Detecting Social Interactions in Indoor Environments with the Red-HuP Algorithm

Paolo Barsocchi, Antonino Crivello, Michele Girolami, Fabio Mavilia
Italian National Council of Research, ISTI-CNR, Pisa, Italy
Email: {name.surname}@isti.cnr.it

***Abstract*—Detecting social interactions among people represents a challenging task. In this study we evaluate the performance of the ReD-HuP algorithm. We study a real-world and useful experimental dataset and we provide a comparison with some classification methods. Interactions are inferred from co-location of people by exploiting Bluetooth Low Energy (BLE) beacons. Our analysis investigates how the different transmission powers affect the overall performance, we also analyze the results by varying the width of the time window used to analyze BLE beacons. Results obtained with the ReD-HuP algorithm have been compared against two well known and wide adopted machine learning classification methods.**

***Keywords*-Social Interactions, Bluetooth Low Energy, Proximity**

I. Introduction

The possibility of monitoring interactions among people is increased with the adoption of unobtrusive sensing units able to sense the environment. With the term *interaction*, we refer to the all-human tendency of establishing social ties with others. The nature of a tie varies according to the multiple factors, e.g. friends, colleagues or couples are all engaged in different kinds of ties. Moreover, the strength of a tie is, commonly, determined by a combination of several markers, as reported in [1]. Among them, we cite the duration, the intimacy and the emotional intensity, all of them determine the strength of such interaction.

Being able to detect and to study the evolution of the social interactions among people is a challenging task. The typical tools adopted for such task include questionnaires, diaries or interviews that subjects monitored are asked to fill. The information collected, in turn, are analyzed by experts in order to detect and to study several features of the interactions. Such tools provide an essential source of information, but we consider that the recent sensing technologies can extend the accuracy and the acceptability in this research domain. In particular, we experienced with the use BLE beacons in order to detect interactions among people. We rely on the observation that co-location of people can be used as "proxy" for a face-to-face social interaction, as discussed in [2].

In this work, we analyze the performance of the ReD-HuP (Remote Detection of Human Proximity) algorithm originally presented in [3] by using a real-world dataset that reproduces social interactions indoor at realistic conditions [4]. ReD-HuP is a distributed algorithm running on multiple stations, each of which collects and analyzes the BLE beacons emitted by tags. Differently from existing some notable solutions for detecting social interactions [5], [6], [7], nodes running the ReD-HuP algorithm detect an interaction by only listening for beacons emitted by wearable tags. Therefore, the user's devices are not used to analyze the proximity with another users, rather they are only used to emit a signal. The fixed wireless receivers, deployed in the environment, are designated to declare if a social interaction occurs.

ReD-HuP has been originally tested with small but significant dataset, we now aim at further testing its performance with a more challenging set of scenarios and to compare the results obtained with some classification methods. Differently from our previous work, the ReD-HuP is used with BLE beacons collected at different powers of emission, raging from -18dBm, +6dBm and +3dBm and by considering more complex patterns of interactions. More specifically, we test ReD-HuP with dyads interacting while resting and walking indoor. Moreover, we also further explore how the width of the time window used to collect BLE beacons affects the overall results. We measure the accuracy of ReD-HuP in detecting social interactions and we compare the results against the Logistic Regression and Random Forest classification methods. In all the tested scenarios, we observe that ReD-HuP obtains values of the accuracy always comparable and higher with respect to the selected classifiers, shewing a robustness the solution proposed but without the need of a traditional training.

The current literature reports several works addressing the problem of automatically detecting social interactions with sensing devices. Some relevant works are based on the use of infrared radio communications and in particular of the use of smart badges as done in [8], [9]. In such works, the hardware adopted relies on a badge emitting RFID sensors and on a receiving device. The emitter sends signals in a range of 1 to 1.5 meters, while the receiver is, commonly, installed on the environment (e.g. on the ceiling). The Copenhagen Networks Study [10], [11] is an interesting project aimed at studying the social interactions of people by means of data collected with a mobile app for smartphone. The application captures multiple signals, including WiFi and Bluetooth scans. More recently, authors of [5] adopt BLE Beacons to infer friendships ties among people. The authors deploy a number of beacons indoor and they exploit some context information in order to classify interactions along the time. Finally, authors of [12], [13], [14] use commercial mobile devices to detect interactions among people. More specifically, in [12] authors use Android Wear and Tizen smartwatches and they present results related to the use of BLE advertising and scan operations implemented on a customized device (developed by the authors) and on two commercial smartwatches. Whereas, the proposed solution exploits the natural feature of the wearable device of emitting beacons. In this way, no specialized hardware is necessary. Moreover, the proposed solution is also energy preserving from the point of view of wearable devices, since the only requirement is to emit and not receive beacons like in all the referenced works.

The rest of the paper is structured as follows. Section II describes the ReD-HuP algorithm Section II-A and the dataset used Section II-B. Finally, Section III provides the experimental results with a comprehensive comparison between ReD-HuP and two other classification methods.

II. The Experimental Settings

A. The ReD-HuP Algorithm

The ReD-HuP algorithm [3] is designed to detect social interactions among people by detecting their proximity with a voting strategy. The algorithm is based on the analysis of the RSSI (Received Signal Strength Indicator) estimated by the receiving devices. RSSI is basically a measurement of a power expressed in decibel-milliwatts (dBm) and it can be correlated with the distance between the transmitter and the receiver [15]. A social interaction happens if a dyad lies for a certain period at a relative short distance, that we estimate in the range 0.5 to 1.5 meters. Such range is defined according to the definition of social distances proposed by [16]. In real-wold settings, the RSSI varies according to multiple factors. Among them, we cite the human's body orientation, the presence of other people in the nearby, any physical obstacle in between the emitting and the receiving device as well as the interference caused by other radio interfaces. However, as a general rule, we can consider that the higher the RSSI the closer the emitting and receiving device. ReD-HuP is a distributed algorithm running on multiple receiving devices. We refer to such devices with the term *anchor*. Each anchor collects beacons and it analyzes the beacon's RSSI on demand, in order to asses the presence or absence of an interaction between a pair of users. Users are supposed to wear a BLE tag emitting beacons at fixed rate with a certain power of emission. More precisely, ReD-HuP is based on a voting strategy, With the term *voting*, we refer to a collaborative process through which all the anchors are asked to vote for the presence or absence of an interaction.

The algorithm is characterized by two phases. Firstly, given the dyad (i, j) each anchor analyzes the beacon's RSSI emitted by tags of users (i, j). The analysis resulting from each station allows to vote for (i, j): *1* for the presence of the interaction, *-1* for absence of the interaction or *0* if the anchor is not able to provide a result. Secondly, all the votes provided by the anchors are combined together in order to produce the final output. Basically, ReD-HuP performs the sum of all the votes:

- if the sum results greater than 0, then the majority of the anchors voted for the interaction. In this case, the ReD-HuP detects an interaction for (i, j);
- if the sum results less than 0, then ReD-HuP detects absence of an interaction for (i, j);
- if the sum is equal to 0 than the majority of the anchors is not able to provide a final vote. In this case, ReD-HuP gives priority to the anchor with an higher channel stability. This last case is introduced in order to allow the anchor with the highest quality to provide the final result for (i, j).

In order to produce a vote, the beacon's RSSI are analyzed by each anchor by collecting sequences of beacons in a time window of duration τ. Each anchor also sets two thresholds, namely σ_{RSSI} and Δ_{RSSI}. σ_{RSSI} is expressed in dBm and it is used in order to exclude those anchors listening beacons with RSSI lower than a threshold. In other words, an anchor returns a vote only if it listens for beacons with a certain quality. The Δ_{RSSI} value is used in order to decide if dyad is interacting or not. Δ_{RSSI} measures the absolute value of the difference between the mean values of RSSI of the dyad (i, j). In particular, anchor x records beacons emitted by i and j, if the difference between the mean value of i's beacons and j's beacons is lower than Δ_{RSSI}, than the dyad (i, j) close enough to interact. In other words, increasing the Δ_{RSSI} value, the range defined in [16] to identify the social interaction increases. At the end of the two phases, ReD-HuP is able to estimate the time intervals during all the dyads interact. A more detailed description of the ReD-HuP algorithm can be found in [3].

B. The Dataset in Brief

We are interested in assessing the performance of the ReD-HuP algorithm with a real-world dataset reproducing social interactions in an indoor environment at realistic conditions. To this purpose, we adopt the dataset described in [4].

The dataset is produced by using a number of stationary anchors deployed in the environments and a set of mobile devices. The BLE beacons are emitted and received by both anchors and mobile devices.

The dataset has been designed by collecting RSSI data with two orthogonal settings:

- *Self Positioning*: what the mobile devices collect.
- *Remote Positioning*: what the anchors collect.

Anchors are based on Raspberry Pi 3 platform equipped with a programmable Bluetooth dongle, namely the BLED112 by Bluegiga. Dongles are able to advertise and to collect beacons. The mobile receiving devices we adopted are Honor 8 by Huawei Technologies, while the mobile emitter we used are RadBeacon Dot produced by Radius Networks. Devices have been configured to transmit advertisements by using the iBeacon[1] protocol at the frequency of 10Hz. Three different experimental campaigns have been performed by varying the transmission power of the emitting devices: -18dBm, -6dBm and +3dBm, respectively. Such variety of settings allows to analyze the performance of the ReD-HuP algorithm at very different conditions. Moreover, the dataset includes six

[1]https://developer.apple.com/ibeacon/

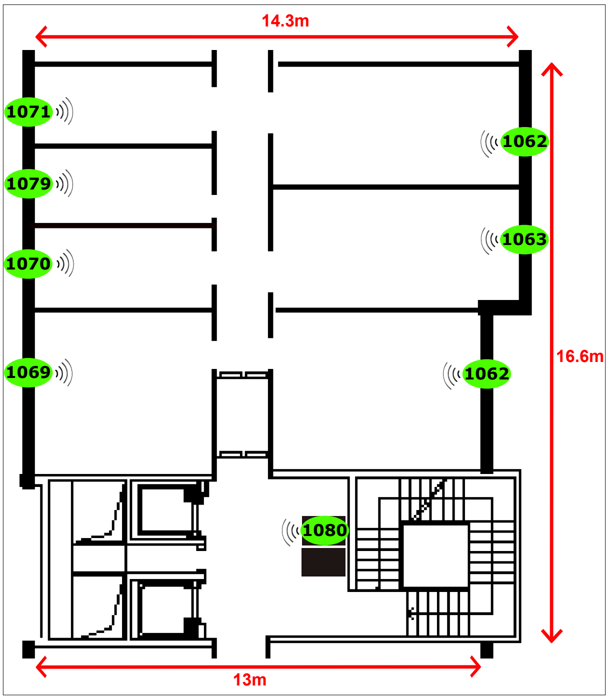

Fig. 1: Map of the testing area.

TABLE I: Overview of the dataset concerning the four social scenarios.

Session	Number of Actors	Number of Meetings	Duration () [min]	Collected beacons		
				-18dBm	-6dBm	+3dBm
1	2	1	14.1 (9)	2237	12137	21192
2	2	1	15.9 (6.6)	2046	14012	20463
3	3	3	36 (18)	9678	71041	104620
4	3	3	33 (18)	10822	63338	87349

reference scenarios designed to mimic some common patterns of interaction among people in an indoor environment: *survey*, *indoor localization* and four different *socialization* scenarios. In total, 18 data collection campaigns have been performed with a result of about 4 millions of beacons collected. We show in Figure 1 the map of the indoor environment used for all the data collection campaigns. The indoor environment is composed of 7 contiguous rooms, a corridor and a small adjacent area housing coffee and vending machines. The testing spans for about $185m^2$ with a maximum vertical span of approximately $16.6m$ and maximum horizontal span of approximately $14.3m$. The whole sensing area is covered by 8 anchors as shown in Figure 1. For what concerns the social interaction scenarios, four different sessions have been performed. As reported in Table I each session varies the number of involved actors, the number of meeting, the session and the meeting duration, and, finally, the number of collected RSSI values for each transmitting power. More specifically, the four sessions have the following features:

- *Session 1,* S_1: actor 1 moves from her workstation in room 1070 to interact with actor 2 in room 1062. Two actors perform a static face-to-face interactions and, at the end, actor 1 walks back to her room.
- *Session 2,* S_2: the same two actors carry out initially a static standing face-to-face meeting in the coffee area (1080). Then, they walk together through the corridor and then they walk back to their respective rooms, 1070 and 1062 for actor 1 and 2, respectively.
- *Session 3,* S_3: the first meeting occurs in the corridor and it involved actor 1 and 2. Later, actor 1 walks in room 1069 in order to meet actor 3. In the meanwhile, actor 2 walks back to her office. Thirdly, actor 1 walks back to her office and then actor 3 moves in room 1062 in order to meet actor 2. Finally, each actor returns to their rooms, 1070, 1062 and 1069 for actor 1, 2 and 3, respectively.
- *Session 4,* S_4: in this final session, different kinds of meetings involving two and three actors were performed. At first, a meeting between actor 1 and actor 2 is performed in room 1062. Then, actor 3 joins the meeting that therefore becomes an interaction between three individuals. Afterwards, the three actors walk together in the corridor and split up with actor 1 coming back to his office while the other two actors go together to the coffee area carrying out the last meeting.

The dataset also provides a ground-truth annotation, obtained by annotating the starting and ending time of each of the social interactions collected. The annotations are recorded by the users involved with the data collection.

III. Experimental Results and Discussion

We now evaluate the performance of the ReD-HuP algorithm (see Section II-A) with the experimental dataset. We first introduce the metrics we used as well as two classification methods for a comparative analysis.

A. Classification Methods and Metrics

Machine learning techniques provide several advanced methods to classify observations according to a number of well-known classes or categories. More specifically, a classification task consists in assigning a class from a set of possible classes to a given observation. For the purpose of this work, we aim to classy the beacons readings obtained from the dataset (see Section II-B) according to two simple classes: 0 = interaction between a dyad or 1 = no interaction between a dyad.

The classification methods generally require to be trained on a set of labelled observations. Labels specify to which of the classes each observation belongs to. The higher the number of labels, the more accurate the classification result. After the model is trained, the model can be used on new observations. In this work, in order to validate our system, two different machine learning models have been applied for automatically detecting interactions, namely Logistic Regression and Random Forest [17], [18]

Logistic regression is a well–known technique based on linear regression in order to produce probabilities. When linear regression is applied for binary classification, a linear function employing regression is calculated and, then, a threshold is applied to decide a 0 or a 1 response. Differently, Random Forest is built on the simple idea to build a tree of decisions in which each internal node is labelled with an input feature. The arcs from a node representing a particular feature are labelled with each of the possible values of that feature. Each leaf of the tree is labelled with a class or a probability distribution over the classes.

We select Logistic Regression and Random Forest because they are representative parametric and non-parametric models respectively. Furthermore, in literature, the efficacy to apply these two models for binary classification was already investigated [19]. If the target variable (i.e., a meeting is occurring or not), is not linearly separable then a more complex model may achieve higher prediction scores. Instead, non-parametric models (e.g., Random Forest) are more complex models and, as a consequence, they can predict through decision boundaries with high variability. However, they have low bias and often they suffer from the overfitting problem. On the other hand, the parametric models (e.g., logistic regression) are generally less complex model resulting in a linear decision boundary. As a consequence, these models can suffer from an higher bias and can led to an under fitting problem. In this work, we show the performances of both models and we compare their performances with the proposed Red-HuP algorithm.

For each of the algorithms we use (classifiers and ReD-HuP), we compute the accuracy metric given by the proportion of correct answers of the algorithm with respect to the total amount of observations. For the purpose of this work, we only focus on the accuracy metric, however other metrics such as precision and recall can be further analyzed.

B. Performance of the ReD-HuP Algorithm

Our analysis starts with a performance assessment of the ReD-HuP algorithm by computing the accuracy in all the experimental sessions described in II-B ($S_1 \cdots S_4$). More specifically we consider the BLE beacons collected with the 3 different power of transmission (-18dBm, -6dBm and +3dBm), and we compute the accuracy score by varying τ, σ_{RSSI} and Δ_{RSSI}. The parameter τ determines the width of the time window during which anchors analyze the BLE beacons received. It ranges from 16 to 30 seconds (with a step of 2 seconds), with a total of 8 different time windows. The parameter σ_{RSSI} ranges from -94dBm to -76dBm (with a step of 2dBm) while the parameter Δ_{RSSI} from 3dBm to 8.4dBm (with a step of 0.2dBm). We experienced that values outside those ranges are not meaningful and do not affect significantly the accuracy metric. As a result, we show the distribution of the accuracy for each of the previous settings (a total of 279 settings), as shown in Figure 2. The figure shows for each step of τ a box plot reporting how the accuracy varies as a function of σ_{RSSI} and Δ_{RSSI}. Each box plot shows the median, the 25^{th} and 75^{th} percentile of the accuracy as well the max and min. Moreover, we also shows for clarity each of the accuracy values as a dots.

The transmission power set to -6dBm represents our worst case from the point of view of performance variability. More specifically, we measure a median value of accuracy with all values of $\tau = [16, 30]$ of 75.14% and a standard deviation of 5.64. As shown by the box plots, we observe a higher dispersion of the accuracy values with respect to the other two settings (-18dBm and +3dBm). As for example, values of accuracy obtained with $\tau = 22s$ span from 82.9% to 57.84% with an difference of 25.11%. Similar considerations also apply for other values of τ such as 26s, and 28s. In all of such cases, we observe a high variability of the accuracy values. In this case, we observe that ReD-HuP more often fails to correctly detect a social interaction. More specifically, tags are set to an intermediate power of emission that lead to a high number of false positive/negative answers from the voting station. In particular, this negative effect is evident for those anchors located far from the tags but still listening for beacons. Such anchors are mole likely to fail in detecting presence/absence of interaction with such an intermediate power of emission.

Values of accuracy obtained with -18dBm provide an intermediate performance. Generally, box plots obtained with such setting show a lower dispersion of points with respect to other settings. The median accuracy obtained with all values of τ is 74.73% (lower than the -6dBm settings) but with an increasing stability of the measured performance (standard deviation of 3.73). Differently from the setting -6dBm, the low power of transmission of tags reduces the number of wrong answers from anchors placed far or at mid-distance from the interaction point. In fact, such anchors do not analyze beacons because the RSSI values are too low (see Section II-A) or they provide abstain from voting.

Finally, the values of accuracy obtained with 3dBm provide our optimal case (better accuracy and low standard deviation). In this setting, we experienced a median accuracy of 79.62% and a standard deviation of 4.89. From the box plot it is possible to observe that the median of the accuracy with all the values of τ is generally higher than that the other settings. As discussed previously, high power of transmission of tags allow anchors to take the correct decision more often. Anchors more easily detect presence or absence of an interaction by limiting those doubtful conditions.

We then compare the performance of ReD-HuP against the two classifiers introduces in Section III-A. To this purpose, adopt the following approach:

- Sessions S_1 and S_2 for calibrating the algorithms;
- Session S_3 and S_4 for evaluating the performance.

By splitting the calibration sessions from the experimental ones, we avoid to use the same data both training and validating even without any cross-validation technique. We simply use different data for the two phases. The training of the two classifiers is performed by providing them as input the average value of the beacon's RSSI grouped every τ seconds. Each of the inputs is labelled with the ground truth, namely presence or absence of an interaction. For what concerns the calibration of ReD-HuP we exploit sessions S_1 and S_2 to

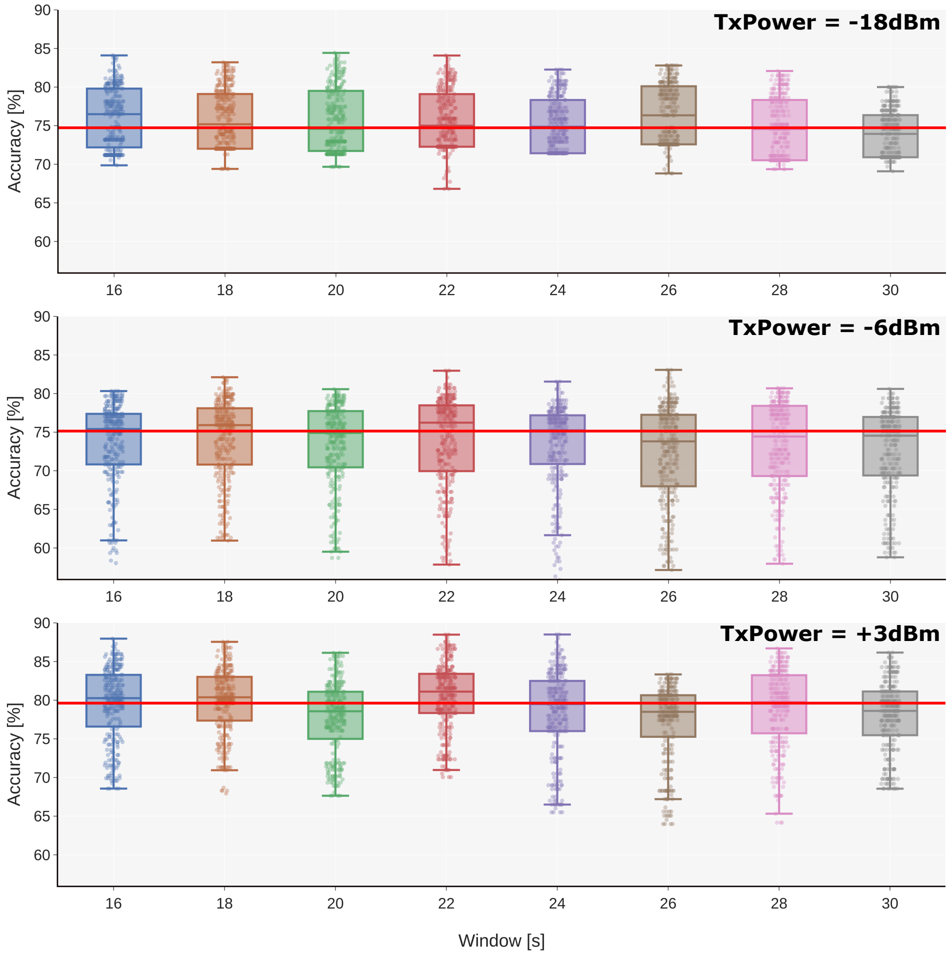

Fig. 2: Distribution of the accuracy metric as a function of τ, σ_{RSSI} and Δ_{RSSI}.

find the optimal configuration of the parameters σ_{RSSI} and Δ_{RSSI}. We consider that, such approach, allows to compare the 3 algorithms at similar conditions.

The comparison of the algorithms is shown in Figure 3. We observe that in all of the settings of τ, Red-HuP obtains values of the accuracy generally higher than that the two classifiers. We also report for each setting, the optimal value of accuracy obtained with the ReD-HuP algorithm. More specifically, with -18dBm and τ= 26s ReD-HuP obtains 77.78% , with -6dBm and $\tau = 26s$ 85.45% while with 3dBm and τ = 21s 89.08%. Therefore ReD-HuP performs better with a time window ranging from from 21s to 26s. Differently, small or high values of τ do not provide optimal results of accuracy.

IV. Conclusions

Social interaction can be detected and analyzed by exploiting wearable sensing units, able to detect proximity among subjects. Under this context, we study in this work the performance of the ReD-HuP [3] algorithm with a real-world dataset. Red-HuP relies on the analysis of BLE beacons by adopting a voting strategy, so that to combine the analysis made by all the receiving devices. ReD-HuP is compared against two common classification methods, namely Logistic Regression and Random Forest. Experimental results show that ReD-HuP algorithm outperforms the selected techniques. We believe that the design of ReD-HuP opens to new perspectives for monitoring human behaviours. In fact, BLE beacons are expected to be even more diffused on most of the commercial devices and their analysis can reveal dynamics of the social interactions with high resolution. We consider that ReD-HuP implements an effective and simple strategy to detect social interactions. However, as reported in Section III, there exist some conditions in which ReD-HuP might fail to correctly reveal the existence of an interaction. In order to mitigate such conditions, we consider as a future line of investigation, the use of a self-calibration procedure, able to re-compute the thresholds used by the algorithm. In this way, ReD-HuP can re-adapt to different conditions, by reducing false positive and negative answers from the voting stations.

References

[1] M. Granovetter, “The strength of weak ties: A network theory revisited,” *Sociological Theory*, vol. 1, pp. 201–233, 1983. [Online]. Available: http://www.jstor.org/stable/202051

[2] M. Génois and A. Barrat, “Can co-location be used as a proxy for face-to-face contacts?” *EPJ Data Science*, vol. 7, no. 1, p. 11, 2018. [Online]. Available: https://doi.org/10.1140/epjds/s13688-018-0140-1

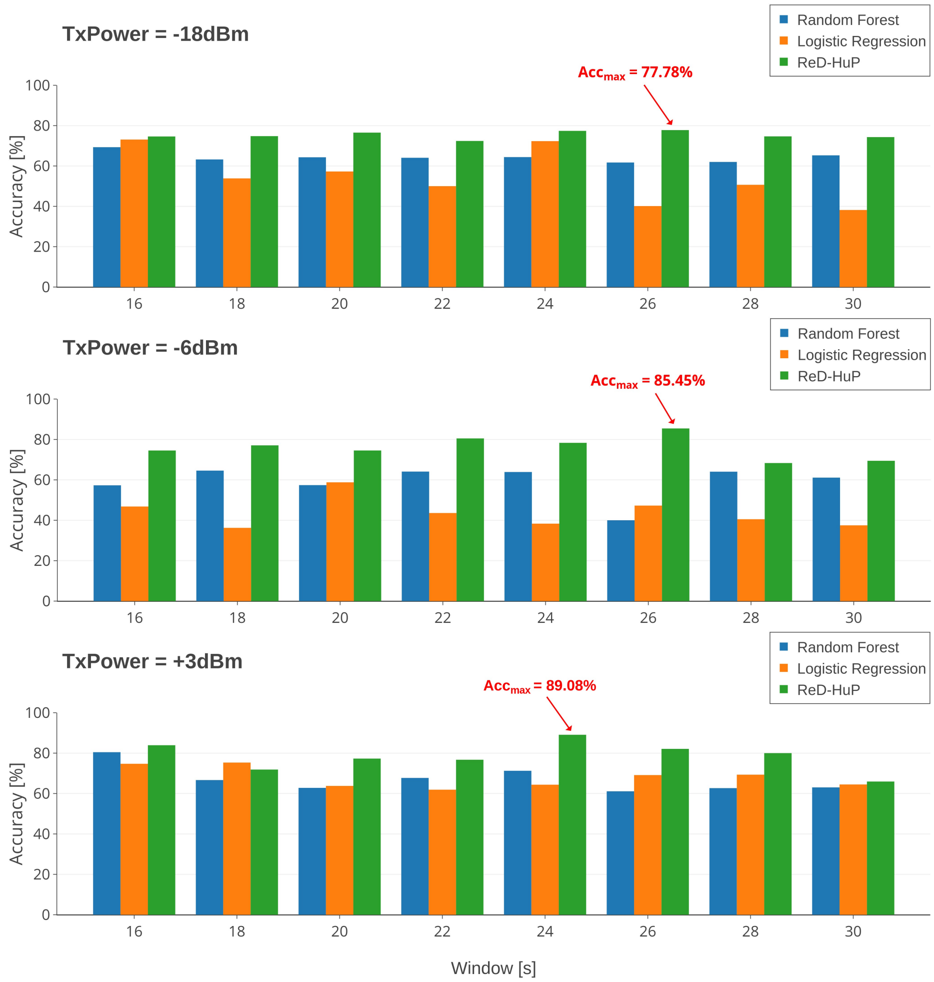

Fig. 3: Comparison of performance of ReD-HuP with respect to Random Forest and Logistic Regression.

[3] F. Mavilia, F. Palumbo, P. Barsocchi, S. Chessa, and M. Girolami, "Remote detection of indoor human proximity using bluetooth low energy beacons," in *2019 15th International Conference on Intelligent Environments (IE)*. IEEE, 2019.

[4] P. Baronti, P. Barsocchi, S. Chessa, F. Mavilia, and F. Palumbo, "Indoor bluetooth low energy dataset for localization, tracking, occupancy, and social interaction," *Sensors*, vol. 18, no. 12, p. 4462, 2018.

[5] R. Purta and A. Striegel, "Predicting friendship pairs from ble beacons using dining hall visits," in *2019 28th International Conference on Computer Communication and Networks (ICCCN)*, July 2019.

[6] S. Liu, Y. Jiang, and A. Striegel, "Face-to-face proximity estimationusing bluetooth on smartphones," *IEEE Transactions on Mobile Computing*, vol. 13, no. 4, pp. 811–823, April 2014.

[7] T. Choudhury and A. Pentland, "Sensing and modeling human networks using the sociometer," in *Proceedings of the 7th IEEE International Symposium on Wearable Computers*, ser. ISWC '03. USA: IEEE Computer Society, 2003, p. 216.

[8] J. Fournet and A. Barrat, "Contact patterns among high school students," *PLOS ONE*, vol. 9, no. 9, pp. 1–17, 09 2014. [Online]. Available: https://doi.org/10.1371/journal.pone.0107878

[9] C. Cattuto, W. Van den Broeck, A. Barrat, V. Colizza, J.-F. Pinton, and A. Vespignani, "Dynamics of person-to-person interactions from distributed rfid sensor networks," *PLOS ONE*, vol. 5, no. 7, pp. 1–9, 07 2010. [Online]. Available: https://doi.org/10.1371/journal.pone.0011596

[10] V. Sekara and S. Lehmann, "The strength of friendship ties in proximity sensor data," *PLOS ONE*, vol. 9, no. 7, pp. 1–8, 07 2014. [Online]. Available: https://doi.org/10.1371/journal.pone.0100915

[11] A. Stopczynski, V. Sekara, P. Sapiezynski, A. Cuttone, M. M. Madsen, J. E. Larsen, and S. Lehmann, "Measuring large-scale social networks with high resolution," *PLOS ONE*, vol. 9, no. 4, pp. 1–24, 04 2014. [Online]. Available: https://doi.org/10.1371/journal.pone.0095978

[12] A. Montanari, S. Nawaz, C. Mascolo, and K. Sailer, "A study of bluetooth low energy performance for human proximity detection in the workplace," in *2017 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, March 2017, pp. 90–99.

[13] M. Girolami, F. Mavilia, F. Delmastro, and E. Distefano, "Detecting social interactions through commercial mobile devices," in *2018 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, March 2018, pp. 125–130.

[14] P. Barsocchi, A. Crivello, M. Girolami, F. Mavilia, and F. Palumbo, "Occupancy detection by multi-power bluetooth low energy beaconing," in *2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, Sep. 2017, pp. 1–6.

[15] F. Potortì, A. Crivello, M. Girolami, E. Traficante, and P. Barsocchi, "Wi-fi probes as digital crumbs for crowd localisation," in *2016 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. IEEE, 2016, pp. 1–8.

[16] E. T. Hall, *The hidden dimension / Edward T. Hall*, [1st ed.] ed. Doubleday Garden City, N.Y, 1966.

[17] J. Friedman, T. Hastie, R. Tibshirani *et al.*, "Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors)," *The annals of statistics*, vol. 28, no. 2, pp. 337–407, 2000.

[18] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[19] K. Kirasich, T. Smith, and B. Sadler, "Random forest vs logistic regression: Binary classification for heterogeneous datasets," *SMU Data Science Review*, vol. 1, no. 3, p. 9, 2018.