Data Cleansing for Indoor Positioning Wi-Fi Fingerprinting Datasets

Darwin Quezada-Gaibor*,†, Lucie Klus†,*, Joaquín Torres-Sospedra‡,
Elena Simona Lohan†, Jari Nurmi†, Carlos Granell*, and Joaquín Huerta*
**Institute of New Imaging Technologies*, *Universitat Jaume I*, Castellón, Spain
†*Electrical Engineering Unit*, *Tampere University*, Tampere, Finland
‡*ALGORITMI Research Centre*, *Universidade do Minho*, Guimarães, Portugal

***Abstract*—Wearable and IoT devices requiring positioning and localisation services grow in number exponentially every year. This rapid growth also produces millions of data entries that need to be pre-processed prior to being used in any indoor positioning system to ensure the data quality and provide a high Quality of Service (QoS) to the end-user. In this paper, we offer a novel and straightforward data cleansing algorithm for WLAN fingerprinting radio maps. This algorithm is based on the correlation among fingerprints using the Received Signal Strength (RSS) values and the Access Points (APs)'s identifier. We use those to compute the correlation among all samples in the dataset and remove fingerprints with low level of correlation from the dataset. We evaluated the proposed method on 14 independent publicly-available datasets. As a result, an average of 14% of fingerprints were removed from the datasets. The 2D positioning error was reduced by 2.7% and 3D positioning error by 5.3% with a slight increase in the floor hit rate by 1.2% on average. Consequently, the average speed of position prediction was also increased by 14%.**

***Index Terms*—Data cleansing, Data pre-processing, Indoor positioning, Localisation, Wi-Fi Fingerprinting**

I. Introduction

Indoor positioning and localization services are becoming increasingly demanded in various applications, including patient monitoring, ambient assisted living, smart parking assistance and indoor navigation apps. Wi-Fi-based deployments are one of the most commonly used infrastructures for Indoor Positioning System (IPS) [1], mainly due to the global availability of Wi-Fi Access Point (AP)s and their standardized characteristics compliant with IEEE 802.11, ensuring a good generalization properties across deployments. The measurements of Wi-Fi Received Signal Strength (RSS) are easily obtainable by any User Equipment (UE), ranging from mobile phones to battery-restricted Internet of Things (IoT) devices such as wearables. The main advantages of utilizing RSS-based fingerprinting include its capability to perform well in environments with rich scattering characteristics and limited Line-of-Sight (LoS) availability, in which the deterministic path-loss models usually fail [2].

Fingerprinting is a simple technique, the position of a fingerprint (array of RSS measurements) can be estimated using the positions of the closest matches from a dataset with pre-recorded fingerprints (i.e., the radio map). The radio map acquisition, pre-processing, training the matching algorithm and its optimization are referred to as the offline phase of fingerprinting. The online phase consists of finding the coordinates of the newly measured fingerprint in a real time.

The achievable positioning performance of the fingerprinting method depends on the scenario and strategy to collect the radio map. The localization algorithm, whether the k-Nearest Neighbors (k-NN), or any alternative, can only fine-tune the positioning, which the training radio map allows it to.

In this work, we focus on improving the quality of the radio map by proposing a data cleansing scheme that is designed to remove the outlier samples from the radio map. The cleansing method calculates the similarity of each sample to the rest of the database based on the detected APs and their signal strength levels and removes the samples dissimilar to the rest. We then evaluate the proposed method on 14 publicly available Wi-Fi fingerprinting datasets and show they remain statistically unchanged. We also perform the fingerprinting-based positioning and show the improved performance of the cleansed databases when compared to the original ones.

The main contributions of this paper are as follows:

- We propose a novel and straightforward algorithm for removing unnecessary samples from Wi-Fi fingerprinting radio maps.
- We evaluate the proposed method and its capabilities on 14 independent open-access datasets.
- We show, that the proposed method not only reduces the size of the datasets, but also improves the building hit, floor hit and positioning accuracy, on average, across all available datasets. Moreover, it reduces the time required to perform the user positioning.

The rest of the paper is structured as follows. In Section II, we discuss the related literature and works connected to our research. Section III introduces the proposed data cleaning approach, which is later evaluated in Section IV. Additional impacting factors and things to consider are further elaborated in Section V and the work is concluded in Section VI.

Corresponding Author: D. Quezada Gaibor (quezada@uji.com)

The authors gratefully acknowledge funding from European Union's Horizon 2020 Research and Innovation programme under the Marie Skłodowska Curie grant agreements No. 813278 (A-WEAR: A network for dynamic wearable applications with privacy constraints, http://www.a-wear.eu/) and No. 101023072 (ORIENTATE: Low-cost Reliable Indoor Positioning in Smart Factories, http://orientate.dsi.uminho.pt).

II. Related work

In this section, we discuss the related literature and outline other data cleaning methods focused on Wi-Fi-based fingerprinting datasets. We also discuss the differences to our work and introduce its main advantages over the current State-of-the-Art.

Indoor positioning using Wi-Fi RSS fingerprinting was broadly addressed across literature, most commonly considering k-NN [3], [4] or various kinds of neural networks [5], [6] as the matching algorithm. Frequently, the individual works consider data pre-processing techniques, such as augmenting the radio map's data representation [3], reducing the number of APs by either removing the redundant ones [7], applying radio map compression [8], [9], or reducing the number of considered samples in the database by e.g., clustering [3], [10]. Nevertheless, improving the quality of the database itself by performing data cleansing was hardly ever addressed. In this work, we evaluate the relevance of each sample in the training database (radio map) and remove the redundant ones.

An example of localization dataset cleaning was proposed in [11]. There, the unlabelled fingerprint was first complemented with additional measurements, in the second iteration the coarse localization was realized, while in the last iteration the probabilistic model predicted the fine location. The work presented improved positioning results, but does not address the question of outliers within the positioning dataset.

The authors of [12] studied the effect of coverage gaps in the RSS positioning datasets by artificially decreasing the database's positioning capabilities. The work showed that removing the samples from the dataset with uniform probability does not have strong diminishing effect. Alternatively, creating the measurement gaps in the training database strongly harmed the overall positioning performance in the deployment. Compared to this work, we eliminate the specific measurements from the database to boost the performance.

Simultaneous localization, outlier detection, and radio map interpolation was realized in [13], which organizes the APs based on their similarity. The work supplements the missing measurements in the fingerprint by interpolating the measured RSS from the neighboring APs. The outlier detection algorithm discards the irrelevant measurements caused by, e.g., adversary attacks. The proposed Group-Sparsity localization system is able to perform even with the reduced database, but the only benchmark utilized in the comparison was compressive sensing, which is not commonly deployed in indoor localization schemes.

The authors of [14] identified several ways to enhance the radio map, including data cleansing and denoising. In [15], the Received Signal Strength Indicator (RSSI) measurements were extracted to overcome sparsity with a stacked Denoising AutoEncoder (DAE). In [16], denoising relied on another neural network architecture which handled not only sparsity but also RSSI fluctuations. In [17], denoising focused on learning the noise characteristics rather than the original characteristics.

The challenge of missing and false values in crowdsensed RSSI sequence data was addressed in [18]. The mapping of RSSI sequences to the floor plan effectively boosts positioning capabilities, yet in many cases, as in this work, the temporal dependencies between samples are not available. Consequently, the conclusions from [18] cannot be applied directly.

The authors of [19] empirically determined the relation between the RSS data and its deviation. The study models the uncertainties in both static and mobile UE situations, but restricts itself to the unobstructed link between the transmitter and the receiver. Nevertheless, uncertainty modelling and its estimation within the fingerprints can enhance the positioning model's knowledge and thus positively impact the positioning accuracy itself.

Compared to the works presented above, we restrict the dataset cleansing approach to directly remove the redundant, irrelevant and confusing samples from the training database, rather than finding the missing values and complementing the radio map, as is the case in many of the aforementioned references. By doing so, this work does not add any synthetically obtained information into the database, and therefore cannot introduce additional bias.

III. Data cleansing

In this section, we provide a general overview of Wi-Fi fingerprinting using the proposed data cleansing algorithm.

A. Overview

WLAN fingerprinting technique has been extensively researched during the last decade for both indoor and outdoor positioning, and it is being used in many commercial and open-source solutions. Generally, this technique consists of two phases - the online and the offline phase. In the offline phase RSS measurements are collected in known reference points to build a radio map. During the online phase, the RSS values collected in an unknown positions are matched with the fingerprints in the radio map using a matching algorithm such as k-NN in order to estimate the device's position.

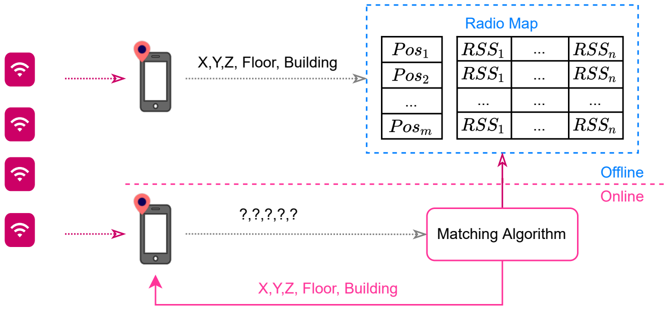

Fig. 1. Wi-Fi Fingerprinting Technique.

Although this technique is considered one of the most robust techniques for indoor positioning, it may be affected by undesirable fluctuations in the signals, leading to an increase in terms of the positioning error. In case these fluctuations are not filtered out during the data collection in the offline phase, they might affect the positioning estimation in the online phase. Therefore, it is desirable to remove these noisy samples from the dataset in order to avoid errors in the position estimation and provide Quality of Experience (QoE) to the end-user.

In order to detect unnecessary fingerprints in the dataset, we propose a new straightforward method to remove outliers and/or unnecessary fingerprints in the radio map. This process consists of five steps detailed in the following paragraphs.

B. Valid RSS values

The first step is to determine the number of valid RSS values per fingerprint. In this case, the non-detected value (γ) has to be set in order to exclude it from the RSS values. Once the number of valid RSS values per fingerprint has been determined, the average or the maximum number of RSS values can be used to determine the correlation between samples. Considering a radio map $\Psi \in \mathcal{R}^{m \times n}$, where m is the number of samples (fingerprints), and n represents the number of APs in the radio map, the average or the maximum number of valid RSS values ($\wp$) can be determined as follows:

$$\begin{aligned} \nu_i &= len(\Psi_i) | \forall \psi_{ij} \neq \gamma \\ \wp &= \lfloor mean(\nu) \rfloor \text{ or } max(\nu) \end{aligned} \tag{1}$$

where $\nu \in \mathcal{R}^m$ is a vector that contains the number of valid RSS values of the i-th sample, $i = 1, 2, 3, ..., m$ and $j = 1, 2, 3, ..., n$. ψ_{ij} is the RSS value in the i-th and j-th position.

C. Sort and Replace

In this step, the RSS values are sorted in descending order and replaced for their corresponding AP identifier. Then, the first $\wp$ columns are selected to compute the match percentage between the fingerprints.

$$\begin{aligned} \mathcal{X}_i &= sort(\Psi_i, descending) \\ x_{ij} &\leftarrow AP_j \end{aligned} \tag{2}$$

where $\mathcal{X} \in \mathcal{R}^{m \times \wp}$ represents the radio map, using the AP identifier instead of the RSS value and x_{ij} is the RSS value in the i-th and j-th position. AP_j is the AP identifier in the j-th position.

D. Compute the match percentage between samples

The next step is to compute the correlation between samples. In this case, it is necessary to set a threshold (ρ) prior to computing the match percentage ($\Im$). This threshold (ρ) represents the minimal match percentage between samples. The match percentage therefore is computed among all sample of the matrix $\mathcal{X}$.

$$\mathcal{X} = \begin{bmatrix} x_{11}, x_{12}, & \dots & x_{1\wp} \\ x_{21}, x_{22}, & \dots & x_{2\wp} \\ \vdots & \ddots & \vdots \\ x_{m1}, x_{m2}, & \dots & x_{m\wp} \end{bmatrix}$$

Thus, the i-th sample is compared with the l-th sample ($l = 1, 2, 3, ..., m$) under the following conditions:

$$\Im_i = \begin{cases} \Im_{i\_old}, & \text{if } \mathcal{X}_i = \mathcal{X}_l, \\ \Im_{i\_old}, & \text{if } \Im_{i\_old} > \Im_i, \\ \Im_{i\_old}, & \text{if } \Im_i < \rho, \\ \frac{len(\mathcal{X}_i \cap \mathcal{X}_l)}{\wp} * 100, & \text{otherwise} \end{cases} \tag{3}$$

where $\Im_{i\_old}$ is the previous match percentage computed between the i-th and the l-th sample ($\Im_i$).

E. Remove unnecessary fingerprints

In the last step of the proposed algorithm, all samples with zero match percentage are removed from the original radio. These samples are considered outliers or unnecessary samples, given that they do not have high enough level of correlation with the rest of the samples. i.e., they may correspond to noisy samples poisoning the radio map.

Algorithm 1: CleanDB

```
Input  : X_train, non_detected_value, threshold
Output: $\Psi_c$
$\Psi \leftarrow X\_train$
$\Psi_c \leftarrow X\_train$
/* Avg. or max. number of valid RSS
   values                                  */
$\nu_i = len(\Psi_i) | \forall \psi_{ij} \neq \gamma$
$\wp = \lfloor mean(\nu) \rfloor$ or $max(\nu)$
/* Sort and replace RSS values             */
$\mathcal{X}_i = sort(\Psi_i, descending)$
$x_{ij} \leftarrow AP_j$
/* Select the first $\wp$ columns of $\mathcal{X}$  */
$\mathcal{X} \in \mathcal{R}^{m \times \wp}$
/* Compute the match percentage            */
for i=1 to m do
    for l=1 to m do
        $\Im'_i = \frac{len(\mathcal{X}_i \cap \mathcal{X}_l)}{\wp} * 100$
        if $\mathcal{X}_i \neq \mathcal{X}_l$ & $\Im_{i\_old} < \Im'_i$ & $\Im'_i > \rho$ then
            $\Im_i = \Im'_i$
    $\Im_{i\_old} \leftarrow \Im_i$
/* Remove samples with zero match
   percentage                              */
for i=1 to m do
    if $\Im_i == 0$ then
        DEL ($\Psi_{ci}$)
```

Algorithm 1 summarizes all the previously explained steps to remove unnecessary fingerprints from the radio map. The algorithm requires three parameters, the training dataset, the non-detected value and the predefined match percentage (ρ). The output is the cleaned training dataset (Ψ_c).

IV. Experiments and Results

This section provides the experiment setup, a brief description of 14 datasets used in the experiment, and the primary outputs of the proposed data cleansing algorithm for indoor positioning radio maps. The source code used in this experiment is available for public usage on Zenodo under the CC BY license [20].

A. Experiment setup

The experiments were performed using a MacBook Pro with an M1 Pro chip packing a 10-core CPU, a 16-core GPU, and 16GB of RAM. The software used for implementation was Python 3.9, and these experiments were carried out using 14 Wi-Fi fingerprinting datasets collected in differing and heterogeneous scenarios. These datasets are: UJI 1–2, LIB 1–2 (collected at Universitat Jaume I, Spain), MAN 1–2 (collected at University of Mannheim, Germany), TUT 1–7 (collected at Tampere University, Finland) and UTSIndoorLoc (collected at University of Technology Sydney, Australia) [3], [21]. These datasets are representatives of multi-floor environments, all apart from MAN 1–2, which consist of measurements from one floor only. Additionally, UJI 1–2 datasets are apart from multi-floor environments also multi-building environments, as they consist of measurements obtained across several buildings.

The core algorithm to estimate the user or device position as well as to classify the fingerprints into buildings and floors was k-Nearest Neighbors (k-NN). It was selected for its good positioning capabilities as previously demonstrated in the literature [3], [22], [23]. The hyperparameters set in the k-NN algorithm are k equal to 1 and Manhattan distance as the distance metric to compute the similarity between the fingerprint vectors. The modules used are *KNeighborsClassifier* and *KNeighborsRegressor* from the *Scikit-learn* (Sklearn) library. Additionally, positive data representation [24] was used in all datasets prior to applying the proposed algorithm.

In order to choose the optimal threshold of the match percentage, the experiments were run using thresholds in intervals of 5%. If the positioning error increases or the floor hit rate decreases within the used interval, intermediate values are selected to run the algorithm. For this reason, there are thresholds of the match percentage equal to 1%, 2%, 20%, 21%, etc. (see Table I). The non-detected value used for all original datasets is 100 dBm and the maximum number of valid RSS samples ($\wp$) is given by Eq. 1.

The results obtained with k-NN using the original dataset and the cleansed dataset were compared in terms of mean 2D positioning error (ϵ_{2D}), mean 3D positioning error (ϵ_{3D}), building hit rate (ζ_b), floor hit rate (ζ_f), testing time (δ) and the size of the training dataset ($\mathcal{T}_{TR}$). Given the heterogeneity of the datasets, the results obtained with the original dataset and the cleansed dataset were normalized in order to be compared, e.g., normalized mean 3D positioning error ($\tilde{\epsilon}_{3D}$). The values reported with the plain 1-NN for the above mentioned metrics have been selected for the normalisation. i.e., normalised values will be relative to that baseline.

B. Results

Table I shows the parameters of each dataset and the main results after running the k-NN algorithm with the original datasets and with the cleansed datasets. $|\mathcal{T}_{TR}|$ represents the number of training samples in the dataset, $|\mathcal{T}_{TE}|$ is the number of testing samples and $|\mathcal{A}|$ is the number of APs in the dataset.

For the baseline method, the 1-NN algorithm, the absolute and normalised values are provided. After using the proposed data cleansing algorithm, most of the training datasets reduced their number of samples, with the exception of TUT 7 dataset in which the cleansing algorithm was not able to detect unnecessary samples. That is why the threshold was set to 0% for TUT 7 dataset. The minimal number of unnecessary fingerprints removed from the datasets were 9 of 3117 in TUT 6 dataset ($\approx$ 0.29%), and the maximum number of removed fingerprints was 237 from the LIB 2 dataset ($\approx$ 41% of the original dataset size). In any case, the positioning error, floor and building hit rate were not negatively affected.

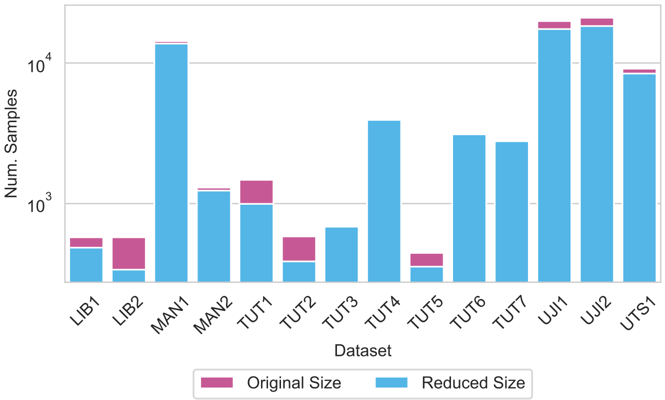

Fig. 2. Dataset size before and after applying the data cleansing algorithm.

Fig. 2 shows the number of fingerprints after (blue) and before the data cleansing (red). In all but one dataset the algorithm achieved at least a small reduction of their original size. Similarly, the prediction time was slightly reduced after applying the data cleansing algorithm by $\approx$ 14% (see Table I).

Additionally, the use of the proposed data cleansing algorithm reveals a slight increment in the average building hit rate ($\tilde{\zeta}_b$) from 1 to 1.004 (0.4%) and the average floor hit rate ($\tilde{\zeta}_f$) from 1 to 1.012 (1.2%). Similarly, the proposed algorithm allowed us to reduce the positioning error in most of the datasets in both 2D ($\tilde{\epsilon}_{2D}$) and 3D ($\tilde{\epsilon}_{3D}$) positioning error. For instance, the normalized mean 3D positioning error in LIB 1 was reduced from 1 to 0.998 without affecting the floor hit rate. In LIB 2, the error was reduced from 1 to 0.858 ($\approx$ 58cm), increasing the floor hit rate from 1 to 1.020 ($\approx$ 2%).

In general, the average normalized 2D positioning error decreased from 1 to 0.973 (2.7%) and the average normalized 3D positioning error from 1 to 0.947 (5.3%). The accuracy of the floor hit increased by 1.2%, and the building hit rate remained almost unchanged.

TABLE I

COMPARISON 1NN ALL DATA VS. 1NN CLEANED DATA

	Parameters				Baseline 1-NN											Cleaned DB + 1-NN					
Database	$\|\mathcal{T}_{TR}\|$	$\|\mathcal{T}_{TE}\|$	$\|\mathcal{A}\|$	$\|\rho\|$	ζ_b [%]	ζ_f [%]	ϵ_{2D} [m]	ϵ_{3D} [m]	δ [s]	$\tilde{\mathcal{T}}_{TR}$ [−]	$\tilde{\zeta}_b$ [−]	$\tilde{\zeta}_f$ [−]	$\tilde{\epsilon}_{2D}$ [−]	$\tilde{\epsilon}_{3D}$ [−]	$\tilde{\delta}$ [−]	$\tilde{\mathcal{T}}_{TR}$ [−]	$\tilde{\zeta}_b$ [−]	$\tilde{\zeta}_f$ [−]	$\tilde{\epsilon}_{2D}$ [−]	$\tilde{\epsilon}_{3D}$ [−]	$\tilde{\delta}$ [−]
LIB1	576	3120	174	33	-	99.84	3.035	3.043	0.531	1	1	1	1	1	1	0.844	-	1.000	0.998	0.998	0.843
LIB2	576	3120	197	40	-	97.724	4.031	4.197	0.608	1	1	1	1	1	1	0.589	-	1.020	0.888	0.858	0.589
MAN1	14300	460	28	34	-	-	2.877	2.877	0.376	1	1	1	1	1	1	0.961	-	-	0.981	0.981	0.914
MAN2	1300	460	28	45	-	-	2.467	2.467	0.034	1	1	1	1	1	1	0.952	-	-	0.989	0.989	0.927
TUT1	1476	490	309	35	-	90	8.623	9.601	0.401	1	1	1	1	1	1	0.674	-	1.014	0.903	0.873	0.769
TUT2	584	176	354	30	-	72.727	11.218	12.893	0.073	1	1	1	1	1	1	0.664	-	1.039	0.964	0.939	0.660
TUT3	697	3951	992	2	-	91.622	8.926	9.594	5.035	1	1	1	1	1	1	0.983	-	1.003	0.990	0.978	0.984
TUT4	3951	697	992	1	-	95.265	6.152	6.406	5.424	1	1	1	1	1	1	0.996	-	1.000	0.998	0.999	0.992
TUT5	446	982	489	21	-	88.391	6.387	6.924	0.393	1	1	1	1	1	1	0.798	-	1.001	0.969	0.956	0.809
TUT6	3116	7269	652	2	-	99.986	1.959	1.959	27.612	1	1	1	1	1	1	0.997	-	1.000	0.990	0.990	0.997
TUT7	2787	6504	801	0	-	99.185	2.110	2.351	27.429	1	1	1	1	1	1	✗	-	✗	✗	✗	✗
UJI1	19861	1111	520	20	99.190	87.759	7.718	10.829	21.674	1	1	1	1	1	1	0.877	1.008	1.030	1.000	0.828	0.877
UJI2	20972	5179	520	20	100.000	85.345	7.742	8.052	108.441	1	1	1	1	1	1	0.873	1.000	1.022	0.978	0.960	0.876
UTS1	9108	388	589	20	-	92.784	7.769	8.757	4.076	1	1	1	1	1	1	0.923	-	1.008	1.002	0.962	0.888
Avg.										1	1	1	1	1	1	0.856	1.004	1.012	0.973	0.947	0.856

"-" indicates single building and/or floor. "✗" represents the dataset where the cleansing algorithm was not able to find any unnecessary fingerprint.

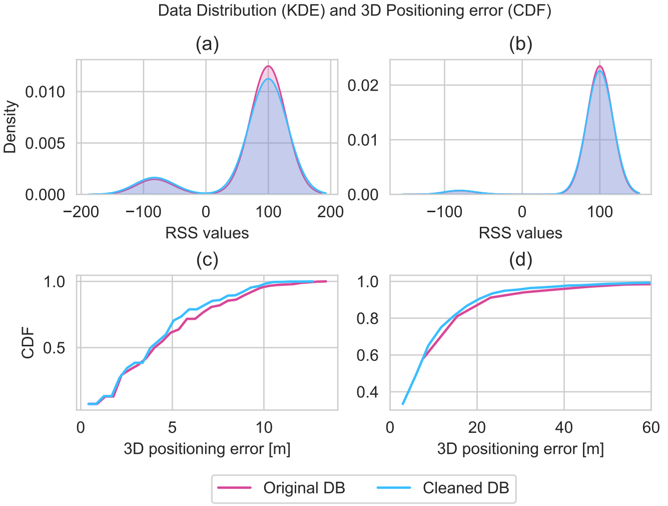

Fig. 3. LIB 2 (a, c) and UJI 1 (b, d). Top (a–b): RSS distribution after and before the data cleansing. Bottom (c–d): CDF of the 3D positioning error.

Fig. 3a shows the distribution of the RSS values using Kernel Density Estimation (KDE) in LIB 2 dataset. KDE is used to visualize the data distribution and its density. The red colour denotes the distribution of the original dataset, whereas the blue colour represents the data distribution after applying the data cleansing. In this case, the vast majority of the values are non-detected values denoted by 100. When the data cleansing algorithm is applied to the dataset, the data distribution is almost the same, but the density in the region of non-detected values was reduced, and the area of valid RSS values slightly increased. As can be observed from the Cumulative Distribution Function (CDF) plot (see Fig. 3c), the proposed data cleansing reduced the positioning error. For instance, the possibility of having a positioning error of less than $4\mathrm{m}$ is 54% before the data cleaning and 60% after it.

Similarly, Fig. 3b shows the distribution of the RSS values in UJI 1 dataset and the CDF of the 3D positioning error in the same dataset (see Fig. 3d). UJI 1–2 are the only datasets with multiple buildings (3 buildings) and floors (4–5 floors). In UJI 1 dataset, we can observe errors over $100\mathrm{m}$, whereas the maximum positioning error obtained after the data cleansing is around $88\mathrm{m}$. The same pattern can also be observed in Table I, where the normalised mean 3D positioning error was reduced by $\approx 17\%$ ($\approx 2\mathrm{m}$).

V. DISCUSSION

Ensuring the quality of the data has become an essential step to provide better analysis and, therefore, better results. Indoor positioning datasets are not an exception; data collected from differing environments may contain irrelevant observations, outliers, missing data or noisy sample that may poison the radio map. Therefore, it is crucial to "cleanse" the datasets to offer high-quality data to any model used to estimate the device position.

The proposed data cleansing algorithm offers a straightforward way of removing irrelevant fingerprints from indoor positioning radio maps without increasing the positioning error. In some cases, the proposed method also helps to provide a better position estimation, showing its potential for data cleansing in Wi-Fi fingerprinting radio maps. However, the complexity of radio maps makes it difficult to detect irrelevant data or outliers in some datasets. For instance, in TUT 4 and TUT 6, the number of unnecessary fingerprints detected was insignificant compared to the size of the dataset.

In the particular case of TUT 7, the proposed algorithm could not detect any unnecessary fingerprints. Even when the threshold was set with a minimal match percentage (less than 5%) between fingerprints, the positioning error was negatively affected.

Although the average or the maximum number of valid RSS samples can be used in the proposed algorithm, the maximum number of valid RSS samples provides better performance than the average in some of the datasets. In some cases, both the average and the maximum can offer the same results but using different thresholds. For instance, in TUT 6 with ρ equal to 5% and average method can obtain the same positioning error as the one reported in Table I.

It is important to highlight that the proposed cleansing algorithm can be complemented with other tools or algorithms to remove unnecessary fingerprints from the radio map.

VI. Conclusions

In this paper, we offer a novel and straightforward algorithm to remove unnecessary samples from Wi-Fi fingerprinting radio maps. This algorithm compares the APs in common between fingerprints to compute the match percentage between each one under predefined conditions. The evaluation comprises 14 multi-storey Wi-Fi datasets taken with different strategies in different locations aiming at obtaining generalizable results.

As a result, the proposed cleansing algorithm was able to remove unnecessary samples, reducing the size of the datasets by more than 14%, with an average improvement in the 2D positioning error of 2.7% and 5.3% in the 3D positioning error. Also, there was a slight improvement in the floor hit rate ($\approx$ 1.2% on average). Additionally, the time required for position prediction was decreased by 14%. i.e., the proposed method is able to improve all metrics.

Future work will analyze new techniques and algorithms to improve the quality of WLAN radio maps, combined with the proposed data cleansing algorithm.

References

[1] A. Ometov, V. Shubina, L. Klus, *et al.*, "A survey on wearable technology: History, state-of-the-art and current challenges," *Computer Networks*, vol. 193, p. 108 074, 2021.

[2] S. Subedi and J.-Y. Pyun, "A survey of smartphone-based indoor positioning system using rf-based wireless technologies," *Sensors*, vol. 20, no. 24, p. 7230, 2020.

[3] J. Torres-Sospedra, P. Richter, A. Moreira, *et al.*, "A comprehensive and reproducible comparison of clustering and optimization rules in wi-fi fingerprinting," *IEEE Transactions on Mobile Computing*, 2020.

[4] H.-A. Pham, T.-V. Le, *et al.*, "An improved weighted k-nearest neighbors algorithm for high accuracy in indoor localization," in *2019 25th Asia-Pacific Conference on Communications (APCC)*, IEEE, 2019, pp. 24–27.

[5] R. Klus, L. Klus, J. Talvitie, *et al.*, "Transfer learning for convolutional indoor positioning systems," in *2021 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, IEEE, 2022, pp. 1–8.

[6] M. Abid, P. Compagnon, and G. Lefebvre, "Improved cnn-based magnetic indoor positioning system using attention mechanism," in *2021 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, IEEE, 2021, pp. 1–8.

[7] S. Eisa, J. Peixoto, F. Meneses, *et al.*, "Removing useless aps and fingerprints from wifi indoor positioning radio maps," in *International Conference on Indoor Positioning and Indoor Navigation*, IEEE, 2013, pp. 1–7.

[8] L. Klus, D. Quezada-Gaibor, J. Torres-Sospedra, *et al.*, "Rss fingerprinting dataset size reduction using feature-wise adaptive k-means clustering," in *2020 12th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, IEEE, 2020, pp. 195–200.

[9] J. Talvitie, M. Renfors, M. Valkama, *et al.*, "Method and analysis of spectrally compressed radio images for mobile-centric indoor localization," *IEEE Transactions on Mobile Computing*, vol. 17, no. 4, pp. 845–858, 2017.

[10] H. Zhou and N. N. Van, "Indoor fingerprint localization based on fuzzy c-means clustering," in *2014 Sixth International Conference on Measuring Technology and Mechatronics Automation*, IEEE, 2014, pp. 337–340.

[11] Y. Lin, D. Jiang, R. Yus, *et al.*, "Locater: Cleaning wifi connectivity datasets for semantic localization," *Proc. VLDB Endow.*, vol. 14, no. 3, pp. 329–341, Nov. 2020.

[12] J. Talvitie, E. S. Lohan, and M. Renfors, "The effect of coverage gaps and measurement inaccuracies in fingerprinting based indoor localization," in *International Conference on Localization and GNSS 2014 (ICL-GNSS 2014)*, 2014, pp. 1–6.

[13] A. Khalajmehrabadi, N. Gatsis, and D. Akopian, "Structured group sparsity: A novel indoor wlan localization, outlier detection, and radio map interpolation scheme," *IEEE Transactions on Vehicular Technology*, vol. PP, Oct. 2016.

[14] N. Singh, S. Choe, and R. Punmiya, "Machine learning based indoor localization using wi-fi rssi fingerprints: An overview," *IEEE Access*, vol. 9, pp. 127 150–127 174, 2021.

[15] R. Wang, Z. Li, H. Luo, *et al.*, "A robust wi-fi fingerprint positioning algorithm using stacked denoising autoencoder and multi-layer perceptron," *Remote Sensing*, vol. 11, no. 11, 2019.

[16] W. Njima, M. Chafii, A. Nimr, *et al.*, "Deep learning based data recovery for localization," *IEEE Access*, vol. 8, pp. 175 741–175 752, 2020.

[17] W.-H. Lee, M. Ozger, U. Challita, *et al.*, "Noise learning-based denoising autoencoder," *IEEE Communications Letters*, vol. 25, no. 9, pp. 2983–2987, 2021.

[18] J. Sun, B. Wang, X. Song, *et al.*, "Data cleaning for indoor crowdsourced rssi sequences," in Aug. 2021, pp. 267–275.

[19] T. Stoyanova, F. Kerasiotis, K. Efstathiou, *et al.*, "Modeling of the rss uncertainty for rss-based outdoor localization and tracking applications in wireless sensor networks," in *2010 Fourth International Conference on Sensor Technologies and Applications*, IEEE, 2010, pp. 45–50.

[20] D. Quezada-Gaibor, L. Klus, J. Torres-Sospedra, *et al.*, *Supplementary Materials for "Data Cleansing for Indoor Positioning Wi-Fi Fingerprinting Datasets"*, version 1.0, Mar. 2022.

[21] X. Song, X. Fan, X. He, *et al.*, "Cnnloc: Deep-learning based indoor localization with wifi fingerprinting," in *2019 IEEE SmartWorld, Ubiquitous Intelligence Computing, Advanced Trusted Computing, Scalable Computing Communications, Cloud Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, 2019, pp. 589–595.

[22] P. Bahl and V. Padmanabhan, "Radar: An in-building rf-based user location and tracking system," in *Proceedings IEEE INFOCOM 2000. Conference on Computer Communications. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (Cat. No.00CH37064)*, vol. 2, 2000, 775–784 vol.2.

[23] J. Ma, X. Li, X. Tao, *et al.*, "Cluster filtered knn: A wlan-based indoor positioning scheme," in *2008 International Symposium on a World of Wireless, Mobile and Multimedia Networks*, IEEE, 2008, pp. 1–8.

[24] J. Torres-Sospedra, R. Montoliu, S. Trilles, *et al.*, "Comprehensive analysis of distance and similarity measures for wi-fi fingerprinting indoor positioning systems," *Expert Systems with Applications*, vol. 42, no. 23, pp. 9263–9278, 2015.