



HAL
open science

An Inductive System Monitoring Approach for GNSS Activation

Shahrooz Abghari, Veselka Boeva, Emiliano Casalicchio, Peter Exner

► **To cite this version:**

Shahrooz Abghari, Veselka Boeva, Emiliano Casalicchio, Peter Exner. An Inductive System Monitoring Approach for GNSS Activation. 18th IFIP International Conference on Artificial Intelligence Applications and Innovations (AIAI), Jun 2022, Hersonissos, Greece. pp.437-449, 10.1007/978-3-031-08337-2_36 . hal-04668669

HAL Id: hal-04668669

<https://inria.hal.science/hal-04668669v1>

Submitted on 7 Aug 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License



This document is the original author manuscript of a paper submitted to an IFIP conference proceedings or other IFIP publication by Springer Nature. As such, there may be some differences in the official published version of the paper. Such differences, if any, are usually due to reformatting during preparation for publication or minor corrections made by the author(s) during final proofreading of the publication manuscript.

An Inductive System Monitoring Approach for GNSS Activation*

Shahrooz Abghari¹[0000-0002-3010-8798], Veselka Boeva¹[0000-0003-3128-191X],
Emiliano Casalicchio¹[0000-0002-3118-5058], and Peter
Exner²[0000-0002-9629-786X]

¹ Blekinge Institute of Technology, Karlskrona, Sweden
{shahrooz.abghari, veselka.boeva, emiliano.casalicchio}@bth.se

² Sony R&D Center Lund Laboratory, Lund, Sweden
peter.exner@sony.com

Abstract. In this paper, we propose a Global Navigation Satellite System (GNSS) component activation model for mobile tracking devices that automatically detects indoor/outdoor environments using the radio signals received from Long-Term Evolution (LTE) base stations. We use an Inductive System Monitoring (ISM) technique to model environmental scenarios captured by a smart tracker via extracting clusters of corresponding value ranges from LTE base stations' signal strength. The ISM-based model is built by using the tracker's historical data labeled with GPS coordinates. The built model is further refined by applying it to additional data without GPS location collected by the same device. This procedure allows us to identify the clusters that describe semi-outdoor scenarios. In that way, the model discriminates between two outdoor environmental categories: open outdoor and semi-outdoor. The proposed ISM-based GNSS activation approach is studied and evaluated on a real-world dataset contains radio signal measurements collected by five smart trackers and their geographical location in various environmental scenarios.

Keywords: Environmental context detection · GNSS signal · Inductive system monitoring · Clustering analysis.

1 Introduction

Global Navigation Satellite System (GNSS) is a positioning technique that can be used for detecting the current position of a portable smart device. It utilizes any satellite constellation capable of providing Positioning, Navigation, and Timing (PNT) services on a global or regional basis. The localization of devices such as smartphones and trackers in outdoor areas typically relies on GNSS such as the Global Positioning System (GPS), which is known to perform well in open

* This work is part of the Sony RAP 2020 Project “Distributed and Adaptive Edge-based AI Models for Sensor Networks”. We thank Hannes Bergkvist (Sony R&D Center Lund Laboratory) for his contribution to the data collection.

sky environments. However, these devices may operate in any environment, e.g., open outdoors, semi-outdoors, and indoors. In addition, battery power is a limitation for any portable smart device such as trackers or smartphones. Therefore, detecting the devices' environment and providing context-aware information in various scenarios can be helpful and lead to battery-saving solutions.

A review of existing GNSS and onboard vision-based solutions for environmental context detection has been conducted in [5]. The main finding of this work is that most of the existing context-aware solutions are based on one type of sensor (GNSS or vision) and are designed basically to perform sensor management in binary indoor/outdoor situations. Other recent works studying indoor/outdoor detection methods based on Machine Learning (ML) techniques are [4, 6, 7, 13, 15, 16]. For example, in [15] the authors investigate a wide range of supervised ML algorithms for detecting four types of indoor/outdoor environments based on an approach using the Global System for Mobile communications (GSM) signal strength. The main idea is that different environmental scenarios lead to different signal strength characteristics. By learning various signal strength characteristics would be possible to determine the smart device's surrounding environment. GNSS measurements from Android smartphones are leveraged in [16] to detect indoor/outdoor complex environments. The authors extract spatial geometry distribution, time sequence, and statistical features from the GNSS measurements through Android smart mobile devices. Supervised ML algorithms are then used to predict indoor/outdoor status, which is interpreted as the observations of the Hidden Markov Model (HMM) to detect the transition between indoor/outdoor in complex scenarios. A similar solution to detect environmental context is proposed in [7]. In [4], an approach is introduced that uses ensemble learning methods such as Random Forest (RF) and AdaBoost to detect indoor/outdoor environments according to the different daily activities of users. Souza et al. [13], propose an interesting approach that integrates supervised and unsupervised ML techniques for indoor/outdoor detection.

In this paper, in contrast to the most published ML-based indoor/outdoor detection methods, we propose using a clustering technique to build a GNSS component activation model. Namely, the proposed approach is based on the Inductive System Monitoring (ISM) method introduced in [8]. Our ISM-based indoor/outdoor detection approach, similar to the idea studied in [15] uses the cellular signal strength to detect the environment. In this context, an ISM model of a given smart device can be created by using the device's historical data marked with geographic locations. The model is built by extracting clusters of related value ranges for the neighboring Long-Term Evolution (LTE) base stations' signal strength. The built model is further refined by using the remaining data without GPS location to discriminate between open outdoor and semi-outdoor scenarios. Each cluster models a specific environmental scenario by defining a range of allowable values for each cellular base station in a given input vector. In that way, our model not only supplies the user with easily interpretable representations of the device's environmental scenarios, but it is also small as it has modest requirements with respect to storage and computations.

2 Related Work

According to [16] the existing indoor/outdoor detection methods can be classified into two main groups: fixed detection rules or threshold-based techniques and ML-based techniques. While approaches in the first group use fixed detection rules and thresholds, making them difficult to adapt to different environments and devices, techniques in the second group use ML algorithms to detect indoor/outdoor status based on features extracted from devices. Since the approach proposed in this study belongs to the second group, ML-based indoor/outdoor environment detection techniques are reviewed in this section.

In [3], an approach, entitled SenseMe, uses the C4.5 algorithm on data generated from GPS, gyroscope, accelerometer, and Bluetooth to module the sensed environmental context, as well as the context-aware location. In [14], the authors proposed a sound-based indoor/outdoor detection method that utilizes binary classification of the environment’s acoustic reverberation features. Reverte et al. [11] employed a binary classification technique on the Received Signal Strength Indicator (RSSI) from 802.11 access points to identify a pedestrian’s indoor or outdoor status. Ashraf et al. [2] proposed MagIO, a solution that utilizes magnetic field signals sensed by smartphones for detecting indoor/outdoor states. Magnetic field features are classified with different ML algorithms, including Support Vector Machines (SVM), gradient booster machines, RF, k -nearest neighbor, and decision trees. In [15], the authors applied an ML algorithm to classify the neighboring GSM base station’s signal in different environments and identify the users’ current context by signal recognition. Radu et al. [9] propose to detect indoor/outdoor context by employing co-training according to the feature of light, magnetic, and cell sensors. The proposed solution can automatically learn the characteristics of new environments and devices, thus providing a detection accuracy that exceeds 90% even in unfamiliar circumstances. Multiple contextual features are also used in [1], which leveraged J48 and other ML algorithms to detect the indoor/outdoor state with high accuracy. In [12], an indoor and outdoor classification system that relies solely on light measurements is introduced. More specifically, the system measures ultraviolet light, color temperature, luminosity, and red, green, blue, and clear components of light at 1-minute intervals. Three ML algorithms, SVM, artificial neural network, and bagged tree, are trained and tested on these measurements. A multi-sensor deep learning model to predict the indoor/outdoor state has been proposed in [17]. An interesting indoor/outdoor detection solution that integrates unsupervised and supervised algorithms and relies on the location, location accuracy, and signal strength is proposed in [13].

Most of the approaches reviewed above propose solutions that use supervised ML techniques and demonstrate higher performance on datasets coming from the same environments/devices as those used for the model training than on new environments/devices. Exceptions are the semi-supervised learning approach proposed in [9] and the hybrid approach, HybridIO, introduced by Souza et al. [13]. The former method can conduct online updates of a Naive Bayes classifier to previously unseen environments by applying co-training. The HybridIO,

on the other hand, utilizes a supervised technique only when the clustering is not able to separate the data well into two (indoor vs. outdoor) groups. Many of the proposed indoor/outdoor detection solutions require a variety of sensors to identify the user’s environmental context. In contrast to the most state-of-the-art indoor/outdoor detection methods, our proposed approach builds an ISM-based GNSS activation model that can discriminate between outdoor and indoor environments and characterizes the most typical outdoor environmental scenarios of a given portable smart device.

3 Methodology

3.1 An Inductive System Monitoring Approach

An ISM method, introduced in [8], provides a technique to automatically produce a health monitoring knowledge base for a given system. Historical datasets are used to extract general clusters of nominal data to characterize the typical system behavior. The system is monitored by comparing real-time operational data with these clusters. The main focus is to build a clustering model using available historical data by considering low and high vectors representing the boundary of each cluster. The low and high vectors are formed by selecting the minimum and maximum values of features by considering all data samples assigned to each cluster. In this case, the first data sample will be considered as a singleton cluster. Every other input data sample is compared to the currently available clusters to identify the closest cluster to it. The distance between a data sample and a cluster’s centroid can be measured by computing the Euclidean distance. In the current context, the cluster’s centroid is calculated by creating a vector from the average of the low and high values for each cluster’s feature. For each new training input data sample, the IMS algorithm finds the closest cluster in the current clustering solution. Then it is determined whether the data sample is contained in the bounding rectangle defined by that cluster’s low and high vectors or if it is close enough to be incorporated into the cluster.

3.2 The proposed ISM-based Approach for GNSS Activation

In this study, we propose a novel GNSS activation approach based on the method developed by Iverson in [8]. The ISM-based GNSS activation model is built in two steps. Initially, the ISM technique (Algorithm 1) is applied to the device’s historical data marked with GPS coordinates to create clusters that model different environmental scenarios. In the second step, the built ISM model is refined by applying Algorithm 2 to the remaining device’s data without GPS location. This procedure allows for identifying those clusters that describe semi-outdoor scenarios. Thus, the built ISM-based model discriminates between two outdoor environmental categories: open outdoor and semi-outdoor. The clusters could be further analyzed and grouped based on their similarities by including additional information on location characteristics such as the city center, suburb, or rural area.

Algorithm 1: ISM Model

Data: data contains seven (un)sorted signal strengths and the number of available cells, and a distance threshold

Result: low_high, centers, labels

- 1 Create an empty list of labels w.r.t to the size of data;
- 2 Assign the first sample to the cluster 0;
- 3 Set the sample to be the centroid of the cluster;
- 4 Set the low and high vectors to be the sample itself;
- 5 **for** *sample 1 to n* **do**
- 6 Compute each sample's Euclidean distance from the available centroids and identify the closest cluster;
- 7 **if** *the sample is between the low and high vectors of the closest clusters* **then**
- 8 Assign the sample to that cluster;
- 9 **else if** *the Euclidean distance of the sample is less than the threshold (= 4.0)* **then**
- 10 Assign the sample to that cluster;
- 11 Update the low and high vectors by computing the min and max vectors;
- 12 Update the centroid by averaging the low and high vectors;
- 13 **else**
- 14 Create a new cluster and assign the sample to it;
- 15 Consider the low and high vectors to be the sample itself and append it to the list of available low and high vectors;
- 16 Set the sample to be the centroid of the cluster and append it to the list of available centroids;
- 17 **end**
- 18 **end**

Our motivation to use the ISM approach [8] for building a GNSS component activation model is to get a better insight into the indoor/outdoor localization problem through data exploration and extraction of environmental scenarios' patterns. In addition, we search for a solution that can run on the device and quickly adapt to a new environment without the need to completely retrain the built model. The aim is to create a model that can identify the availability of GPS coverage with the help of a cellular network. The model created by applying the ISM algorithm (Algorithm 1) describes its environment through nested clusters. Namely, it consists of two categories, standing for open outdoor and semi-outdoor, being outer clusters, each one containing nested clusters modeling the corresponding category's typical environmental scenarios. Each inner cluster is represented by its low and high feature vectors, which facilitates its interpretation, visualization, and comparison with other clusters that represent environmental scenarios from the same category or the other. Note that the indoor environmental scenarios are not present in the built ISM-based GNSS activation model. In that way, during the monitoring phase (Algorithm 2), all data samples that do not fit any of the nested clusters of the two categories are

interpreted as belonging to the indoor environment. All these keep the model small, having modest requirements with respect to storage and computations.

Algorithm 2: ISM Monitor

Data: data contains seven (un)sorted signal strengths and the number of available cells, low and high vectors, and centroids
Result: labeled data

- 1 Create an empty list of labels w.r.t to the size of data;
- 2 **for** *sample 0 to n* **do**
- 3 Compute each sample’s Euclidean distance from the available centroids
 and identify the closest cluster;
- 4 **if** *the sample is between the low and high vectors of the closest clusters* ;
- 5 **then**
- 6 | Assign the sample to that cluster;
- 7 **end**
- 8 **end**

4 Experimental Design

4.1 Dataset

The participants in the data collection process were equipped with trackers to record data during normal daily activities. The trackers are based on a Nordic Semiconductors nRF52832 paired with a Quectel BG96 for integrated GNSS positioning and cellular connectivity capabilities. The modem used for LTE connectivity is based on a Qualcomm mdm9206. The trackers are held in hand by participants or placed in a pocket or backpack.

During data collection, the trackers were configured to capture data within a 1-second interval, recording cellular network parameters along with positioning parameters. Cellular parameters included signal strength, channel quality, and bandwidth descriptors, while positioning parameters consisted of latitude and longitude coordinates. Parameters from the serving cell station to which the trackers were connected and neighboring cell stations were recorded. Participants were instructed to collect data from indoors and outdoors while considering corner cases, such as overpasses and building corners, and varying the type of building and areas visited, such as commercial, residential, and industrial. No other constraints were given regarding collection patterns, and participants were told to walk freely as they would normally. Data collection took place over two months, and in total more than 707k samples amounting to 200hrs data were collected.

4.2 Data Preprocessing

Among the available data features, we chose the signal strengths of six neighboring cells and the serving cell, together with the trackers’ longitude and latitude. The trackers’ GPS was enabled during data collection. As a result, these two columns might be empty based on the unavailability of GPS signals. In addition, we considered the number of available cells visible by the tracker at each time, t . The data further was labeled based on the availability of the GPS coverage into class 0 in the presence of GPS coverage and class -1 otherwise. Table 1 summarizes the total number of samples and percentages of samples with/without GPS coverage for each tracker.

Table 1: Collected data samples by five trackers

Tracker	No. of samples	GPS coverage(%)	
		with	without
d1	176 528	59.79	40.21
d2	136 290	61.17	38.83
d3	129 342	66.75	33.25
d4	129 707	64.91	35.09
d5	135 476	65.15	34.85
Total	707 343		

For each tracker, we used class 0 to build a tracker’s model and class -1 to check the model’s performance in predicting whether GPS should be activated. For both training and testing, only eight features, i.e., the signal strengths of the six neighboring cells and the serving cell, and the number of available cells were considered. The signal strengths collected by the trackers are between 0 and 63. This range translates into -111 and -48 dBm by subtracting 111 from each value, representing the weakest and strongest signals. As the signal strengths are in the range $[0, 63]$, we fill in the missing signal strengths with a significant negative value, i.e., -100 . Finally, all eight features are standardized by removing the mean and scaling to unit variance using z -score, calculated by the following equation $z = (x - \bar{x})/\sigma$, where \bar{x} represents the mean of the samples and σ is the standard deviation of the samples.

4.3 Evaluation Metrics

We apply a symmetric external validation index, *Rand Index (RI)*, to evaluate the pairwise similarity between built models for different trackers. RI [10] measures the similarity between pairs of models considering the level of agreement between the two. RI is calculated as the ratio between the number of agreeing pairs and the total number of pairs. The RI score is bound between 0 and 1. A score close to 0 represents random labeling, and 1 stands for a perfect match. We also use *accuracy* to evaluate the performance of the built models in classifying

data samples into classes 0 and 1, standing for the presence and absence of GPS coverage, respectively.

5 Results and Discussion

We performed three experiments using data collected by five trackers. In *Experiment 1*, we applied 10-fold cross-validation on the data collected by each tracker, i.e., nine parts of the data with GPS coverage were used for training, while the 10th part together with all the data without GPS coverage were used for testing. We explored shuffling the data with GPS coverage in another setup before applying 10-fold cross-validation. In addition, either the collected signals strengths by the trackers were sorted in descending order, or they were left as initially received. In total, a combination of four different setups was studied. Each setup was performed three times. Figure 1 summarizes the obtained results. The main finding is that sorting the collected signals' strengths by the trackers in the unshuffled scenario led to lower number of clusters and higher average accuracy for most trackers.

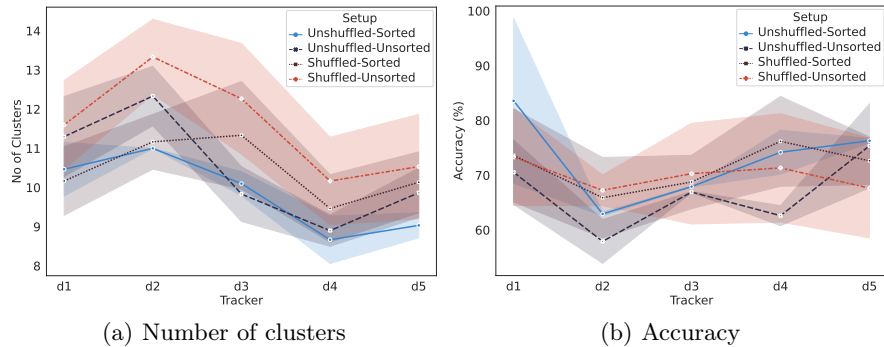


Fig. 1: *Experiment 1* – Results of the 3-time 10-fold cross-validation. The dark and bright colors represent the mean and standard deviation, respectively.

Table 2 represents the built model for tracker d1 in fold 9 with unshuffled and sorted setups in *Experiment 1*. The used training data contains 94 981 data samples with GPS coverage, and the test data contains 81 541 data samples, where 10 553 samples have GPS coverage, and 70 988 do not have GPS coverage. This model contains 11 clusters. Each cluster in Table 2 is described by the minimum and maximum number of cells, the total number of samples, the number of samples with and without GPS coverage, and their percentages.

After performing the test, most of the data with GPS coverage fitted into clusters 3, 10, 8, and 0. The parenthesized numbers under GPS Coverage, Count column, in Table 2 represent test data with GPS coverage. Regarding the data without GPS coverage, 99.98% of the data did not match any clusters except 81 data samples that fit into cluster 0. Therefore, this cluster is considered to represent a semi-outdoor scenario.

Table 2: *Experiment 1* – Built model with accuracy 99.89 % in fold 9 for tracker d1 (unshuffled data samples and sorted signal strengths)

Cluster	Total	GPS Coverage		No GPS Coverage		Cells	
		Count	%	Count	%	Min	Max
0	17 991	17 910 (1 552)	99.55	<u>81</u>	0.45	4	7
1	374	374 (31)	100.00	0	0.00	3	3
2	762	762 (76)	100.00	0	0.00	2	2
3	53 704	53 704 (3 769)	100.00	0	0.00	5	7
4	5 524	5 524 (302)	100.00	0	0.00	4	4
5	97	97 (5)	100.00	0	0.00	1	1
6	1 145	1 145 (101)	100.00	0	0.00	3	3
7	95	95 (10)	100.00	0	0.00	1	1
8	15 905	15 905 (1 736)	100.00	0	0.00	4	5
9	484	484 (301)	100.00	0	0.00	4	4
10	9 522	9 522 (2 658)	100.00	0	0.00	7	7
-1	70 919	<u>12</u> (12)	0.02	70 907	99.98	1	7

In *Experiment 2*, we used the data with GPS coverage collected by each tracker to build a model representing the tracker’s behavior. For testing, we used data without GPS coverage. Table 3 represents the accuracy of the models for each tracker with and without sorting the signal strengths. As one can notice, overall the models’ accuracy of all trackers, except d5, is higher while the signal strengths are sorted. Therefore, we used the models with sorted signals for the third experiment.

Table 3: *Experiment 2* – Model’s accuracy on data without GPS coverage for sorted and unsorted trackers’ signal strengths

Model	Accuracy (%)	
	Sorted signals	Unsorted signals
M-d1	99.89	64.44
M-d2	57.15	49.82
M-d3	61.49	60.40
M-d4	68.81	56.04
M-d5	71.94	72.83

As an example, we present in Table 4 the statistics regarding the tracker d2 model, which achieved the lowest accuracy among other models with the signal strengths sorted. As one can see, the built model contains 11 clusters, where cluster 0 represents the largest cluster with 52 099 data samples with GPS coverage and cluster 6 with only 28 data samples stands for the smallest cluster. After applying the model to the test samples, which only consist of data without GPS coverage, three clusters (0, 3, and 7) are identified as ones modeling semi-outdoor scenarios. Note that all these three clusters have something in common,

namely the minimum and maximum number of cells in all are between 5 and 7. On the other hand, approximately 57% of the samples are not fitted into any clusters, suggesting the tracker’s GPS should not be turned on. Another interesting observation was that when the minimum and the maximum number of cells for a cluster is in a range of [4, 7] or [5, 7], it represents a semi-outdoor scenario.

Table 4: *Experiment 2* – Built model with accuracy 57.15 % for tracker d2 (sorted signal strengths)

Cluster	Total	GPS Coverage		No GPS Coverage		Cells	
		Count	%	Count	%	Min	Max
0	70 236	52 099	74.18	<u>18 137</u>	25.82	5	7
1	1 793	1 793	100.00	0	0.00	4	4
2	258	258	100.00	0	0.00	3	3
3	19 872	15 947	80.25	<u>3 925</u>	19.75	5	7
4	101	101	100.00	0	0.00	1	1
5	68	68	100.00	0	0.00	2	2
6	28	28	100.00	0	0.00	1	1
7	12 427	11 811	95.04	<u>616</u>	4.96	5	7
8	724	724	100.00	0	0.00	4	4
9	264	264	100.00	0	0.00	3	3
10	273	273	100.00	0	0.00	2	2
-1	30 246	0	0.00	30 246	100.00	1	7

In *Experiment 1*, the accuracy of the models is evaluated based on data with and without GPS coverage. In contrast, in *Experiment 2*, we studied how well a tracker’s model can classify samples without GPS coverage collected by the same tracker. As can be observed in Table 3, the most precise model in the case where signal strengths are sorted belongs to the tracker d1, while the tracker d2 generates the least accurate model in both scenarios.

In *Experiment 3*, we applied each tracker’s model to the data collected by the other four trackers. This experiment aimed to show to what extent the built model for a specific tracker can be used for accurate predictions on the data samples (with and without GPS coverage) collected by another tracker. Table 5 shows the performance of each model on other trackers’ data. As one can see, the results vary, e.g., the tracker d1 model, denoted by M-d1, achieved an accuracy of 55.71% on the tracker d4 data. On the other hand, M-d4 attained an accuracy of 42.68% on the tracker d1 data and performed much better (51.89%) on the tracker d5 data. Overall, M-d2 achieved the best accuracy on the tracker d3 data (65.27%). In addition, both M-d3 and M-d4 achieved their highest performance on the tracker d5 data compared to other trackers’ data.

Tacker d1 data was collected within 56 days from the city center and a small rural area. On the other hand, other trackers’ data was collected within 27 days and only limited to the city center. In addition, all five trackers were only

Table 5: *Experiment 3* – Models’ accuracy (%) on the collected data by other tracker’s (sorted signal strengths)

	d1	d2	d3	d4	d5
M-d1	-	35.41	30.69	55.71	33.33
M-d2	54.58	-	65.27	38.27	62.14
M-d3	56.58	48.93	-	45.98	64.59
M-d4	42.68	51.23	46.73	-	51.89
M-d5	54.56	48.75	51.53	39.56	-

used together for 24 days, explaining the different performances of the models. According to a domain expert, the tracks’ model performance variation may also be due to trackers’ orientation.

As part of *Experiment 3*, we also performed a pairwise comparison of the built model (see Table 6). For this reason, we used RI to measure the similarity between two models by considering the level of agreement between them. RI scores close to 1 represent perfect agreements between the two models in clustering pairs of data samples together, while scores close to 0 explain strong disagreements among the models. As can be seen, M-d2 and M-d3 achieved the highest level of agreement, 0.77. In addition, both of these models received RI scores of 0.71 and 0.73 when compared with M-d5, respectively.

Table 6: *Experiment 3* – Similarity of the models in terms of RI score

	M-d1	M-d2	M-d3	M-d4
M-d2	0.49			
M-d3	0.49	0.77		
M-d4	0.53	0.52	0.52	
M-d5	0.48	0.71	0.73	0.53

6 Conclusions and Future Work

In this paper, we have proposed a novel GNSS component activation model that automatically detects indoor/outdoor environments. We used radio signals received from LTE base stations to build an ISM-based model that describes different environmental scenarios captured by a smart tracker. The built model is capable of discriminating between two outdoor environmental categories: open outdoor and semi-outdoor. During the monitoring phase, data samples that do not fit any of the clusters of these two categories are interpreted to belong to the indoor environment. In that way, the model is kept small, with modest requirements regarding storage and computations.

We have evaluated the proposed ISM-based GNSS activation model on real-world data provided by a partner company in three different experiments. The obtained results have been analyzed, and interesting patterns about the GNSS activation problem have been extracted. For example, the models built on sorted

and unshuffled data have shown higher performance. The proposed model supplies the user with easily interpretable representations of the environmental scenarios. These representations can be further analyzed to gain additional knowledge for further refinement of the built model. Our ISM-based model has produced a high accuracy result (99.89%) for the tracker trained on the largest dataset covering the longest period. This confirms that the quantity and quality of the training data matter for the model performance.

The built ISM model can easily be adapted to new environments by applying the same learning procedure to newly arrived data as the one used for the initial building of the model. This is one of the directions that is planned to be explored in the future. Another ambition is to develop an integration technique that can create an overall ISM model from ones built on different devices/locations.

References

1. Anagnostopoulos, T., et al.: Environmental exposure assessment using indoor/outdoor detection on smartphones. *Pers Ubiquit Comput* **21**, 761–773 (2017)
2. Ashraf, I., Hur, S., Park, Y.: Magio: Magnetic field strength based indoor- outdoor detection with a commercial smartphone. *Micromachines* **9**(10) (2018)
3. Bhargava, P., et al.: Senseme: a system for continuous, on-device, and multi-dimensional context and activity recognition. In: *MobiQuitous* (2014)
4. Esmaeili Kelishomi, A., Garmabaki, A., Bahaghighat, M., Dong, J.: Mobile user indoor-outdoor detection through physical daily activities. *Sensors* **19**(3) (2019)
5. Feriol, F., Vivet, D., Watanabe, Y.: A review of environmental context detection for navigation based on multiple sensors. *Sensors* **20**(16) (2020)
6. Gao, H., Groves, P.D.: Environmental context detection for adaptive navigation using gnss measurements from a smartphone. *NAVIGATION* **65**(1), 99–116 (2018)
7. Gao, H., Groves, P.D.: Context detection for advanced self-aware navigation using smartphone sensors. *arXiv: Signal Processing* (2020)
8. Iverson, D.L.: Inductive system health monitoring. In: *IC-AI*. pp. 605–611 (2004)
9. Radu, V., Katsikouli, P., Sarkar, R., Marina, M.K.: A semi-supervised learning approach for robust indoor-outdoor detection with smartphones. In: *Proceedings of the 12th ACM Conf. on Embedded Network Sensor Systems*. p. 280–294 (2014)
10. Rand, W.M.: Objective criteria for the evaluation of clustering methods. *J. of the American Statistical association* **66**(336), 846–850 (1971)
11. Reverte, Ó.C., de Teruel, P.E.L., Ruiz, A.: Wifiboost: a terminal-based method for detection of indoor/outdoor places. In: *MobiQuitous* (2014)
12. Rhudy, M.B., et al.: Indoor and outdoor classification using light measurements and machine learning. *Applied Artificial Intelligence* pp. 1–14 (2021)
13. Souza, R.P., et al.: A big data-driven hybrid solution to the indoor-outdoor detection problem. *Big Data Research* **24**, 100194 (2021)
14. Sung, R., et al.: Sound based indoor and outdoor environment detection for seamless positioning handover. *ICT Express* **1**(3), 106–109 (2015)
15. Wang, W., Chang, Q., Li, Q., Shi, Z., Chen, W.: Indoor-outdoor detection using a smart phone sensor. *Sensors* **16**(10) (2016)
16. Zhu, Y., et al.: A fast indoor/outdoor transition detection algorithm based on machine learning. *Sensors* **19**(4) (2019)
17. Zhu, Y., Luo, H., Zhao, F., Chen, R.: Indoor/outdoor switching detection using multisensor densenet and lstm. *IEEE IoT Journal* **8**(3), 1544–1556 (2021)