEE International Symposium on Wearable Computers*, pages 1–4, Washington, DC, USA, 2007.

[16] L. Vu, K. Nahrstedt, S. Retika, and I. Gupta. Joint bluetooth/wifi scanning framework for characterizing and leveraging people movement in university campus. In *Proceedings of the 13th ACM international conference on Modeling, analysis, and simulation of wireless and mobile systems*, New York, NY, USA, 2010.

[17] G. Yan, J. Lin, D. B. Rawat, and W. Yang. A geographic location-based security mechanism for intelligent vehicular networks. In R. Chen, editor, *Intelligent Computing and Information Science*, volume 135 of *Communications in Computer and Information Science*, pages 693–698. Springer Berlin Heidelberg, 2011.

[18] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to information retrieval. *ACM Transactions on Information and System Security*, 22(2):179–214, 2004.