Improving BLE Distance Estimation and Classification Using TX Power and Machine Learning: A Comparative Analysis

Mimonah Al Qathrady, Ahmed Helmy
Computer and Information Science and Engineering Department
Gainesville, Florida, USA
{mimonah,helmy}@ufl.edu

ABSTRACT

Distance estimation and proximity classification techniques are essential for numerous IoT applications and in providing efficient services in smart cities. Bluetooth Low Energy (BLE) is designed for IoT devices, and its received signal strength indicator (RSSI) has been used in distance and proximity estimation, though they are noisy and unreliable. In this study, we leverage the BLE TX power level in BLE models. We adopt a comparative analysis framework that utilizes our extensive data library of measurements. It considers commonly used state-of-the-art model, in addition to our data-driven proposed approach. The RSSI and TX power are integrated into several parametric models such as *log shadowing* and *Android Beacon library models*, and machine learning models such as *linear regression*, *decision trees*, *random forests* and *neural networks*. Specific mobile apps are developed for the study experiment. We have collected more than 1.8 millions of BLE records between encounters with various distances that range from 0.5 to 22 meters in an indoor environment. Interestingly, considering TX power when estimating the distance reduced the mean errors by up to 46% in parametric models and by up to 35% in machine learning models. Also, the proximity classification accuracy increased by up to 103% and 70% in parametric and machine learning models, respectively. This work is one of the first studies (if not the first) that analyze in depth the TX power variations in improving the distance estimation and classification.

1 INTRODUCTION

Internet of Things (IoT) is the expected architecture of our interaction with the physical world in the smart cities, and billions of dollars are allocated for its solutions [7]. Distance estimation and proximity classification are major building blocks in numerous IoT applications such as crowd monitoring, infection tracing [14], localization, and mapping. BLE is built specifically for IoT [5]. It is an energy efficient version of Bluetooth and overcomes its limitation. BLE advertisements can be sent without establishing a connection, and they are used for proximity and range estimation. As a result, BLE has the potential to become an alternative for indoor localization [3, 8]. Much of the previous BLE models have used RSSI as the primary parameter in their positioning or distance estimation [4, 10, 19]. However, experimental studies show that RSSI is unreliable in determining the distance accurately [1]. On the other hand, BLE's transmission (TX) power level can be included in the advertisement. Consequently, we extensively investigate the gain of integrating TX power level with RSSI into the BLE models. We target two main goals in this paper; first to improve encounter distance estimate using BLE, and second to improve encounter proximity classification using BLE.

First, in terms of distance estimation, this study covers several parametric and machine learning models. Parametric models are *log shadowing*[1], as the more general model that describes the relationship between distance and the RSSI [9], and *Android Beacon library models*[2] that got over 150 million installations [12]. Machine learning models include *linear regression* [6], *decision trees* [17], *random forests* [11] and *neural networks* [15]. *Linear regression* is used to model the relationship between variables. *Decision trees* and *random forests* are not affected by the nonlinear relationship between parameters. *Random forests* are more complicated than *decision tree*, but they are more powerful. *Neural networks* are capable of understanding the patterns and generalizing the result.

The second main goal of this paper is proximity classification. Identifying nearby objects provides valuable contextual knowledge to the smart applications. The distances where the encounters are classified to be proximate depends on the application needs, and it varies from one study to another. For example, it is defined as the distance within three meters for social interaction [16], while 3 feet is known to be the unsafe distance from an infected patient [18], and has to be identified during infection tracing. There are some BLE proximity studies, such as how its parameters affect the detection system [16]. However, the level of comparative analysis in this paper has not been previously presented in any of the BLE proximity studies. In our work, we have leveraged BLE TX power levels in the models and evaluated them under different definitions of proximity classifications that range from encounter within half a meter to encounters within three meters. Overall, our work contributions are summarized as follows:

- We have integrated BLE TX power levels with RSSI in several BLE distance estimation and proximity classification models: parametric and machine learning models.
- We have built a framework that systemically studies and analyzes in-depth incorporating Tx power into the models, and evaluated them under various TX power levels with two states: as regular models that use only RSSI, and when they incorporate the TX power. They are assessed for their distance estimation and proximity classification accuracy.

MSWiM '17, November 21–25, 2017, Miami, FL, USA

ACM ISBN 978-1-4503-5162-1/17/11…$15.00
https://doi.org/10.1145/3127540.3127577

[1] It is also called path-loss model
[2] It is also called Alt Beacon Library

- We have collected over 1.8 million BLE records with known distances from an indoor environment. We developed two specific Android applications for this study: a scanner and an advertiser. The advertiser sends BLE advertisements with various TX power levels: high, medium, low and ultra low. The scanner scans TX power along with RSSI. The distances ranges are up to 22 meters, which is longer than many of the previous studies. The tools and data will be shared with the community to establish reproducibility.
- Significant results have been reached. For instance, the mean distance estimation error is reduced by up to 46% in parametric models and by up to 35% in machine learning models when considering TX power. Also, the proximity classification accuracy has been increased by up to 103% and 70% in parametric and machine learning models respectively.

The analysis results in hybrid models which consist of several other models that produced the best result under each TX power level. The rest of the paper is organized as follows: In section 2, we discuss the comparative analysis framework. The results are presented in section 3. The conclusion and future work are provided in section 4.

2 COMPARATIVE ANALYSIS FRAMEWORK

As in figure 1, our proposed framework consists of four main components: the models, the training and testing data, the evaluation and the adaptive hybrid models. These components are discussed in more details in the rest of this section.

2.1 Models

The investigated models include parametric models and machine learning models.

2.1.1 *Parametric Models.*

They use equations to estimate the distance. We analyze two parametric models: *log shadowing*, and *Android Beacon library models* (old and new versions). For the purpose of this study, we define two kinds of the referenced *RSSI*: $RSSI_1$ and $RSSI_{TX}$. $RSSI_1$ is used with regular models, and it is the average *RSSI* when the distance is one meter between the BLE devices regardless of the advertised TX power. $RSSI_{TX}$ is used with TX power integrated models, and it is the average RSSI when the distance is one meter between the devices, but it depends on the TX power. As a result, we have a vector of $RSSI_{TX}$, a particular value for each TX power.

A. Regular Parametric Models: They use *RSSI* and $RSSI_1$ as the main parameters in their equation. The *log shadowing model (LogR)* uses the following equation:

$$distance = 10^{\frac{RSSI - RSSI_1}{-10n}}, \tag{1}$$

where n is the path-loss exponent, n = 2 in free space [3].

Android Beacon Models use the following formula:

$$distance = A \times (\frac{r}{t})^B + C, \tag{2}$$

where r is the RSSI measured by the device, and t is the referenced RSSI. A, B, and C are constants. We have found two library versions based on the coefficients. The old version *(oldBconR)*, as its coefficients were presented in the previous research [2, 4], and assigns the following values to the coefficient: A=0.89976, B= 7.7095, and C= 0.111. The new version *(newBconR)* is the most recent version of the default coefficients as in the library code [12], where A=0.42093, B=6.9476 and C= 0.54992.

B. TX Power Integrated Parametric Models: We have adjusted the models to consider TX power level. Therefore, the scanner reads TX power in the received BLE advertisement along with RSSI. The TX power *log-shadowing* model *(LogTx)* estimates the distance using the following equation:

$$distance = 10^{\frac{RSSI - RSSI_{TX}}{-10n}}, \tag{3}$$

where $RSSI_{TX}$ depends on the scanned TX power. For example, when TX power=high, the model then uses the value in $RSSI_{TX}$ vector that corresponds to the high TX power. The *Android libraries* are also adjusted in this work to consider the TX power in their equations. The libraries check the received BLE TX power and RSSI to estimate the distance. Then, they use the $RSSI_{TX}$ value that corresponds to the TX power level. The old version *(oldBconTX)* estimate the distance using the following equation:

$$distance = 0.89976 \times (\frac{RSSI}{RSSI_{TX}})^{7.7095} + 0.111. \tag{4}$$

While the new version *(newBconTX)* use the equation:

$$distance = 0.42093 \times (\frac{RSSI}{RSSI_{TX}})^{6.9476} + 0.54992. \tag{5}$$

Note that the *Android Beacon Library* currently does not scan for TX power [12]. However, in this study, we have implemented the TX power integrated model and compared it to the regular version of the library under each TX power level.

2.1.2 *Machine Learning Models.*

We study four machine learning models for distance estimation between BLE devices. The models are the *linear regression (LR), decision trees (DT), random forests (RF) and neural networks (NN).* There are two modes of the machine models: regular and TX power integrated.

A. Regular Machine Learning Model: The models are trained on RSSI regardless of the TX power. As a result, they only use BLE advertisement RSSI to estimate the distance. For example, in *linear regression*, the distance is computed as:

$$distance = A \times RSSI + C, \tag{6}$$

where A and C are defined from the training phase. *Decision trees and random forests* split based on RSSI values, and the *neural networks* will have RSSI as an input layer.

B. TX Power Integrated Machine Learning Models: TX power level is combined in the machine learning distance estimation models using two different ways:

1) RSSI and TX power as input features: the models are trained on RSSI and TX power. For instance, the *linear regression* equation will contain TX power and RSSI variables.

$$distance = A \times TXpower + B \times RSSI + C \tag{7}$$

Also, the *decision trees and random forests* have TX power and RSSI in their decision nodes, while the *neural networks* will include TX power in their input layers. To estimate the distance, both RSSI and TX power are plugged into the trained model.

[3] n=2 provides better estimation with our data than n=1.6 to n=1.8

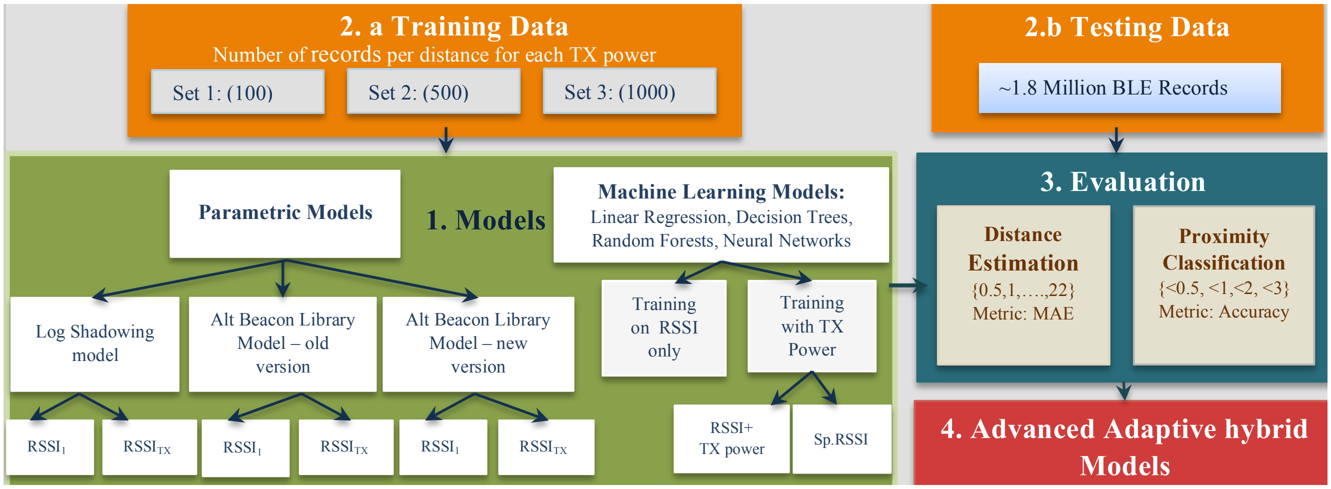


Figure 1: The Comparative analysis framework. It consists of four main components.1. Models: parametric or machine learning, each model is evaluated as a regular model and when TX power is combined with it. 2. Datasets including a. training dataset, and b. the testing dataset. 3. Evaluation: models are evaluated by using a) *MAE* for distance estimation, and b) accuracy of classifying, and 4. Hybrid Models: consist of the best models performed in each TX power.

2) Training specific model for each TX power data (sp.RSSI): the RSSI training data are separated based on their TX power. Then, a particular model is trained for each group using the RSSI only. Therefore, we will have four models from each machine learning model. For example, we will have four *linear regression* models: one model for each TX power level. To estimate the distance, the TX power is checked, and then its corresponding model is used to calculate the distance based on RSSI. Also, there will be four *decision trees and four random forests*: a tree and a forest for each TX power level. Also, we will have *four different neural networks*. For instance, the neural networks that are used when TX power is high will be different than the *neural networks* that will be used when the BLE TX power = medium.

2.2 Collected Data

2.2.1 Data Collection and Experiments. Two specific Android applications are developed to collect BLE records with known distances: an advertiser and a scanner. The advertiser sends the advertisements using different TX power levels. The scanner records the BLE RSSI along with TX power. The experiment is run in an indoor environment, specifically in the CSE department hallway, with no obstacle between the devices, which have been placed at predefined distances from a half to 22 meters {0.5,1,2,....,22}. More than 1.8 million records that are sent from our advertisers are collected.

2.2.2 Training Data. They consist of three sets of data with different sizes. The purpose is to investigate if the training size affects the modeling result. The number of records for each TX power (total 4) at each distance (total 23) is shown in figure 1. There are 9,200, 46,000, and 92,000 records in the 1^{st}, 2^{nd} and 3^{rd} set, respectively. These training sets are used to train the machine learning models. Also, the parametric models parameters are extracted from the training sets such as $RSSI_1$, and $RSSI_TX$.

2.2.3 Testing Data. The testing data are around 1.8 million BLE records. They do not include the data used for training.

2.3 Evaluation

2.3.1 Distance estimation. The mean absolute error (*MAE*) is used to measure how close the estimated distance is to the real one.

2.3.2 Proximity classification. We consider the following distance classification in meters: half or less, one or less, two or less and three or less. The result of distance estimation is used for classification. If the distances set is *Pd={0.5,1,2,3,....,N}*, and the proximate distance of interest is Pd_i, then the estimated distance '*d*' between objects is classified as proximate if:

$$d <= \frac{Pd_i + Pd_{i+1}}{2} \tag{8}$$

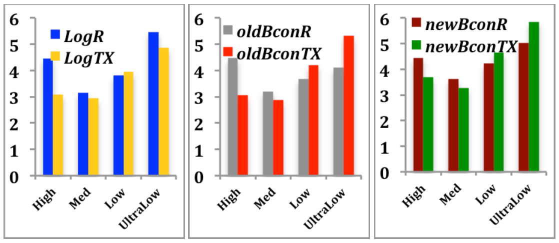
The models' accuracy is their ability to classify the proximate distance accurately, which is given by:

$$Accuracy = \frac{TP + TN}{S}, \tag{9}$$

where *TP* is the true positive cases with the distances less than or equal the proximate distance, and they are classified as *proximate*. *TN* is the true negative cases with distances more than the proximate distance, and they are classified as *not proximate*. *S* is the testing data size.

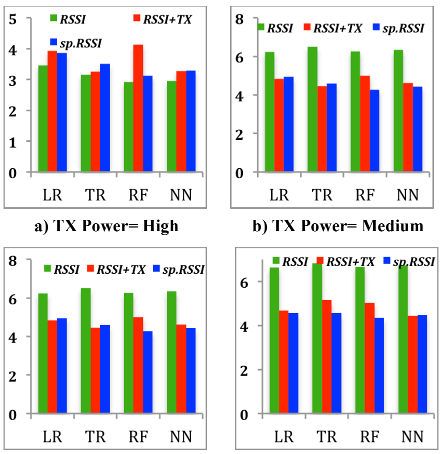
2.4 Advanced Adaptive Hybrid Models

Our comparative analysis resulted in hybrid models that are sensitive to the TX power. One is used when the goal is to reduce the distance estimation error. The other is used when the proximity classification is the primary goal. Details of the hybrid models are omitted for brevity. For details please see [13].



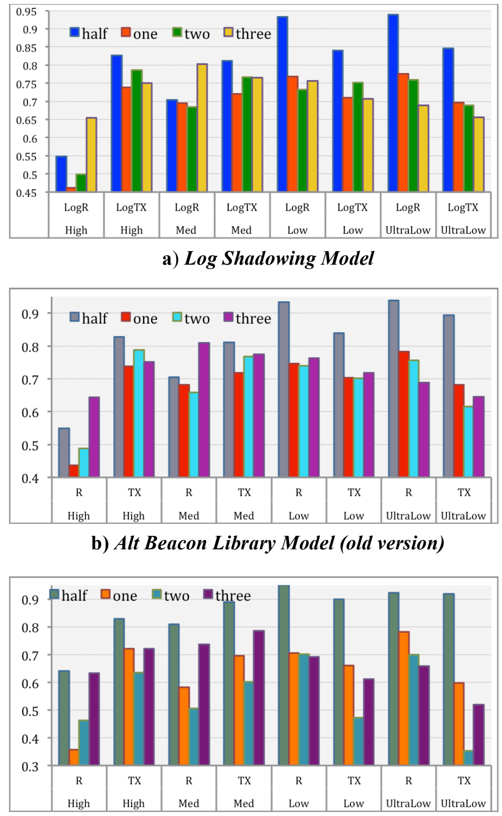
a) Log Shadowing *b) Beacon Library-old* *c) Beacon Library-new*

Figure 2: Distance Estimation *MAE* for Parametric Models.



a) TX Power= High
b) TX Power= Medium
c) TX Power= Low
d) TX Power= Ultra Low

Figure 3: Distance Estimation *MAE* for Machine Learning Models. LR *Linear Regression*, DT: *Decision Trees*, RF: *Random Forests*, NN: *Neural Networks*.



a) *Log Shadowing Model*
b) *Alt Beacon Library Model (old version)*
c) *Alt Beacon Library Model (new version)*

Figure 4: Proximity Classification *Accuracy* for Parametric Models. R: Regular Models, TX: TX Power Integrated Models.

3 COMPARATIVE ANALYSIS RESULT

This section discusses the models' distance estimation *MAE* and classification accuracy.

3.1 Distance Estimation

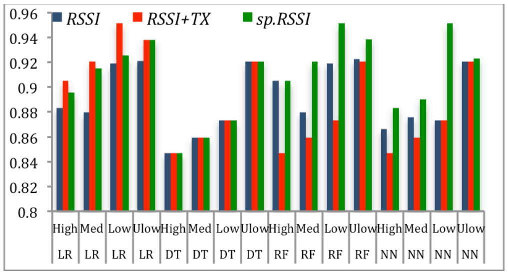
3.1.1 Parametric Models. As shown in figure 2, the models have similar patterns in most cases. TX power integrated model with medium TX power reduced *MAE* by 46%, 35% and 30% in *log shadowing, Android Beacon library (new), Android Beacon library (old)* models, respectively, than using the models' regular versions with ultra low power. Consequently, if an IoT application needs distance estimation, our result suggests using medium TX power with TX power integrated model.

3.1.2 Machine Learning Models. They have similar *MAE* pattern in most cases as in figure 3. Considering TX power improved the models with all TX power levels, with the exception of the high level. For example, with ultra low power, TX power integrated model *MAE* is reduced by up to 35%.
Random Forest that trained on RSSI only produces the least mean error with high TX power. However, parametric models outperform machine learning with other TX power levels. *TX power integrated Android Beacon library*-old is the best with *medium* TX power, while its regular version is the best when TX power is *low or ultra low.*

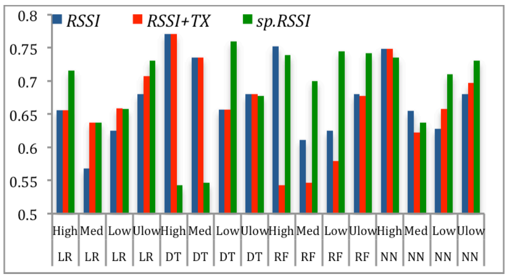
3.2 Proximity Classification

3.2.1 Parametric Models. Low and ultra low TX power levels yield the best result, up to 95% accuracy, when classifying encounters using regular *Android Beacon library* (new) within half a meter as in figure 4. Therefore, if an IoT application requires classifying close encounters, then our result suggests using low or ultra low TX power. Another finding is regular models with high TX power produced unreliable classification. However, integrating TX power in the model improved their accuracy. For example, *Android Beacon library* (new) classifying encounters within one meter improved by 103% accuracy when combining TX power into the same model. The models can classify encounters that are within half a meter accurately, but their accuracy is decreased with other distances classification.

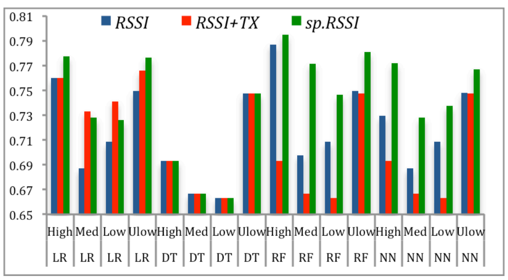
3.2.2 Machine Learning Model. Figure 5 shows their classification accuracy. As the case with parametric models, the encounters within half a meter can be classified with up to 95% accuracy when using low or ultra low power with TX power integrated *linear*



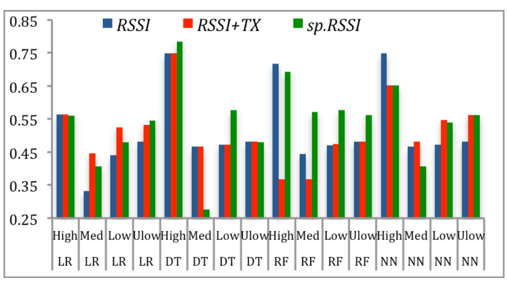
a) Within Half a Meter



b) Within One Meter



c) Within Two Meters



d) Within Three Meters

Figure 5: Proximity Classification *Accuracy* for Machine Learning Models. LR: *Linear Regression*, DT: *Decision Trees*, RF: *Random Forests*, NN: *Neural Networks*.

regression, *random forests* or *neural networks* models. Also, the accuracy decreased when classifying other distances. Integrating TX power improved the accuracy in some cases, and the maximum improvement has been observed when classifying encounters that are within three meters. For example, in *linear regression*, the accuracy is improved by 70% when considering TX power and using high level than using a regular model with medium TX power.

3.3 Training Data Size

The different training sets results have similar pattern in most cases. Though, a slight improvement was noticed with the largest set.

4 CONCLUSION AND FUTURE WORK

This work combines TX power into BLE distance estimation models. Then, extensive comparative analysis between regular and TX power integrated models is conducted. The distance estimations *MAE* is reduced by up to 46%. Also, considering TX power improved the classification accuracy in some cases by up to 103% accuracy. Our study reached significant conclusions; for example, the medium TX power is the best with parametric models when the distance estimation is of interest. However, low and ultra low TX power provide the best accuracy classification of encounters within half a meter. Also, the result of this investigation prompts a hybrid model that is adaptive to the TX power and consists of other models. While our experiment is performed indoor with no obstacle between objects, we expected leveraging TX power would also improve the distance estimation and classification in other indoor environments. We plan to study this in the future.

ACKNOWLEDGMENTS

This work was partially funded by Najran University, Saudi Arabia, and NSF 1320694.

REFERENCES

[1] A.Kwiecien, M.Mackowski, M.Kojder, and M.Manczyk. 2015. Reliability of bluetooth smart technology for indoor localization system. In *International Conference on Computer Networks*. Springer, 444–454.

[2] M. S. Aman, H. Jiang, C. Quint, K. Yelamarthi, and A. Abdelgawad. 2016. Reliability evaluation of iBeacon for micro-localization. In *(UEMCON)*. IEEE, 1–5.

[3] S Bertuletti, A Cereatti, U Della, M Caldara, and M Galizzi. 2016. Indoor distance estimated from Bluetooth Low Energy signal strength: Comparison of regression models.. In *Sensors Applications Symposium (SAS)*. IEEE, 1–5.

[4] H. Cho, J. Ji, Z. Chen, H. Park, and W. Lee. 2015. Accurate Distance Estimation between Things: A Self-correcting Approach. *Open Journal of Internet Of Things (OJIOT)* 1, 2 (2015), 19–27.

[5] Bluetooth Low Energy. *https://www.bluetooth.com/what-is-bluetooth-technology/how-it-works/low-energy*.

[6] Julian J Faraway. 2002. Practical regression and ANOVA using R. (2002).

[7] Here's how the Internet of Things will explode by 2020. *www.businessinsider.com/iot-ecosystem-internet-of-things-forecasts-and-business-opportunities-2016-2*.

[8] Z. Jianyong, L. Haiyong, C. Zili, and L. Zhaohui. 2014. RSSI based Bluetooth low energy indoor positioning. In *(IPIN)*. IEEE, 526–533.

[9] J.Xu, W.Liu, F.Lang, Y.Zhang, C.Wang, et al. 2010. Distance measurement model based on RSSI in WSN. *Wireless Sensor Network* 2, 8 (2010), 606–616.

[10] K.Urano, K. Hiroi, K. Kaji, and N. Kawaguchi. 2016. A Location Estimation Method using BLE Tags Distributed Among Participants of a Large-Scale Exhibition. In *MOBIQUITOUS*. ACM, 124–129.

[11] Andy Liaw and Matthew Wiener. 2002. Classification and regression by randomForest. *R news* 2, 3 (2002), 18–22.

[12] Android Beacon Library. *https://altbeacon.github.io/android-beacon-library/*.

[13] M. Al Qathrady. *https://www.cise.ufl.edu/qathrady/reports/BLEreport.pdf*.

[14] M. A. Qathrady, A. Helmy, and K. Almuzaini. 2016. Infection tracing in smart hospitals. In *(WiMob)*. IEEE, 1–8.

[15] Brian Ripley, William Venables, and Maintainer Brian Ripley. 2016. Package 'nnet'. *R package version* (2016), 3–7.

[16] R.Tabata, A.Hayashi Arisa, S.Tokunaga, S.Saiki, M.Nakamura, and S.Matsumoto. 2016. Implementation and evaluation of BLE proximity detection mechanism for Pass-by Framework.. In *Computer and Information Science*. IEEE/ACIS, 1–6.

[17] Terry M Therneau, Beth Atkinson, Brian Ripley, et al. 2010. rpart: Recursive partitioning. R package version. *R package version* 3 (2010), 1–46.

[18] www.cdc.gov/hicpac/2007IP/2007ip part1.html. *CDC*.

[19] Y. Zhuang, J. Yang, Y. Li, L. Qi, and N. El-Sheimy. 2016. Smartphone-based indoor localization with bluetooth low energy beacons. *Sensors* 16, 5 (2016), 596.