High-Performance Features in Generalizable Fingerprint-Based Indoor Positioning

Andrea Brunello[1], Angelo Montanari[1], Nicola Saccomanno[1(✉)], and Joaquín Torres-Sospedra[2]

[1] Data Science and Automatic Verification Lab, University of Udine, Udine, Italy
{andrea.brunello,angelo.montanari,nicola.saccomanno}@uniud.it
[2] Centro ALGORITMI, Universidade do Minho, Guimarães, Portugal
jtorres@algoritmi.uminho.pt

Abstract. In the context of WiFi fingerprint-based indoor localization, the present work systematically investigates the generalization capabilities, within a k-Nearest Neighbor framework, of meta-distances learned through a genetic programming approach, considering sixteen well-known and widely used benchmark datasets. Our study reveals clear variations in meta-distances performance, emphasizing that some training datasets lend themselves to superior generalization capabilities compared to others. We identify salient features, independent from the parameter k, that appear to contribute to such a superior performance: geographically large radio-map where the average number of access points detected per fingerprint is not too high. Finally, we propose a simple approach to combine the most successful meta-distances derived from our investigation. The latter leads to an overall improvement in positioning accuracy with respect to classical distances used the literature and the previously constructed individual meta-distances, confirming the soundness of the overall experimental workflow.

Keywords: Genetic programming · WiFi · Machine learning · Localization

1 Introduction

The digital age has made indoor localization a vital technology, widely used in many areas, ranging from logistics to emergency response systems [23]. Over the years, indoor localization has seen a variety of techniques being developed and explored [12,36] and, among these, WiFi fingerprint-based indoor positioning systems have gained considerable attention thanks to their ease of deployment [18]: WiFi access points (APs) are, nowadays, ubiquitous and ad-hoc equipment is not required to exploit them. In addition, signal strengths received from

Andrea Brunello and Nicola Saccomanno are co-first authors.

A. Zaslavsky et al. (Eds.): MobiQuitous 2023, LNICST 593, pp. 46–67, 2024.
https://doi.org/10.1007/978-3-031-63989-0_3

the APs, rather than their exact location, are enough to derive position information [36].

Classical WiFi fingerprinting uses the Received Signal Strength (RSS) from one or more APs for positioning. The overall process involves two phases. In the *offline phase*, a site survey collects RSS vectors from predetermined reference points (RPs) in the indoor environment, with each vector capturing the RSSs from visible APs at the specific RP. Fingerprints are (RSS vector, position) pairs, which are stored in a database called *radio-map*. The exploitation of the positioning system by a user happens in the *online phase*, where an RSS vector sampled at an unknown location is matched against the radio-map elements to estimate the position. In this paper, we focus on deterministic fingerprinting. It often relies on the k-Nearest Neighbor (k-NN) algorithm [2], which builds its predictions based on the k most similar candidates. The reason behind such a choice is that, even in the most recent literature, k-NN still emerges as one of the most simple and yet competitive in terms of performance approaches [1,28]. Similarity can be computed using predefined and fixed formulas such as the *Euclidean*, *Cosine*, and *Matusita* distances. Alternatively, machine learned distances or meta-distances, that may better adapt to the characteristics of the considered scenarios, can be employed [6]. While a typical distance function takes the RSS values of two fingerprints as input, a meta-distance function works at a higher level of abstraction by taking as input two or more pre-calculated distance values between these fingerprints.

Despite the growing body of research in the field [26], the issue of the generalizability of machine-learned (meta) distances used for fingerprint-based indoor positioning is still a critical concern, since the ability to accurately perform on unseen data, or generalizing, is crucial for the practical implementation of indoor positioning systems. Here, we focus on this issue. To the best of our knowledge, the only related work is one of ours [6], where we evaluate the generalization capability of meta-distances trained on the UJIIndoorLoc dataset [32] using machine learning and symbolic regression via genetic programming. We also show how the meta-distance function derived from genetic programming is the best solution, among tested ones, when used in a (k-NN) positioning framework.

Starting from [6], we consider 16 well-known and widely used indoor positioning benchmark datasets, learning a different meta-distance on each of them and then analyzing their generalization capabilities across the same sources of data. Our analyses not only unveil significant variations in the performance of the meta-distances, but they also identify specific datasets that, when used for learning a meta-distance, lead to better results. Furthermore, we shed light on the dataset characteristics that underpin these superior outcomes, providing valuable insights to the research community about the collection of novel test beds, as well as to the industrial practitioners when building a radio-map. Finally, we straightforwardly combine the most successful individual meta-distances, building a model that leads to an improvement in positioning performance, paving the way for further research in this direction. Summarizing, we cover the following:

– is there any variability in the datasets, concerning the performance of the meta-distances learned over them?

– if the answer is yes, what are the most convenient datasets to be used for meta-distance learning? Can they be characterized by any features?
– is it possible to combine the best performing meta-distances to obtain an even better performance?

The paper is organized as follows. Section 2 provides background knowledge and a description of the datasets that we consider in our study. Section 3 discusses the experiments and their outcomes. Conclusions provide an overall assessment of the work done.

2 Materials and Methods

In this section, first, we present the fingerprint distance functions under consideration; they constitute the input of the learned meta-distances, as in the work of [6]. Then, an overview of the reference datasets follows. We conclude with a brief account of the fundamental concepts of symbolic regression, evolutionary computation, and genetic programming, and of their application to our case study.

2.1 Distance Functions

Literature [31,33] reports the use of various fingerprint distance functions for indoor localization. A thorough study of their ability to maintain real-world spatial relationships and to foster positioning accuracy can be found in [25]. Here, we focus on three distances that, irrespective from the positioning task, exhibited distinct behaviors, namely, *Euclidean*, *Matusita*, and *Cosine*.

Let $AP_1, \ldots, AP_n$ be the list of all the access points that can be detected in a given scenario, let $\mathbf{p}$ and $\mathbf{q}$ be two array-like fingerprints, collected in the environment, with $|\mathbf{p}| = |\mathbf{q}| = n$, and, for all $i \leq n$, let $p_i \in \mathbf{p}$ (resp., $q_i \in \mathbf{q}$) be the RSS value (which is negative) in fingerprint $\mathbf{p}$ (resp., $\mathbf{q}$) for the access point with identifier i. The first considered distance is the *Euclidean* one:

$$d_{\text{euclidean}}(\mathbf{p}, \mathbf{q}) = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2}. \tag{1}$$

Similar, except for the square root applied to the single elements, is the *Jeffries-Matusita* distance:

$$d_{\text{matusita}}(\mathbf{p}, \mathbf{q}) = \sqrt{\sum_{i=1}^{n} (\sqrt{p_i} - \sqrt{q_i})^2}. \tag{2}$$

The third distance is the *Cosine* one:

$$d_{\text{cosine}}(\mathbf{p}, \mathbf{q}) = 1 - \frac{\sum_{i=1}^{n} p_i \cdot q_i}{\sqrt{\sum_{i=1}^{n} p_i^2} \cdot \sqrt{\sum_{i=1}^{n} q_i^2}}. \tag{3}$$

Note that, to evaluate fingerprint distances, APs not detected by fingerprints need to be dealt with, as well as RSS negative values, which may not be supported by some distance functions, like *Matusita*. To overcome these issues, a rescaling step is applied to the fingerprint values before computing distances. Here, following the suggestion in [25], we proceed as follows. Given a fingerprint, we define function *Positive*, that subtracts from each AP the minimum recorded RSS, selected from all the observations, after subtracting 1 to it (such a value is called *min*). As a result, all fingerprint values are made positive in the interval $[1, |min| + 1]$, and the lowest admissible value 0 is used to represent those APs that have not been detected in a fingerprint. Formally, let $\mathbf{p}$ be a fingerprint and i be an AP, with RSS value p_i. Then:

$$Positive_i(\mathbf{p}) = \begin{cases} (p_i - min) & \text{if AP } i \text{ is detected,} \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

Zero-to-One rescaling is defined as follows:

$$Zero\text{-}to\text{-}One_i(\mathbf{p}) = \frac{Positive_i(\mathbf{p})}{-min}. \tag{5}$$

Note that the function maps all RSS values in the range $[0, 1]$. We refer to the rescaled RSS values as RSSI (Received Signal Strength Indicator).

2.2 Datasets

In this work, we base our analyses on a selection of 16 datasets, well-known and widely used within the indoor positioning community. Table 1 reports their distinctive features, which are quite heterogeneous, distinguishing between the official training and test splits. As we will see in Sect. 3, apart from these features, which have been reported from previous literature, we computed an extended set of attributes to determine possible relationships linking them to their generalizability performance. In the following, we give a short account of the datasets.

DSI1 and DSI2 [20] have been collected in the same building at the University of Minho, Portugal. The first is obtained from the second by removing FPs sampled at the same RP. LIB1 and LIB2 [17] model a library at the Universitat Jaume I, Spain. They differ in the acquisition date: 2016 for the former, and 2017 for the latter. MAN1 and MAN2 have been collected on a single floor at the University of Mannheim, Germany [13]. Specifically, MAN2 is obtained from MAN1 by applying an additional post-processing step that averages FPs in 10 blocks of 10 fingerprints each. SIM001 is a synthetic dataset, generated according to a simple path loss model with additive Gaussian noise [33]. All TUT datasets have been collected at Tampere University, Finland. TUT1 [27], TUT3 [16], TUT4 [16] (which is TUT3 with swapped training and test sets), and TUT6 [15] all consider the same premises, and differ from one another in their collection strategies. TUT1 relies on cell averaging with a 1-meter grid size, TUT3 exploits crowdsourcing, and TUT6 is collected in a classical way (like, e.g., DSI1). As for

Table 1. Characteristics of the considered datasets

DB	Dim/Area	Setting	#u	#d	#b	#f	#r	FP x RP	#FP	#$\widetilde{\text{FP}}$	#AP	#$\widetilde{\text{AP}}$	#RP	FPρ	$\widetilde{\text{FP}}\rho$	Valid APs	Valid $\widetilde{\text{APs}}$
DSI1	100 m × 18 m	university floor	1	1	1	1	25	1 to 9	1369	348	152	125	230	0.73 ± 0.27	0.70 ± 0.27	24.8 ± 8.6	23.6 ± 7.8
DSI2	100 m × 18 m	university floor	1	1	1	1	25	1 to 5	576	348	152	125	230	0.31 ± 0.11	0.29 ± 0.12	24.8 ± 8.6	23.6 ± 7.8
LIB1	15 m × 10 m	two-floor library	1	1	1	2	2	12 or 36	576	3120	77	174	48	2.42 ± 0.59	2.41 ± 0.70	14.2 ± 4.6	21.0 ± 6.3
LIB2	15 m × 10 m	two-floor library	1	1	1	2	2	12 or 36	576	3120	119	165	48	2.42 ± 0.59	2.41 ± 0.70	21.6 ± 7.0	18.8 ± 5.1
MAN1	50 m × 36 m	office hallway	1	1	1	1	–	10 or 110	14300	460	27	24	130	20.88 ± 4.48	20.55 ± 5.12	10.3 ± 2.5	10.5 ± 2.5
MAN2	50 m × 36 m	office hallway	1	1	1	1	–	10	1300	460	26	25	130	1.90 ± 0.41	1.87 ± 0.47	14.1 ± 2.9	14.1 ± 2.7
SIM001	50 m × 20 m	simulated	–	–	1	1	–	1 or 10	10710	1000	8	8	1071	8.86 ± 1.80	8.86 ± 1.50	8.0 ± 0.0	8.0 ± 0.0
TUT1	124 m × 57 m	university building	1	1	1	4	<822	1 to 17	1476	490	309	183	1476	0.33 ± 0.13	0.41 ± 0.10	32.9 ± 12.1	25.0 ± 7.3
TUT2	145 m × 88 m	university building	1	1	1	3	NA	1	584	176	353	248	584	0.11 ± 0.05	0.12 ± 0.05	47.9 ± 19.0	21.9 ± 7.1
TUT3	130 m × 62 m	crowdsourced	8	21	1	5	822	1 to 14	697	3951	779	989	694	0.17 ± 0.07	0.16 ± 0.08	48.0 ± 38.4	49.7 ± 38.7
TUT4	130 m × 62 m	crowdsourced	8	21	1	5	822	1 to 14	3951	697	989	779	3843	0.96 ± 0.50	0.91 ± 0.48	49.7 ± 38.7	48.0 ± 38.4
TUT5	85 m × 145 m	university building	NA	NA	1	3	NA	1	446	982	489	296	446	0.08 ± 0.03	0.07 ± 0.02	42.5 ± 21.1	34.8 ± 13.5
TUT6	135 m × 62 m	university building	NA	NA	1	4	<822	1	3116	7269	562	566	3116	0.65 ± 0.31	0.63 ± 0.31	34.9 ± 15.9	34.7 ± 15.9
TUT7	88 m × 137 m	university building	NA	NA	1	3	NA	1	2787	6504	767	770	2787	0.48 ± 0.30	0.47 ± 0.29	27.0 ± 11.4	27.1 ± 11.1
UJI1	108703 m^2	university campus	19	25	3	4 or 5	≫933	1 to 80	19861	1111	465	367	933	2.46 ± 1.84	2.45 ± 1.62	18.1 ± 7.3	16.5 ± 6.9
UTS	44000 m^2	university building	1	1	1	16	NA	1 to 35	9108	388	557	500	1466	0.46 ± 0.29	0.51 ± 0.3	35.1 ± 13.5	42.7 ± 10.7

DB = dataset name; Dim/Area = size of the premises; Setting = collection environment; #u = number of involved users; #d = number of employed devices; #b = number of buildings; #f = number of floors; #r = number of rooms; FP x RP = fingerprints sampled per position; #FP = number of fingerprints; #AP = number of APs seen at least one time; #RP = number of distinct positions (x, y, z); FP_ρ = average FP density within a 5 m radius from each RP; Valid APs = average number of detected APs per FP; superscript ˜denotes test data, – denotes meaningless information, NA denotes not available information.

TUT2 [27], TUT5 [24], and TUT7 [15], they have been independently collected in the same university building. TUT5 follows an approach similar to TUT1, considering cell averaging with a 5-meter grid size, while the way in which TUT2 and TUT7 have been collected is the same as TUT6. UJI1 [32] has been again collected at the Universitat Jaume I. It is the largest and most complex dataset among the considered ones, as it is multi-device and the only one that considers a multi-building scenario. Finally, UTS [29] describes the FEIT Building at the Sidney University of Technology, Australia (a single building with a high number of floors).

2.3 Genetic Programming for Symbolic Regression

Drawing inspiration from [6], we rely on a symbolic regression approach, based on genetic programming, to learn a meta-distance from each of the 16 considered datasets, starting from the three distances *Euclidean*, *Cosine*, and *Matusita*. The aim is to use the meta-distance for fingerprint distance evaluation, to support k-NN based positioning.

Symbolic regression (SR) is an optimization technique that explores the space of mathematical expressions to find an optimal model, e.g., in terms of predictive capacity, for a given dataset. Unlike classical regression, which optimizes the parameters of a pre-defined model structure, SR learns both the model structure and its parameters [21]. The SR algorithm typically begins with a set of fully random expressions, formed from a combination of mathematical operators, constants, and variables, rather than a specified starting model. Each expression can be visually depicted as a tree, where leaves represent input values (here, either

constant values or the results of the three considered fingerprinting distance functions) and the root represents the output value.

We solve the aforementioned optimization problem through the *Evolutionary algorithm* outlined in [6], that implements a genetic programming task. Evolutionary algorithms (EAs) are population-based metaheuristics inspired by biological evolution and genetics, that proved themselves effective in solving combinatorial optimization problems [9]. Unlike random search, EAs leverage historical information to guide the exploration towards the most promising search space regions, by simulating natural adaptive evolution processes. In this context, the evolution process can be considered as a sequence of generations. Each generation in the EA consists of a population of candidate solutions (aka individuals) which undergo processes analogous to those in biological evolution, such as selection, crossover, and mutation. During each generation, the selection process favours individuals with higher fitness values, thereby simulating survival of the fittest. These selected individuals then proceed to crossover and mutation stages. As generations advance, the algorithm converges towards an optimal or near-optimal solution, guided by the continuous improvement of individual fitness within the population.

In the context of EA, *Genetic Programming* (GP) is a technique that evolves programs from a population of random solutions [22]. In GP, an individual is represented as a computation tree with internal nodes encoding primitives (like mathematical operators) and leaves indicating input values. The output is given by the root-encoded primitive. Due to its formulation, GP is well-suited for symbolic regression tasks.

Meta-distance Learning Task. Given that the meta-distance learning task is at the core of our work, in the following we briefly summarize the evolutionary algorithm implementation we adopted. For further details, we refer the interested reader to our previous work [6]. The proposed solution is based on the DEAP (Distributed Evolutionary Algorithms in Python) framework [10], which facilitates the quick prototyping of custom evolutionary algorithms. DEAP's core comprises two structures: the *creator* that aids in creating individuals and populations of any data structure, thus supporting all kinds of evolutionary algorithms, and the *toolbox*, a container for operators which can be used for operations like genetic mutation and individual selection.

Overall Task. The objective is to symbolically learn a distance function between two fingerprints. This function takes as input the results of the *Euclidean*, *Cosine*, and *Matusita* distance calculations made over the two fingerprints. Since the learned distance is an algebraic combination of values obtained through the traditional distances, we refer to the learned distance as a *meta-distance* (a distance over distances) in this work. For instance, Fig. 1 depicts the meta-distance learned, in [6], over the dataset UJI1, where the leaves either represent a fixed numerical coefficient or the distance value provided by the Euclidean distance (EUC), Cosine distance (COS) or Matusita distance (MAT); ST (·) nodes represent parameterized instantiations of the ST (A) subtree. Please, note that the

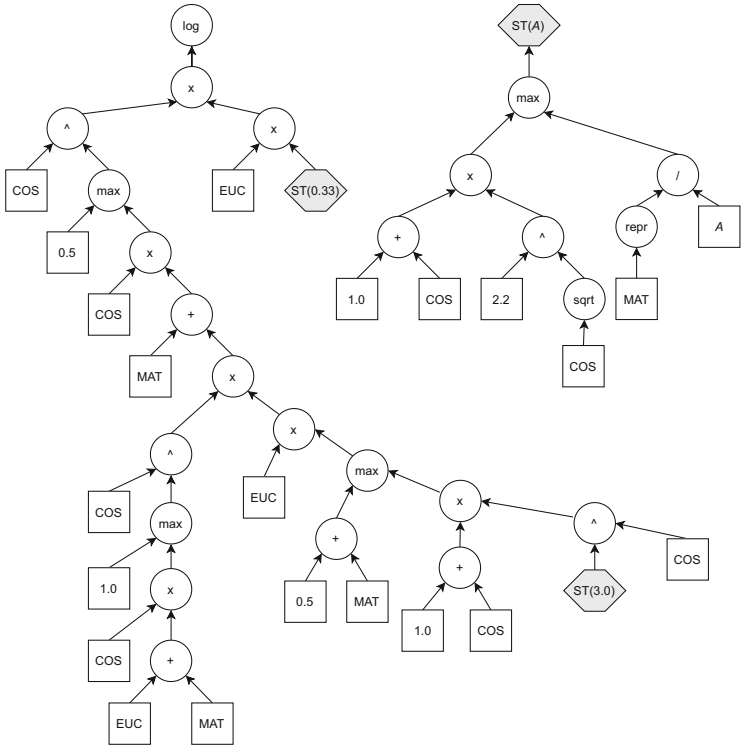


Fig. 1. The computation tree generated by the evolutionary algorithm when applied on dataset UJI1, where $ST(\cdot)$ nodes represent parameterized instantiations of the $ST(A)$ subtree. The meta-distance makes use of *Euclidean* (EUC), *Cosine* (COS), and *Matusita* (MAT) distance functions (image from [6]).

meta-distance do not use explicitly any RSSI values. Of course, such a meta-distance can then be used within any positioning framework, as it happens with classical distances. Here, as already mentioned, we will consider a k-NN approach. Figure 2 provides a schematic account of the whole genetic algorithm operation, whose single blocks are described in the following.

Population and Initialization. Each individual in the population is a computation tree that combines the values derived from *Euclidean*, *Matusita*, and *Cosine* fingerprint distances, to produce a fingerprint meta-distance. The population is initialized using DEAP's *genHalfAndHalf* method, producing random computation trees with a maximum height of 6 [14]. Half of the time, a tree with leaves at the same depth is generated; otherwise, leaves may have varying depths.

Primitives. A computation tree node can represent a single value - either one of the chosen fingerprint distance functions, a DEAP's *EphemeralConstant* (a random value from $[-10, 10]$), or a constant from $-1, 0.5, 1, 2, 3$ (square boxes

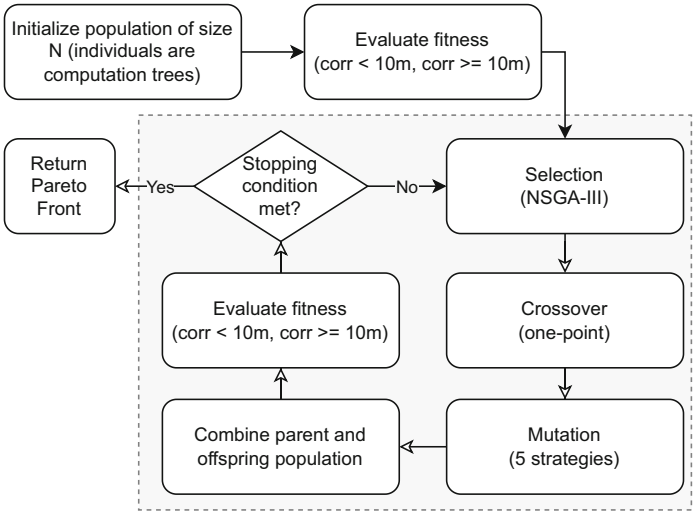


Fig. 2. Overview of the evolutionary algorithm, where the term *corr* stands for the Pearson correlation between distances in the fingerprint and real-world space. The shaded area includes the steps performed during an EA generation.

in Fig. 1). It can also denote a unary function applied to a value x, chosen from $\{log_e(x),\ \sqrt{x},\ x^2,\ |x|,\ -x,\ 1/x$ (reciprocal, repr in Fig. 1), $sin(x),\ cos(x)\}$ as well as a binary function applied to a pair of values x, y, from $x+y, x-y, x \cdot y, x/y, x^y,\ max(x,y), min(x,y)$ (circles with one or two inputs respectively in Fig. 1). To prevent invalid calculations, such as $x/0$, each function has a 'protected' implementation that substitutes NaN values with 0, positive infinite values with 1, and negative infinite values with -1.

Training Data. Fingerprint pairs are extracted from a given dataset's official training split. For each pair, *Euclidean*, *Matusita*, and *Cosine* fingerprint distance functions are calculated, along with the real-world 3D Euclidean distance between their associated locations.

Fitness Function and Best Solution Extraction. A bi-objective fitness function is used. Given an individual of the population, each objective determines the Pearson correlation between the distance values computed by such an individual over all the fingerprint pairs and the corresponding real-world Euclidean distances where such fingerprints were collected (thus, no positioning task is considered within the EA). Specifically, the first objective considers fingerprint pairs with real-world distances in the $[0, 10)$ meters range, while the second handles distances in the $[10, +\infty)$ meters range, aligning with the results presented in [25]. The two objectives are to be maximized. In computing the fitness, counter-overfitting measures are undertaken, based on the concept of training instances rotation [11]. The rationale behind this is that the fitness function should foster

the creation of individuals that encode fingerprint distances that, to the greatest extent possible, retain the real-world distances within the fingerprint space. This ensures that the relationships and structure present in the real-world are effectively mirrored within the fingerprint-based representation.

Given the two objectives, no unique best solution can be directly selected from a population. Instead, a *Pareto front* of optimal solutions is identified, including all *non-dominated* ones.[1] From the front, we select the solution with the highest hypervolume[2], among those that perform better than *Cosine* with respect to both the objectives.

Termination Criteria. They include an upper limit on the number of generations (500) and an *early stopping* strategy based on the increase of the *hypervolume* measure along the generations.

Crossover and Mutation. Given two parent solutions, two offspring are produced via one-point crossover (DEAP's *cxOnePoint*), which selects a node within each individual and exchanges the subtrees originating from that point. In addition, several mutation operators from DEAP are considered, each selected uniformly at random. These include *mutUniform*, which swaps a node in an individual with a randomly generated mathematical expression; *mutInsert*, which adds a new subtree at a random position in an individual; *mutNodeReplacement*, which replaces a randomly selected node in the individual; *mutEphemeral*, which changes the value of a single constant within an individual; and *mutShrink*, which randomly selects a subtree in an individual and replaces it with one of the subtree's arguments, also randomly chosen. Note that, in performing both crossover and mutation, anti-bloat strategies that limit the height of the generated computation trees are employed [14].

Selection. In order to foster diversity in the population, we adopt the elitist strategy implemented in NSGA-III [8], based on the concepts of *reference points* and *niche preservation*.

3 Experimental Evaluation

As already stated, we consider 16 indoor positioning datasets, each partitioned into an official training and test split. On each training set, following the symbolic regression approach described in [6], we learned a meta-distance (namely, MD_TRAINDATASETNAME) for fingerprint distance evaluation that combines the results computed by *Euclidean*, *Cosine*, and *Matusita* distances.

Then, for positioning and evaluation purposes, we rely on k-NN with $k = 3$ and the 16 above-mentioned learned meta-distances. For comparison purposes,

[1] A set $\mathcal{S}$ of solutions is said to be *non-dominated* if no solution in $\mathcal{S}$ is entirely superior to any other in terms of all objectives.

[2] The hypervolume measures the volume of the search space, bounded by a given reference point, that is weakly dominated by the points on the Pareto front [7].

the individual *Euclidean*, *Cosine* and *Matusita* distances are also taken into consideration in the evaluation. Thus, for each dataset, using the instances from the corresponding training split as the radio-map, we have run 19 (16 + 3) k-NN models. We collect the positioning performance in terms of success rate (fraction of instances for which the building and floor were correctly identified) and average positioning error in meters.

With those results at our disposal, in the subsequent sections, we will address the following research questions:

RQ1 Is there any variability in the datasets, concerning the performance of the meta-distances?
RQ2 What are the most convenient datasets to be used for meta-distance learning?
RQ3 Are there any features that characterize the datasets that perform well for meta-distance learning?
RQ4 Is it possible to combine the best performing meta-distances to obtain an even better performance?

Note that, to make our results comparable with [6], we have maintained a consistent choice of three neighbors ($k = 3$) for all results presented hereafter. While our primary objective in this research is comparative analysis rather than the pursuit of optimal positioning performance, we still recognize the potential limitation inherent in maintaining a fixed value for k. Consequently, Sect. 3.5 is devoted to comprehensively addressing this particular issue.

3.1 RQ1: Variability in the Datasets

When assessing the performance of a positioning system, a high success rate is desirable, together with a low positioning error. The two dimensions are strongly intertwined, as one cannot be evaluated aside from the other. However, a unanimous consensus about how to evaluate indoor positioning systems with a single measure is still to be reached [34]. In light of this, in analogy to [19,34], we proceed as follows. Given a positioning approach p evaluated over a dataset, we start from its average positioning error $err(p)$, calculated over all instances. Then, we consider the fraction of instances which were predicted a wrong building $w_{bd}(p)$ and a wrong floor $w_{fl}(p)$. Finally, we adjust the average positioning error:

$$err_{adj}(p) = err(p) + w_{bd}(p) \cdot dist_{p50} + w_{fl}(p) \cdot fl_{dist_{p50}},$$

where, for the given dataset, $dist_{p50}$ and $fl_{dist_{p50}}$ represent the median of the distances among all radio-map fingerprints and the median of the distances among radio-map fingerprints on a same floor of a same building, respectively. The rationale, without any other information on the topology of the premises, is that if an user were randomly positioned in any location within a floor or building, the worst-case situation would find he/she at one end and the staircase (or the correct building) at the other, necessitating a traversal of the entire floor or scenario. Conversely, the best-case situation would place the user right next to

a staircase or the correct building. Averaging these two extremities provides a median or most probable estimate, hence the half-floor or half-scenario traversal. In essence, by using this average, a balanced viewpoint that is neither overly optimistic nor pessimistic is considered.

The above described methodology allows us to easily compare approaches over the same dataset. Now, to make such a comparison fair among datasets, it should be considered that reducing the positioning error of 0.5 m may be negligible in some settings, and a considerable achievement in others. Thus, we rescale $err_{adj}(p)$ on the basis of the performance exhibited by the individual classical distances *Euclidean*, *Cosine* and *Matusita* over the same dataset, obtaining our final score:

$$score(p) = \frac{err_{adj}(c_{min}) - err_{adj}(p)}{err_{adj}(c_{max}) - err_{adj}(c_{min})},$$

where c_{min} and c_{max} are the lowest and the highest err_{adj} among the classical distances. The idea is that, in a dataset where it is difficult to reduce the positioning error, all classical distances should perform in a similar way. As for the subtraction at the numerator, it gives a positive score to an improvement over the best performing classical distance.

We are now ready to explore the results of our experiments. To begin with, Fig. 3 reports, for each dataset, the boxplot and the scores reported by the 16 meta-distances. The single scores are represented by a colored dot, being the learned meta-distance for the dataset highlighted with a cross (e.g., MD_UTS applied on UTS). Boxplots[3] as well as single data points are shown. Datasets are sorted according to the variability exhibited in the scores.

As it can be seen, especially for the datasets TUT6, TUT4, UTS, and TUT7 the choice of the meta-distance seems to play a major role when it comes to positioning performance, given the high variability exhibited by the scores. This can be easily explained for UTS, as it models a very rich scenario that, from an empirical point of view, seems more difficult to deal with than another complex dataset, UJI1.

Usually, the distance derived from a training set achieves the highest rank for its corresponding dataset (e.g., MD_UJI1 is top ranked for UJI1). Interestingly, there are some notable exceptions. For some datasets, like LIB1, the performance from its own training set (represented by a cross) is not the best. In fact, it's near the 1st quartile, indicating other training sets might offer better performance in these cases. Note also how LIB1 and LIB2 exhibit a rather different score distribution, which may be tied to peculiarities connected to their time of collection. Different behaviours are exhibited by MAN1 and MAN2 as well, probably due to the post-processing step employed in the latter. DSI1 and DSI2 are instead quite similar, and this is not surprising as the former is obtained from the latter just by removing redundant fingerprints. Finally, TUT datasets exhibit rather het-

[3] As in all following graphs, boxes extend from the 1st to the 3rd quartile values of the data, with a line at the median. Whiskers extend to the smallest and largest observations which are not outliers (assuming 1.5 times the interquartile range).

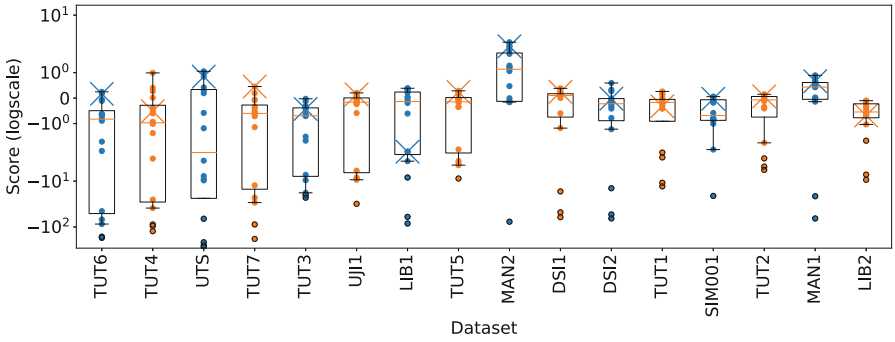


Fig. 3. Score values from the 16 learned meta-distances across each dataset, with higher scores being better. The X-axis is sorted based on score variability (higher - left, lower - right). Meta-distances derived from their respective datasets are marked with a cross.

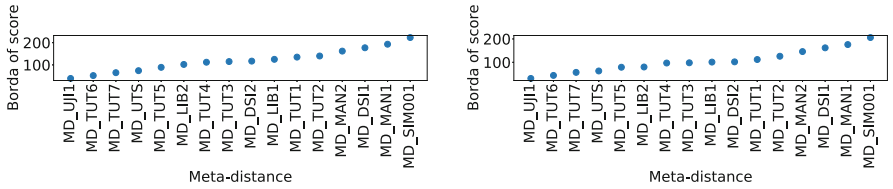
erogeneous behaviours, due to their different collection strategies and considered premises.

Answer to RQ1: there is indeed variability in the datasets concerning the meta-distances performance. For some of them, due to their peculiarities, the choice of the meta-distance severely affects the positioning outcome.

3.2 RQ2: Best Performing Datasets for Meta-distance Learning

Given the answer to *RQ1*, we now investigate which dataset(s) perform best when it comes to the meta-distance learning task. We proceed as follows. For each dataset, we sort the meta-distances according to their score, from the highest (best performance, rank position 0) to the lowest (worst performance, rank position 15) one. Then, we calculate the Borda count [4] pertaining to each meta-distance, where the voting contributions are expressed by the positions in the ranks (thus, a lower Borda corresponds to a better overall performance, and the highest/worst possible Borda is $15 \times 16 = 240$).

Figure 4 reports the results obtained for each trained meta-distance, considering all test datasets (Fig. 4a) as well as excluding the test split related to the training data of each meta-distance (e.g., MD_UJI1 is not tested on UJI1, Fig. 4b). In each graph, meta-distances are sorted from the lowest (best performing) to the highest (worst performing) Borda count. As it can be seen, MD_UJI1 performs the best, probably thanks to the complexity and richness of the premises considered in UJI1. The worst result is by far that of MD_SIM001, which raises concerns about the adequacy of the artificial generation strategy used to create the corresponding dataset. Further studies in this direction are therefore warranted, considering other synthetic datasets and approaches. Focusing on MD_DSI1 and MD_DSI2 it emerges that, at least for learning purposes, removing redundant instances brings to adverse results. Comparing MD_MAN1 and MD_MAN2, instead, the post-processing step carried out in



(a) All test datasets. (b) Without same test dataset.

Fig. 4. Borda counts for learned meta-distances. Lower is better.

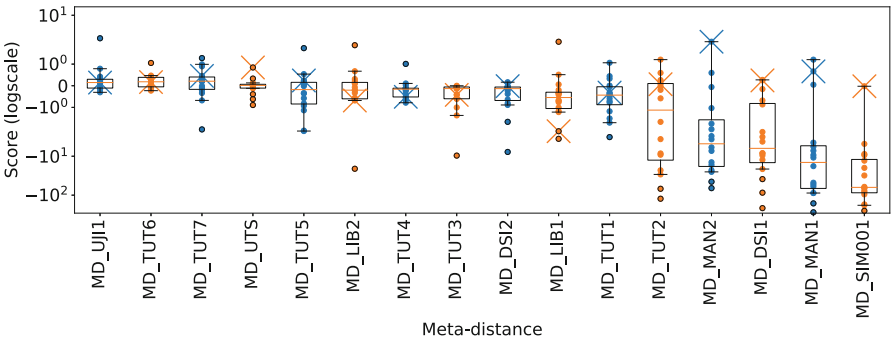


Fig. 5. Score values for each meta-distance, higher is better. Results of the meta-distances when applied on the same test set are highlighted with a cross.

MAN2 (based on multiple fingerprint average) seems to be effective. Finally, observe how the order meta-distances seems to be rather stable between the two pictures (with the exception of a close swap between MD_DSI2 and MD_LIB1), meaning that the generalization capabilities of the meta-distances are not confounded/contaminated by the presence of their corresponding test datasets. For this reason, for the remaining part of the analyses, we are going to consider the Bordas of Fig. 4a.

After the Bordas overview, we now consider some more detailed results. Figure 5 shows all scores grouped by meta-distance, the latter being sorted according to their Borda count of Fig. 4a. Results over the same dataset are highlighted with a cross. Here, the UJI1-based meta-distance emerges as the best, i.e., MD_UJI1 fits generally well in all datasets, closely followed by MD_TUT6, MD_TUT7, and MD_UTS. Meta-distances MD_TUT2, MD_MAN2, MD_DSI1, MD_MAN1, and MD_SIM001 are instead clearly under-performing. Interestingly, MD_TUT3 and MD_TUT4 behave similarly, with the former only marginally worse, despite the corresponding datasets including far less FPs (recall that the two datasets just swap their training and test splits).

To formally determine which learned meta-distance performed better than the others, we used a statistical procedure based on the pairwise evaluation of Wilcoxon signed-rank test [35], with Benjamini-Hochberg (BH) procedure to

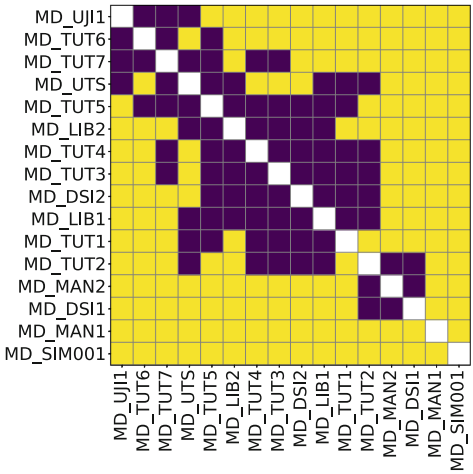


Fig. 6. Outcome of the pairwise statistical procedure based on Wilcoxon signed-rank (BH correction, α=0.05). White: same dataset; Dark: no evidence that $meta\text{-}distance_{row} \neq meta\text{-}distance_{column}$; Yellow: evidence that $meta\text{-}distance_{row}$ has a different behaviour than $meta\text{-}distance_{column}$.

account for multiple tests [3], considering a false discovery control level $\alpha = 0.05$. Figure 6 reports the overall results. Notice how the matrix confirms our previous intuitions: MD_UJI1's performance cannot be considered to be statistically different from that of MD_TUT6, MD_TUT7, and MD_UTS, and is instead superior with respect to the remaining meta-distances (considering the order provided by Fig. 4a). MD_SIM001 is statistically different (and inferior, again considering the order of Fig. 4a) with respect to all the other meta-distances. Finally, the difference between MD_TUT3 and MD_TUT4 is not statistically significant; this also holds between MD_TUT1 and MD_TUT5, which are indeed very similar considering the collection strategies and modeled premises of the corresponding datasets.

Answer to RQ2: there is a clear difference between the training sets when it comes to the performance of the meta-distances learned over them. Datasets UJI1, TUT6, TUT7, and UTS are better to generate generalizable meta-distances than the others; SIM001 is the worst one.

3.3 RQ3: Characterization of Well Performing Training Sets

Since the dataset used for meta-distance learning matters, what are the key features characterizing the well performing ones? To answer that, we define a set of attributes over the training splits of the 16 datasets. We start from the features of Table 1, disregarding those with a too-low variance, such as *#b,* or those with unknown values, like *#r*. In addition, to better describe some of the attributes, we calculate statistics over them, i.e., percentiles (perc), mean, and standard

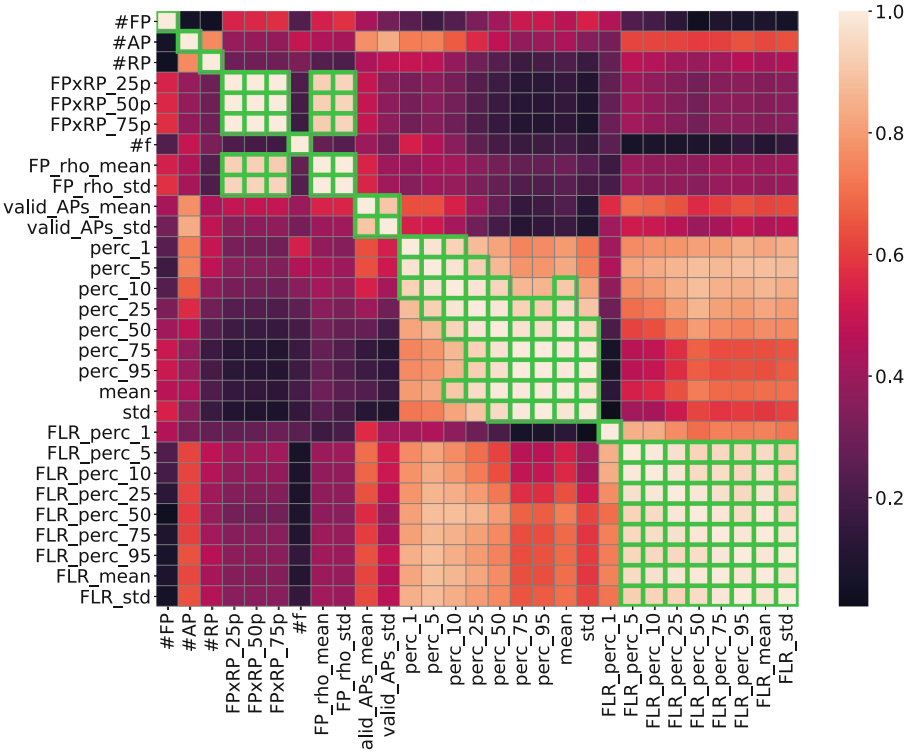


Fig. 7. Attributes and their Pearson correlations (color: darker=lower correlation; lighter=higher correlation). Correlations ≥ 0.9 are highlighted in green. (Color figure online)

deviation (std). Finally, we add features pertaining to the spatial characteristics of the radio-maps: 1st, 5th, 10th, 25th, 50th, 75th, 95th percentiles, mean, and standard deviation of the distances between the collected fingerprints, both overall and considering those on a same floor.

Figure 7 provides an overview of the considered attributes, with their pairwise Pearson correlations (colors). Correlations ≥ 0.9 are highlighted in green. The matrix is rather self-explanatory. As an example, it shows that the attributes related to the radio-map fingerprint distance values over a same floor are highly correlated, with the notable exception of the 1st percentile (robust approximation for the *minimum* value).

Before proceeding with our analyses, given the evident redundancies, we perform a feature selection phase consisting of two steps. First, we iteratively remove attributes so to break all correlations ≥ 0.9, i.e., near collinearities. Then, to solve possible multi-collinearity issues, we rely on the Variance Inflation Factor (VIF) to filter the remaining features, considering, as it is typically done [5], a threshold of 10. We obtain the following subset of attributes: *#AP*, *#RP*, *#f*, *FP_rho_mean*, *valid_APs_mean*, *perc_50*, and *FLR_perc_1*.

Analyzing how, for each dataset, the value of the selected attributes relates to the Borda counts of the corresponding meta-distances, some relationships emerge (see Fig. 8). Attributes *FP_rho_mean* and *FLR_perc_1* exhibit a weak correlation. As for the remaining ones, their increase seems to be connected with a decrease in the Borda scores and, thus, better learned meta-distance performance.

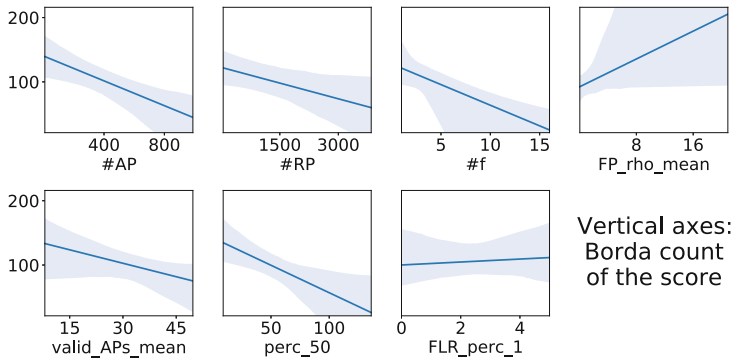


Fig. 8. Different attributes vs. meta-distances Borda count. Lines have been obtained through linear interpolation of the 16 points pertaining to the different training datasets. The 95% confidence interval is also depicted.

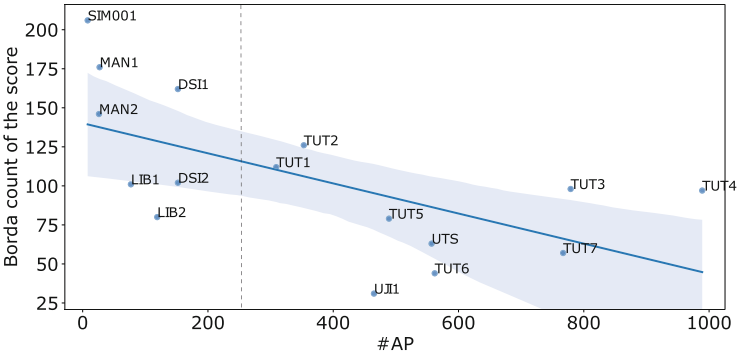


Fig. 9. Number of APs seen at least one time vs. meta-distances Borda count. The solid line has been obtained through linear interpolation. The 95% confidence interval is also depicted. The dashed line separates two groups of datasets according to their number of APs.

Let us now focus on *#AP*. Figure 9 shows that, besides the previously-described general trend, some datasets have a divergent behaviour, as in the case of DSI1 and LIB2. The two have a very similar number of APs, but the corresponding meta-distances are characterized by rather different Bordas. It might be that such discrepancies are explained by taking into account more than one attribute. To verify that, we build a Sklearn's *DecisionTreeClassifier* which,

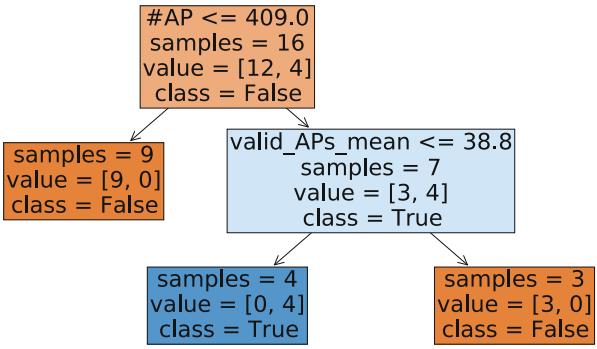


Fig. 10. Decision tree that predicts good vs bad datasets.

based on the selected features, tries to discriminate the datasets into two classes: those that in Fig. 5 led to a learned meta-distance with a median score greater than zero (class = True) and the others (class = False). Intuitively, the former represents good datasets, that are expected to produce meta-distances that perform better, on average, than the best classical distances, namely, UJI1, TUT6, TUT7, and UTS. Figure 10 shows the generated decision tree, which achieves an accuracy of 100% and, thus, perfectly explains the instances.[4] Interestingly, good datasets are characterized by a rule of the form *"in their radio-map, the number of APs seen at least one time is large enough, and the average number of APs detected per fingerprint is not too high"*. The first condition is strongly linked to the size of the geographical area covered by the radio-map: indeed, ordering the datasets by $\#AP$ (i.e., scanning Fig. 9 left to right), it can be noticed that UJI1, UTS, and all TUT datasets are clearly distinguished with respect to DSI, LIB, MAN, and SIM001 datasets (i.e., first considerable gap according to x-axis in Fig. 9, highlighted by a dashed vertical line). This is a sensible result, as larger radio-maps are expected to represent more heterogeneous scenarios, which can in turn support the meta-distance learning tasks. As for the other condition, it seems, at first glance, to be linked with the *curse of dimensionality*: distances may be less effective when working with vectors composed of several non-zero values, producing less meaningful distance evaluations that hinder the meta-distance learning phase. From a broader perspective, the learning degradation is supposed to happen also when a large number of common APs is witnessed in several, spatially dispersed fingerprints. This point, pertaining to the APs distribution among fingerprints and spatial locations, seems to be linked to the concept of information entropy, and it surely deserves attention in future work.

Answer to RQ3: the well performing training sets are characterized by a geographically large radio-map where the average number of APs detected per fin-

[4] Note that here we use the decision tree model to perform *explanation*, not *predict* over new instances: thus, we are not interested in its generalization capability.

gerprint is not too high. Notably, the sheer number of fingerprints is not enough to characterize a high-quality dataset.

3.4 RQ4: Meta-distances Combination

In this section, we build a straightforward combination of the meta-distances based on UJI1, TUT6, TUT7, and UTS datasets (i.e., MD_UJI1, MD_TUT6, MD_TUT7 and MD_UTS), which we determined to be the best-performing ones. We proceed as follows. Let us recall that meta-distances are used within a 3-NN algorithm which, given a test set fingerprint, determines the 3 closest fingerprints in the corresponding training set radio-map according to a given distance function. Thus, considering four meta-distances, each test set instance has four distances with respect to every radio-map fingerprint. To derive a single value, we simply calculate the l_2-norm of the vector composed of the four distances. The underlying reason is that the different meta-distances, being trained independently on different datasets, produce distance values with differing ranges. Utilizing the l_2-norm allows us to seamlessly combine them without being affected by such range discrepancies.

To assess the achieved localization performance, results in terms of success rate and positioning error are reported in Table 2. Here, we consider also distance *Cosine*, since it exhibited the best positioning behavior in [25], and the meta-distances based on UJI1, i.e., the winner of our previous Borda score experiments, TUT6, TUT7, and UTS, i.e., the others involved in the combination process. In the table, the number of times each method scored as the best one (i.e., the row denoted with "# 1st") is determined separately for the two evaluation parameters. As it can be noticed, the simple combination already leads to an overall better positioning performance, given that the combined metric ("Comb."in Table 2) ranks first with the highest frequency. This paves the way for further research to enhance the performance of localization systems, possibly by means of more sophisticated ensemble techniques.

Answer to RQ4: Despite the very simple combination strategy, a gain in performance is obtained with respect to the *Cosine* distance and all the single meta-distances involved in the combination.

3.5 On the Number of k Nearest Neighbors

As previously mentioned, all the results obtained so far in this study were based on a fixed number of $k = 3$ neighbors within a k-NN framework. It is generally understood that different datasets may require a specific number of neighbors k for optimal positioning performance (see, for instance, [30]). However, our primary objective here is to examine the relative performance of the meta-distances across various datasets, rather than to achieve the absolute best in localization accuracy. Therefore, for our research objectives, it is sufficient to demonstrate that, even with a different k value, the observed similarities and differences among the meta-distances persist.

Table 2. Positioning performance of meta-distances combination

Dataset	Success rate						Avg Positioning Error [m]					
	Cosine	MD_TUT6	MD_TUT7	MD_UJI1	MD_UTS	Comb	Cosine	MD_TUT6	MD_TUT7	MD_UJI1	MD_UTS	Comb
DSI1	–	–	–	–	–	–	5.4187	5.1501	5.2443	5.1711	5.3266	**4.9248**
DSI2	–	–	–	–	–	–	4.7864	4.9256	4.9169	4.7817	**4.5835**	4.6458
LIB1	**0.9990**	0.9987	**0.9990**	**0.9990**	0.9984	0.9984	2.7940	2.7214	2.7241	2.7330	2.7759	**2.6910**
LIB2	0.9997	**1.0000**	0.9968	0.9997	0.9984	**1.0000**	2.8403	3.0970	3.2323	**2.8286**	2.9551	3.0229
MAN1	–	–	–	–	–	–	2.7242	2.5560	**2.3733**	2.5517	2.7136	2.4565
MAN2	–	–	–	–	–	–	2.6138	2.4425	2.2171	2.0456	2.5752	**1.9692**
SIM001	–	–	–	–	–	–	2.9739	2.7946	3.2019	2.9579	2.9503	**2.7907**
TUT1	0.9061	0.9061	0.9061	**0.9102**	0.9041	0.9082	6.4019	6.8945	7.0979	6.2686	**6.1739**	6.3588
TUT2	0.8864	0.8807	0.8466	**0.9318**	0.8977	0.8977	11.1418	9.4256	9.5493	**9.0762**	10.7162	10.1477
TUT3	0.8932	0.9015	0.8884	0.8881	0.8927	**0.9036**	**7.8789**	7.9837	8.2745	8.3370	8.2318	8.1098
TUT4	0.9498	0.9584	**0.9656**	0.9555	0.9527	0.9584	5.5748	5.6628	5.5302	5.6289	5.6651	**5.4385**
TUT5	0.8921	0.9318	0.9226	**0.9593**	0.8870	0.9155	6.3685	**5.8878**	5.9721	6.4486	6.3387	5.9040
TUT6	0.9986	0.9996	0.9997	0.9996	0.9975	**1.0000**	2.4867	2.5008	**2.4814**	2.7035	2.6469	2.5004
TUT7	0.9906	**0.9926**	0.9924	0.9903	0.9889	0.9917	2.5727	2.5904	**2.3249**	2.6483	2.7153	2.4988
UJI1	0.9199	0.9199	0.9145	0.9226	**0.9298**	0.9217	7.5601	**6.7333**	7.1024	6.8471	7.4413	6.8007
UTS	0.9235	0.9367	0.9367	0.9261	0.9340	**0.9446**	7.4480	7.2641	7.6870	7.8682	**7.1957**	7.4349
# 1st	1	2	2	**4**	1	**4**	1	2	3	2	3	**5**

Table 3. Pearson correlations between the scores, varying k

k	1	3*	5	7	9	11
Correlation	0.9938	1.0	0.9994	0.9985	0.9977	0.9975
P-value	1.38e−244	0	0	0	1.27e−298	2.23e−293

*: For validation purposes, we also report the correlation between the 3-NN scores and themselves.

To do that, we proceed as follows. We rerun the experimental procedure outlined in Sect. 3, applying the same set $k = \{1, 3, 5, 7, 9, 11\}$ employed in [30]. For each k, we collect the *scores* of each meta-distance with respect to every test set. Then, for each k, we sort the scores according to the meta-distance and test set, obtaining k lists ordered in the same fashion. As the final step, we calculate the Pearson correlation between the list corresponding to $k = 3$ and every other list. Results are reported in Table 3. As it can be seen, all correlation values are statistically significant and exceed 0.99, meaning that the relationships between the scores are linearly preserved when changing the value of k.

This should not come as a surprise, as the meta-distances were trained by means of the evolutionary algorithm described in Sect. 2.3, optimizing the correlation between the fingerprint distance values and those of the corresponding real-world locations, irrespective of any specific positioning task. This confirms the intuition reported in [6]: optimizing the correlation, or in other words, learning to retain real-world distances within the fingerprint space, proves to be effective, regardless of the specific downstream tasks (such as positioning) to be subsequently undertaken.

4 Conclusions

In this work, we systematically investigated, based on 16 well-known datasets, the generalization capabilities of meta-distances learned through a genetic programming approach in the context of WiFi fingerprint-based indoor positioning. We found marked variations: some datasets lend themselves to superior generalization capabilities compared to others when used for meta-distance training. Moreover, we identified salient features linked to such superior results: datasets characterized by a geographically large radio-map and a moderate average number of APs detected per fingerprint tended to perform well. Finally, we demonstrated the potential of combining the best performing meta-distances through a simple approach, which outperformed all distance and meta-distance contenders. Our research highlights the role of training set selection in meta-distance learning for WiFi fingerprint-based indoor localization. Moreover, the identified features can inform dataset selection and creation phases for indoor positioning.

As for future work, besides those already mentioned in the paper, we will investigate the effectiveness of genetic programming in learning distances directly from RSS values. Additionally, we will explore the potential of other symbolic regression approaches, such as those based exclusively on or in conjunction with deep learning techniques. Finally, note that, despite its simplicity and intuitiveness, the method we considered here to build the ranks, i.e., Borda count, does not take into account the magnitude of the differences between the elements. Further studies will be conducted considering approaches capable of dealing with continuous data.

Acknowledgements. Angelo Montanari and Nicola Saccomanno acknowledge funding from *ublox AG*, project *Towards a uniform modeling framework for fingerprint-based positioning*; Andrea Brunello, Angelo Montanari, and Nicola Saccomanno also acknowledge funding from the Italian INdAM-GNCS, 2023 project *Analisi simbolica e numerica di sistemi ciberfisici*, CUP_E53C22001930001.

References

1. Alitaleshi, A., Jazayeriy, H., Kazemitabar, J.: EA-CNN: a smart indoor 3D positioning scheme based on Wi-Fi fingerprinting and deep learning. Eng. Appl. Artif. Intell. **117**, 105509 (2023)
2. Bahl, P., Padmanabhan, V.N.: RADAR: an in-building RF-based user location and tracking system. In: Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (IEEE INFOCOM), pp. 775–784. IEEE Computer Society (2000)
3. Benjamini, Y., Hochberg, Y.: Controlling the false discovery rate: a practical and powerful approach to multiple testing. J. R. Stat. Soc. Ser. Stat. Methodol. **57**(1), 289–300 (1995)
4. Borda, J.D.: Mémoire sur les élections au scrutin. Histoire de l'Academie Royale des Sciences pour 1781 (1784)
5. Brunello, A., Civilini, M., De Martin, S., et al.: Machine learning-assisted environmental surveillance of Legionella: a retrospective observational study in Friuli

Venezia Giulia region of Italy in the period 2002–2019. Inform. Med. Unlock. **28**, 100803 (2022)
6. Brunello, A., Montanari, A., Saccomanno, N.: A genetic programming approach to WiFi fingerprint meta-distance learning. Pervasive Mob. Comput. **85**, 101681 (2022)
7. Cao, Y., Smucker, B.J., Robinson, T.J.: On using the hypervolume indicator to compare Pareto fronts: applications to multi-criteria optimal experimental design. J. Stat. Plan. Inference **160**, 60–74 (2015). https://doi.org/10.1016/j.jspi.2014.12.004
8. Deb, K., Jain, H.: An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part I: solving problems with box constraints. IEEE Trans. Evol. Comput. **18**(4), 577–601 (2013). https://doi.org/10.1109/TEVC.2013.2281535
9. Eiben, A.E., Smith, J.E.: Introduction to Evolutionary Computing. Springer, Heidelberg (2003)
10. Fortin, F.A., De Rainville, F.M., Gardner, M.A., Parizeau, M., Gagné, C.: DEAP: evolutionary algorithms made easy. J. Mach. Learn. Res. **13**, 2171–2175 (2012)
11. Gonçalves, I., Silva, S.: Balancing learning and overfitting in genetic programming with interleaved sampling of training data. In: Krawiec, K., Moraglio, A., Hu, T., Etaner-Uyar, A.Ş, Hu, B. (eds.) EuroGP 2013. LNCS, vol. 7831, pp. 73–84. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-37207-0_7
12. Khalajmehrabadi, A., Gatsis, N., Akopian, D.: Modern WLAN fingerprinting indoor positioning methods and deployment challenges. IEEE Commun. Surv. Tutor. **19**(3), 1974–2002 (2017)
13. King, T., Kopf, S., Haenselmann, T., et al.: CRAWDAD dataset mannheim/compass (v. 2008-04-11) (2008). https://doi.org/10.15783/C7JS3P
14. Koza, J.R.: Genetic programming as a means for programming computers by natural selection. Stat. Comput. **4**(2), 87–112 (1994)
15. Lohan, E.S.: Additional TAU datasets for Wi-Fi fingerprinting- based positioning (2020)
16. Lohan, E.S., Torres-Sospedra, J., Leppäkoski, H., et al.: Wi-Fi crowdsourced fingerprinting dataset for indoor positioning. Data **2**(4), 32 (2017)
17. Mendoza-Silva, G.M., Richter, P., Torres-Sospedra, J., et al.: Long-term WiFi fingerprinting dataset for research on robust indoor positioning. Data **3**(1) (2018)
18. Mendoza-Silva, G.M., Torres-Sospedra, J., Huerta, J.: A meta-review of indoor positioning systems. Sensors **19**(20), 4507 (2019)
19. Moreira, A., Nicolau, M.J., Meneses, F., Costa, A.: Wi-Fi fingerprinting in the real world-RTLS@ UM at the EvAAL competition. In: Sixth International Conference on Indoor Positioning and Indoor Navigation (IPIN), pp. 1–10. IEEE (2015)
20. Moreira, A., Silva, I., Torres-Sospedra, J.: The DSI dataset for Wi-Fi fingerprinting using mobile devices (2020)
21. Orzechowski, P., La Cava, W., Moore, J.H.: Where are we now? A large benchmark study of recent symbolic regression methods. In: Genetic and Evolutionary Computation Conference (GECCO), pp. 1183–1190 (2018)
22. Poli, R., Langdon, W., Mcphee, N.: A Field Guide to Genetic Programming. Lulu Enterprises, UK Ltd. (2008)
23. Potortì, F., Crivello, A., Palumbo, F., Girolami, M., Barsocchi, P.: Trends in smartphone-based indoor localisation. In: Eleventh International Conference on Indoor Positioning and Indoor Navigation (IPIN), pp. 1–7. IEEE (2021)
24. Richter, P., Lohan, E.S., Talvitie, J.: WLAN (WiFi) RSS database for fingerprinting positioning (2018)

25. Saccomanno, N., Brunello, A., Montanari, A.: What you sense is not where you are: on the relationships between fingerprints and spatial knowledge in indoor positioning. IEEE Sens. J. **22**, 4951–4961 (2021)
26. Shang, S., Wang, L.: Overview of WiFi fingerprinting-based indoor positioning. IET Commun. **16**(7), 725–733 (2022)
27. Shrestha, S., Talvitie, J., Lohan, E.S.: Deconvolution-based indoor localization with WLAN signals and unknown access point locations. In: International Conference on Localization and GNSS (ICL-GNSS), pp. 1–6. IEEE (2013)
28. Song, X., et al.: CNNloc: deep-learning based indoor localization with WiFi fingerprinting. In: IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation Conference (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI), pp. 589–595. IEEE (2019)
29. Song, X., Fan, X., Xiang, C., et al.: A novel convolutional neural network based indoor localization framework with WiFi fingerprinting. IEEE Access **7**, 110698–110709 (2019)
30. Torres-Sospedra, J., Gaibor, D.P.Q., Nurmi, J., Koucheryavy, Y., Lohan, E.S., Huerta, J.: Scalable and efficient clustering for fingerprint-based positioning. IEEE Internet Things J. **10**(4), 3484–3499 (2022)
31. Torres-Sospedra, J., Montoliu, R., Trilles, S., et al.: Comprehensive analysis of distance and similarity measures for Wi-Fi fingerprinting indoor positioning systems. Expert Syst. Appl. **42**(23), 9263–9278 (2015)
32. Torres-Sospedra, J., Montoliu, R., Usó, A.M., et al.: UJIIndoorLoc: a new multi-building and multi-floor database for WLAN fingerprint-based indoor localization problems. In: Fifth International Conference on Indoor Positioning and Indoor Navigation (IPIN), pp. 261–270. IEEE (2014)
33. Torres-Sospedra, J., Moreira, A.J.C.: Analysis of sources of large positioning errors in deterministic fingerprinting. Sensors **17**(12), 2736 (2017)
34. Torres-Sospedra, J., Silva, I., Klus, L., et al.: Towards ubiquitous indoor positioning: comparing systems across heterogeneous datasets. In: Eleventh International Conference on Indoor Positioning and Indoor Navigation (IPIN), pp. 1–8. IEEE (2021)
35. Woolson, R.F.: Wilcoxon signed-rank test. Wiley Encycl. Clin. Trials 1–3 (2007)
36. Xia, S., Liu, Y., Yuan, G., et al.: Indoor fingerprint positioning based on Wi-Fi: an overview. ISPRS Int. J. Geo-Inf. **6**(5), 135 (2017)