Wi-Fi-Based Indoor Positioning Using Human-Centric Collaborative Feedback

Yan Luo*, Yuanzhu Peter Chen†, Orland Hoeber‡
Department of Computer Science
Memorial University
St. John's, NL, A1B 3X5, Canada
Email:{*yanl, †yzchen, ‡hoeber}@mun.ca

***Abstract*—In recent years, "folksonomy"-like systems such as Wikipedia and Delicious Social Bookmarking have achieved huge successes. User collaboration is the defining characteristic of such systems. For indoor positioning mechanisms, we argue that it is also possible to incorporate collaboration in order to improve system performance, especially for fingerprinting based approaches. In this paper, we propose a robust and efficient model for integrating human-centric collaborative feedback within a baseline Wi-Fi fingerprinting-based indoor positioning system. Experimental results show that the baseline system performance (i.e., positioning error) is improved by collecting both positive and negative feedback from users. Moreover, the feedback model is robust with respect to malicious feedback, quickly self-correcting based on subsequent helpful feedback from users.**

I. Introduction

Mobility and location-awareness are two essential characteristics of mobile services. After over a decade of research and development, location-aware services have gradually penetrated into real life. They assist human activities in a wide range of applications, from productivity and goal fulfillment to social networking and entertainment. Traditionally, location-aware applications have been confined to outdoor environments, mostly using GPS. Relatively less research has explored the potential applicability of similar services for indoor settings. However, in large indoor environments such as airports, libraries, or shopping centres, location awareness can increase the quality of service provided by these facilities. The lack of research and development on the indoor aspect of this problem is a result of two technical challenges. First, GPS signals cannot reach indoor receivers. Second and more importantly, the complexity of indoor environments makes triangulation based approaches (i.e., the approach used for GPS) much less effective. On the other hand, such challenges provide us great opportunities for innovative indoor positioning techniques. For example, some early indoor positioning technologies used infrared, laser, and/or ultrasonic range finders, yielding fairly good system performance in field tests [1]. The disadvantages of such an approach are its size, complexity, and cost, which render it infeasible for mobile devices.

A number of researchers have been working on using Wi-Fi infrastructure for indoor positioning even though it was not specifically designed for this purpose [2]. Due to the infeasibility of indoor triangulation, most of these systems use a fingerprinting approach based on the Received Signal Strength (RSS) transmitted by nearby Wi-Fi access points [2]. Typically, such an approach consists of a training phase and a positioning phase. In the training phase, each survey position is characterized by location-related Wi-Fi RSS properties called *Wi-Fi RSS fingerprints* [3]. During the positioning phase, the position likelihood is calculated based on the current Wi-Fi RSS measurement. That is, the system estimates the position by comparing the current measurement to the fingerprints in the system to generate the most likely match. Here, fine-grained system training is normally required to achieve high accuracy and resolution.

Due to the large signal variance, changing infrastructure, or insufficient measurement data, such an approach may deliver inaccurate and unreliable results. In these circumstances, adding a compensation mechanism to modify the results can improve the robustness. For some types of mobile devices carried by people, such compensation can come in the form of feedback from end users [4].

In this article, we propose a Wi-Fi based indoor positioning system that includes an integrated human-centric collaborative feedback model. In the proposed prototype, we define an efficient and robust user feedback model, where the initial likelihood vector calculated by the positioning system will be compensated before being presented to the user. Further, the user can participate in how the compensation works by providing feedback.

The rest of this article is organized as follows. The next section gives a short overview of Wi-Fi RSS fingerprinting based positioning. In Section III, we describe a baseline Wi-Fi fingerprinting framework. Next, we define our detailed user feedback model in Section IV. The feedback model is tested in comparison to the baseline method as reported in Section V. This article is concluded in Section VI with discussion and an overview of future work.

II. Related Work

During the training phase of Wi-Fi fingerprinting, beacon frames are collected from nearby access points at each survey position. The MAC address, RSS, and timestamp are extracted from each beacon. The collection of beacons in a single scan by the device form a vector. Typically, multiple scans are conducted to constitute a fingerprint for this survey position. As such, a fingerprint database is created for future queries in the positioning phase. However, this collection process can be very time-consuming and laborious, especially for future updates and maintenance. Thus, streamlining such a training phase is very important for its commercialization. According to Chai and Yang [5] and Lemelson et al. [6], pre-processing fingerprints can significantly reduce the system training costs. At the extreme, "zero-configuration" can be achieved by only involving user updates without system training [7]. Furthermore, commercial Wi-Fi infrastructure is usually deployed with a large number of relatively dense access points. It may seem that a higher positioning accuracy can always be achieved if more access points are utilized. However, this is not the case as indicated by Kaemarungsi and Krishnamurthy [8]. Instead, a subset of access points can be used for the same level of system performance at a much reduced overhead. A straightforward approach would be to select the subset of access points with the highest observed RSS. More intelligently, Chen et al. [9] provide a novel selection strategy based on the discriminant power of each access point using an information gain criterion. As a result, the access points that best differentiate the survey positions are selected for positioning services.

Calculation of the distances between an RSS observation and the fingerprints stored in a database is the essence of fingerprint-based techniques. Multiple methods exist to calculate such a distance (e.g., Euclidean distance and cosine similarity). The returned result is either the survey point with the smallest distance to the observation (i.e., nearest neighbor (NN) classification) or a representative of k closest survey points (i.e., k-nearest neighbor (KNN) classification) [2]. Another flavour of Wi-Fi positioning methods rely on probabilistic techniques like Bayesian Networks or Gaussian kernels to handle the uncertainty in RSS measurements [10], [11], [12], [13]. Here, positions are estimated using posterior probability density functions.

The accuracy of Wi-Fi fingerprinting thus designed is highly dependent on the number of survey positions employed during the training phase. This not only implies a high system overhead and training cost, but also be vulnerable to environmental changes. Indeed, maintaining such a system would require retraining the system almost from scratch. On the other hand, if the system is augmented with learning or compensation capabilities, it will be able to update its own knowledge. Since these systems may provide services to many mobile users, such a learning capability can be obtained via user feedback for free during the positioning phase. *Active Campus* [14] is an early system integrating user feedback. It allows users to incrementally update the training data for future usage. Similarly, *Redpin* [7] used a "folksonomy"-like approach, where many users train the system while using it. A potential pitfall in this approach is that the model constructed during the training phase could also be adversatively affected by unreliable or misleading user feedback. Thus, it is crucial that the feedback from users should be given proper weights or credibility, rather than blind acceptance or rejection. This idea was first proposed in Hossain et al. [15], where a simple credibility rating of positive user feedback is incorporated. That is, when the user does not believe in the position returned by the system, an alternate position can be suggested. In their system, positive user feedback is given a higher credibility weight if the suggested position has a small discrepancy with the system.

In this work, we devise a more general framework using a wider variety of user feedback. Such a framework is endowed with a high degree of system robustness when a large number of users provide correct feedback. Even when incorrect feedback is provided, the system is able to quickly recover by incorporating subsequent corrective feedback.

III. Position Estimation Baseline

We start by first introducing a baseline Wi-Fi fingerprinting system. The implementation of this baseline system is similar in many respects to the systems reviewed in Section II. However, it is also refined to be more robust and suitable for integrating and processing user feedback. A summary of notation used in this article is listed in Table I.

TABLE I
List of notation

Symbol	Meaning
C_i	Number of occurrences of access point i
σ_i	RSS variance of access point i
F	Wi-Fi RSS fingerprint
σ_F	Variance of the Wi-Fi RSS fingerprint
A_s	System anchor
p_s	The physical coordinates of system anchor
F_s	Wi-Fi RSS fingerprint of system anchor
$d(i,j)$	Euclidean distance from anchor point i to j
L_i	Likelihood of anchor i
L	Likelihood vector
A_u	User anchor
p_u	The physical coordinates of user anchor
F_u	Wi-Fi RSS fingerprint of user anchor
α	Positive user feedback compensating factor, $\alpha = \frac{1}{a+e^{-x}}$
x	Independent variable accumulating via similar user feedback
T	Current Wi-Fi scan times
T_s	Wi-Fi scan times used in system training phase
Δx	Increment derives from a similar user feedback
a	Parameter adjusting the initial and maximum value of α
b	Parameter adjusting the increasing velocity of x
β	Negative user feedback compensating factor, $\beta = e^{-x}$
L'_i	Compensated likelihood of anchor i

A. Training Phase

In the training phase, a set of grid points within the study area were selected as survey positions. The training is conducted for each survey point in a two-step process.

1) Collect raw Wi-Fi measurement data: For each survey position, a trainer uses a mobile device to scan for the beacons transmitted by nearby Wi-Fi access points. Each scan is recorded as a list of 2-tuples, where a 2-tuple element is a pair of the MAC address of the access point and its RSS in decibels. Since the RSS can fluctuate drastically, the trainer usually conducts multiple scans in order to collect sufficient data. As a result, in a given period of sampling, the device logs a time series of these data.

2) Generate Wi-Fi RSS fingerprint: We next extract some statistics from the raw Wi-Fi measurement data to generate an RSS fingerprint for each survey position. We define a Wi-Fi RSS *fingerprint* as a vector of 5-tuples (i.e., *MAC*, *RSS Mean*, *RSS Variance*, *Timestamp*, and *Count*), describing a set of access points. The *MAC* field contains the MAC address of the access point, denoted herein as i. *RSS Mean* is an average of the Wi-Fi RSS over the sampling period. The time of the fingerprint creation is stored in the *Timestamp* field. The value of *Count* is the number of occurrences of the access point during the sampling period, denoted C_i, which is a very important indicator for the reliability of this access point. For a fixed number of Wi-Fi scans, a large *Count* value means that the access point can be heard for most of the time, indicating that the access point will have a more reliable estimation of its RSS value. *RSS Variance* contains the variance of the measured RSS from the access point, denoted σ_i.

At the end of the training phase, each survey position is associated with an RSS fingerprint containing the access points that best describe the specific location. For each survey position p_s in the system, we define a *system anchor* A_s as (p_s, F_s), where F_s is its RSS fingerprint. The collective RSS variance for this fingerprint is defined as

$$\sigma_{F_s} = \frac{\sum_{i \in F_s} \sigma_i C_i}{\sum_{i \in F_s} C_i} \tag{1}$$

B. Positioning Phase

1) Calculate likelihood: For experimental purposes, the prototype implementation allows for a variable number of Wi-Fi scans. After each scan, we send the generated RSS vector to the system for position estimation. Each subsequent scan leads to a cumulative estimation result with a decreasing error. For position estimation, we employ the Gaussian kernel [16], which is commonly used to calculate the likelihood between an RSS fingerprint in system anchors and the current RSS measurement. Thus, each system anchor will then have a likelihood for being the estimated position of the device. We use the top-k anchors to determine the user's position, where $k = 4$ by default in our experiments.

2) Present position: A naïve approach is to use the "centre" of the top-k anchors as the estimation for the position. Usually, these k survey points are close to each other in the physical space, and they can be considered as a cluster. Thus, their weighted mean position is a reasonable representative. However, if one or more outliers exist, the weighted mean position could be pulled far away from the cluster formed by other system anchors. As a result, this mean position could be a meaningless point in the physical space.

Instead, we can use an approach to the *vertex* p-centres problem [17] to determine the representative of the top-k anchors. In particular, the vertex 1-centre is the system anchor point that minimizes the maximum distances from itself to the other top-$(k-1)$ anchor points. These distances are weighted with the likelihood estimated as above. For two indices $i, j = 1, 2, \ldots, k$, we minimize the following over all values for i

$$\max_{j \neq i} \frac{d(i,j)}{L_i} \tag{2}$$

The $d(i,j)$ is the Euclidean distance between anchor A_{S_i} and A_{S_j} and L_i is the likelihood of A_{S_i}. The resulting anchor point becomes the estimation for the location.

IV. User Feedback Model

The focus of this work is on providing position information on mobile devices, where users can provide feedback to the positioning service. That is, users can choose to accept or reject the system estimation, or even suggest an alternative location based on their knowledge of the surroundings. In this section, we define a user feedback model in order to improve system performance. It is useful to begin by identifying the three types of user input that can be collected within a human-centric collaborative feedback system:

- *Positive* feedback is generated when a user rejects the estimated position and suggests a location based on his/her knowledge. In such a case, the system can accept the updated information from the users. The result is that the system may create a new anchor from the user's suggestion, called a *user anchor*.
- *Negative* feedback indicates that the estimated position is not believed by the user, who is unable to make any suggestion as to his/her current location. In this case, the system should reduce the positioning likelihood of the returned location in the future.
- *Null* feedback occurs when a user chooses not to provide any feedback. The assumption here is that the estimated position is accurate, and that there is no need to make any modification to the positioning model.

Assume that the model has N (system and user) anchors, and the likelihood of the i-th ($i = 1, 2, \ldots, N$) anchor is denoted as L_i. Before ranking these anchors based on the estimated likelihood, our user feedback model compensates L_i with two factors, α_i and β_i. These factors are controlled by the history of positive and negative user feedback, respectively. The likelihood of anchor L_i then becomes

$$L_i' = \begin{cases} \beta_i L_i & \text{if } L_i \text{ is system anchor.} \\ \alpha_i \beta_i L_i & \text{if } L_i \text{ is user anchor.} \end{cases} \tag{3}$$

A. Positive User Feedback

Whenever the system receives a user-suggested location associated with its RSS measurement, the system creates a temporary user anchor (A_u). If this anchor is sufficiently similar to an existing user anchor in the model, it is merged with it, and the α factor is updated. Otherwise, it becomes a new user anchor, with the associated α factor set to a small initial value.

1) Temporary user anchor: Since a user's suggested position could be arbitrary, saving these suggestions separately would bloat the model significantly. Therefore, we use discrete locations by dividing the space into an $m \times n$ grid. That is, any position within a grid cell is represented by the centre of

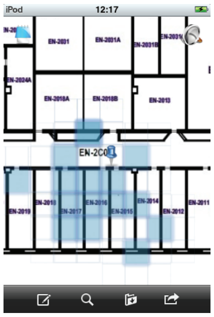

Fig. 1. The user interface allows the user to select grid cells for positive feedback, confirming this choice with a double tap.

the cell. This grid-based selection of the position is enabled directly in the interface provided to the user (see Figure 1).

We define the user anchor A_u as:

$$A_u = (p_u, F_u) \tag{4}$$

The p_u is the cell centre that contains the user suggested position and F_u is the fingerprint summarized from the current Wi-Fi RSS measurement.

2) Anchor merge: A temporary user anchor A_{u_i} is merged with the an existing user anchor A_{u_j} in the same cell if their fingerprints are sufficiently similar. Specifically, we use cosine similarity to compare two fingerprints. If multiple anchors already exist in the same cell as A_{u_i}, we only consider the most similar one, denoted A_{u_j}. If the similarity between A_{u_i} and A_{u_j} is greater than a threshold s, the temporary user anchor is regarded as the same as the existing one.

3) α factor: Whenever a temporary user anchor is merged with an existing user anchor in the system, the associated α factor is updated. For user anchor A_{u_i}, we define α_i as

$$\alpha_i = \frac{1}{a + e^{-x}}, \text{ with } x \geq 0 \text{ and } 0 < a \leq 1 \tag{5}$$

The variable x has a cumulative effect and a is a parameter controlling the initial and maximum values of α_i. As more positive feedback is provided in support of user anchor A_{u_i}, its α factor gradually increases until it reaches an upper limit. Here, the magnification capability of the α factor is $\frac{a+1}{a}$. The increment of x is defined as

$$\Delta x = \frac{\frac{T}{T_s} + e^{-\sigma_F}}{b} \text{ with } b > 0 \tag{6}$$

The pace of the increase of x is controlled by a few aspects. 1) An independent parameter b. 2) The variance of the current RSS fingerprint, σ_F. 3) If T is the number of Wi-Fi scans used in the positioning query and T_s is the number of Wi-Fi scans used during system training, their ratio $\frac{T}{T_s}$ also reflects the credibility of this positive feedback.

As a result, the α factor increases fastest with the first few instances of the user anchor, becoming stable once a sufficient number of feedback events are received. The rationale for this design is to allow the system to quickly adapt to new information provided by the users, but without this feedback overpowering the system.

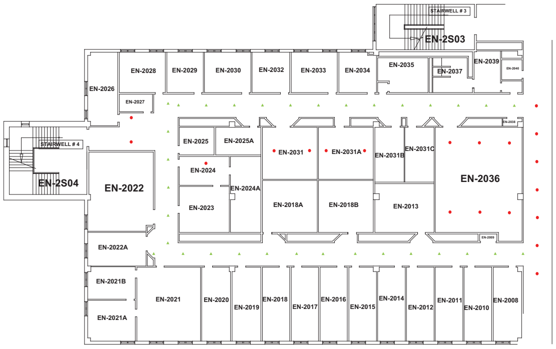

Fig. 2. The experimental field includes both the training cells (green triangles) as well as measurements taken outside of the training area (red discs).

B. Negative User Feedback

Suppose the system delivers a position from top-k anchors according to their likelihood ranking, but the user believes this location to be incorrect and cannot provide any further information regarding the actual location. Negative user feedback on this estimated position can also provide valuable information to the system. Typically, when a user rejects the position estimated by the system, the reason could be that the user is nowhere near any of the anchors known by the system. In this case, none of the top-k anchors would truly represent a good estimate. Therefore, we should try to decrease their likelihoods simultaneously.

Given an anchor A_i, we use a negative user feedback factor β_i to reduce its likelihood according to the accumulation of negative feedback received. Similar to the positive feedback model, the negative factor model also has fast adaptability. Accordingly, we define β_i as

$$\beta_i = e^{-x} \tag{7}$$

When an anchor is given a negative feedback, we give x in above formula the same increment Δx used in the positive user feedback. The value of β is inversely related to x, such that β will decrease from the initial value 1 to its limit zero as x increase from zero to infinity. As a result, if more and more users reject the same set of anchors, they will never be chosen as the top-k due to the small value of the β factor.

V. Evaluation

Experiments and evaluations with this feedback model were conducted in an office setting. The space was divided into a grid using a 3×3m cell size. 33 positions were selected within the hallways for training the baseline system (denoted the training area), and additional 20 positions were selected as untrained positions for testing purposes (denoted the non-training area). A diagram of the setting is provided in Figure 2.

The prototype system was developed for iPhone OS 3.1.2; experiments were conducted using Apple iPhones and iPod Touch devices. The evaluation represents three levels of increasingly complex scenarios, (i.e., no feedback, knowledgable and helpful feedback, and mixed feedback).

As mentioned earlier, the parameters in the feedback model are used to adjust the rate of change of the α and β factors (i.e., the sensitivity of our user feedback model). In production environments, the sensitivity of the user feedback model will depend on the number of users. However, for the purpose of evaluation, extra weight is given to the user feedback information in order to speed up the system evolution.

A. No Feedback

Since the time that a user is willing to spend waiting for a positioning result could affect the service quality, we need to conduct experiments to investigate the relationship between time (e.g., Wi-Fi scans) and system accuracy. We use the baseline system to determine the smallest number of Wi-Fi scans (measured at one scan per second) needed for the system to produce a reasonably accurate result.

In the training area, for each survey point, we collected 20 scans of the Wi-Fi RSS, using these incrementally to query the positioning system. The average positioning error after each scan is plotted as the bottom curve in Figure 3. We can observe that for a small number of scans, the system has an error between 2 and 4m. As more scanned RSS data are used (i.e., greater than four), the accuracy stabilizes at around 2m. Similarly, in the non-training area, we also collected 20 scans for each position. We plotted the positioning accuracy for the number of scans as the top curve in the same figure. In this case, the absolute error is significantly greater than in the training area. However, in either case, four scans provides a reasonable trade-off between accuracy and time. Thus, we use this as the number of Wi-Fi scans in the rest of our experiments.

B. Knowledgable and Helpful Feedback

Next, we investigated how the user feedback model improves the system performance. In this scenario, whenever the system returns a position that does not match the true position of the user, feedback was provided. We modelled the user as being knowledgable and helpful; whenever the position was inaccurate, the user suggested positive feedback 80% of the time, and negative feedback 20% of the time.

Within the training area, we define a *round* as a traversal of all 33 positions. In a round, the user stopped at each survey position to scan the RSS for the nearby access points (using four scans). If the result was correct, the user moved to the next position. Otherwise, the user provided feedback before moving on. The average positioning accuracy after nine such rounds of visiting and testing each position is plotted in Figure 4. In the course of providing this user feedback, the positioning error improved from approximately 2.5m to 1.5m after just four rounds. From there, little change was observed.

Within the non-training area, the experiment following the same procedure, with the data plotted in the same figure. Because there was no training data in these regions, the initial positioning error was rather large. However, after 13 rounds of collecting user feedback, the error decreased from 9m to 2m. As a result, the positioning error in an area that had not been previously trained became comparable to the training area.

At the beginning of the testing within the non-training area, the model contained only system anchors, and therefore could only return the position of a system anchor (i.e., within the training area) to the user. These positions were often far from the true position of the user. As a result of the positive feedback, user anchors were added and the relative weight of these anchors were enhanced by the α factor. Similarly, with the negative feedback, the weight of the system anchors were reduced by the β factor. As a result, the positioning accuracy increased as more user anchors become valid candidate positions. What this means for indoor positioning systems is that the system training and maintenance cost can be reduced significantly by relying on knowledgable and helpful end users working on a partially trained system, eventually achieving the same level of accuracy as a fully trained system.

C. Mixed Feedback

In a real environment, user feedback can be either helpful or malicious. In this experiment, we test the model to determine its ability to recover from incorrect feedback. In particular, we model the user feedback as completely malicious at the beginning and as completely informative thereafter. Such a

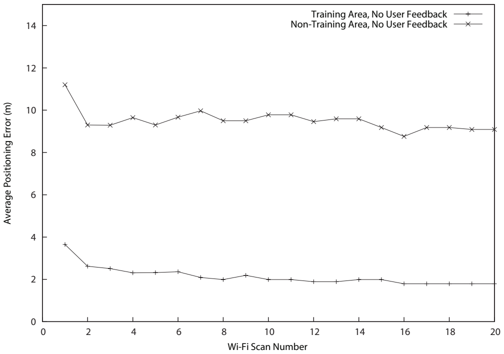

Fig. 3. Using the baseline system, the positioning error becomes relatively stable using just four Wi-Fi scans. Note that the system is significantly more accurate within the training area.

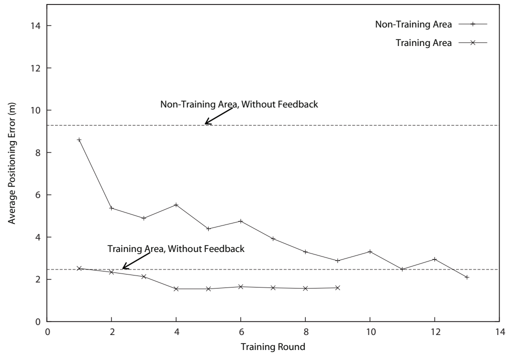

Fig. 4. The system accuracy is improved when integrating knowledgeable and helpful user feedback. With sufficient feedback, the performance in the non-training area approaches that of the training area.

behaviour is not typical but it provides a "worst case scenario" study of the system and its ability to recover from incorrect or malicious feedback.

Our focus here is on the training area only. As seen in the previous experiments, the non-training area can become nearly as good as the training area with sufficient user feedback. As such, we expect similar results within the non-training area as the training area with respect to mixed feedback.

During the initial phase of this experiment, whenever the system returns a correct position estimation, the malicious user has a 50% chance of either providing negative feedback of suggesting a random false position. When the system is incorrect, the malicious user provides null feedback. Following a similar methodology as the previous experiments, such malicious feedback was provided for four rounds. Another eight rounds of feedback from a knowledgeable and helpful user was then collected.

The position errors for this experiment are plotted in Figure 5. We observe that the system error starts out with around 4m and quickly increases to 14m as a result of the malicious feedback. The helpful feedback quickly corrects the significant positioning errors, recovering to the starting accuracy after five rounds of feedback, and below 3m after eight rounds.

VI. Concluding Remarks

In this paper, we presented a model for using human-centric collaborative feedback for Wi-Fi-based indoor positioning. This model allows for the collection and use of both positive feedback (i.e., adding and re-weighting user anchors) and negative feedback (i.e. re-weighting both system and user anchors). Experiments illustrate the ability of the system to improve upon the positioning error in both regions that have been trained, as well as in nearby regions that do not include any training data. The system was also shown to be robust with respect to malicious feedback, quickly recovering based on helpful user feedback.

This feedback model may be extended in a number of interesting ways. For example, the model could take advantage of the timestamp within the RSS fingerprint, limiting the candidate fingerprints to those that were created at approximately the same time of the day. This could increase the accuracy of the system in locations with time-dependant changes in human activity. The introduction of a "forgetting factor" could also be used to address situations where malicious feedback has been received but subsequent helpful feedback is not available.

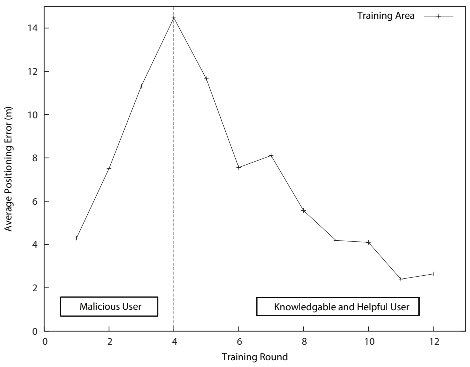

Fig. 5. Providing malicious user feedback, followed by knowledgeable and helpful user feedback illustrates the ability of the model to self-recover.

References

[1] A. Harter, A. Hopper, P. Steggles, A. Ward, and P. Webster, "The anatomy of a context-aware application," *Wireless Networks*, vol. 8, no. 2/3, pp. 187–197, 2002.

[2] P. Bahl and V. N. Padmanabhan, "RADAR: An in-building RF-based user location and tracking system," in *Proceedings of 19th IEEE Conference on Computer Communications*, 2000, pp. 775–784.

[3] K. Kaemarungsi and P. Krishnamurthy, "Properties of indoor received signal strength for WLAN location fingerprinting," in *Proceedings of the 1st Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services*, 2004, pp. 14–23.

[4] H. Lemelson, S. Kopf, T. King, and W. Effelsberg, "Improvements for 802.11-based location fingerprinting systems," in *Proceedings of the 33rd Annual IEEE International Computer Software and Applications Conference*, 2009, pp. 21–28.

[5] X. Chai and Q. Yang, "Reducing the calibration effort for location estimation using unlabeled samples," in *Proceedings of the 3rd IEEE International Conference on Pervasive Computing and Communications*, 2005, pp. 95–104.

[6] H. Lemelson, T. King, and W. Effelsberg, "Pre-processing of fingerprints to improve the positioning accuracy of 802.11-based positioning systems," in *Proceedings of the 1st ACM International Workshop on Mobile Entity Localization and Tracking in GPS-less Environments*, 2008, pp. 73–78.

[7] P. Bolliger, "Redpin-adaptive, zero-configuration indoor localization through user collaboration," in *Proceedings of the 1st ACM International Workshop on Mobile Entity Localization and Tracking in GPS-less Environments*, 2008, pp. 55–60.

[8] K. Kaemarungsi and P. Krishnamurthy, "Modeling of indoor positioning systems based on location fingerprinting," in *Proceedings of 23rd the IEEE Conference on Computer Communications*, 2004, pp. 1012–1022.

[9] Y. Chen, Q. Yang, J. Yin, and X. Chai, "Power-efficient access-point selection for indoor location estimation," *IEEE Transactions on Knowledge and Data Engineering*, vol. 18, no. 7, pp. 877–888, 2006.

[10] A. Haeberlen, E. Flannery, A. M. Ladd, A. Rudys, D. S. Wallach, and L. E. Kavraki, "Practical robust localization over large-scale 802.11 wireless networks," in *Proceedings of the 10th Annual International Conference on Mobile Computing and Networking*, 2004, pp. 70–84.

[11] T. Roos, P. Myllymäki, H. Tirri, P. Misikangas, and J. Sievänen, "A probabilistic approach to WLAN user location estimation," *International Journal of Wireless Information Networks*, vol. 9, no. 3, pp. 155–164, 2002.

[12] T. King, S. Kopf, T. Haenselmann, C. Lubberger, and W. Effelsberg, "COMPASS: A probabilistic indoor positioning system based on 802.11 and digital compasses," in *Proceedings of the 1st International Workshop on Wireless Network Testbeds, Experimental Evaluation and Characterization*, 2006, pp. 34–40.

[13] M. Youssef and A. Agrawala, "The Horus WLAN location determination system," in *Proceedings of the 3rd International Conference on Mobile Systems, Applications, and Services*, 2005, pp. 205–218.

[14] E. S. Bhasker, S. W. Brown, and W. G. Griswold, "Employing user feedback for fast, accurate, low-maintenance geolocationing," in *Proceedings of the Second IEEE International Conference on Pervasive Computing and Communications*, 2004, pp. 111–120.

[15] A. M. Hossain, H. N. Van, and W.-S. Soh, "Utilization of user feedback in indoor positioning system," *Pervasive and Mobile Computing*, vol. 6, no. 4, pp. 467–481, 2010.

[16] A. Kushki, K. N. Plataniotis, and A. N. Venetsanopoulos, "Kernel-based positioning in wireless local area networks," *IEEE Transactions on Mobile Computing*, vol. 6, no. 6, pp. 689–705, 2007.

[17] O. Kariv and S. L. Hakimi, "An algorithmic approach to network location problems. I: The p-Centers," *SIAM Journal on Applied Mathematics*, vol. 37, no. 3, pp. 513–538, 1979.