Classification based Symbolic Indoor Positioning

Thesis for the Degree of Doctor of Philosophy (PhD)

By: Judit KUNNÉ TAMÁS

Supervisor: Zsolt TÓTH, PhD

UNIVERSITY OF DEBRECEN
Doctoral Council of Natural Sciences and Information Technology
Doctoral School of Informatics

Debrecen, 2021

Hereby I declare that I prepared this thesis within the Doctoral Council of Natural Sciences and Information Technology, Doctoral School of Informatics, University of Debrecen in order to obtain a PhD Degree in Informatics at Debrecen University.
The results published in the thesis are not reported in any other PhD theses.
Debrecen, 2021.

.....
signature of the candidate

Hereby I confirm that **Judit Kunné Tamás** candidate conducted her studies with my supervision within the Data science and visualization Doctoral Program of the Doctoral School of Informatics between 2020 and 2021. The independent studies and research work of the candidate significantly contributed to the results published in the thesis. I also declare that the results published in the thesis are not reported in any other theses. I support the acceptance of the thesis.
Debrecen, 2021.

.....
signature of the supervisor

Classification based Symbolic Indoor Positioning

Dissertation submitted in partial fulfilment of the requirements for the doctoral (PhD) degree in Informatics

Written by Judit Kunné Tamás certified computer science engineer

Prepared in the framework of the Informatics doctoral school of the University of Debrecen
(Data science and visualization programme)

Dissertation advisor: Zsolt Tóth, PhD

The official opponents of the dissertation:

	Dr.	.....
	Dr.	.....
	Dr.	.....

The evaluation committee:

chairperson:	Dr.	.....
members:	Dr.	.....
	Dr.	.....
	Dr.	.....
	Dr.	.....

The date of the dissertation defence: .....

Acknowledgements

Firstly, I would like to express my sincere gratitude to my advisor Dr. Zsolt Tóth for the continuous support of my Ph.D. study and related research, for his patience, motivation, and immense knowledge. His guidance helped me in all the time of research and writing of this thesis.

Furthermore, I would like to thank my co-authors, my co-workers and the rest of the undergraduate research team for their collaborative effort during the research.

Finally, I would like to thank my parents, my grandparents, my brother, my parents-in-law and especially my husband Dániel for their support and patience throughout the years.

Köszönetnyilvánítás

Elsőként szeretnék köszönetet mondani Dr. Tóth Zsolt témavezetőmnek a PhD tanulmányom és a kapcsolódó kutatások folyamatos támogatásáért és a türelméért, motivációjáért és a hatalmas ismereteiért. Az útmutatása folyamatosan segített a kutatások és a disszertáció írása során.

Ezenkívül szeretnék köszönetet mondani a társszerzőimnek, a munkatársaimnak és az egyetemi kutatócsoport többi tagjának a kutatás során végzett közreműködést.

Végül szeretnék köszönetet mondani a szüleimnek, a nagyszüleimnek, a testvéremnek, az anyósomnak és apósomnak, és különösen a férjemnek, Dánielnek az évek során nyújtott támogatásukért és türelmükért.

"It always seems impossible until it's done."

Nelson Mandela

Contents

List of Figures

List of Tables

List of Abbreviations

ANN	**A**rtificial **N**eural **N**etwork
AP	**A**ccess **P**oint
BIM	**B**uilding **I**nformation **M**odel
CAD	**C**omputer-**a**ided **D**esign
CSV	**C**omma **S**eparated **V**alues
GNSS	**G**lobal **N**avigation **S**atellite **S**ystem
GPS	**G**lobal **P**ositioning **S**ystem
IIS	**I**nstitute of **I**nformation **S**cience
ILONA	**I**ndoor **Lo**calisation and **Na**vigation
IndoorGML	**Indoor G**eographic **M**arkup **L**anguage
IoT	**I**nternet **o**f **T**hings
IPS	**I**ndoor **P**ositioning **S**ystem
k–NN	**k**–**N**earest **N**eighbour
GLONASS	**GLO**bal **NA**vigation **S**atellite **S**ystem
GNSS	**G**lobal **N**avigation **S**atellite **S**ystem
MLP	**M**ulti**L**ayer **P**erceptron
NIST	**N**ational **I**nstitute of **S**tandards and **T**echnology
OGC	**O**pen **G**eospatial **C**onsortium
RSSI	**R**eceived **S**ignal **S**trength **I**ndicator
SSID	**S**ervice **S**et **Id**entifier
UPGMA	**U**nweighted **P**air **G**roup **M**ethod with **A**rithmetic Mean
UPGMC	**U**nweighted **P**air **G**roup **M**ethod with **C**entroid
Weka	**W**aikato **E**nvironment for **K**nowledge **A**nalysis
WPGMA	**W**eighted **P**air **G**roup **M**ethod with **A**rithmetic Mean
WPGMC	**W**eighted **P**air **G**roup **M**ethod with **C**entroid
XML	E**x**tensible **M**arkup **L**anguage

Chapter 1

Introduction

Positioning has a wide range of applications and people are interested in this topic since ancient times. Positioning can help during an emergency, like the response time can be decreased by automatically determining the location of the caller using the cellular network. Positioning is also essential to navigation, surveying, tracking in warehouses, routing and consumer advertisement. Monitoring and controlling the movement of a person can help them reach a given destination through an unfamiliar environment. The first form of positioning relies on the recognition of landmarks, however, recently the positions are determined electronically. For example, Global Navigation Satellite System (GNSS) [HC08] is a navigation system that uses satellites with known position and trilateriation to determine any position on the globe.

Global Positioning System (GPS) [BS93] is a satellite-based navigation system that is available to civil usage since the $70's$. Its accuracy is between 1 and 5 meters for civil usage. GLONASS was developed and financed by the Russian Federation [Ros01], which can achieve 4 to 7 m accuracy. Galileo is the programme of the European Union for a global positioning system. During the design process, the goal was to have a high accuracy positioning system, which is independent of the US GPS and the Russian GLONASS. It can achieve less than 1 m accuracy for general uses. However, the Galileo system was completed in 2018, the EU promotes the creation of new services around the system.

GPS is the most popular and common positioning system momentarily. Its popularity is well illustrated by the fact that cars can be ordered with a built-in onboard computer with GPS capability. Truck tracking solutions are used by large international transportation companies, the location of the trucks can be even monitored on the internet. Moreover, agricultural machinery, like automatic

irrigation machines use GPS for the accurate area and row management. There are GPS-based games, like Geocaching [SH05], in which the goal is to find another player's hidden object by visiting the given map coordinate. There are countless other applications, which uses GPS, hence its leading role in the market is indisputable.

Unfortunately, Global Navigation Satellite Systems cannot be used for indoor positioning due to the unique properties of the indoor environments. Signal attenuation and reflection are two of the many reasons that make the indoor positioning challenging. Signal attenuation is caused by walls and humans, and it limits the applicability of received signal strength-based distance calculations. The reflection is caused by metal objects and it yields multipath effects. That causes difficulty in using positioning systems, that are based on the time-of-arrival of the signal. Moreover, the line-of-sight requirement for proper localization of the GPS is not satisfied. These effects limit the applicability of traditional triangulation-based outdoor positioning techniques and methods in the indoor environment.

Indoor Positioning Systems (IPS) have been considered as an active research field since the early 1990$'s$. However, the topic gained popularity in the 2010$'s$ with the widespread of smartphones and it is still a hot topic these days. Indoor positioning systems are used to determine the position of people or objects in buildings and closed areas. The smart environment can be enhanced with indoor positioning and navigation services. In addition, IPS attracted the business and software industry. For example, the model of big, frequently visited public spaces, such as railway stations and airports, can be found in Google Maps [Goo]. With Industry 4.0 [Las+14], the phenomenon of Smart Factories [Car+18; Lu+17; Dav+12] has emerged. Indoor positioning is also fundamental for Smart Factories because IPS is the base of other smart services such as tracking or intelligent traffic control. Smart Factory is a term by the National Institute of Standards and Technology (NIST) of the U.S Department of Commerce. Tracking workers and devices in the facility can improve efficiency and control. Logistics automation can reduce standby times of equipment and increase agility and accuracy of prediction. Both fields require the location information of materials, objects or workers. Therefore, location awareness is significant for Smart Factories. Indoor Positioning Systems are usually classified by the used technologies and the type of location.

Although different indoor positioning systems can be found in the literature [KY10; Liu+07], there is no single widely accepted solution like GPS for the outdoor environment. Existing indoor positioning systems can be based on various technologies such as Infrared [WH92], Ultrasonic [WJH97], GSM [WGM00], Bluetooth [Wei04; New14], RFID [Ni+04] and WLAN [BP00] technology, and on the magnetic field [Sär+15; Bra+14] and visible light [Li+14]. In the 2010′*s*, hybrid indoor positioning systems have emerged that simultaneously use various sensors and technologies [Bor+05; Wu+13] to determine the position, so they can enjoy the advantages of technologies applied. Developers have to make trade-offs between accuracy and cost when they choose a technology. For example, systems that use ultrasonic technology can achieve high accuracy, but they also have huge installation costs and may require specific client devices. On the other hand, systems that use WLAN for positioning have low installation costs and the client device can be an arbitrary smartphone, but their accuracy is lower than that of the systems based on ultrasonic technology. Although there have been numerous attempts to provide an accurate, robust and widely available indoor positioning system, a sufficiently precise, easily accessible and sustainable industrial standard has not been created yet.

Proximity based, Absolute and Symbolic positions [BD05] can be determined by Indoor Positioning Systems. The selection of the position type of indoor positioning system highly depends on the application area. Proximity-based positioning can be suitable for applications where not the exact location, but closeness to a well-known place or object, like Beacon, is required. For instance, the advertising systems broadcast to local devices and services with permissions could work this way. Absolute position is given by coordinates and it is used by highly accurate and robust systems. Global Positioning Systems determine absolute position and can achieve 1 cm accuracy for military purposes, but it has high installation costs. Symbolic positioning determine the location of an object as a well-defined part of a building. It would be suitable for asset tracking, where the user needs only a well-defined position of the object. Symbolic positioning can give us room-level results, which can be useful in indoor positioning.

Symbolic positions can be considered as a category label, thus the symbolic positioning can be converted into a classification problem. Classification is a well-studied part of data mining, therefore numerous well-known classification methods could be applied to

the indoor positioning. Euclidean distance is a common measure for error calculation in indoor positioning systems, although it is only convenient for absolute positions. Hence, the evaluation of the symbolic indoor positioning methods is based on the CRISP approach. The CRISP approach does not differentiate between the wrongly predicted classes, hence every misclassification is equally wrong. Besides CRISP logic, application-specific approaches are usually used for the evaluation of classifiers.

Application-specific methods use some domain knowledge. For example, there are different evaluation methods in computer vision [FM04; KEP99], speech recognition [JHL97; McD+07] and document classification [MY01; HK00]. For symbolic indoor positioning, a part of the domain knowledge can be incorporated into the description of the environment. To the best of our knowledge, there is no application-specific classification evaluation method for symbolic indoor positioning.

1.1 Research Goals

In my research, the following three goals were set. The first goal was to create a data set, which can be used to compare solutions for symbolic indoor positioning purposes. It should contain measurements from multiple sensors. Based on the created data set, various classifiers planned to be tested. To the best of our knowledge, there was no other such data set available. The second goal is to create an application-specific approach for classification error calculation for indoor positioning purposes. It should incorporate the domain knowledge, the topology, to the calculation. The usability of the proposed method needs to be examined, and compared to the traditional, CRISP approach. Application-specific evaluation of classification error is usually used in other fields, but as far as we know, this kind of method was not available for symbolic indoor positioning. The third goal is to develop a novel method to improve the classification accuracy for symbolic indoor positioning purposes using environment topology as domain knowledge. Most of the classification-based solutions do not use domain knowledge. In other words, they do not consider the layout and placement of the rooms. Applying this knowledge, we can provide a more error-tolerant method.

1.2 Dissertation Guide

The structure of the dissertation is the following. Chapter 1 introduces the topic of the dissertation and illustrates the motivations of the theses. Chapter 3 overview the related works and the existing indoor positioning systems, and shows the theoretical background of the work. Chapter 4 contains the data set created, and the benchmarking of classifiers based on this data set. In Chapter 5, the need for a topology-based classification evaluation is presented. Also, the novel gravitational force-based approach is proposed and validated by both test and real-life environments. Moreover, the comparison of the proposed method and the CRISP approach is studied. Chapter 6 describes the hierarchical grouping of topology information, the novel concept of enhanced classification. The concept is validated in both test and real-life environments. Chapter 7 summarizes the dissertation in English and in Hungarian.

Chapter 2

Indoor Positioning

2.1 Indoor Positioning Technologies

Although there are many different indoor positioning solutions in the literature, see [Liu+07; KY10] surveys for reviews, there is no single widely accepted solution such as GPS for the outdoor environment. The unique properties of the indoor environment make positioning challenging and limit the usability and performance of the existing global positioning methods. For example, GPS signals are absorbed by the walls. Hence, the existing indoor positioning solutions rely on different technologies such as infrared [WH92], ultrasonic [WJH97; Hoe+19], magnetometer [Sär+15], cellular [WGM00; Riz19a; Riz19b], visible light [Li+14; Gua+20], or other radio frequency [YA05; Ni+04; Wei04; Abb+19; Gro+19; Xu+17; CGOC17] signals.

Active Badges [WH92] was one of the first indoor positioning systems and was developed by AT&T Cambridge. Active Badges used infrared beacons carried by the users and the reader devices were installed into the building. It could reach an acceptable accuracy, but the evaluation of the indoor positioning was not performed by the authors.

Active Bats [WJH97] was also an early solution developed by AT&T Cambridge but it used ultrasound technology. The beacons of the Active Bats system were installed into the ceiling and the receivers were carried by the users. The system is evaluated as 95% of the raw readings lie within 14 cm of the true position, and the averaged readings lie within 8 cm of true position. Although Active Bats has good accuracy, its drawbacks are the special hardware requirement and the high installation cost.

IndoorAtlas [Sär+15] system is created by the University of Oulu

in Finland, which operates with the Earth's magnetic field for positioning. The magnetic field can be also applied to tracking purposes [Bra+14]. This technology requires a magnetometer that is usually built into modern smartphones. The accuracy of the system is claimed to be less than 2 meters.

Bluetooth is an emerging technology among indoor positioning systems. The Topaz [Wei04] system is based on Tadlys' Bluetooth infrastructure. The BlueTag is a typical small transceiver, and it has a unique ID [Liu+07]. The Topaz system is made up of a positioning server, wireless Access Points, and wireless tags. Apple's iBeacon solution [New14] is also based on Bluetooth technology. The evaluation of the system is based on the distance of the true and the calculated position.

The LANDMARC [Ni+04] system is based on RFID technology. The RFID tags are installed in known positions of the building. Users carry the reader device and use a database about the RFID tags installed for positioning. The high installation cost and high energy consumption of the RFID readers limit the usability of this technology. On the other hand, RFID based solutions can achieve high accuracy. The evaluation of the system is based on the distance of the true and the calculated position.

Epsilon [Li+14] is a visible light-based localization system, that uses LED bulbs. The system has different approaches to determining the location, depending on the number of detected LEDs. It can achieve sub-meter accuracy in a typical office environment. However, the light has to stay on always, which is not energy efficient, and the phone has to be held in the hand. The evaluation of the system is based on the distance of the true and the calculated position.

Hybrid systems were designed to overcome the limitations of standard indoor positioning systems that are based on a single technology. Hybrid systems can be based on the fingerprinting [KK04] approach, which is popular in indoor positioning systems [KY10].

2.2 Symbolic Positioning Systems

Symbolic positioning systems determine the location of an object as a well-defined part of a building, such as room, office or floor.

The Active Badges [WH92] system is the first symbolic positioning system. It uses the characteristic of the infrared, that the

infrared signals can not get through the walls. Thus, one badge in each room can provide a symbolic, room-level positioning.

The RADAR [BP00] system is based on WLAN technology. It determines position by coordinates, however, the system provides a list for each room, which contains the coordinates that belong to them. Hence, the RADAR system can also be considered as a symbolic positioning system.

WALRUS [Bor+05] is a room–level indoor positioning system which uses ultrasound signals and a WLAN network to determine the location of an object. The ultrasound signals are emitted by PCs installed in each room, and the clients do not have an additional requirement. The installation cost of the WALRUS system mainly consists of a PC per room. The WALRUS system uses the CRISP approach to evaluate the localization.

WILL [Wu+13] is a logical indoor localization system that uses WiFi RSSI and accelerometer information. WILL is based on the fingerprinting approach, and in the training phase, it uses data mining techniques to determine virtual rooms based on the data. This data mining enhanced process makes the training phase faster. The WILL system uses the CRISP approach to evaluate the positioning.

2.3 Fingerprinting Approach

The fingerprinting approach was presented in the RADAR System [BP00]. Fingerprinting approach based indoor positioning systems use data mining and various heuristic methods to estimate the position. These systems usually distinguish two phases that are called off-line and on-line phases. In the off-line phase, a site survey is performed, and the database is populated with measurements. The measurements are recorded in known positions that are also stored in the database. In the on-line phase, users perform measurements at an unknown location and send them to the system for the estimated position.

2.4 ILONA System

The ILONA (Indoor Localization and Navigation) System [Tot16] was used in the research presented to record measurements. The main functions of the positioning system are seen in Figure 2.1. The architecture of the system is client-server model, where the client

is an Android device chosen due to its programming simplicity. The ILONA System uses the fingerprinting approach for positioning. The collected sensor data from the client is sent to the server, where the positioning component determines the location with an integrated classifier method. The Miskolc Institute of Information Science Hybrid Indoor Positioning System Dataset was created using the ILONA System.

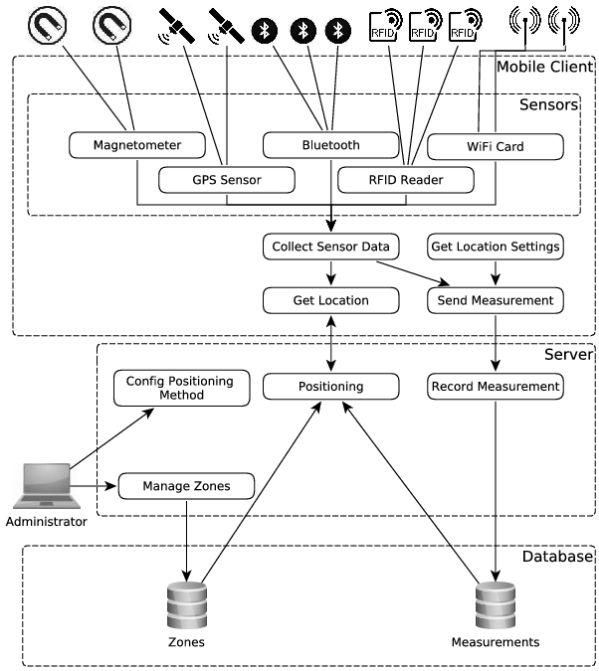


FIGURE 2.1: The architecture of the ILONA System [Tot16]

The system is based on client-server model and built from loosely coupled components. The positioning methods, storage and most of the business logic are implemented at the server. The clients were designed to run with low resources and to be easy to implement due to the huge variety of the smartphones. Thus, the clients are used to read sensor data and send the measurement to the server for further processing.

The web application was implemented in Java and Spring. It consists of loosely coupled components that have specific tasks such as measurements, positioning, navigation, and tracking. Only the measurement component was used by the system to create the data set[6].

Chapter 3

Theoretical Background

3.1 CRISP approach

The CRISP approach is the classical way to determine the distances of the original and the predicted symbolic positions. To easily access and store the distances of symbolic positions, a distance matrix is used to represent these values.

$$A = \begin{pmatrix} 0 & x_{1,2} & x_{1,3} & \dots & x_{1,n} \\ x_{2,1} & 0 & x_{2,3} & \dots & x_{2,n} \\ x_{3,1} & x_{3,2} & 0 & \dots & x_{3,n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_{n,1} & x_{n,2} & x_{n,3} & \dots & 0 \end{pmatrix} \tag{3.1}$$

The distance matrix is an $n \times n$ matrix, where n is the number of symbolic positions. The formal description of the distance matrix can be seen in Equation 3.1. In the distance matrix, the values on the main diagonal 0 for every symbolic position.

If the original and the predicted positions are not matching, in other words, the case is misclassified, the distance is 1, as seen in Equation 3.2.

$$x_{i,j} = 1,\ 1 \leq i, j \leq n,\ i \neq j \tag{3.2}$$

The CRISP approach does not differ among the predicted misclassified symbolic positions, hence it considered each other symbolic position equally wrong. The performance of the classification method, that is calculated by the CRISP approach, can be represented in a confusion matrix.

3.1.1 Confusion Matrix

The confusion matrix is a special contingency table, where both dimensions contain the same class elements. The confusion matrix displays the frequency distribution of the cases.

TABLE 3.1: Confusion Matrix

Predicted	Actual			Precision
	C_1	C_2	C_3	
C_1	.	.	.	%
C_2	.	.	.	%
C_3	.	.	.	%
Recall	%	%	%	

Table 3.1 shows an example of a confusion matrix with header information. There are numerous statistical indicators based on the confusion matrix [Pow11] such as accuracy, recall, and precision.

$$Accuracy = \frac{\sum_{c \in C} |c_{True}|}{Population} \tag{3.3}$$

Equation 3.3 shows the accuracy of a classifier method, where c_{True} stands for the cases when the actual c is predicted as c for every $c \in C$, and $Population$ denotes all the cases. Thus, the accuracy is the proportion of correctly predicted cases. It does not contain additional information about the classifier method.

$$Recall(c) = \frac{|c_{True}|}{|c_{True}| + |c_{MisPredicted}|} \tag{3.4}$$

Equation 3.4 shows the recall of a given c class, where c_{True} stands for the cases when the actual c is predicted as c, and $c_{MisPredicted}$ denotes the cases when actual c is predicted as not c. The recall is the probability of detection. Thus, it denotes the proportion of the correctly predicted c cases, and the not retrieved c labeled actual cases.

$$Precision(c) = \frac{|c_{True}|}{|c_{True}| + |c_{False}|} \tag{3.5}$$

Equation 3.5 shows the formula for calculating the precision of a given c class, where c_{True} stands for the cases when the actual c is predicted as c, and c_{False} denotes the cases when actual not c is

predicted as c. The precision denotes the proportion of the correctly predicted c cases, and the c predicted, but not actual c cases.

The recall and precision describe the classification method more detailed than the accuracy.

3.2 Capacity

Capacity function determines the size of a room. The capacity function can be defined in both two and three dimensions.

3.2.1 Plane

The capacity of a room in two-dimensional space is called the area. The shape of a room is assumed to be a simple polygon since this is a typical shape, but in other cases a rough approximation can be provided. The Shoelace or Surveyor's area formula [Bra86] calculates the area of a simple polygon with given points in two dimensions as seen in (3.6).

$$A = \frac{1}{2}\left| \sum_{i=1}^{n-1} x_i y_{i+1} + x_n y_1 - \sum_{i=1}^{n-1} x_{i+1} y_i - x_1 y_n \right| \tag{3.6}$$

3.2.2 Space

In three-dimensional space, the capacity of a room is the volume. Each room is treated as a prism shape whose base is a regular n-sided polygon to allow non-cuboid room shapes. The volume of a prism can be calculated based on the area of the base and the distance between the two base faces.

3.3 Distance

Distance functions are used to express the similarity between objects. Similar objects are closer to each other, and their distance approaches zero. Equation (3.7) shows the general form of distance functions.

$$d : O^2 \to \mathbb{R}, \qquad O : Set\ of\ Objects \tag{3.7}$$

A distance function is called metric if it fulfils the following criteria:

$$
\begin{aligned}
&i \quad && d(o_1, o_2) \geq 0 \\
&ii \quad && d(o_1, o_2) = 0 \leftrightarrow o_1 \equiv o_2 \\
&iii \quad && d(o_1, o_2) = d(o_2, o_1) \\
&iv \quad && d(o_1, o_2) + d(o_2, o_3) \geq d(o_1, o_3)
\end{aligned} \tag{3.8}
$$

The distance between two rooms can be measured by their physical location or the length of the path which brings us from one room to the other room.

3.3.1 Coordinate System Distance

Various distance functions can be defined in a coordinate system. In this study, the Euclidean distance is used for calculating the distance of rooms. Two types of reference points can be distinguished to a room.

Centroid

The centroid reference point is the geometric centre of a room, which is a global feature. The coordinates of the centroid can be calculated using (3.9). The centroid is primarily used in two dimensions, but (3.9) can be extended to n dimensions.

$$A = \frac{1}{2} \sum_{i=0}^{n-1} (x_i \, y_{i+1} - x_{i+1} \, y_i) \tag{3.9a}$$

$$C_{\mathrm{x}} = \frac{1}{6A} \sum_{i=0}^{n-1} (x_i + x_{i+1})(x_i \, y_{i+1} - x_{i+1} \, y_i) \tag{3.9b}$$

$$C_{\mathrm{y}} = \frac{1}{6A} \sum_{i=0}^{n-1} (y_i + y_{i+1})(x_i \, y_{i+1} - x_{i+1} \, y_i) \tag{3.9c}$$

Nearest Point

The nearest points are local reference points, hence the nearest point depends on the placement of the two rooms. It can be either the nearest corner points from a room to another or any points of two parallel line segments. The advantage of this reference point is to improve the distance calculation in case of narrow, long rooms.

3.3.2 Graph Model

Topology defines both the room and their connections. So the indoor environment could be mapped to a graph where the nodes are the rooms, and the edges are their connections. In the graph, the distance of two nodes can be measured with their shortest path. The shortest path between two nodes can be found by different algorithms in different graphs. The algorithms can be categorized based on the selection of nodes as single-source or all-pairs algorithms, or based on the direction and the weight of the edges, as seen in Table 3.2.

TABLE 3.2: Shortest path-finding algorithms

Algorithm	Nodes	Edge type	Edge weight
Dijkstra [Lei+01]	Single-source	Directed Undirected	Weighted (Non-negative)
Bellman-Ford [Bel58]	Single-source	Directed	Weighted Unweighted
Floyd-Warshall [Flo62]	All-pairs	Directed Undirected	Weighted

Dijkstra's shortest path algorithm

The most frequently used algorithm is called Dijkstra's algorithm [Lei+01] and is not applicable in negative-cycled graphs. It is a greedy algorithm, which finds the shortest path from a single source node to all other nodes in the graph. It uses two sets, one for visited nodes and the other for not-visited nodes. At every step, the algorithm finds a node that is in the not-visited set and has a minimum distance (weight) from the source, and then the node is moved to the first set. The algorithm can be easily modified to determine the shortest path from source to the target node.

Bellman-Ford shortest path algorithm

The Bellman–Ford algorithm find the global shortest path of a directed graph in maximal $|V| - 1$ number of iterations. The algorithm initializes the distance to the source to 0, and all other nodes to infinity. Then for all edges, if the distance to the destination node can be shortened by choosing the edge, the distance is updated to

the new value. At the *i*th iteration, the edges are examined, and the algorithm finds all shortest paths of maximal *i* length edges. The algorithm stops if in the last iteration, none of the distances could be updates.

Floyd-Warshall shortest path algorithm

The Floyd–Warshall algorithm finds the shortest path of all pairs of nodes in graphs with no negative cycles. The original algorithm only focuses on the length of the shortest path, however the algorithm can be simply modified to get the paths. The shortest path sizes are initialized with the weight of the edges between connected nodes. If two nodes are not connected, infinity is selected to be the size value, and if the two nodes are the same, 0 is the length of the shortest path. The algorithm selects the *k* node to find path from two nodes connected by this node. If the current distance of two nodes are greater, than the sum of the distances between the first and k node, and the last and k node, the sum of the distances will update the distance of the two nodes

3.4 Clustering

Clustering is an unsupervised learning method, whose goal is to discover large groups of objects in the dataset. A group is called a cluster, and the clusters contain similar objects. Cluster elements that belong to the same cluster are more similar while the elements of different clusters are different. This similarity is often measured with some kind of distance function. Objects from different groups are diverse. Clustering methods can be categorized based on their cluster model.

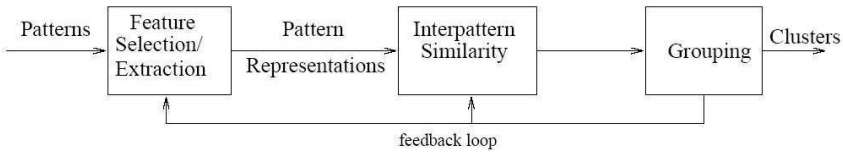


FIGURE 3.1: The steps of clustering process

The clustering process can be seen in Figure 3.1, where the patterns are the objects. The clustering process has three steps which

are feature selection and extraction, interpattern similarity and grouping.

The feature selection process identifies the most effective subset for clustering. The feature extraction make transformations of available features to create new important features. Pattern or object representation refers to "the number of classes, the number of available patterns, and the number, type, and scale of the features available to the clustering algorithm" [JMF99].

The interpattern similarity is based on pattern proximity. The pattern proximity is measured by a distance function defined between two patterns. Different distance function can be defined, but in most cases, the Euclidean distance is used.

The grouping step can be performed in various ways. The hard and fuzzy output of clustering algorithms are distinguished. In hard clustering, each pattern is part of a group exclusively, while in fuzzy clustering, each pattern has a certain degree of membership in each group.

3.4.1 Hierarchical clustering

Hierarchical clustering is a method of cluster analysis, which organizes the elements into a tree structure. It is usually a greedy approach when in each iteration, the local best is selected. The hierarchical clustering is visualized in a dendrogram. Two types of hierarchical clustering can be found, namely top-down and bottom-up.

The top-down type of hierarchical clustering is also called a divisive approach. The top-down approach starts the hierarchical clustering with all the objects in one cluster. Based on different criteria, the cluster is recursively split.

The bottom-up type of hierarchical clustering is also called as agglomerative approach. The bottom-up approach starts the hierarchical clustering with all the objects as an own cluster. Based on different criteria, the cluster is recursively merged.

Linkage method

The linkage method is used to determine the distance between two clusters. It requires the definition of the used distance function, detailed in Section 3.3.. There are several commonly used linkage methods [BA78].

Single linkage Single linkage method is also known as Nearest Point Algorithm. It calculates the distance of two clusters as presented in Equation 3.10, where i is an object in cluster u and j are the objects in cluster v.

$$D(u, v) = \min(d(u[i], v[j])) \tag{3.10}$$

It is a suitable linkage method in the case of well-separable clusters. Due to the usage of minimal distance, if clusters get too close to one another, it tends to link them and possibly form a split within the cluster. It is sensitive to outliers.

Complete linkage Complete linkage method is also known as Farthest Point Algorithm. It calculates the distance of two clusters as presented in Equation 3.11, where i is the objects in cluster u and j is the objects in cluster v. It is sensitive to outliers.

$$D(u, v) = \max(d(u[i], v[j])) \tag{3.11}$$

Average Linkage Average linkage method is also known as Unweighted Pair Group Method with Arithmetic Mean (UPGMA). It calculates the distance of two clusters as presented in Equation 3.12, where i is the objects in cluster u, j is the objects in cluster v and $|u|$ is the cardinality of cluster u and $|v|$ is the cardinality of cluster v. It is less affected by outliers.

$$D(u, v) = \sum_i \sum_j \frac{d(u[i], v[j])}{|u| * |v|} \tag{3.12}$$

Weighted Linkage Weighted linkage method is also known as Weighted Pair Group Method with Arithmetic Mean (WPGMA). It calculates the distance of two clusters as presented in Equation 3.13, where s and t are the two nearest clusters, u is a higher-level cluster combined cluster of s and t and v is the remaining cluster in the forest.

$$D(u, v) = \frac{d(s, v) + d(t, v)}{2} \tag{3.13}$$

Centroid linkage Centroid linkage is also known as Unweighted Pair Group Method with Centroid (UPGMC). In a cluster of points,

the centroid is the point that has the average coordinates of all the objects of the cluster. Instead of calculating the distance based on all the cluster objects, only the centroid point is used. Equation 3.14 shows the distance calculation of cluster s and cluster t, where c_s is the centroid of cluster s and c_t is the centroid of cluster t.

$$D(s,t) = ||c_s - c_t|| \tag{3.14}$$

The new centroid point is calculated using all the objects in the new cluster.

Median linkage Median linkage is similar to the centroid linkage and it is also known as Weighted Pair Group Method with Centroid (WPGMC). It calculates the distance by the same formula as seen in Equation 3.14.

When s and t clusters are merged into a new cluster u, the centroid of the new cluster is the average of the centroid of cluster s and the centroid of cluster t.

Ward linkage The ward linkage method aims to minimize the total within-cluster variance. It is also known as the incremental algorithm. Equation 3.15 shows the formula of the distance calculation, where cluster s and cluster t is merged into cluster u, v is an unused cluster in the forest and $T = |v| + |s| + |t|$ and $|x|$ is the cardinality of any cluster.

$$d(u,v) = \sqrt{\frac{|v|+|s|}{T}d(v,s)^2 + \frac{|v|+|t|}{T}d(v,t)^2 - \frac{|v|}{T}d(s,t)^2} \tag{3.15}$$

Chapter 4

Symbolic Indoor Positioning as Classification Task

Thesis 1. *Room-level indoor positioning can be considered as a classification problem. I created a data set, which allows benchmarking of classification-based symbolic indoor positioning methods.*

Indoor Positioning Systems are usually classified based on the applied technology. The early systems required specific hardware devices for positioning. Active Badge [WH92] system used infrared to locate people. The badges emit unique signals which were received by the installed readers. Active Bats [WJH97] was based on ultrasound technology and it used specific devices that were installed into the ceiling. WALRUS system [Bor+05] also uses ultrasound for room–level positioning.

Smartphones became ubiquitous these days, therefore they are the main client devices of IPS. Smartphones have various built-in sensors which affect on the performance of the system. Movement sensors [Taj14], Magnetometer [Li+12; Bra+14] were used to track the movement of the device in the building. Bluetooth [Fel+03; CGOC17; Gro+19] and WLAN [YA05; Abb+19] interfaces are both used for positioning purposes. Although RFID based solutions [HC10; HWB00; Hig+01; Ni+04; Xu+17] have promising results, RFID reader has not been integrated into smartphones yet.

WiFi-based indoor positioning systems are popular due to their

low installation cost and wide availability. These systems are generally based on client-server architecture and the positioning is performed by the server. The fingerprinting method was presented in the Radar [BP00] system first. The Horus [YA04; YA05] was also based on fingerprinting and it showed that the performance of the system can be improved by client-side filtering techniques. Offline and online phases are usually distinguished by these systems. Site survey is performed in the offline phase which is tedious, time-consuming and costly. Localization service is available in the online phase of the system. Efforts are made to merge the offline and online phases [Wu+13]. Some popular WLAN based indoor positioning system are compared in [KT14].

Data sets are widely used for evaluation and comparison of various machine learning algorithms. The UCI Machine Learning Repository [Lic13] contains more than 300 data sets for various tasks such as classification, clustering and regression. This repository contains two data sets [TS+14; TS+15] related to indoor positioning and the measurements were performed in a multi–building and multi–floor environment. One of these data sets [TS+14] contains approximately 20.000 instances of WiFi fingerprints on almost $110000m^2$. The other data set [TS+15] has about 40.000 instances and it shows the variation of the Earth's magnetic field on a 15 $m \times$ 20 m office space. However these data sets allow the comparison of various indoor positioning methods, but they are limited to one technology.

Hence, a new data set is required, which allows the comparison of indoor positioning algorithm based on multiple sensors. In the following, the construction of the Miskolc IIS Hybrid IPS Data Set is described.

4.1 Data Set

There are various built-in sensors for mobile phones, and the composition of the sensor set depends on the type and the manufacturer of the mobile phone. This composition changes over time due to the current trends and technology innovations.

Table 4.1 shows the usually available sensors of mobile platforms, which is considered in the data set. GPS, Magnetometer, Bluetooth, WiFi and RFID sensors had been used in the data set.

TABLE 4.1: Common Sensors of Android Smart Phones

Sensors	
Name	On Android
GPS	Yes
Infrared	No
Ultrasonic	No
Magnetometer	Yes
Bluetooth	Yes
WiFi	Yes
RFID	No
NFC	Yes

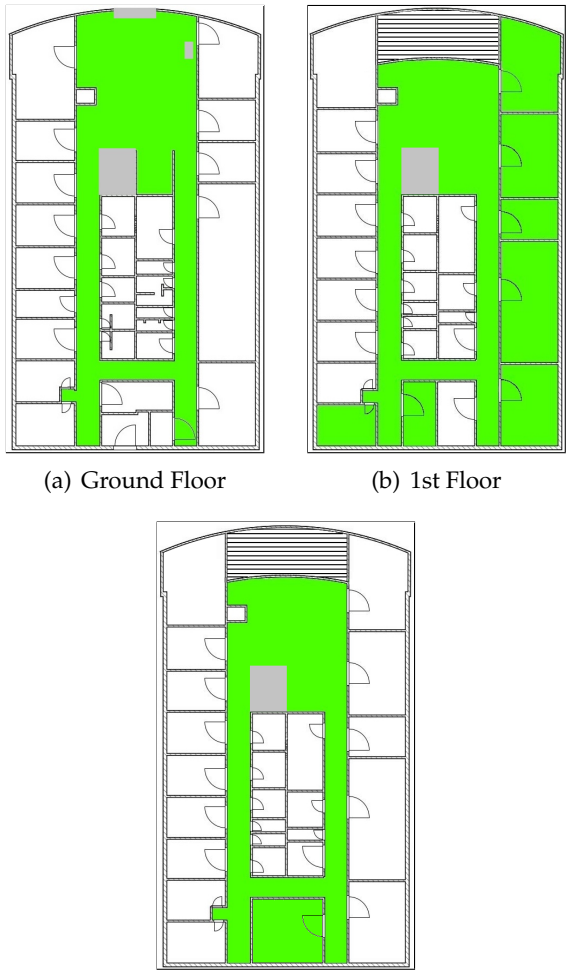
GPS is the only supported sensor for the determination of the building in case of a multi-building environment. The Magnetometer is used due to the magnetic field in any location on the Earth. Due to the wide range of device availability of the Bluetooth sensor, it can provide useful information by scanning the near devices for localization. The established infrastructure of WiFi Access Points makes the usage of the WLAN sensor undeniable. The integration of the NFC sensor has been rejected since the range of the sensor is 10 cm.

Miskolc IIS (Institute of Information Science) Hybrid Indoor Positioning System Data set [7] is available in the UCI Machine Learning Repository [3]. This data set was used to compare the different classification methods in the same environment. The measurements were recorded in a three-story building of the University of Miskolc. The ILONA System was used to record and store the measurements in a database. The construction of the data set was made at the end of February, 2016. The measurements were performed in a weekend in order to reduce the noise.

4.1.1 Environment

The measurements were performed in the Institute of Information Science at the University of Miskolc. Figure 4.2(d) shows the structure of the building. The walls are made of reinforced concrete so GPS signal is usually unavailable within the building. Furthermore, the installation of numerous WiFi Access Points was necessary to provide WLAN access in the building. Four parts of the

floor plan can be distinguished based on their purpose. The offices and the laboratories are placed at both sides of the building and their access is restricted, while the hall and the corridors have public access. Finally, storages and rest rooms are in the centre of the floor and the stairs are on the opposite side of the hall to the entrance.



(a) Ground Floor (b) 1st Floor

(c) 2nd Floor

FIGURE 4.1: Covered area of the Miskolc IIS Building

The data set does not cover the entire building because access to the offices and storage rooms is restricted. Thus, the data set cover

about 50% of the building which is shown in Figure 4.1. Table 4.2 sums up the total and the covered area and their ratio for each floor. The covered area is in between the area of the previously examined databases.

TABLE 4.2: Coverage of the building

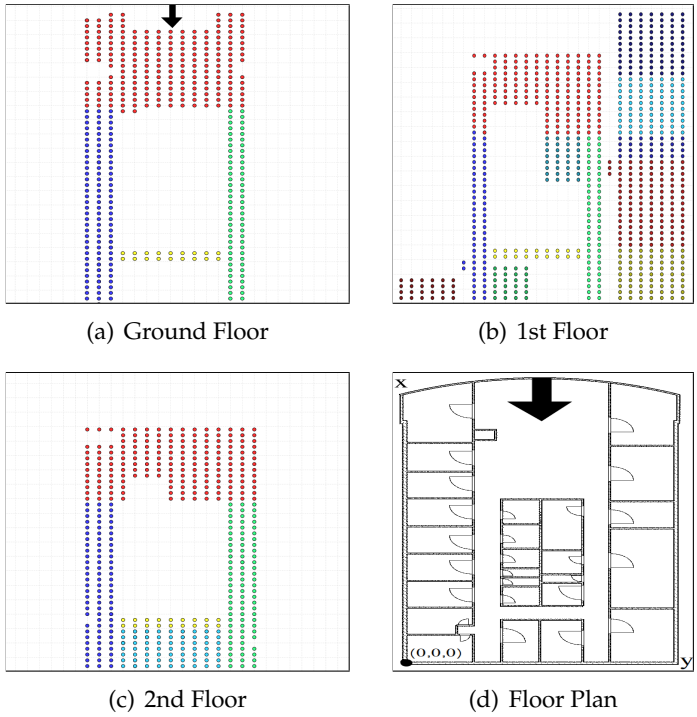
Zone	Area (m^2)	Available (m^2)	Covered (m^2)	Ratio
Ground Floor	1425	1356	436	32.15 %
First Floor	1425	1270	812	63.93%
Second Floor	1425	1270	407	32.04%
Institute	4275	3896	1655	42.48%

The coloured points of Figure 4.2 show the unique absolute position of each measurement. The location of measurements follows a 1 $m \times 1$ m grid-like layout on the building. The colours of Figure 4.2 are used to separate the Zones. The uncovered parts were not available during the recording due to security reasons. There were further limitations on the possible positions for measuring on each floor. The elevator, stairs and built-in cupboards reduce the available area on each floor. Figure 4.2(a) shows the ground floor, Figure 4.2(b) the first floor, Figure 4.2(c) the second floor and Figure 4.2(d) shows the general floor plan of the building.

Figure 4.2(a) shows the positions of the ground floor. The corridors and the hall are approximately $465.75m^2$ and cover about 33% of the building. The reception counter, the elevator, the stairs, and the main entrance reduce the number of measurable positions. The soaring atrium lobby limits the area of the lobbies on the upper floors, thus the corridors and the hall are approximately $345.75m^2$.

Figure 4.2(b) shows the first floor, where approximately half of the measurements were recorded. Laboratories and a lecture hall are placed on one side of the first floor, and they were included in the measurement so the data set covers about 64% of the first floor. The laboratories can be seen on the right side of Figure 4.2(b). The built-in cupboards of the laboratories made a few points inaccessible. There were offices where we had access, which can be seen on the left side of Figure 4.2(b). Finally, a storage room was also measured in the middle of the building, where no GSM signals were available. The stairs and the elevator also occupy an area of the first floor which could not be used during measuring.

Figure 4.2(c) shows the points of the second floor. The second floor also consists of corridors and the lobby area without stairs and elevator. In addition, a lecture hall can be found in the back of the building on the second floor.



(a) Ground Floor (b) 1st Floor

(c) 2nd Floor (d) Floor Plan

FIGURE 4.2: The locations of the measurements and the floor plan, where the colours distinguish each room

The measurements were taken in 21 symbolic positions in the building, called Zones, hence 21 classes are distinguished. The data set contains 1539 measurements, 32 WiFi Access Points, and 20 Bluetooth devices. The distribution of measurements in Zones can be seen in Table 4.3.

4.1.2 Infrastructure

ILONA system [Tot16] was used to record the measurements which is a centralized indoor positioning system detailed in Chapter 2.4 The clients were designed to run with low resources and to be easy to implement due to the huge variety of the smart phones. Thus,

TABLE 4.3: Number of Measured Points in the Zones

Zone Name	# of Positions	Zone Name	# of Positions
Ground Floor West Corridor	68	Ground Floor Lobby	208
Ground Floor North Corridor	18	1st Floor Lobby	151
Ground Floor East Corridor	103	2nd Floor Lobby	177
2nd Floor West Corridor	87	Lecture Hall XXVI	77
2nd Floor North Corridor	18	Lab 101	70
2nd Floor East Corridor	86	Lab 102	28
1st Floor West Corridor	56	Lab 103	108
1st Floor North Corridor	18	Lab 104	63
1st Floor East Corridor	60	Lab 106	24
Lecture Hall 205	63	Office 107b	24
Lab 115	32		

the clients are used to read sensor data and send the measurement to the server for further processing. The measurements are stored in a MySQL database [6].

Samsung Galaxy Young GT-S5360 smart phones were used as clients to collect the measurements. Android 4.4.4 running on the client devices with CynagenMod updates. The application performed a measurement in tree steps. First, the corresponding sensor data was read. Then, the read data were converted into a suitable format for the server and were wrapped. Finally, the wrapper object was serialized in JSON and sent to the server via HTTP.

4.1.3 Data set Description

The measurements were recorded in a MySQL database by the ILONA system [6]. The data set is available as an SQL script so it can be used to create the same environment in other systems. The

database was exported in a standard CSV format which is preferred by many data mining tools.

MySQL Database

Figure 4.3 shows the schema of the database, where the main components are the `Measurement` and the `Position` tables. The measurements contain a timestamp, the positions, and data from various sensors. The timestamp is in the format of `YYYY-MM-DD hh:mm:ss`, and it is generated as the measurement had been stored in the database. In the data set, each position record contains both absolute and symbolic position, both require the manual set by the user. The (x, y, z) absolute coordinates are defined with the base point of the building as seen in Figure 4.2(d). The `Zone` is the symbolic position, which is given by the identifier and the name of the zone. A symbolic position refers to a disjunct part of the building such as "Lab 101", "Office 107B" or "Ground Floor West Corridor". The measurements, positions, and zones have unique identifiers, because the client device generated the identifiers by Java `UUID` class.

The following sensors are supported by the schema. The GPS sensor can determine an approximate position on Earth using the longitude, latitude, and altitude. The magnetometer's measurement is stored as the three components of the direction vector of the magnetic field and the rotation. Some device is able to correct the direction vector by the rotation. Bluetooth devices had a unique hardware address given in hexadecimal format. The RFID reader can scan all the RFID Tag identifier within a range. The result of RFID reader and Bluetooth scan are stored as a set of sensed devices. Finally, the WiFi Received Signal Strength Indicator (RSSI) values are stored as a key-value pair, where the Service Set Identifier (SSID) is the identifier of the WiFi Access Point (AP). The RSSI is represented in a negative form, where the closer the value to zero, the stronger the signal. The RSSI [Wwr] is presented on 8 bit, hence the range of RSSI values is $[-255, 0]$.

The database store the measurements in a normalized schema and the usage of an RDBMS is required. Because these data tend to be analyzed, the database was exported and converted into a CSV format which is easily distributable and supported by almost any data analyst tool.

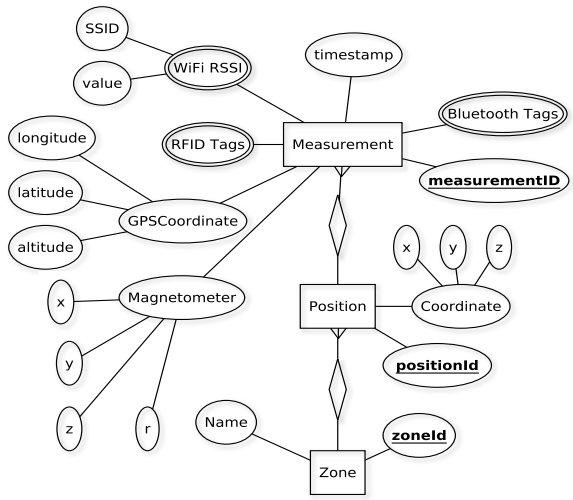


FIGURE 4.3: Schema of the Database

CSV Format

In order to facilitate the data mining, evaluation, and processing tasks, the database of measurements was exported into CSV (Comma-Separated Values) format. The CSV schema for the measurements can be seen in Figure 4.4. However the model supports the RFID values, the client device had no RFID reader, hence it is omitted in the export. Currently, the data set contains only the measurements of a building, hence the GPS values are ignored during the export. The rotation of the magnetometer has been already corrected by the used smartphone. Hence, the rotation has been left out, as the phone always senses it as 0.

Measurement Information		Position Information		Measurements		
ID	Timestamp	Absolute	Symbolic	Magnetometer	WiFi APs	Bluetooth
1	2	3-5	6-7	8-10	11-42	43-65

FIGURE 4.4: CSV schema for the Measurements

Each row represents a measurement except the first row, which is the header. The header contains the following fields: the *ID* of the measurement, *timestamp*, the absolute coordinates called x, y, z, the *name* of the symbolic position, the *Zone ID* of the symbolic position and the magnetic field coordinates denoted with *magnetometerX*, *magnetometerY*, *magnetometerZ*.

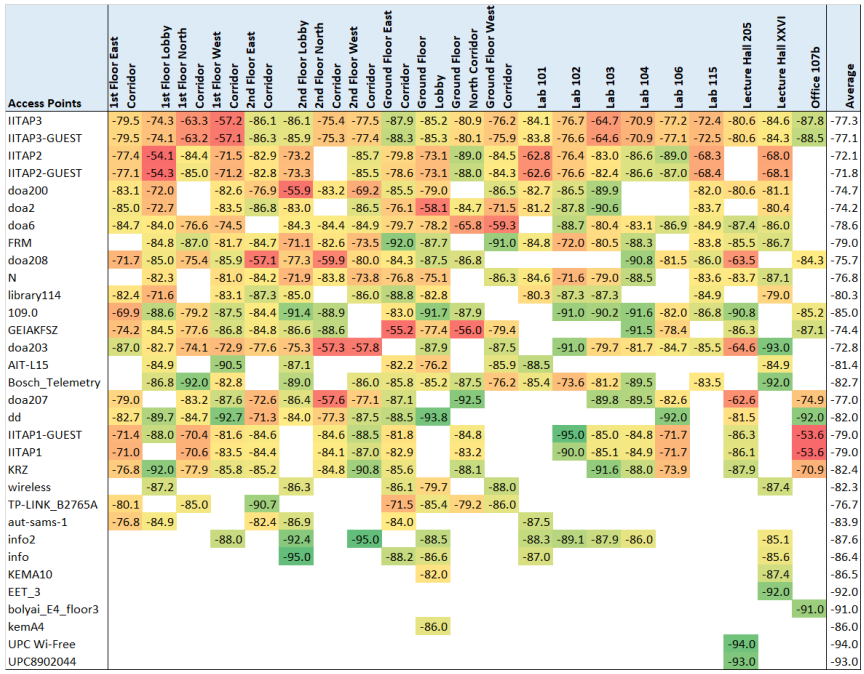
The list of the WiFi Access Points is fixed for each measurement, and the Access Points are presented by their SSID. Fields indexed between 11–42 represent the RSSI value of the corresponding Access Point. In the data set, *null* denotes the missing values in the WiFi RSSI fields.

Access Points	1st Floor East Corridor	1st Floor Lobby	1st Floor North Corridor	1st Floor West Corridor	2nd Floor East Corridor	2nd Floor Lobby	2nd Floor North Corridor	2nd Floor West Corridor	Ground Floor East Corridor	Ground Floor Lobby	Ground Floor North Corridor	Ground Floor West Corridor	Lab 101	Lab 102	Lab 103	Lab 104	Lab 106	Lab 115	Lecture Hall 205	Lecture Hall XXVI	Office 107b	Count
IITAP3	54	148	18	56	26	118	18	85	25	137	18	66	64	28	108	63	24	32	60	39	12	1199
IITAP3-GUEST	54	151	18	56	25	107	18	85	27	134	18	64	65	28	108	63	24	32	61	41	12	1191
IITAP2	48	151	7	56	21	177		45	51	208	1	31	70	28	99	14	1	32		77		1117
IITAP2-GUEST	44	151	7	56	19	177		45	48	207	1	26	70	28	94	11	1	32		77		1094
doa200	35	148		39	82	177	14	87	39	202		11	68	19	10			27	50	76		1084
doa2	12	151		30	5	159		2	103	208	13	68	69	14	5			27		77		943
doa6	3	81	16	51		15	7	56	89	187	18	68		7	102	58	18	18	9	1		804
FRM		78	1	34	38	177	12	85	1	23		2	54	28	104	7		31	47	7		729
doa208	57	88	18	28	85	170	18	77	55	24	8					8	24	1	63		4	728
N		46		23	22	158	9	75	43	155		3	29	26	76	2		17	26	9		719
library114	10	149		20	3	127		2	11	130			70	13	10			18		67		630
109.0	60	86	18	39	64	27	11		79	3	13			1	60	20	24	27	20		16	568
GEIAKF5Z	45	36	18	11	35	7	7		103	183	18	53				4	24		6		11	561
doa203	3	36	11	38	59	143	18	84		14		2		1	49	25	6	6	58	1		554
AIT-L15		106		2		58			60	208		14	13							44		505
Bosch_Telemetry		35	1	25		3		1	32	135	4	55	36	28	95	2		24		1		477
doa207	32		18	18	85	80	18	65	22		2				4	11	24		63		24	466
dd	49	18	3	3	82	160	18	47	13	4							5		62		1	465
IITAP1-GUEST	39	1	18	18	38		13	2	44		14			1	25	34	24		43		24	338
IITAP1	39		18	17	39		13	2	45		11			1	24	37	24		42		24	336
KRZ	49	1	18	19	21		17	4	37		16				5	31	24		36		24	302
wireless		13				22			29	157		4								25		250
TP-LINK_B2765A	32		1		3				102	38	17	1										194
aut-sams-1	25	63			22	40			27				6									183
info2				2		5		1		4			25	9	65	1				37		149
info						1			9	42			9							42		103
KEMA10										1										5		6
EET_3																				2		2
bolyai_E4_floor3																					1	1
kemA4										1												1
UPC Wi-Free																			1			1
UPC8902044																			1			1

FIGURE 4.5: The number of seen WiFi Access Points per zone

The WiFi Access Points were already installed and the WLAN was used for communication too. The number of seen Access Points per zone can be seen in Figure 4.5. The 32 Access Points seem to be far more than necessary to cover the building. But a few of these Access Points belong to the nearby student hostel or other buildings. The others are owned by one of the three departments which are placed in the building. Six of these Access Points were sensed less than 10 times and the most frequently available Access Point was sensed nearly 1200 times. There are Access Points, which had been sensed in every zone, while 5 Access Point was only sensed in one zone. A zone sensed at least 11 Access Points, but the number of detected Access Points does not exceed 24.

The range of the RSSI values is $[-96, -34]$. The average RSSI values of Access Points per zone can be seen in Figure 4.6. The



Access Points	1st Floor East Corridor	1st Floor Lobby	1st Floor North Corridor	1st Floor West Corridor	2nd Floor East Corridor	2nd Floor Lobby	2nd Floor North Corridor	2nd Floor West Corridor	Ground Floor East Corridor	Ground Floor Lobby	Ground Floor North Corridor	Ground Floor West Corridor	Lab 101	Lab 102	Lab 103	Lab 104	Lab 106	Lab 115	Lecture Hall 205	Lecture Hall XXVI	Office 107b	Average
IITAP3	-79.5	-74.3	-63.3	-57.2	-86.1	-86.1	-75.4	-77.5	-87.9	-85.2	-80.9	-76.2	-84.1	-76.7	-64.7	-70.9	-77.2	-72.4	-80.6	-84.6	-87.8	-77.3
IITAP3-GUEST	-79.5	-74.1	-63.2	-57.1	-86.3	-85.9	-75.3	-77.4	-88.3	-85.3	-80.1	-75.9	-83.8	-76.6	-64.6	-70.9	-77.1	-72.5	-80.6	-84.3	-88.5	-77.1
IITAP2	-77.4	-54.1	-84.4	-71.5	-82.9	-73.2		-85.7	-79.8	-73.1	-89.0	-84.5	-62.8	-76.4	-83.0	-86.6	-89.0	-68.3		-68.0		-72.1
IITAP2-GUEST	-77.1	-54.3	-85.0	-71.2	-82.8	-73.3		-85.5	-78.6	-73.1	-88.0	-84.3	-62.6	-76.6	-82.4	-86.6	-87.0	-68.4		-68.1		-71.8
doa200	-83.1	-72.0		-82.6	-76.9	-55.9	-83.2	-69.2	-85.5	-79.0		-86.5	-82.7	-86.5	-89.9			-82.0	-80.6	-81.1		-74.7
doa2	-85.0	-72.7		-83.5	-86.8	-83.0		-86.5	-76.1	-58.1	-84.7	-71.5	-81.2	-87.8	-90.6			-83.7		-80.4		-74.2
doa6	-84.7	-84.0	-76.6	-74.5		-84.3	-84.4	-84.9	-79.7	-78.2	-65.8	-59.3		-88.7	-80.4	-83.1	-86.9	-84.9	-87.4	-86.0		-78.6
FRM		-84.8	-87.0	-81.7	-84.7	-71.1	-82.6	-73.5	-92.0	-87.7		-91.0	-84.8	-72.0	-80.5	-88.3		-83.8	-85.5	-86.7		-79.0
doa208	-71.7	-85.0	-75.4	-85.9	-57.1	-77.3	-59.9	-80.0	-84.3	-87.5	-86.8					-90.8	-81.5	-86.0	-63.5		-84.3	-75.7
N		-82.3		-81.0	-84.2	-71.9	-83.8	-73.8	-76.8	-75.1		-86.3	-84.6	-71.6	-79.0	-88.5		-83.6	-83.7	-87.1		-76.8
library114	-82.4	-71.6		-83.1	-87.3	-85.0		-86.0	-88.8	-82.8			-80.3	-87.3	-87.3			-84.9		-79.0		-80.3
109.0	-69.9	-88.6	-79.2	-87.5	-84.4	-91.4	-88.9		-83.0	-91.7	-87.9			-91.0	-90.2	-91.6	-82.0	-86.8	-90.8		-85.2	-85.0
GEIAKFSZ	-74.2	-84.5	-77.6	-86.8	-84.8	-86.6	-88.6		-55.2	-77.4	-56.0	-79.4				-91.5	-78.4		-86.3		-87.1	-74.4
doa203	-87.0	-82.7	-74.1	-72.9	-77.6	-75.3	-57.3	-57.8		-87.9		-87.5		-91.0	-79.7	-81.7	-84.7	-85.5	-64.6	-93.0		-72.8
AIT-L15		-84.9		-90.5		-87.1			-82.2	-76.2		-85.9	-88.5							-84.9		-81.4
Bosch_Telemetry		-86.8	-92.0	-82.8		-89.0		-86.0	-85.8	-85.2	-87.5	-76.2	-85.4	-73.6	-81.2	-89.5		-83.5		-92.0		-82.7
doa207	-79.0		-83.2	-87.6	-72.6	-86.4	-57.6	-77.1	-87.1		-92.5				-89.8	-89.5	-82.6		-62.6		-74.9	-77.0
dd	-82.7	-89.7	-84.7	-92.7	-71.3	-84.0	-77.3	-87.5	-88.5	-93.8							-92.0		-81.5		-92.0	-82.0
IITAP1-GUEST	-71.4	-88.0	-70.4	-81.6	-84.6		-84.6	-88.5	-81.8		-84.8			-95.0	-85.0	-84.8	-71.7		-86.3		-53.6	-79.0
IITAP1	-71.0		-70.6	-83.5	-84.4		-84.1	-87.0	-82.9		-83.2			-90.0	-85.1	-84.9	-71.7		-86.1		-53.6	-79.0
KRZ	-76.8	-92.0	-77.9	-85.8	-85.2		-84.8	-90.8	-85.6		-88.1				-91.6	-88.0	-73.9		-87.9		-70.9	-82.4
wireless		-87.2				-86.3			-86.1	-79.7		-88.0								-87.4		-82.3
TP-LINK_B2765A	-80.1		-85.0		-90.7				-71.5	-85.4	-79.2	-86.0										-76.7
aut-sams-1	-76.8	-84.9			-82.4	-86.9			-84.0				-87.5									-83.9
info2				-88.0		-92.4		-95.0		-88.5			-88.3	-89.1	-87.9	-86.0				-85.1		-87.6
info						-95.0			-88.2	-86.6			-87.0							-85.6		-86.4
KEMA10										-82.0										-87.4		-86.5
EET_3																				-92.0		-92.0
bolyai_E4_floor3																					-91.0	-91.0
kemA4										-86.0												-86.0
UPC Wi-Free																			-94.0			-94.0
UPC8902044																			-93.0			-93.0

FIGURE 4.6: The average RSSI of WiFi Access Points per zone

loudest average RSSI could be sensed with IITAP1 and IITAP1-Guest in the Office 107b. The average RSSI values of the Access Points are in the range $[-94, -71.8]$.

Results of the Bluetooth scan are placed in the last part of the record. Each position between 43 and 65 has a corresponding device that is identified by a string, which contains its name and MAC address. In these positions, the record contains 1 if the device was within range and 0 otherwise.

Each measurement sensed 0 to 10 Bluetooth enabled devices. On average, about 4 Bluetooth enabled devices were sensed by a measurement. This phenomenon fits the setup of the measurements and can be explained with the relatively short range of Bluetooth. Nine of the Bluetooth devices were installed for the measurement and the others were used by visitors.

The number of seen Bluetooth devices per zone can be seen in Figure 4.7. All of the Bluetooth devices were seen at least 100 times. A Bluetooth device was sensed at least 4 different rooms, while the

Bluetooth Devices	1st Floor East Corridor	1st Floor Lobby	1st Floor North Corridor	1st Floor West Corridor	2nd Floor East Corridor	2nd Floor Lobby	2nd Floor North Corridor	2nd Floor West Corridor	Ground Floor East Corridor	Ground Floor Lobby	Ground Floor North Corridor	Ground Floor West Corridor	Lab 101	Lab 102	Lab 103	Lab 104	Lab 106	Lab 115	Lecture Hall 205	Lecture Hall XXVI	Office 107b	Count
00:16:53:4C:B4:EB	27	82		28	8	168		4	14	103		33	35	13	66			8		37		626
00:16:53:4C:F2:6A	21	82		28	5	163			50	111		33	6	3						25		527
00:16:53:4C:F5:2D	29	82	18	27		7	18	19	2	23	15	13		7	68	35	24	10	36	29	3	465
00:16:53:4C:FA:60	7	82		25	14	176		8		89		33										434
00:16:53:4C:E9:1D		59	13	19	10	159	18	19		21						25			46			389
MrEv3 00:16:53:4C:B4:EB	25	68		27	15			42	5	76		10	33	9				11		30		351
48:5A:B6:54:35:DC		61				122			7	112	9	33										344
00:16:53:4C:B1:F9	26	27	16		13	140	18	19	3								10		45		4	321
00:16:53:4C:B2:02			11	12		90		11	40	65	18	33				23						303
00:16:53:4C:F9:A4	29	45	18	13	9		12	18	43	6	18					31	24		20		7	293
EV3BD 00:16:53:4C:F5:2D	29	36		28	17			50		1		15	28	6	35	27		17	1			290
EV3 00:16:53:4C:F2:6A	18	62			23				15	89		10	31							13		261
00:16:53:4C:FA:67	28	27	14						67	56	18	13					22					245
EV3 00:16:53:4C:FA:60	11	50			64			55		38		10						3		7		238
EV3 00:16:53:4C:E9:1D		1		13	62			64		10					29	14			17			210
6B:C2:26:12:62:60	29	27	18										35				24	13		42	2	190
IZE 00:16:53:4C:B1:F9	29				70			51											17		1	168
JOE 00:16:53:4C:F9:A4	31	11		3	59				22										16		15	157
DM06082 48:5A:B6:54:35:DC		50							2	93		11										156
DANI 6B:C2:26:12:62:60	26				7					1		9	34			1		18		34	2	132
MEGAROBOT 00:16:53:4C:B2:02				13					6	35		35			29	3						121
EV3 00:16:53:4C:FA:67	31								27	34		15									12	119

FIGURE 4.7: The number of seen Bluetooth devices per zone

most seen Bluetooth device was seen in 19 rooms. In a room, the number of detected devices is between 4 and 18.

4.2 Benchmarking with Classifiers for Indoor Positioning

To develop a classification-based room level indoor positioning method, the performance evaluation of some well-known classifiers is necessary. The Miskolc IIS Hybrid Indoor Positioning System Data set [7; 3] was used as the data set during the evaluation process. The following classification methods were tested: *k*–NN, Naive Bayes, Decision Tree, ANN, and Rule induction. This section gives a brief overview of the classifiers tested and the setup of the training and validation data sets.

4.2.1 Evaluation Process

The evaluation process was performed with RapidMiner with the exception of the Multilayer Perceptron (MLP) because its topology was optimized with a genetic algorithm. The Multilayer Perceptron was implemented by the Weka Framework.

Figure 4.8 shows the flowchart of the evaluation process. The process retrieves the Miskolc IIS Hybrid IPS data set detailed in

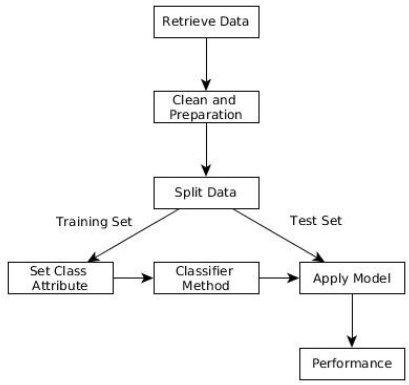


FIGURE 4.8: The steps of the evaluation process

Section 4.1.3 in CSV format. The redundant, technical or not relevant attributes are eliminated by dimension reduction. The missing RSSI values are replaced with an out-of-range value. In the data set, -100 was chosen to represent unreachable Access Points. However, the measured values are accepted in their original form, the inaccuracy of the measured values is not examined. Then the data set is split into training and test sets. The label role is set on the Zone ID field, which marks the class attribute, then the classifier model is built with the training set. In the Apply Model step, a category is predicted for each record of the validation set based on the classifier model. Then the performance vector of the classifier is calculated.

The main factor during the evaluation of each model was the classification accuracy based on the validation samples. It provides us with more certainty that an unknown measurement will be classified correctly.

$$Accuracy(c) = \frac{|c_{Correct}|}{|c_{Correct}| + |c_{Incorrect}|} \tag{4.1}$$

The formula in Equation 4.1 is the accuracy of a given c class. $c_{Correct}$ stands for the cases when the actual c class is predicted as c, and $c_{Incorrect}$ denotes the cases when actual c is predicted as not c. To determine the accuracy of the model based on the performance vector, the proportion of correct predictions is calculated. The comparison is based on these accuracy rates.

Training and validation sets

The data set is partitioned into the training and the validation subsets. The schema for training and validation sets can be seen in Figure 4.9. The records of each set contain attributes related to the sensors and the id of the symbolic position as category label. The training set is used for building each classifier, and the classifier is tested by the validation set.

Attributes			Category
Magnetometer	WiFi Access Points	Bluetooth Devices	Zone ID
1-3	4-32	33-54	55

FIGURE 4.9: CSV schema for the training and validation set

Due to the small data set, the training and validation sets were built by the stratified sampling of the data set with a 0.9 and 0.1 ratio. The classifiers are not tested with records contained in the training set. So the evaluation shows how well the classifier works with unknown objects.

RapidMiner

RapidMiner [HK13] is an open-source, cross-platform data mining software implemented in Java. RapidMiner can be used for statistical analysis, data mining and predictive analytics. The evaluation process can be implemented with the built-in components of RapidMiner. The graphic user interface can visualize the performance with a confusion matrix [Con] and the accuracy of each method tested. RapidMiner was also used to create the training and validation sets for programmatic use of the Weka.

Weka

The Artificial Neural Network-based classifiers were tested in RapidMiner but the optimization of the topology was considered to be difficult by the software. Thus a custom Java application was developed to optimize the topology of ANN. The implementation of ANN was provided by the Weka (Waikato Environment for Knowledge Analysis) [Hal+09] Framework. The Weka Framework provides access to a collection of machine learning algorithms and

tools for evaluation. To optimize the topology of ANN a genetic algorithm was used.

TABLE 4.4: Parameters of Genetic Algorithm

Number	Population Number	Iteration	Mutation Rate
1.	400	50	0.8
2.	400	50	0.6
3.	400	25	0.8
4.	200	25	0.9
5.	200	50	0.4

Table 4.4 shows the parameters used during genetic algorithms. The number of elite entities is 5% of the population and these entities are carried over to the next population unaltered. The genetic algorithm uses roulette-wheel selection to determine the entities for breeding. *Training time*, *learning rate*, *momentum*, and *hidden layers*, the parameters of the multilayer perceptron, are optimized with genetic algorithms.

4.2.2 Tested Classifiers

This work focuses on the performance analysis of well-known classification methods for room-level indoor positioning. Decision tree, *k*–NN, Rule Induction, Naive Bayes, and Artificial Neural Network classifiers were analysed and evaluated. These well-known classifiers can be categorized into Instance-based and Model-based learning approaches.

An instance-based classifier predicts the class based on the unmodified training instances. Therefore the classifier does not need retouch in case of new training instances. However, the time complexity is growing with the increasing number of training instances ($O(n)$). The *k*–NN and the Naive Bayes classifiers are instance-based.

A model-based classifier constructs a deterministic model by the training instances. The time complexity of model-based classifiers is $O(1)$ or $O(log\ n)$. Although a new training instance requires the rebuilding of the model. The Decision Tree, the ID3, the Rule Induction, and the Artificial Neural Network classifiers belong to this category.

The following part of this section gives a brief overview of the selected methods.

Naive Bayes

The Naive Bayes classifier [JL95] is a group of probabilistic classifiers based on Bayes' theorem. Naive Bayes is a conditional probability model that assumes the independence of the random variables. The given measurement is represented by a $X = (X_1, X_2, ..., X_n)$ vector. Equation 4.2 shows the calculation of the probability that the X observation is classified as k class, where C_k represents the kth class. The observed object is classified as the class with the maximum probability value.

$$p(C_k|\mathbf{x}) = \frac{p(C_k)\ p(\mathbf{x}|C_k)}{p(\mathbf{x})} \tag{4.2}$$

The *Naive Bayes (kernel)* uses a weighting function in non-parametric estimation techniques. The *Naive Bayes* and the *Naive Bayes (kernel*) components require a training set, and the output of each component is a classifier model. The *Naive Bayes* component only supports the *Laplace correction* parameter, which indicates if the zero probabilities should be prevented. The *Naive Bayes kernel* component also defines additional parameters. The *estimation mode* parameter sets the kernel density estimation mode. The *minimum bandwidth*, the *bandwidth* and the *number of kernels* are further parameters of estimation modes. The *use application grid* indicates if the kernel density function grid should be used.

k–Nearest Neighbour

The *k*–Nearest Neighbour [CD07] is a non-parametric method used for classification. The *k*–NN method searches for the *k* most similar samples to the input in the sample set. The similarity is calculated based on an arbitrary distance function such as Euclidean distance. The category is determined by a majority vote. The *k* parameter is a positive integer that determines the number of the neighbours, and it is suggested to be odd to prevent the bi-valence of the major vote. If *k* is set to 1, the output is simply the class of the nearest neighbour, thus the classification problem becomes a minimum search task.

The input of *k–NN* component is a training set, and the output is a classifier model. The following parameters of the *k–NN* are adjusted to achieve higher accuracy. The *k* parameter is the number of nearest neighbours, which are used to determine the category by

majority voting. The *weighted vote* indicates the usage of different voting, by calculating the weight for each neighbour based on its distance from the unknown object. The *measure types* parameter is used for selecting the type of measure to be used for finding the nearest neighbours. The *mixed measure, nominal measure, numerical measure* parameters declare the distance function used.

Multilayer Perceptron

The Multilayer Perceptron [PM92] is a feed-forward artificial neural network that has various applications such as function approximation and classification. Neural networks are usually considered as a black box whose input is a vector that represents the object, and the output is the estimated value or category.

Multilayer Perceptron is a Fully Connected Network, where each node is a neuron with a non-linear activation function, except the input nodes. The input layer is the known attributes of the object. The output layer contains nodes for each class attribute. Artificial Neural Networks are popular because they are easy to use, but their tuning can be difficult and time-consuming.

The *Neural Net* component is based on a multilayer perceptron and requires a mapped training set with numerical categories, and the output is a classifier model. The neural net classifier model-based prediction requires remapping for calculating the performance. There are numerous parameters of the Neural Network. The *hidden layers* parameter determines the number of nodes in each hidden layer, separated with a comma. The training cycles or *training time* is the number of training epochs, which is a forward and a backward pass of all the training examples. The *learning rate* determines the convergence of the multilayer perceptron. The *momentum* parameter is used to reduce the fluctuations in weight changes.

Decision Tree and ID3

The decision tree is a decision support tool, with the advantage to visualize the decision-making process. The internal nodes test attributes and each branch represent a decision, while the leaves denote categories. ID3 [Hss+14] is an algorithm that is used to generate a decision tree, but it cannot handle continuous attributes. For each unused attribute it calculates the entropy of the subset, then selects the attribute with the smallest entropy value, and splits the

subset based on it. It repeats the previous step on every subset until one of the following exit criteria is fulfilled: there are no more attributes to select, the subset belongs to the same class or no examples are left in the subset. It uses a greedy approach by selecting the best variation in every step, which can lead to local optima.

The *Decision Tree* and the *ID3* components require a training set, and the output of each component is a classifier model. The *Decision Tree* component is based on the C4.5 [Hss+14] algorithm, which can handle missing values. The *Decision Tree* component and the *ID3* component have common parameters. The *criterion* parameter selects the metric of attributes for splitting, like *information gain* or *gain ratio*. The *minimal gain* parameter determines the threshold for split based on the gain value of the node. The *minimal leaf size* is the number of minimal examples in its subset. The *minimal size for split* determines that only those nodes are split whose size is no less than this parameter. The *ID3* component learns an unpruned decision tree. The *Decision Tree* and the *ID3* components have the same parameters, which are the following. The *maximal depth* is the termination condition of the tree building process, which determines the maximal number of levels of the tree. *Apply pruning* and *apply prepuning* parameters enable or disable the pruning and prepruning process on the tree. The *number of prepuning alternatives* adjusts the number of alternative nodes tried for splitting when the split of a certain node is prevented. The *confidence* parameter is the confidence level used in the pruning process.

Rule Induction

Rule Induction [GB05] is a supervised learning method, where each case has a labelled class attribute. The outcome of the rule is the predicted class, and the conditions are the path along the leaf node of the prediction. Starting with the less dominant classes, the algorithm iteratively expands and abridges rules until there are no positive examples left or the error rate is greater than 50%.

The *Rule Induction* component requires a training set, and the output of the component is a classifier model. The parameters of Rule Induction are the following. The *criterion* parameter selects the metric of attributes for splitting, like *information gain* or *accuracy*. The *sample ratio* defines the ratio of training data for growing and pruning. The *pureness* is the minimum ratio of the major class

in the covered subset. The *minimal prune benefit* parameter determines the amount of benefit which is required to be pruned over unpruned. The *use local random seed* enables or disables the usage of *local random seed* for randomization.

4.2.3 Experimental Results

This section presents the achieved accuracy of the selected methods. The Artificial Neural Network is denoted as ANN, the *k*–NNW represents the weighted vote variant of the *k*–NN. Decision tree with C4.5 algorithm is denoted as Decision Tree, while with the ID3 algorithm is denoted by ID3 according to the RapidMiner component name.

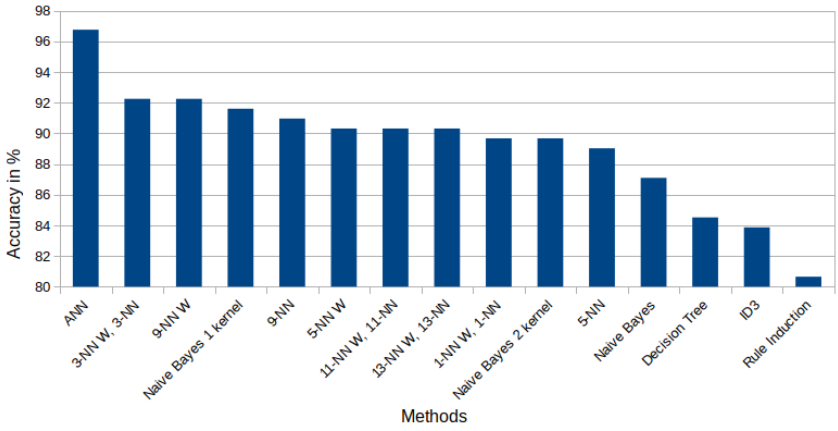


FIGURE 4.10: Performance of the *k*–NN, Rule Induction, Decision Tree, Artificial Neural Network and Naive Bayes

Figure 4.10 shows the accuracies achieved during the experiments with *k*–NN, Naive Bayes, Rule Induction, Artificial Neural Network, and Decision Tree classification methods.

Most of the methods tested performed between 85% and 90% accuracy. Among the methods tested in Rapid Miner, the *k*–NN classifier had the best performance (92.26%) when the *k* parameter was 3 and uses weighted major vote, as is shown in Figure 4.10 and denoted by 3-`NN W`. The Artificial Neural Network could perform 96.77% accuracy with 0.9 learning rate, 0.5 momentum, 380 training time and 1 hidden layer with 30 neurons.

TABLE 4.5: Ranking of the tested classifiers

#	Method	*Accuracy*	#	Method	*Accuracy*
1.	ANN	96.77	7.	13–NN	90.32
2.	3–NN W	92.26	12.	1–NN W	89.68
2.	3–NN	92.26	12.	1–NN	89.68
2.	9–NN W	92.26	14.	Naive Bayes 2 kernel	89.68
5.	Naive Bayes 1 kernel	91.61	15.	5–NN	89.03
6.	9–NN	90.97	16.	Naive Bayes	87.10
7.	5–NN W	90.32	17.	Decision Tree	84.52
7.	11–NN W	90.32	18.	ID3	83.87
7.	11–NN	90.32	19.	Rule Induction	80.65
7.	13–NN W	90.32			

Naive Bayes

As can be seen in Figure 4.10 the Naive Bayes classifier performed at 87.1% accuracy, which could be increased 91.61% accuracy by the application of one kernel density function. The number of the individual kernel density functions determines the smoothing. The values of the parameters do not change the accuracy significantly; most of the built models resulted in 91.61% accuracy.

k–Nearest Neighbour

The performance of the *k*–NN algorithm depended on the k parameter as seen in Table 4.5. Due to the characteristics of the training and validation sets, each method tested was used with the Mixed Euclidean Distance function.

The 3–NN method with and without weighted voting achieved the highest, 92.26% accuracy among the *k*–NN methods tested. The weighted versions of *k*–NN methods reached at least the same accuracy as the majority vote versions. The amount of increase achieved by weighted vote is 1.5% in the case of 5–NN and 1.4% of 9–NN method. The 3–NN, the 11–NN performed the accuracy regardless of the method of voting. The *k*–NN method had the lowest performance with 89.03% when the k parameter was chosen to be 5 and the vote type is simple major.

Multilayer Perceptron

The multilayer perceptron was first tested with RapidMiner, and it achieved 96.77% with 0.9 learning rate, 0.5 momentum, 380 training times 380 and 30 hidden nodes in one layer. To optimize the topology and learning parameters of the perceptron, a genetic algorithm, seen in Table 4.4, the Weka framework was used. Weka framework provides the `MultiLayerPerceptron` class that implements an Artificial Neural Network based classifier.

Five genetic algorithms were used to determine the best global parameters of Multilayer Perceptron, as it can be seen in Table 4.4. The result of these genetic algorithms can be seen in Table 4.6, where the 100 most accurate neural networks of the last iteration for each genetic algorithm were analysed. The best neural network of all cases achieved 100% with 0.464 learning rate, 0.458 momentum, 217 training epoch and 24 hidden nodes.

TABLE 4.6: Top 100 Most Accurate Artificial Neural Network of the Genetic Algorithms

		Case 1	Case 2	Case 3	Case 4	Case 5
Accuracy (%)	Min	93.35	99.49	99.64	99.42	99.21
	Max	99.78	99.86	100	99.86	99.71
	# of Max	9	2	1	3	4
Hidden Nodes (#)	Min	15	16	21	20	14
	Max	17	18	24	24	17
	Most Frequent	16	17	22	22	15
Training Time (epoch)	Min	180	189	198	206	171
	Max	218	217	230	224	213
	Most Frequent	195	200	210	214	196
Learning Rate $]0,1]$	Average	0.488	0.48	0.463	0.497	0.456
	Most Frequent	0.488	0.481	0.464	0.497	0.46
Momentum $]0,1]$	Average	0.435	0.407	0.457	0.452	0.375
	Most Frequent	0.433	0.405	0.454	0.452	0.377

The first genetic algorithm achieved the accuracy in a range of $99.35 - 99.78\%$, and 9 neural networks achieved the maximum accuracy of this case. The number of hidden nodes ranged between

$15 - 17$, and the most frequent hidden node number was 16. The training time of these Multilayer Perceptrons is 180 to 218 epochs, while the most frequent parameter is 195. The average learning rate is 0.488 with 0.488 as the most frequent learning rate, while the average momentum is 0.435 with 0.433 as the most frequent momentum.

The second genetic algorithm achieved the accuracy in a range of $99.49 - 99.86\%$, and 2 neural networks achieved the maximum accuracy of this case. The number of hidden nodes ranged between $16 - 18$, and the most frequent hidden node number was 17. The training time of these Multilayer Perceptrons is 189 to 217 epochs, while the most frequent parameter is 200. The average learning rate is 0.48 with 0.481 as the most frequent learning rate, while the average momentum is 0.407 with 0.405 as the most frequent momentum.

The third genetic algorithm achieved the accuracy in a range of $99.64 - 100\%$, and 1 neural network achieved the maximum accuracy of this case. The number of hidden nodes ranged between $21 - 24$, and the most frequent hidden node number was 22. The training time of these Multilayer Perceptrons is 198 to 230 epochs, while the most frequent parameter is 210. The average learning rate is 0.463 with 0.464 as the most frequent learning rate, while the average momentum is 0.457 with 0.454 as the most frequent momentum.

The fourth genetic algorithm achieved the accuracy in a range of $99.42 - 99.86\%$, and 3 neural networks achieved the maximum accuracy of this case. The number of hidden nodes ranged between $20 - 24$, and the most frequent hidden node number was 22. The training time of these Multilayer Perceptrons is 206 to 224 epochs, while the most frequent parameter is 214. The average learning rate is 0.497 with 0.497 as the most frequent learning rate, while the average momentum is 0.452 with 0.452 as the most frequent momentum.

The fifth genetic algorithm achieved the accuracy in a range of $99.21 - 99.71\%$, and 4 neural networks achieved the maximum accuracy of this case. The number of hidden nodes ranged between $14 - 17$, and the most frequent hidden node number was 15. The training time of these Multilayer Perceptrons is 171 to 213 epochs, while the most frequent parameter is 196. The average learning rate is 0.456 with 0.46 as the most frequent learning rate, while the

average momentum is 0.375 with 0.377 as the most frequent momentum.

Decision Tree

As can be seen in Figure 4.10, the Decision Tree method performed 84.52% while the ID3 method achieved 83.87% accuracy. The split criteria is set to *gain_ratio*, which solves the drawback of information gain, that attributes with a large number of distinct values might learn the training set too well. The set of minimal leaf size parameter is 2, while the minimal gain is 0.1 in both classifiers. The Decision tree are parametrized with 0.01 confidence level and the minimal size for split is 4 in ID3.

Rule Induction

Figure 4.10 shows that the Rule Induction method achieved 80.65% accuracy. The split criterion of Rule induction is *information_gain*. The pureness parameter is set to 1.0 so only one class is covered in the subset, and the sample ratio is also set to 1.0. The minimal prune benefit is set to 0.9 to be the threshold of pruning a rule.

4.2.4 Discussion

Based on the experimental results, the following observations can be drawn for each classifier tested. Simulations show that Rule Induction and Decision Tree classifiers are not suggested due to their low accuracy. The Naive Bayes achieved an acceptable accuracy, while the *k*–NN and ANN managed to earn the best results.

The Decision Tree can handle both nominal and numerical types. Accuracy is influenced by the variation in data types. Decision Tree using nominal WiFi RSSI values achieved higher accuracy than using numerical WiFi RSSI values. Even with the highest accuracy that the Decision Tree achieved, it underperformed 55% of tested methods.

The ID3 algorithm can only handle continuous attributes, and the WiFi RSSI attributes of the measurement are continuous numerical values. Hence, conversion to nominal type is required in the classifier building process. The usage of the ID3 classifier is limited due to the fact that the conversion of unknown measurements is also necessary for predicting the category. In the case of

fulfilled conditions, the ID3 tree could not outperform the other well-known methods.

Rule Induction can handle both nominal and numerical types. Rule Induction performed better with numerical WiFi RSSI values. However, Rule Induction has the lowest accuracy of the methods tested.

Although the Naive Bayes classifier achieved 87.10% accuracy during the test, 79% of the tested methods outperformed it. The usage of kernel functions could increase the accuracy up to 91.61%. This performance was enough to be only the fourth worst choice among the selected methods during the tests, regardless of the application of kernel functions.

The *k*–NN algorithm outperformed most of the other classification methods during the tests and achieved 92.26% accuracy. The performance of the *k*–NN classifier strongly depends on the *k* parameter. 3–NN achieved the highest accuracy among the *k*–NN methods and even among all tested methods. Furthermore, distance-based weighting could increase its accuracy. Based on the experimental results, the *k*–NN is a good candidate to be used for indoor positioning purposes.

The usage of ANN method is highly recommended based on the experimental results. Artificial Neural Networks achieved 100% and 99.86% accuracy, although their training and the finding of the best parameters could be time-consuming and challenging. On the other hand, ANN classifiers have to be retrained if the training set changes. The training time of ANN strongly depends on the topology and the training time. The topology is the number of hidden layers and nodes. The training time denotes the number of epochs during the training.

As an overall observation, one classifier can be highlighted from each classifier category for indoor positioning purposes. From the instance-based classifiers, the *k*–NN method seems to be the best classifier. In the case of model-based classifiers, the ANN achieved outstanding results among them.

From the viewpoint of performance of Indoor Positioning Systems, the time complexity of the classification methods can be important too. The time complexity of *k*–NN classification algorithm is $O(n)$, where n is the number of samples that could limit its applicability in real-life scenarios. The time complexity of prediction of the ANN classifiers is constant ($O(1)$), i.e. the time cost of classification is independent of the number of instances in the database.

Although the time complexity of the classification methods was not analysed in the current work, it was taken into account during the making of suggestions. The ANN classifier significantly faster than the *k*–NN, which can improve the user experience in a real-life scenario. The *k*–NN algorithm is recommended because it could achieve high accuracy, it is simple and does not require modification when the training set changes. In addition, the ANN method is also recommended due to its high accuracy and fast response time. In cases when the data set is dynamically changing, *k*–NN is the preferable classifier choice due to the building time of the ANN. However, if the data set is static, and the fast response time is a criterion, ANN is the recommended classifier.

To sum up the observations, based on the test performed the usage of *k*-NN and ANN classifiers are recommended for indoor positioning purposes.

4.3 Conclusions

In the time of the construction of the Miskolc IIS Hybrid IPS Data Set, there was a need for such a data set, because no data set was available that contains multiple sensor data for indoor positioning purposes. The Miskolc IIS Hybrid IPS Data Set was cited by researchers from Spain, Ecuador, France and USA, and it was recognised by other researchers from Italy, Iran, Pakistan, and Malaysia since 2017. The data set allows the comparison and evaluation of indoor positioning algorithms.

Well-known classifiers were evaluated over the Miskolc IIS Hybrid IPS Data set. Both instance-based and model-based approaches were examined, namely Decision Tree, *k*–NN, Rule Induction, Naive Bayes and Artificial Neural Network. Experimental results showed that the *k*–NN and the ANN classifiers could be used for indoor positioning purposes. The *k*–NN with k set as 3 achieved 92.26% accuracy, but the time complexity is $O(n)$. The ANN optimized with genetic algorithm could achieve 96.77%, although it requires the rebuilding of the model in case of new training instance. Hence, both an instance-based and a model-based classifier could be applied efficiently to indoor positioning purposes.

Since the construction of the Miskolc IIS Hybrid IPS Data Set, there are new data set for indoor positioning purposes, but they mainly use one technology. For example, the MagPIE [Han+17]

data set uses magnetometer measurements and inertial measurement unit values. The measurement was performed in a multi-building environment in a large, $960m^2$ total test area. Another example is a crowdsourced WiFi data set [Loh+17], which contains 4648 fingerprints collected with 21 devices. It was recorded in a five-floor building with a footprint of about $22570m^2$. The last example is a data set [Bar+16], which uses multiple sensor and multiple source. It uses WiFi and geo-magnetic field fingerprints with additional inertial sensor data from smartwatch and smartphone.

Thesis 1.

Room-level indoor positioning can be considered as a classification problem. I created a data set, which allows benchmarking of classification-based symbolic indoor positioning methods.
Related Publications: [6], [9], [7], [4].
Citations: [CGOC18], [CGOC17], [MS+19], [Sat18], [MS+18], [Bog17], [Fen+20], [YZZ20], [SS+20], [NMN20], [AM20], [Elg+20], [Ara+20], [MGT18]

Chapter 5

Topology-based Evaluation

Thesis 2. *I proposed a topology-based evaluation method for classification-based indoor positioning algorithms which allows a more detailed evaluation.*

In a further examination of experimental results produced by symbolic indoor positioning methods, a remarkable behaviour can be detected. A more accurate classification method predicts further the symbolic positions from the original location when it is misclassified, than a less accurate classifier. Furthermore, less accurate classifiers often predict the neighboring, and nearby symbolic position. In conclusion, a different approach is recommended to be considered as an alternative of CRISP.

5.1 Further Experiment

To revise the conclusion about the most suitable classifier, the extended evaluation process is performed in each case of classifier tested. The Miskolc IIS Hybrid IPS Dataset, detailed in Section 4.1.3 was also the base of the extended evaluation. The confusion matrix created by RapidMiner software was used to calculate the metrics.

The dataset is partitioned into training and test set, and the distribution of instances in the training and test sets is proportional to the Zone distribution of the whole dataset. The comparison rests on the accuracy metric of the CRISP approach defined in Equation 3.3 based on the confusion matrix shown in Table 3.1.

Besides the overall accuracy, small areas of the environment are chosen to observe the classification errors more detailed. To detect the challenging areas, the base of selection is the number of training

points and the density of the Zones. The vertical aligned Zones are not considered to be neighbours and the lobbies are dismissed due to the soaring atrium. We choose two, disjunct groups of Zones to examine the tested classifiers. Each group consists of four Zones, where at least one Zone is enclosed by the others.

5.1.1 Extended Results

The overall result of the experiment can be seen in Table 5.1. The classifiers are in descending order based on overall accuracy. There are additional information about the classifiers, like the number of misclassified cases. The misclassified cases are categorized based on the distance of the actual and predicted Zones, namely *Close* and *Far* prediction, using domain knowledge. A misclassification is called *far*, when the predicted zone has no common neighbour with the actual zone. In the case of *k*–NN classifiers, the *W* suffix denotes the distance-based weighted variant.

TABLE 5.1: Summary of Tested Classifiers

Name	Accuracy in %	Miss	Close	Far	Close ratio	Far ratio
ANN	96.77	5	4	1	0.8	0.2
3NN W	92.26	12	8	4	0.67	0.33
9NN W	92.26	12	8	4	0.67	0.33
Naive Bayes 1 kernel	91.61	13	10	3	0.77	0.23
9NN	90.97	14	8	6	0.57	0.43
5NN W	90.32	15	10	5	0.67	0.33
11NN W	90.32	15	12	3	0.8	0.2
13NN W	90.32	15	11	4	0.73	0.27
13NN	90.32	15	11	4	0.73	0.27
1NN W	89.68	16	12	4	0.75	0.25
5NN	89.03	17	11	6	0.65	0.35
Naive Bayes	87.10	20	15	5	0.75	0.25
Decision Tree	84.52	24	11	13	0.46	0.54
ID3	83.87	25	15	10	0.6	0.4
Rule Induction	80.65	30	18	12	0.6	0.4

The most accurate classifier is the Artificial Neural Network in the experiment. It misclassified the least cases among the tested

classifiers, and the number of far prediction is 1. The 3–NN *W* and 9–NN *W* classifiers achieved the highest accuracy among the instance-based classifiers. Each classifier fairly missed 4, and slightly missed 8 cases. The classifier with the lowest accuracy is Rule Induction. It misclassified 30 cases, where 12 was a far miss.

The *9NN* classifier missed more cases far away, than the *Naive Bayes* classifier, although the *9NN* achieved 3.87% higher accuracy. In the following sections, the examination of these two classifiers is taken place in two different selected areas.

5.1.2 1st Case

The first case was selected to be on the first floor, because it is the most covered floor. Besides the lobbies, the *Lab 103* has the most measured points. Thus, the *Lab 103*, and its direct neighbours have been highlighted. The selected Zones, and their layout can be seen in Figure 5.1. These Zones are called *Lab 102*, *Lab 103*, *Lab 104* and *1st Floor West Corridor*.

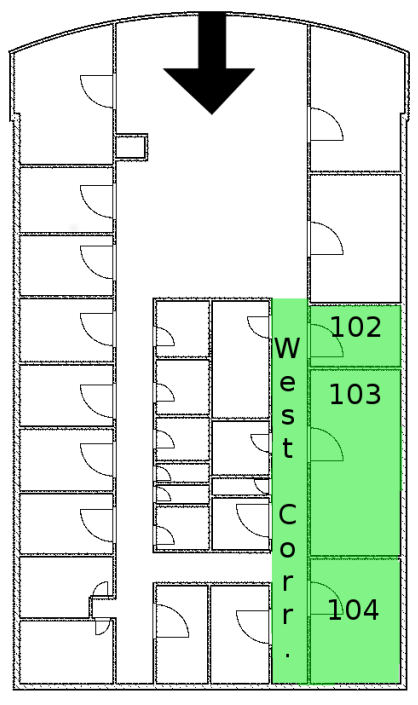


FIGURE 5.1: Selected Zones of 1st Case

Evaluation

The observed part of the confusion matrix of 9–NN classifier can be seen in Table 5.2, and the confusion matrix for Naive Bayes classifier can be seen in Table 5.3. The *1st Floor West Corridor* is denoted in the Table 5.2 with its abbreviated form. The four highlighted Zones have a total of 26 test cases, where the number of test cases for each Zone is proportional to its area.

TABLE 5.2: Confusion Matrix of 1st Case Selected Zones with 9NN classifier

Actual	Predicted						Total Result
	1st Floor West Corr.	Lab 102	Lab 103	Lab 104	Other Close	Far	
1st Floor West Corr.	**4**				1	1	6
Lab 102	1	**0**	1	1			3
Lab 103			**10**	1			11
Lab 104				**6**			6
Total Result	5	0	11	8	1	1	26

9-NN As can be seen in Table 5.2, the number of misclassified cases is 6, thus the accuracy of these four Zones is 79.62%. Although, only one case had been predicted to be a far Zone among the highlighted Zones. The lowest recall valued Zone is the *Lab 102* with 0%, thus all the cases are misclassified. And none of the cases are classified as this Zone. The *Lab 104* Zone achieved the highest recall with 100%, so each case with actual *Lab 104* is classified correctly. However, two cases are predicted incorrectly as the *Lab 104* Zone.

Naive Bayes As can be seen in Table 5.3, the number of misclassified cases is 7, thus the accuracy of these four Zones is 73.08%. Although, only one case had been predicted to be a far Zone among

TABLE 5.3: Confusion Matrix of 1st Case Selected Zones with Naive Bayes classifier

Actual	Predicted						Total Result
	1st Floor West Corr.	Lab 102	Lab 103	Lab 104	Other Close	Far	
1st Floor West Corr.	**2**				3	1	6
Lab 102		**3**					3
Lab 103			**11**				11
Lab 104			3	**3**			6
Total Result	2	3	14	3	3	1	26

the highlighted Zones. The lowest recall valued Zone is the *1st Floor West Corridor* with 33%, thus two-third of the cases are misclassified. The *Lab 102* and the *Lab 103* Zones achieved the highest recall values with 100%, so each case is classified correctly. However, three cases are predicted incorrectly as the *Lab 103* Zone.

5.1.3 2nd Case

The second case was selected to be Zones from the top floor, where a Lecture Hall was accessible. Thus, the *Lecture Hall 205*, and its neighbours, namely *2nd Floor East Corridor*, *2nd Floor West Corridor* and *2nd Floor North Corridor* have been highlighted.

Evaluation

The observed part of the confusion matrix of 9–NN classifier can be seen in Table 5.4, and the confusion matrix of Naive Bayes can be seen in Table 5.5. The West, East and North corridors of the 2nd floor are displayed with an abbreviated form. The four highlighted Zones have a total of 26 test cases, where the number of test cases for each Zone is proportional to its area.

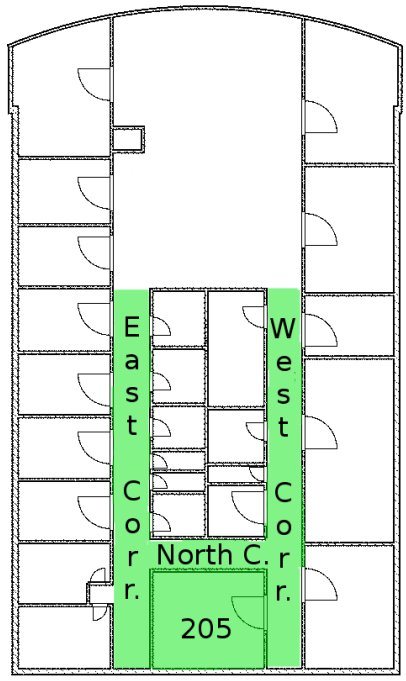


FIGURE 5.2: Selected Zones of 2nd Case

9-NN As can be seen in Table 5.4, the number of misclassified cases is 1, thus the accuracy of these four Zones is 96.15%. Although, that one case had been predicted to be a far Zone among the highlighted Zones. The lowest recall valued Zone is the *2nd Floor East Corridor* with 88.89%, thus only one case is misclassified. The *Lab 102*, *Lab 103* and *Lab 104* Zones achieved the highest recall with 100%, so each case classified correctly. Moreover, no case is predicted incorrectly as one of the highlighted Zones.

Naive Bayes As can be seen in Table 5.5, the number of misclassified cases is 5, thus the accuracy of these four Zones is 80.77%. Although, only one case had been predicted to be a far Zone among the highlighted Zones. The lowest recall valued Zone is the *Lecture Hall 205* with 55%, thus half of the cases are misclassified. The *2nd Floor North Corridor* and the *2nd Floor West Corridor* Zones achieved the highest recall values with 100%, so each case is classified correctly. However, three cases are predicted incorrectly as the *2nd Floor North Corridor* Zone, and 1 as the *Lecture Hall 205*.

TABLE 5.4: Confusion Matrix of 2nd Case Selected Zones with 9NN classifier

Actual	Predicted					Total Result
	2nd Floor East Corr.	2nd Floor North Corr.	2nd Floor West Corr.	Lecture Hall 205	Far	
2nd Floor East Corr.	**8**				1	9
2nd Floor North Corr.		**2**				2
2nd Floor West Coor.			**9**			9
Lecture Hall 205				**6**		6
Total Result	8	2	9	6	1	26

5.1.4 Discussion

Based on the experimental results, the following two remarks can be made. Firstly, the CRISP based accuracy is not a sufficient indicator for compare indoor positioning methods. Secondly, the CRISP approach does not take into account the topology.

The *k*–NN variants and Artificial Neural Network classifiers achieved the highest accuracies during the experiments. The Naive Bayes classifier with one kernel function performed in the top five of tested classifiers. The Decision Tree, the ID3, and the Rule Induction performed the least accuracy. The Rule Induction achieved approximately 3.2% less than the ID3. The accuracy deviation among the first and last classifier is more than 16%.

The four most accurate classifiers achieved more accuracy than 91.6%. These classifiers fairly far misclassified maximum 5 cases. However, the fifth most accurate classifier, the 9–NN achieved 90.97% accuracy with 6 fairly far misclassified cases. The next classifier in

TABLE 5.5: Confusion Matrix of 2nd Case Selected Zones with Naive Bayes classifier

Actual	Predicted					Total Result
	2nd Floor East Corr.	2nd Floor North Corr.	2nd Floor West Corr.	Lecture Hall 205	Far	
2nd Floor East Corr.	**7**			1	1	9
2nd Floor North Corr.		**2**				2
2nd Floor West Coor.			**9**			9
Lecture Hall 205		3		**3**		6
Total Result	7	5	9	4	1	26

the order to fairly far miss this amount of cases is the 5–NN classifier with 89.03%, which is the eleventh in the order by accuracy. The Naive Bayes classifier is the twelfth in the order with 87.1%, and it missed one less case fairly far than the 9–NN. Therefore, the comparison of the 9–NN and the Naive Bayes classifiers in a more detailed view could explain the nature of these classifiers.

The Naive Bayes classifier misclassified 6 more cases than the 9–NN classifier. However, these additional errors are not increased the number of fairly far misclassified cases, thus this 6 case is missed in a close range. Moreover, the number of fairly far misclassified cases is decreased by one. Thus, the size of the error should be measured. Hence, the CRISP based accuracy is not the most suitable indicator for comparing indoor positioning method.

Furthermore, the two highlighted part of the building, detailed in Section 5.1.2 and Section 5.1.3, shows that the two classifiers behaved differently in denser areas.

In the 1st Case, the accuracy of 9–NN classifier is 79.62%, while

the Naive Bayes classifier achieved 73.08%. In the 2nd Case, the 9–NN classifier correctly classified 96.15% of the cases, while the Naive Bayes could only 80.77%. It follows that the 9–NN classifier could overperform its overall accuracy in one of the dense areas.

The 9–NN classifier is not be able to predict the position for *Lab 102* Zone, despite this, the recall of *2nd Floor North Corridor* is 100%. As Figure 4.2 shows, the *2nd Floor North Corridor* contains fewer points, than the *Lab 102*, hence we expect the classifier to recall the *Lab 102* at least the same.

The 9–NN and Naive Bayes classifiers in the two highlighted areas misclassified fairly far the same amount, 2 cases We can conclude from this, that the 9–NN classifier makes 4 faults in less dense areas, while the Naive Bayes only makes 3.

Consequently, the CRISP approach is not sufficient to evaluate classifiers for indoor positioning purposes, because it does not take into account the topology.

Topology-based evaluation of symbolic indoor positioning methods requires two things: a classification error calculation method and a formal description of the indoor environment as domain knowledge to quantify the classification error.

5.2 Requirements for Topology-based Classification Error Calculation

The topology of the building defines the Zones and their sizes and arrangement. The topology-based approach should measure the similarity of the Zones based on their distance and size. We can establish requirements, which seems to be essential for topology-based classification error calculation for symbolic indoor positioning purposes. Firstly, the error values should be proportional to the sizes of the Zones. Secondly, the layout of the Zones should have a high impact on the error values. Lastly, the classification error should not be symmetric due to the size differences of the Zones.

5.3 Proposal of Gravitational force-based Approach

The gravitational force-based approach [10] was designed to consider the topology in the classification error calculation. The main inspiration of this approach is that the rooms can be considered as a 3 dimensional shape, and mass values can be assigned to them.

Then the interaction that exists between the two bodies of mass and causes acceleration in both directions of the centre of mass can be used.

Let Z be the finite set of the rooms. The method assumes the disjunction of the rooms. The approach requires the determination of capacity and distance functions. The capacity $(V : Z \rightarrow \mathbb{R}^+)$ function maps each room to a positive real value. The distance $(d : Z^2 \rightarrow \mathbb{R})$ function determines how far a room is from another or it measures the dissimilarity of the rooms.

The gravitational force $(F_g : Z^2 \rightarrow \mathbb{R})$ [New99] measures the similarity between two rooms. The gravitational force is proportional to the product of their capacity and inversely proportional to the square of their distance. Hence, the gravitational force is non-negative and symmetric derived from the symmetry of distance function and the formulation as seen in (5.1).

$$F_g(Z_i, Z_j) = \frac{V(Z_i)V(Z_j)}{d(Z_i, Z_j)^2} \tag{5.1}$$

While the gravitational force represents the similarity between rooms, their difference is required for error calculation. The δ $(\delta : Z^2 \rightarrow \mathbb{R}^+)$ function is introduced to represent the difference of two rooms. The δ function is the reciprocal of the F_g function; however, the denominator is increased by 1 in order to avoid division by zero. The δ function can be calculated as seen in (5.2) and it ranges in $]0, 1[$.

$$\delta(Z_i, Z_j) = \frac{1}{1 + F_g(Z_i, Z_j)} \tag{5.2}$$

The classification error should also be proportional to the sizes of the rooms and asymmetric due to the size differences. The ϵ $(\epsilon : Z^2 \rightarrow \mathbb{R})$ function is introduced to measure the classification error while fulfils these requirements. The ϵ function weights the classification error with the size of the first room and divides by the joint size of the rooms. The ϵ is non-symmetric in the $[0, 1]$ range, as can be seen in (5.3).

$$\epsilon(Z_i, Z_j) = \frac{V(Z_i) * \delta(Z_i, Z_j)}{V(Z_i) + V(Z_j)} \tag{5.3}$$

In other words, the greater the distance of the rooms, the higher

the classification error, because the F_g decreases and the δ is in inverse relation to F_g. In addition, the classification error of a bigger room misclassified as a smaller room should be higher than in the opposite case due to the direct proportionality between the error and the weighting of the δ function with the size of the actual room. The gravitational force-based approach fulfils the requirement to consider the topology in the classification error calculation [2]. Hence, the gravitational force-based approach can be used to evaluate the classification methods.

The above detailed topology-based classification error calculation approach requires the determination of capacity (V) and distance (d) function. The capacity of a room can be calculated in two or three dimensions. The distance of two rooms can be calculated in a coordinate system or a graph model. The capacity and distance functions are detailed in Section 3.2 and 3.3.

5.4 Experiment in Test Environment

Applicability of the proposed error calculation method is demonstrated with two experiments in this section. The error calculation method has three parameters, which are the capacity function, the reference points and the distance function. Therefore, we set up two experiment cases with different reference points, and with the same distance function and capacity function. With the distance and the area values, the gravitational force, then the δ values can be calculated. Based on the δ values and the areas of the Zones, the error matrix can be constructed which was presented in Section 5.3. The experiment was presented in a two-dimensional space and the simulation was implemented in Python.

5.4.1 Test Environment

The test environment was given in a two-dimensional space. The environment consists of 10 Zones, and we assume that they are rectangular without overlapping, as it can be seen in Figure 5.3. This layout allows us to simulate three major categories of Zones. Firstly, there are long narrow Zones, that can be considered as corridors. Secondly, the huge Zones represent atrium hall, for example Z_8. Finally, the small Zones represents offices and other rooms.

The Zones are defined by their two diagonally opposite points, which can be seen in Table 5.6 and marked with dots in Figure 5.3.

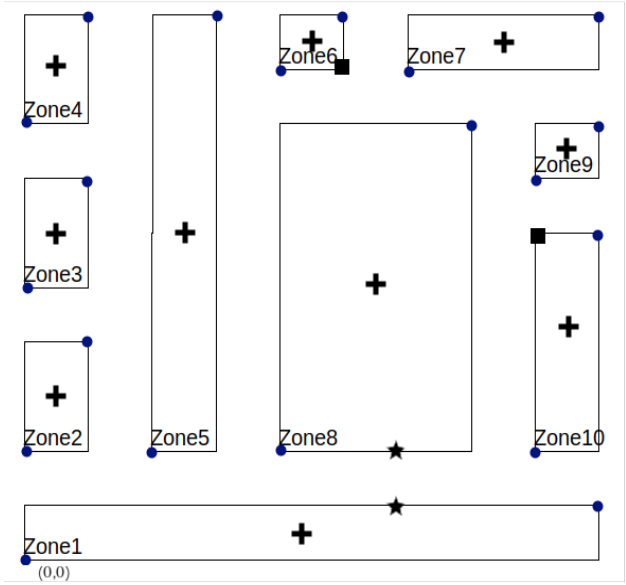


FIGURE 5.3: The Layout of the Test Environment

Each Zone in Figure 5.3 has a corresponding Z_i notation. The base of our coordinate system is selected to be the bottom left corner, the horizontal axis is the x-axis, the vertical axis is the y. The coordinates of the Zones are measured in units.

In both experiments, the capacity function is selected to be the area of the Zones. Due to the characteristic of the Zones, the area can be calculated easily. Table 5.6 also shows the areas of the Zones in the V column, and their relative size in the $V\%$ column.

The distance of the two Zones is calculated based on the Euclidean distance function. The distance of the two Zones is the distance of their reference points. The selection of the reference points distinguishes the two experiments, that are the centroid and the boundary distance cases as detailed in Section 3.3.1.

5.4.2 Centroid Distance Case

In the first case, the centroid of each Zone is selected to be the reference point for the Euclidean distance function. Because each Zone is assumed to be rectangular-shaped, the centroid lies where the two diagonals intersect each other. The centroid of each Zone are

TABLE 5.6: The Zones of the Test Environment

Z_i	p_1		p_2		V	V%
	x	y	x	y		
Z_1	0	0	9	1	9	0.18
Z_2	0	2	1	4	2	0.04
Z_3	0	5	1	7	2	0.04
Z_4	0	8	1	10	2	0.04
Z_5	2	2	3	10	8	0.16
Z_6	4	9	5	10	1	0.02
Z_7	6	9	9	10	3	0.06
Z_8	4	2	7	8	18	0.36
Z_9	8	7	9	8	1	0.02
Z_{10}	8	2	9	6	4	0.08

marked with a cross in Figure 5.3. Based on these reference points, the distance can be calculated for each Zone pair.

With the known distance values, the gravitational force matrix can be determined, which represents the similarity of the Zones. With the gravitational force values, the δ matrix can be constructed to express the dissimilarity of the Zones. Based on the δ matrix and the capacity function, the error matrix is produced, which can be seen in Table 5.7.

TABLE 5.7: Error Matrix with Euclidean Distance of Centroids

	Z_1	Z_2	Z_3	Z_4	Z_5	Z_6	Z_7	Z_8	Z_9	Z_{10}
Z_1	0	0.1699	0.2244	0.2806	0.0398	0.45	0.195	0.0092	0.4253	0.0891
Z_2	0.0378	0	0.2143	0.3	0.0368	0.5282	0.2457	0.013	0.5474	0.1673
Z_3	0.0499	0.2143	0	0.2143	0.0222	0.4844	0.2264	0.0124	0.5352	0.1692
Z_4	0.0624	0.3	0.2143	0	0.0368	0.4456	0.2156	0.0151	0.5352	0.1804
Z_5	0.0354	0.1471	0.0889	0.1471	0	0.2978	0.1475	0.0066	0.3876	0.11
Z_6	0.05	0.2641	0.2422	0.2228	0.0372	0	0.125	0.0107	0.4086	0.1259
Z_7	0.065	0.3685	0.3396	0.3235	0.0553	0.375	0	0.0119	0.3203	0.1362
Z_8	0.0184	0.1171	0.1117	0.1359	0.0149	0.1932	0.0716	0	0.1689	0.0344
Z_9	0.0473	0.2737	0.2676	0.2676	0.0484	0.4086	0.1068	0.0094	0	0.0933
Z_{10}	0.0396	0.3346	0.3384	0.3608	0.055	0.5037	0.1816	0.0076	0.3733	0

The rows of the matrix contain the actual Zones, while the elements of the columns are the predicted Zones. As it was presented in Section 5.3, the elements of the main diagonal are nearly zeros.

The standard deviation of the error matrix except the main diagonal values is 0.1536, and the average of the matrix is 0.1927. The elements of the error matrix are in the $[0.0066, 0.5474]$ range except for the main diagonal zeros. The highest error is determined in the case of actual Z_2 is misclassified as Z_9. The lowest error value, except the zeros, occurs when the Z_5 is misclassified as Z_8.

The Z_3 is between Z_2 and Z_4, and their sizes are the same, thus the two misclassifications have the same error value. Moreover, these three Zones are symmetrical to the Z_3, hence their error values are equals with the reversed cases.

Most of the misclassification to Z_9 and Z_6 results in a relatively high error value. These two Zones are near Z_8, which is the largest Zone, and the misclassification of Z_8 to Z_6 is 0.1932 and to Z_9 is 0.1689. The average of error values of both Z_6 and Z_9 is 0.41.

The Z_8 is the largest Zone, and it is almost neighbouring with all the other Zones, as can be seen in Figure 5.3. The maximum error value of misclassification of any other Zone to Z_8 is 0.0151 from Z_4, while the maximum of the neighbouring Zones is 0.0119 from Z_7. It has the lowest error values in the case of some Zone misclassified as Z_8.

The Z_5 is in the middle of the other Zones representing a corridor, and it has relatively low error values if it is wrongly predicted. The worst error value of Z_5 column is 0.0553 in the case of Z_7 actual Zone. The lowest error is the misclassification from Z_8 with 0.0149. The Z_1 is also representing a corridor, in contrast with Z_5, the Z_1 is placed on the edge of the test environment. The maximum of the error values is 0.065 in the case of actual Zone Z_7, and the lowest error is 0.0184 in the case of Z_8. Although the relative area of Z_5 is 0.16%, which is smaller than Z_1 with 0.18%, Z_1 has slightly worse error values than Z_5. These Zones are long and narrow, hence the centroid of Z_5 is almost in the middle of the y-axis, while the centroid of Z_1 is at the half of the x-axis.

5.4.3 Boundary Distance Case

In the second case, the nearest boundary points are chosen to be the reference points for the distance function. Due to the disjunction of the Zones and the presence of wall thickness, the distance of the Zones must be greater, than zero. In Figure 5.3, the nearest boundary points of Z_6 and Z_{10} pair are marked with squares, and for the Z_8,Z_1 Zone pair, they are denoted with stars.

Based on the distance of these pairs of reference points and the capacities of the Zones, the elements of the gravitational force matrix can be calculated. While the gravitational force matrix represents the similarity of the Zones, the δ matrix is constructed to be inversely proportional to the gravitational force values. Using the δ matrix and the capacities of the Zones, the ϵ matrix is created to represent the error values for each misclassification cases. The error matrix can be seen in Table 5.8.

TABLE 5.8: Error Matrix with Euclidean Distance of Nearest Boundary

	Z_1	Z_2	Z_3	Z_4	Z_5	Z_6	Z_7	Z_8	Z_9	Z_{10}
Z_1	0	0.0431	0.1488	0.2291	0.0073	0.4235	0.1714	0.002	0.36	0.0187
Z_2	0.0096	0	0.1	0.25	0.0118	0.4964	0.2164	0.0077	0.528	0.1556
Z_3	0.0331	0.1	0	0.1	0.0118	0.4288	0.1892	0.0077	0.5185	0.1556
Z_4	0.0509	0.25	0.1	0	0.0118	0.4	0.1818	0.0077	0.5185	0.1588
Z_5	0.0064	0.0471	0.0471	0.0471	0	0.0988	0.0808	0.0021	0.3419	0.0901
Z_6	0.0471	0.2482	0.2144	0.2	0.0123	0	0.0625	0.0028	0.3799	0.1029
Z_7	0.0571	0.3246	0.2838	0.2727	0.0303	0.1875	0	0.0026	0.1875	0.0857
Z_8	0.0041	0.0692	0.0692	0.0692	0.0048	0.0499	0.0156	0	0.0499	0.0112
Z_9	0.04	0.264	0.2593	0.2593	0.0427	0.3799	0.0625	0.0028	0	0.04
Z_{10}	0.0083	0.3111	0.3111	0.3176	0.045	0.4118	0.1143	0.0025	0.16	0

The rows of the matrix contain the actual Zones, while the columns represent the predicted Zones. The standard deviation of the error matrix is 0.1466 without the zero elements of the main diagonal, and the average error is 0.1471. The elements of the error matrix are in the $[0.002, 0.528]$ range besides the 0. The highest error is determined in the case of actual Z_2 is misclassified as Z_9. The lowest error value, except the diagonal zero values, is 0.002 when Z_1 is wrongly predicted as Z_8.

The Z_3 is equally near Z_2 and Z_4, and their sizes are the same, thus the two misclassifications have the same error value. Moreover, these three Zones are symmetrical to the Z_3, hence their error values are equals in both directions.

Most of the misclassification to Z_9 and Z_6 results in a relatively high error value. These two Zones are near Z_8, which is the largest Zone, and the misclassification of Z_8 to Z_6 and to Z_9 is 0.0499. The average error value of Z_6 is 0.3196 and 0.3382 for Z_9.

The Z_8 is the largest Zone with 0.36% relative area, and it is almost neighbouring with all the other Zones, as can be seen in Figure 5.3. The maximum error value of misclassification of any other Zone to Z_8 is 0.0077 from Z_2, Z_3 and Z_4, while the maximum

of the neighbouring Zones is 0.0028 from Z_6 and Z_9. It has the lowest error values in the case of other Zones misclassified as Z_8.

Both Z_5 and Z_1 are long and narrow Zones, representing corridors in the test environment. The Z_5 is in the middle of the other Zones, conversely the Z_1 is placed on the edge of the test environment, as seen in Figure 5.3. The worst error value of Z_5 column is 0.045 in the case of Z_{10} actual Zone, and the highest error of Z_1 is 0.0571 in the case of actual Z_7. The lowest error is the misclassification from Z_8 to Z_5 with 0.0048, and to Z_1 this value is 0.0041. Although the relative area of Z_5 is 0.16%, which is smaller than Z_1 with 0.18%, Z_1 has a slightly higher maximum error value than Z_5, however, the minimal error is lower.

5.4.4 Conclusion

Based on the experiment results, the presented method fulfills the requirements. The classification error should be proportional to the sizes of the Zones, take into count the layout of the Zones, and it should be asymmetric.

In the test environment, the Z_2, Z_3 and Z_4 are a relatively dense area of the environment, and their size is the same. The Z_3 is between the two other Zones, and their distance is equal. In the error matrix of both experiments, the same values are assigned to these cases, and they are symmetric. In a real life scenario, the misclassification from Z_3 to either Z_2 and Z_4 is the same amount of error. In both experiments, the highest error value has occurred in the case of Z_2 misclassified to Z_9. The Z_2 and Z_9 Zones are relatively small with 0.04% and 0.02%. The two Zones are on the opposite edge of the layout, thus they considered very far from each other in the experiments. In real life, this misclassification would be also considered as the worst case. Hence, the proposed method fulfills the requirement to consider the layout of the Zones when calculating the error values.

The largest Zone in the environment is Z_8, which is in a central position of the layout. The error values are relatively low for the neighbouring Zones misclassified as Z_8, which coincides with the expectation. The errors of misclassification Z_8 to other Zones are not significantly higher due to its placement. The Z_1 and Z_5 are relatively large narrow Zones, and they are placed close to each other. The misclassification to neighbouring, Z_8 Zone should result in low error values. In both experiments, the lowest error value occurs

when one of the corridor-like Zones are misclassified to Z_8. Thus, the presented method is proportional to the sizes of the Zones, and it is asymmetric.

Based on our experimental results, the presented method results proportional error values to the Zone sizes, takes the layout of the Zones into the count and the error values of two Zones are asymmetric. Therefore the presented classification error calculation method considers the topology.

5.5 Experiment in a Real-life Environment

To further test the applicability of the proposed gravitational force-based method, the experiment in a real-life environment is performed. The topology of the building defines the rooms, their arrangements and their connections in the buildings. Building topology can be modeled with a wide range of tools. Computer-aided Design (CAD) tools and Building Information Model (BIM) are used by architects, construction workers and interior designers. The CAD model is primarily two-dimensional, which contains lines, arcs and circles. However, BIM is in two to six-dimensional space, and it consists of walls, windows, floors and roofs. IndoorGML (Indoor Geographic Markup Language) [Lee+14; Ogc] is a standard defined by the Open Geospatial Consortium (OGC), which is an open format to describe the topology. It represents a model of the building by the shapes defined in the XML (Extensible Markup Language) format.

5.5.1 IndoorGML

IndoorGML represents a model of the building by the shapes defined in the XML Schemas, which provides data in XML. Indoor spaces are non-overlapping closed objects, and they are bounded by physical or fictional boundaries. OGC also provides Java classes for converting the XML files into objects for further, higher-level processing.

The topology of the environment is stored in a constructed IndoorGML document. The `id` property of `cellSpace` tag the unique identifier of the cellSpace. The name of the room is added to the `metaDataProperty` tag. This document describes the rooms by two forms in three-dimensional space, one is with vertices and the other

is with a bounding box for speed-up purposes. Hence, the IndoorGML provides the possibility of the usage of coordinate-based distance calculation. In addition, the document contains transitions between rooms, which can be mapped to a graph.

The Institute of Information Science (IIS) Building was modeled with IndoorGML [1] standard. The `id` property of `cellSpace` tag is chosen to be the same as in the Miskolc IIS data set with an uuid prefix. The XML files can be created in both automatic and manual ways. The model could be generated from the construction plan, but this plan is not available in the case of this particular building.

The manual creation requires a grid on the building, whose base point is in the bottom left corner of Figure 4.2(d). The horizontal axis is the y, and the vertical axis is the x. Currently, the building has a 1 $m \times 1$ m measured grid in the accessed areas. The coordinates are determined based on the available grid. Thus, the data set and the IndoorGML use the same coordinate system.

5.5.2 Results

For the implementation of the topology-based classification error calculation a Java application had been developed. It converts the data from the IndoorGML XML document to Java classes, both provided by IndoorGML. A zone is represented as CellSpace object, which means that each zone contains the name and the ID of the zone, the bounding and two diagonal cornerstone coordinates.

In this paper, the three-dimensional lower- and upper corner coordinates are used in distance and capacity calculation. For distance calculation, the Euclidean distance had been chosen, and the distance is specified by the length of the straight line between the middle points of the two zones. To calculate the capacity of a zone, the benefit of cuboid property had been applied to calculate the volume. Based on the distance and the capacity function, the classification error can be calculated for each zone pair. Table 5.9 shows some examples of the classification error.

The overall average classification error is 0.0088 with 0.0271 standard deviation. The highest error calculated is 0.3641 in the case of *Ground Floor Elevator* actual zone is misclassified as *Overhead of Office 206 and 207*. The volume of *Ground Floor Elevator* is 17.5 m^3, while the *Overhead of Office 206 and 207* is 2.8 m^3.

The two farthest zones are *Lab15* on the front of the ground floor and *Lecture Hall 205* on the back of the second floor. The volumes

TABLE 5.9: Examples of the Topology-Based Classification Error

Actual			Predicted			Error
Name	Capacity (m^3)	Centroid	Name	Capacity (m^3)	Centroid	
Ground Floor Elevator	17.5	(40.8, 8.3, 4.5)	Overhead of Office 206 and 207	2.8	(5.5, 6.5, 4.5)	0.3641
Lab15	169.4	(45.5, 2.8, 1.4)	Lecture Hall 205	156.8	(3.5, 14, 7.6)	0.0009
Lab100	241.5	(44.8, 24.8, 4.5)	Lab101	172.9	(33.8, 25.3, 4.5)	0.0002
2nd Floor West Corridor	231	(16.5, 19.8, 7.6)	1st Floor West Corridor	231	(16.5, 19.8, 4.5)	0.0001

of these zones are 169.4 and 156.8 m^3. The classification error calculated in this case is 0.0009.

The *Lab100* and *Lab101* are neighbouring zones with 241.5 and 172.9 m^3 volume. The classification error in this case is 0.0002. The *2nd Floor West Corridor* and the *1st Floor West Corridor* zones are congruent, they only differ in the z coordinate. The misclassification in both directions results the 0.0001 value.

5.5.3 Conclusion

As it can be seen in Figure 5.1, the *Ground Floor Elevator* and the *Overhead of Office 206 and 207* both relatively small zones, and they are very far from each other. Hence, the classification error in this case is high. Otherwise, the *Lab15* and the *Lecture Hall 205* are also very far from each other, but they are both relatively large, thus it has significantly lower error value than in the case of *Ground Floor Elevator* and *Overhead of Office 206 and 207*. Therefore the method considers the size and the layout of the zones, thus the misclassification of smallest, farthest zones results in the highest error values.

The misclassification of *Lab100* to *Lab101* zones has a relatively small error value. *Lab100* is larger than *Lab101*, but both are considered as relatively large zones and they are neighbouring.

The *2nd Floor West Corridor* and *1st Floor West Corridor* zones are congruent in size and location beside the z coordinate. As it would be expected, the classification errors of these two zones are symmetric, and small.

The results of the classification error calculation show that the gravitational force-based approach considers the topology in classification error calculation. However, the calculated error values have a very low average, and the standard deviation should be higher, and the highest error value is lower than the half of the possible range. Therefore, the gravitational force-based approach can be an alternative to the CRISP approach in the evaluation of symbolic indoor positioning methods.

5.6 Comparison of the gravitational force-based and the CRISP approach

Experiments were performed in order to compare the CRISP and the proposed topology-based classification evaluation method. Comparison was performed over a dataset and map. More than 20 well-known classifiers were evaluated for location estimation. The classifiers were ranked based on CRISP and three variants of the proposed topology-based method.

5.6.1 Test Environment

The CRISP and the topology-based classification evaluation methods were performed over the Miskolc IIS Hybrid IPS data set, detailed in Section 4.1.3. The data set was recorded in the Institute of Information Science Building at the University of Miskolc, whose topology is given in IndoorGML format as stated in Section 5.5.1.

5.6.2 Comparison Process

The comparison process has seven major steps.

1. The data set and the topology are loaded.
2. The data set is split into training and validation sets using a stratified sampling method [Gro+11].
3. For each classifier and evaluation method

   (a) the training set is used to train the current classifier.
   (b) the validation set is used to measure the accuracy of the classifier.

4. The classifier variants are ranked based on the sum of the error values for each evaluation method.

5. The error values are divided by the best value to represent the distribution and the relative performance of the evaluation method.

In previous works [9; 8], Decision Tree, *k*–NN, Rule Induction, Naive Bayes, and Artificial Neural Network (ANN) classifiers were analysed and evaluated with CRISP approach. The selection criterion of the tested classifiers was the representation of both model-based and instance-based classifiers, including the most popular methods. The training of these classifiers was executed in RapidMiner [HK13] and the evaluation of the classification results was implemented in Java.

5.6.3 Evaluation methods

In this study, the CRISP approach and the proposed gravitational force-based approach are compared in order to determine the usefulness of topology-based evaluation.

CRISP approach

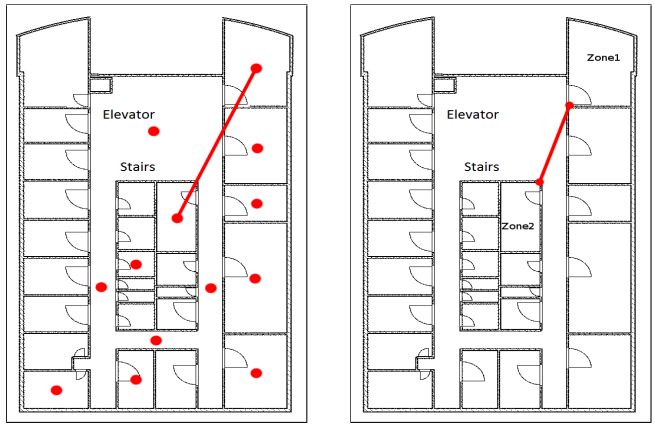
Traditionally, the accuracy of the classifier is determined by using the CRISP approach. The CRISP approach can be applied to any classification task. Because CRISP is a general and widely accepted evaluation method, it was selected for comparison.

Gravitational force-based approach

Gravitational force-based approach has two parameters, which are the capacity and distance functions. The capacity function calculates the area of the room in the floor plan. Based on the mentioned capabilities of the IndoorGML in Section 5.5.1, three different distance functions were used during the experiments.

Euclidean Distance of Centroids This method projects the vertices to two dimensions, assuming the third dimension to be the same in each vertex. This mechanism can simplify the calculation of the centroid of the room as seen in Figure 5.4(a). Then the distance between the centroids is calculated with the Euclidean distance function. Therefore, the centroid point can be considered a global feature of the room.

Euclidean Distance of Nearest Points This method uses the nearest points of two selected rooms, which is not limited to the closest vertices. The nearest point of a room depends on the other room; thus, this cannot be treated as a room feature, as can be seen in Figure 5.4(b). The advantage of this method is that the distance of two neighbouring rooms is the thickness of the wall and it reduces their classification error.



(a) Centroids (b) Nearest Points

FIGURE 5.4: Examples of the two types of reference points

Length of Shortest Path in Graph This method uses the transitions of the IndoorGML document to generate a graph where the transitions are the edges and their connected states are the rooms. In this method, the weight for each edge is assumed to be equal and the type of the edge can be both directed and undirected, hence the Dijkstra's shortest path algorithm is selected among the algorithms presented in Table 3.2. The algorithm calculates the route of the

actual room to the predicted room as the lowest number of passed rooms. Hence, the shortest path would approximate the walking distance between the rooms.

5.6.4 Experimental Results

The result of the CRISP and topology-based classification error evaluation methods are investigated by two viewpoints. The first is the ranking of the classifiers and the second is the relative performance of the evaluation method.

Ranking

The tested classifiers were ranked based on the calculated classification error. The most accurate classifier is the first, while the least accurate is the last. In the case of exact classification error values, the classifiers are grouped. ANN 1 and ANN 2 are two settings of the Artificial Neural Network as presented in Table 5.10. The *k* Nearest Neighbour classifier and its weighted version are denoted by *k*NN and *k*NNW. The Naive Bayes 3 Kernel denotes the extended Naive Bayes classifier with 3 kernel functions. Table 5.11 shows the ranks of each tested classifier in the different evaluations.

TABLE 5.10: Settings of Artificial Neural Network

Notation	Learning rate	Momentum	Number of training cycles	Number of nodes in the hidden layer
ANN 1	0.9	0.5	380	24
ANN 2	0.464	0.454	210	22

The experimental results show that some classifiers resulted in the same classification error value. The classifiers were grouped based on their performance. The evaluation method could not distinguish the elements of the groups. The highest number of groups is in the case of nearest point distance, with 20 groups, after the centroid distance stands with 18 groups, then the CRISP approach with 13 groups, and the lowest number of groups is 12 in the graph distance. The average group size is 1.77 in the case of CRISP, while the graph distance case could exceed the average group size to 1.9. However, the centroid distance and the nearest point distance resulted in an average group size of 1.277 and 1.15, respectively. The

TABLE 5.11: Rankings of the Tested Classifiers

#	CRISP	Gravitational Centroid	Gravitational Graph	Gravitational Nearest
1	ANN 1	ANN 1	ANN 1	ANN 1
2	ANN 2 9NNW 3NN 3NNW	9NNW Naive Bayes 1 Kernel	ANN 2 Naive Bayes 1 Kernel	ANN 2
3				Naive Bayes 1 Kernel
4		3NN 3NNW	Naive Bayes 2 Kernel	Naive Bayes 2 Kernel
5			9NNW 13NN Naive Bayes 3 Kernel Naive Bayes	9NNW
6	Naive Bayes 1 Kernel	ANN 2 Naive Bayes 2 Kernel		Naive Bayes 3 Kernel
7	9NN			11NNW 11NN
8	13NN 11NN 11NNW 13NNW 5NNW	13NN		
9		11NN 11NNW	ID3 Tree 11NNW 11NN	13NNW
10				3NNW 3NN
11		Naive Bayes 3 Kernel		
12		13NNW	13NNW	Naive Bayes
13	Naive Bayes 2 Kernel 1NNW 1NN Naive Bayes 3 Kernel	5NNW	7NNW 3NNW 3NN	13NN
14		Naive Bayes		1NNW 1NN
15		1NNW 1NN		
16			5NNW 7NN	5NNW
17	5NN	9NN		7NNW
18	7NNW	7NNW	5NN 9NN	7NN
19	7NN	7NN		Decision Tree
20	Naive Bayes	5NN	Decision Tree	5NN
21	Decision Tree	ID3 Tree	1NNW 1NN	ID3 Tree
22	ID3 Tree	Decision Tree		9NN
23	Rule Induction	Rule Induction	Rule Induction	Rule Induction

largest group in the CRISP evaluation consists of 5 classifiers, with

4 classifiers in the case of graph distance, while both centroid distance and nearest point distance produced the largest group with only 2 classifiers.

Rank-based ordering of the classifiers was also analysed. Two kinds of classifiers can be distinguished. The classifiers with fixed positions form the first type, the second group consists of classifiers with diverse positions.

In every evaluation method, two classifiers are in fixed places in the order. The Artificial Neural Network with setting 1 is in the first place in every tested case. However, the last place is always obtained by the Rule Induction in the experiments. In addition, the places of two other classifiers can be narrowed to five positions. The 9–NN classifier with weighted votes is in the top 5, while the Decision Tree classifier is in the bottom 5 in every evaluation case.

Among the classifiers in the highly diverse group, there are ones with significant diversity. For example, the ID3 Tree classifier is in the bottom 3 in the CRISP, the centroid and the nearest point distance, while the graph distance ranks the ID3 Tree in the 9th place. Another example is the 9–NN classifier, which is the 7th in the CRISP evaluation. However, when the topology is considered in the classification error calculation, it falls to the last third of the ranking.

In the comparison, classifiers with slightly diverse positions are the most frequent. As an illustration, the ANN with setting 2 is in the second place in most of the evaluation methods except for one evaluation. In the centroid distance evaluation, the ANN 2 is ranked as the 6th. To give another example, the Naive Bayes with 1 kernel function classifier is among the top 3 classifiers in the three new evaluation methods. However, in the CRISP approach, it takes the 6th place. As a last instance, the 1–NN and the weighted version are in the middle section of the rankings except for the graph distance evaluation. In that evaluation, among the Rule Induction, the nearest neighbour method is at the bottom of the list.

Sensitivity

Besides the ranking of the classifiers, their relative performance to the most accurate one also can be analysed. For each evaluation method, the classification error values are divided by the best value, which is the lowest error value of the given evaluation. Hence,

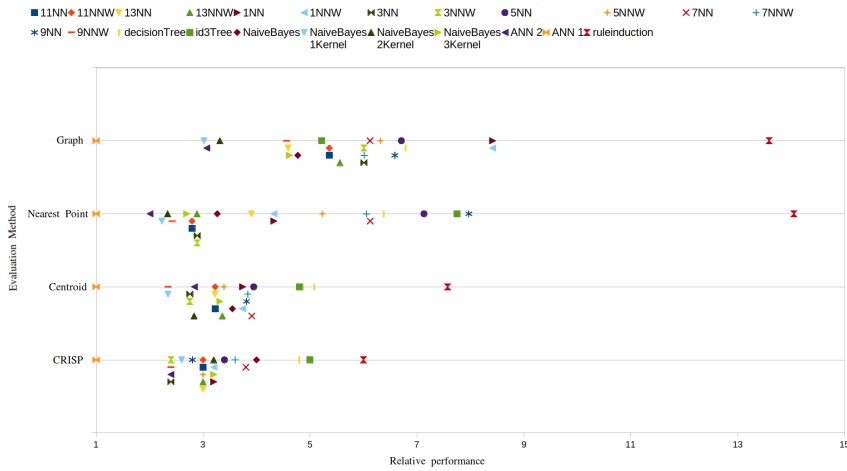


FIGURE 5.5: Relative performance of tested classifiers in evaluation cases

the minimum of the relative performance values is 1 for each evaluation method. Therefore the relative performance of the classifiers can be considered as the sensitivity of the evaluation method. This relative performance of the classifiers can be seen in Figure 5.5, where the x-axis shows the multiplier, the number of times the classifier has more error, than the best classifier in the evaluation method.

The proportions of the CRISP approach range from 1 to 6 with a 1.03 standard deviation and the average proportion is 3.23. In the case of the Euclidean distance of the centroids, the upper bound of the range is 7.57, while the standard deviation is 1.22 and the average is 3.5. The method with Euclidean distance of the nearest point resulted in the upper bound being 14.05, the standard deviation 2.87 and the average 4.49. The upper bound of the graph distance method is 13.59 with 2.41 standard deviation and 5.71 as an average value. Grouping of classifiers based on their performance also shows the sensitivity of the evaluation method.

5.6.5 Discussion

Two general observations can be drawn. The first observation can be derived from the detailed examination of the proposed four evaluation methods. The second observation is related to the selection of classifiers for symbolic indoor positioning purposes.

Firstly, the sensitivity analysis of the evaluation methods shows that the presented approach gives a better, more detailed comparison than CRISP. However, the distance function has a high impact of sensitivity. As can be seen in Figure 5.5, the two methods that is calculating in the coordinate system have a remarkable sensitivity difference. However, the nearest point distance and the graph distance methods are more alike, while the CRISP and the centroid distance methods also show more similarity.

The centroid distance function variant achieved the lowest proportion range among the methods that consider the topology. Although it results in more groups of classifiers than CRISP, the standard deviation and the range show minor improvements. Thus, the usage of the centroid distance could not give additional information about the tested classifiers. As a consequence, the centroid of a room seems to be an unworthy choice of reference point in topology-based evaluation.

Nearest point distance function method developed the largest range between the best and the worst method. The highest standard deviation value also occurs in the nearest point distance case. In addition, the number of created groups is significantly higher than with the other methods and the average group size is the closest to 1. It follows that the gravitational force-based approach with nearest point distance seems to be a good option for topology-based evaluation.

The graph distance function method also resulted in a similar range as the nearest point distance method. Besides, the standard deviation of the graph distance method slightly below the highest deviation. However, the number of groups does not exceed the same feature of CRISP. Despite this, the gravitational force-based approach with graph distance can be considered as a solution for topology-based evaluation.

As a general result, even the poorly performing variant of gravitational force-based approach was able to slightly outperform the results of the CRISP approach.

Secondly, the presented experimental results indicate whether Artificial Neural Network and Rule Induction are suitable or insufficient for symbolic indoor positioning problems. The Artificial Neural Network with setting 1 is in the first place of rankings of each evaluation method. The Rule Induction classifier causes the highest error values in the experiment. Moreover, the proportion of the Rule Induction to the ANN1 is significantly larger in the

methods with nearest point or graph distance. The findings of this study suggest that the Artificial Neural Network classifier is the best choice for symbolic indoor positioning purposes. In addition, our findings suggest that Rule Induction is not very suitable for indoor positioning purposes.

5.7 Conclusion

The proposed gravitational force-based approach fulfills the requirements to consider the topology in classification error calculation. The usage of the proposed method had benefits over the classic CRISP approach. As an overall conclusion, the proposed gravitational force-based approach seems to be a good candidate to be used in topology-based evaluation calculation for symbolic indoor positioning purposes.

Thesis 2.

I proposed a topology-based evaluation method for classification-based indoor positioning algorithms which allows a more detailed evaluation.
Related Publications:[9], [11], [8], [10] ,[1], [2].
Citations: [HHAR19],[Wan+20],[YZZ20],[Fen+20],[Yan+21]

Chapter 6

Hierarchical Grouping enhanced Classification

Thesis 3. *I designed a classification-based symbolic indoor positioning method enhanced by hierarchical clustering, which considers the topology of the building based on the confidence of the classification.*

Indoor positioning is challenging due to the unique properties of the indoor environment. Developers have to make trade-offs between accuracy and cost when they choose a technology. A sufficiently precise, easily accessible and sustainable industrial standard has not been created yet. Symbolic positions can be considered as categories, thus symbolic positioning can be converted into a classification problem. Well-known classifiers accept classes as a prediction based on the confidence values. There are some cases when the confidence for each class is relatively small. Hence, the accuracy of these classifiers can vary in a moderate range. To boost the performance of these classifiers for symbolic indoor positioning purposes, a hierarchical grouping of class categories can be introduced.

6.1 Hierarchical Clustering of rooms

The creation of the tree structure of the hierarchical grouping can be manual or automatic. The manual creation requires firsthand domain knowledge of the given environment. However, hierarchical clustering algorithms can be applied using topological description to generate the tree.

6.1.1 Clustering

In the experiment, different hierarchical clustering methods with different distance functions are examined. The clustering process is implemented in Python. The IndoorGML document is loaded, and converted into a DataFrame using *pandas* package. The grouping of the objects is performed using *SciPy* package, while the tree model is presented using the *tanglegram* package [Tan].

Room representation

The room representation is performed using IndoorGML standard. The IndoorGML document is transformed to be used for clustering purposes.

Feature Selection The features are examined based on the practicability for clustering purposes.

Some metadata of the rooms is retained in this step for the evaluation of constructed hierarchies. The physical and virtual boundaries defined using coordinates are selected for presentation. Hence, the lower and upper corners of the bounding box and all the vertices of the rooms are preserved. Contrary to the layout of the rooms, their transitions are not convenient to create a hierarchy among rooms. However, they can be appropriate for way-finding purposes. In this study, the permeability of the room borders is not incorporated.

Feature Extraction After the elimination of unnecessary features, a new feature is introduced. The new feature is the capacity of each room. In this experiment, the volumes of each room are added. The volume is calculated from the lower and upper corners of the bounding box.

Furthermore, the identifier and the name are merged for indexing purposes. It is required to eliminate these features from the actual grouping process.

Similarity

Various distance functions can be defined in a coordinate system. In this study, two distance functions are applied, euclidean and gravitational distance. Euclidean distance is detailed in Section 3.3,

in addition, some linkage methods only accept this distance function as its metric. The δ can be considered as a virtual distance among the rooms presented in Section 5.3. Euclidean distance is used as a distance metric and volume is used as capacity function for the gravitational force-based approach.

Grouping

The grouping is performed in the bottom-up or agglomerative way. The linkage method and the distance metric is the parameter of the grouping. The result of the grouping is the $(n-1) \times 4$ linkage matrix. The n is the number of original objects. The first and second columns of the linkage matrix are the ids of the clusters, which will be merged to create a new cluster with an incremented id. The third column is the distance between the two merged clusters. The fourth column is the number of original objects assigned to the newly formed cluster. The linkage matrix is visualised using a dendrogram.

Comparison of dendrograms

The created cluster models are compared based on the generated dendrograms. A pair of cluster models are represented in a tanglegram. Tanglegram is used to compare tree diagrams. It measures the quality of the two dendrogram alignment as entanglement. For each object, a vector can be established between the two dendrograms. The entanglement is the L norm distance between these vectors. The number of optimization iterations can be specified in order to minimize the entanglement. The entanglement value is the base of the evaluation.

6.1.2 Evaluation of cluster hierarchies

During the experiments, the euclidean distance function and the gravitational force-based distance is examined. The average, centroid, complete, median, single, ward and weighted linkage methods had been tested. In the case of Euclidean distance function, all the linkage methods could operate. However, centroid, median and ward linkage methods can not be applied with gravitational force-based distance or any distance function other than the Euclidean distance.

The dendrogram of the generated tree using complete linkage method with Euclidean distance can be seen in Figure 6.1. The leaf nodes of the tree are the room identifiers, namely the id and the name. Two main groups of rooms can be distinguished. The height of the dendrogram represents the distance between the clusters.

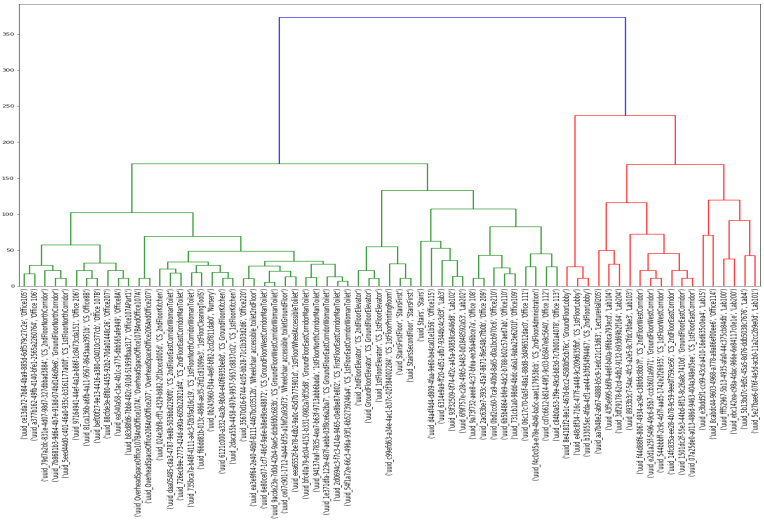


FIGURE 6.1: Dendrogram of Hierarchical Clustering using Complete Linkage Method

The dendrogram shown in Figure 6.1 is an example in the paper. The trees generated by the same distance function are compared to each other, and then optimized to minimize the entanglement value. These entanglement values are not necessary symmetric due to the implemented brute force approach.

The entanglement values of the generated tree with Euclidean distance function can be seen in Table 6.1. The values are optimized in 10000 iterations. The lowest, 1.0704 entanglement value occurs in the case of centroid and weighted linkage methods. The highest entanglement value in this scenario is single and complete linkage method with value 7.2676. Most of the linkage methods resulted in entanglement value between 3 and 4. However, single linkage has almost two times higher average entanglement value than the other methods.

Table 6.2 shows the entanglement values using Gravitational force-based distance. The iteration number is 10000 in this case. The centroid, median and ward linkage methods are not shown in

TABLE 6.1: Entanglement of Methods using Euclidean Distance Optimized with 10000 Iterations

linkage methods	average	centroid	complete	median	single	ward	weighted
average		2.507	2.7606	1.9155	7.493	3.0704	2.1127
centroid	1.831		3.9437	2.2817	5.662	3.4085	2.0282
complete	3.3803	4.4507		4.0282	7.2676	3.2958	3.7746
median	2.8732	1.8592	3.8873		6.9014	3.3803	2.3099
single	6.1127	6.7887	6.6761	5.2958		5.7183	4.5352
ward	2.5915	3.493	2.3944	2.9859	6.4225		3.2676
weighted	2.338	1.0704	3.2676	2.4225	6.9014	4.1408	
average value	3.1878	3.3615	3.8216	3.1549	6.7747	3.8357	3.0047

TABLE 6.2: Entanglement of Methods using Gravitational force-based Distance Optimized with 10000 Iterations

linkage methods	average	complete	single	weighted
average		5.4366	8.9296	4.5634
complete	5.7183		11.9155	5.1549
single	8.1127	12.2254		12.7324
weighted	5.4366	5.2676	11.0704	
average value	6.4225	7.6432	10.6385	7.4836

the Table due to the incompatibility of the distance function. The lowest entanglement value is 4.5634 in the case of weighted and average linkage methods. The highest entanglement occurs in the case of weighted and single linkage methods with value 12.7324. The difference between the average entanglement values of single linkage method to other methods is significant.

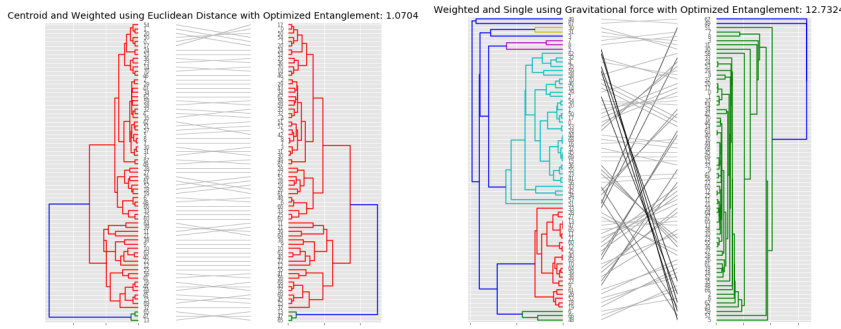
Table 6.3 shows the entanglement values using Gravitational force-based distance. The iteration number is 100000 in this case. The centroid, median and ward linkage methods are not shown in the Table due to the incompatibility of the distance function. The lowest entanglement value is 2.9577 in the case of weighted and average linkage methods. The highest entanglement occurs in the case of single and complete linkage methods with value 11.7183. The difference between the average entanglement values of single linkage method to other methods is significant.

TABLE 6.3: Entanglement of Methods using Gravitational force-based Distance Optimized with 100000 Iterations

linkage methods	average	complete	single	weighted
average		4.3099	9.1268	2.9577
complete	4.5915		11.7183	4.5915
single	8.6197	11.5775		10.1127
weighted	3.2676	4.3099	10.3944	
average value	5.4929	6.7324	10.4132	5.8873

In Table 6.2 and Table 6.3, the increment of the iteration number could decrease the average entanglement value. While in the case of average, complete and weighted methods the average values lessen by at least 0.91, the single method could decrease its averages by 0.22.

The lowest entanglement value in the experiment resulted by centroid and weighted linkage methods using Euclidean distance, and its tanglegram can be seen in Figure 6.2(a). The highest entanglement value in the experiment was achieved by the weighted and single linkage methods with gravitational force-based distance. The tanglegram of the two dendrograms can be seen in Figure 6.2(b).



(a) Centroid and Weighted linkage methods using Euclidean distance with 10000 iterations

(b) Weighted and Single linkage methods using Gravitational force-based distance with 10000 iterations

FIGURE 6.2: Tanglegram of dendrograms: best and worst cases

The two distance functions can be examined by using the same linkage method. The weighted linkage had been selected for the comparison due to the compatibility and lower entanglement values. The tanglegram of the distance functions can be seen in Figure 6.3 with 9.3803 entanglement value.

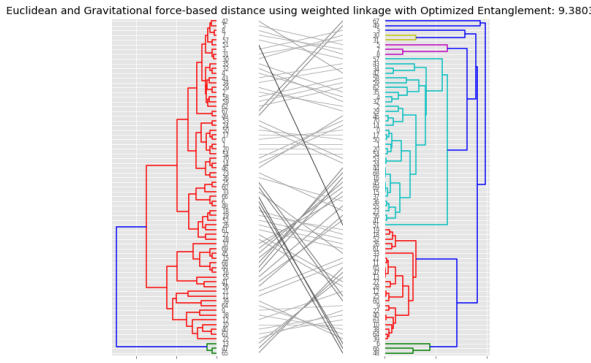


FIGURE 6.3: Euclidean and Gravitational force-based distance using weighted linkage method with 10000 iterations

Discussion

Based on the experimental results, two observations can be drawn, one about the linkage methods and one about the distance functions.

The first observation is about the sameness of the linkage methods. Experimental results show, that most of the linkage methods achieved similar average entanglement values. However, the single linkage method shows a significant difference in both distance functions. Hence, the single linkage is an outlier method. In Figure 6.2(b), the single linkage method does not create an explicable hierarchical grouping among the rooms of a building.

The other one is about the usability of the distance functions. Figure 6.3 shows that the two dendrograms are highly diverse. Euclidean distance can result in similar hierarchical clustering using most of the linkage methods. However, these linkage methods seem understandable to the given purpose, the gravitational force-based distance could reflect the topology, the arrangement of the rooms by the distances in the dendrogram.

6.1.3 Conclusion

The possibilities of hierarchical clustering for symbolic indoor positioning enhancement purposes are examined. The physical space is described using IndoorGML. Euclidean distance function and gravitational force-based distance were used as distance function for the clustering. Average, centroid, complete, median, single, ward and weighted linkage methods had been tested in this experiment. Experimental results show that the single linkage method does not behave like other linkage methods tested. The gravitational force-based distance could reflect the topology more detailed in the dendrogram.

6.2 Enhanced classification

Using hierarchical clustering information of symbolic positions, the accuracy of symbolic indoor positioning algorithms can be improved in case of a low confidence level.

The concept of enhanced classification requires the following parameters:

- Classifier
- Threshold
- Dendrogram

The classifier is the method for supervised learning based on the training set and dataset detailed in Section 4.2.2. The threshold is a real value between 0 and 1, which determines whether the prediction is accepted or the proposed concept is used. If the confidence value of the predicted class is equal to or higher than the threshold, the classifier method return with the class. The dendrogram can be predefined by a linkage matrix or it is produced by linkage and distance methods parameters from the topology information. The linkage parameter is detailed in Section 3.4.1 and the distance function is detailed in Section 3.3.

The tree structure generated by the hierarchical clustering can be seen in Figure 6.4. The leaf nodes are the rooms, while the root node is the whole described environment.

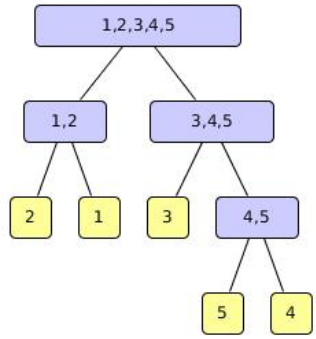


FIGURE 6.4: Concept base structure

The tree structure had been modified to include additional information using Python language. The representation of the dendrogram is created with treelib. The identifier of each node is derived from the dendrogram. Each node contains pointers for its parent and its child nodes. The nodes contain data object, which contains three information. It contains the uuid for searching purposes, the set of the contained zones, and the size of this set.

Based on the improved tree structure, the following process of the enhancement concept is performed.

1. The prediction is performed with the classifier.
2. If the confidence of the predicted class is equal to or higher than the threshold, the process terminates by returning the class as the result.
3. The leaf node in the tree is located by the uuid.
4. Until the confidence of the current node is not reaching the threshold or the root node is reached.
   (a) The parent of this node is selected for examination.
   (b) Its confidence is calculated as the sum of the confidence values of its descendant leaf nodes.
5. The process terminating by returning the contained zones of the lastly examined node.

The predicted room of the classification can be identified as a node in the tree.

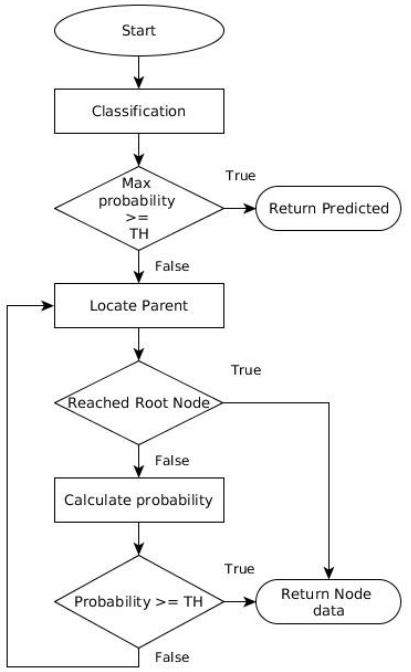


FIGURE 6.5: Flowchart of the process

6.2.1 Experiment

In the experiment, the k–NN and the Naive Bayes classifiers are used to the available functionality to return the class probabilities. These classifiers are selected, because they are easy to parametrize. The threshold is noted as TH, and $TH \in \{0.6, 0.7, 0.8, 0.9, 1\}$. In this experiment, each linkage method is performed for each classifier and threshold. The linkage methods in the experiment are average, complete, single and weighted. The distance function is selected to be the dissimilarity value of the gravitational force-based approach detailed in Section 5.3. The environment is narrowed to rooms on the same level for understandable examination. Different cases can be found in the test, which can present the benefit of the presented concept.

Environment

The narrow the scope of the experiment, the environment is chosen to be the second floor of Miskolc IIS Building defined in IndoorGML. The environment contains 20 zones. The environment can be seen in Figure 6.6.

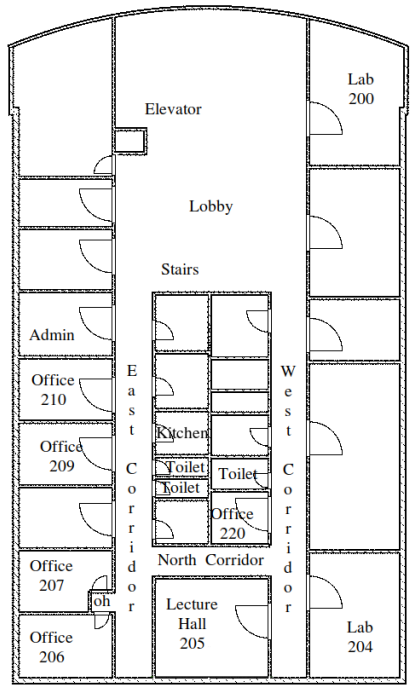


FIGURE 6.6: Second floor of the Miskolc IIS Building

However, the Miskolc IIS Hybrid Dataset contains measurements taken in only 5 of these rooms, namely the *East Corridor*, *West Corridor* and *North Corridor*, the *Lobby* and the *Lecture Hall 205*.

Case

To verify the usability of the presented concept, a beneficial case scenario is presented. Although there are cases, where the enhancing concept is not required or applied. For example, 1–NN will always result in 1 probability in the prediction.

Based on the environment, the weighted linkage method and the gravitational force-based distance, the hierarchical clustering resulted in the dendrogram shown in Figure 6.7.

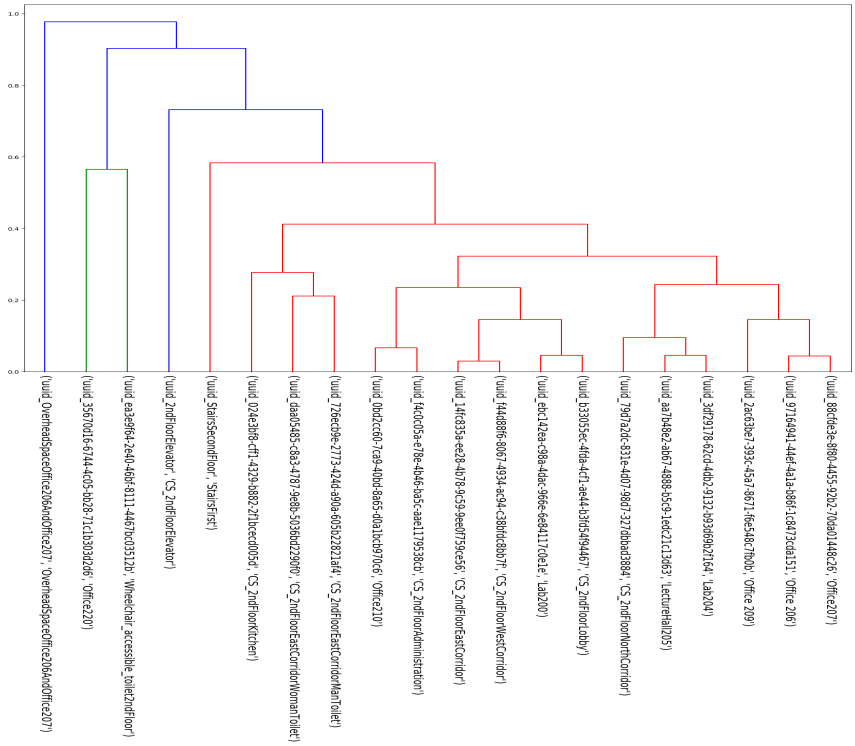


FIGURE 6.7: Dendrogram generated by using Gravitational force-based distance and weighted linkage method

The 9-NN classifier was used a without weighted vote to predict the class using the measured values. Based on the dendrogram presented in Figure 6.7, a graph can be constructed as seen in Figure 6.8. In this graph, the leaf nodes presented in the dataset have probability values for the given measurement. But only two of these nodes have a non-zero value. The first is marked with 11, and it represents the *East Corridor* room. This room has a 0.328 probability in the classification. The second is the *Lobby* denoted by number 14 with 0.672 probability. The actual class node is *East Corridor* marked by green background colour on the graph. A traditional classifier would return with the *Lobby*, because it has the highest probability value.

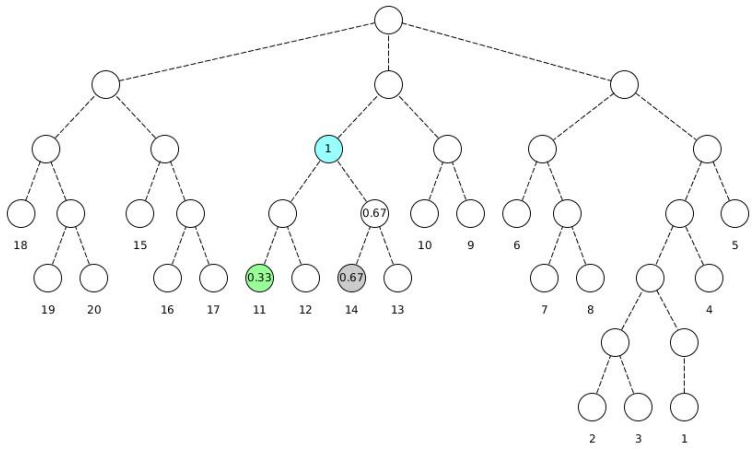


FIGURE 6.8: Example case for advantage of enhancement

However, the concept presented in Section 6.2, instead of returning the predicted class, check whether the probability of the predicted class reaches the given threshold. With 0.7 or above threshold, the enhanced classifier locates the predicted class in the graph, and it examines its parent. Hence, the parent is not the root node, the process continues. As the predicted node has only one sibling with zero probability, the parent also has the probability value below the threshold. For this reason, the searching process moves up one level to the parent. The sum of the probabilities of each descendant leaf node is 1, which could pass any threshold. Thus, the last examined node, with the blue background, is the terminating node, which returns the list of its descendant leaf nodes. The result of the classification process consists of only 4 rooms, namely *East Corridor*, *West Corridor*, *Lab200* and *Lobby*. As it can be seen in Figure 6.8, the actual class is the descendant of the terminating node. Thus, the enhanced concept correctly classified the measurement using only 4 rooms.

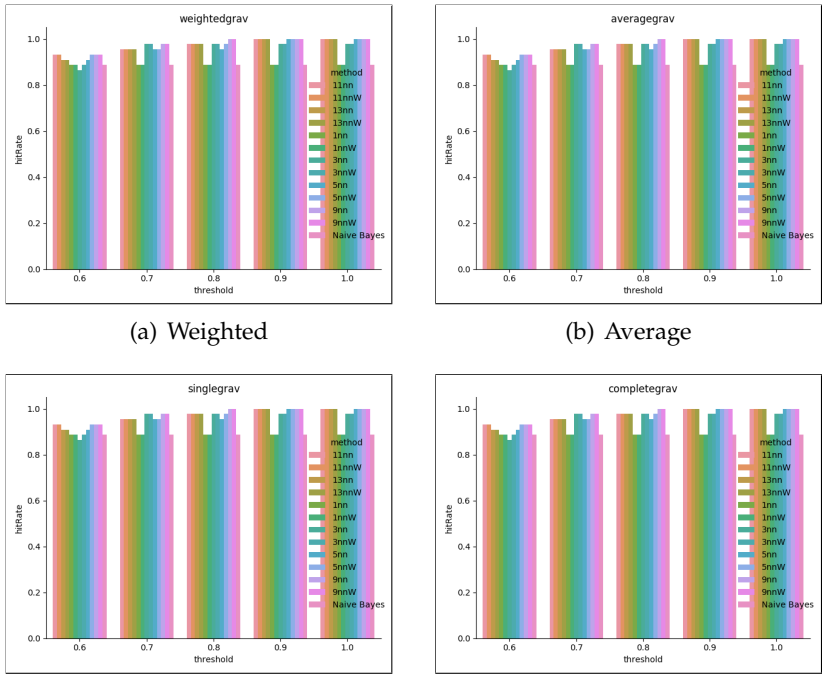
6.2.2 Results

The results are stored in a `csv` file for further processing. The file name contains meta information about the setup, namely the classifier, the linkage method and the threshold. The file contains the following fields. `Correct Classification` can be True or False

based on the containment of the `Actual ID` in the `Predicted IDs` set. `Confidence` is a real value between the threshold and 1, including both values, which represents the accepted confidence of the result. The cardinality of the `Predicted IDs` is stored in the `Set Size` column. The transformation of the selected properties is required for comparison.

Hit

Hit is the associated value for the True or False of `Correct Classification`. Derived from this property of the results, `hitRate` can be calculated for a setup. It is the rate of the correctly classified cases and all the cases to represent the accuracy. Hence, the `hitRate` is a real number in the $[0, 1]$ interval. The goal function is to maximize the `hitRate`.



(a) Weighted (b) Average

(c) Single (d) Complete

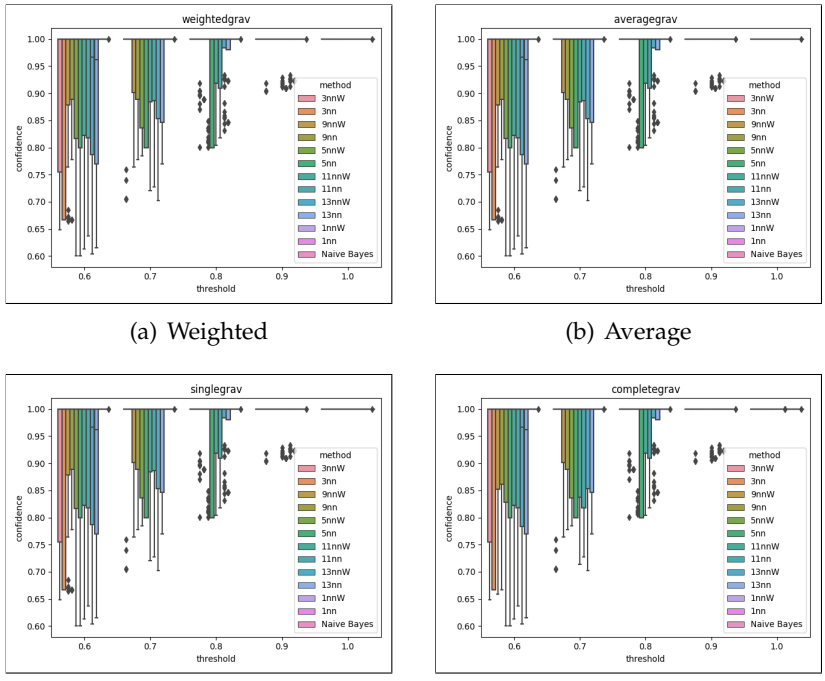
FIGURE 6.9: Hit rates of classifiers tested

The `hitRate` values can be seen in Figure 6.9 for each classifier tested. The values are grouped by both linkage method and threshold. As it can be seen, the linkage method does not have a high impact on the `hitRate` in this test. The graph shows, that 1 `hitRate`

was not achieved using 0.6 or 0.7 threshold. With 0.8 threshold, the 9–NN and 9–NNW were the few classifiers to achieve 1. Moreover, the set of fully correct classifiers does not differ using 0.9 or 1 as threshold. 1–NN 1–NNW and Naive Bayes classifiers did not use the enhancement in the experiment. Although, 3–NN and 3–NNW were able to increase the `hitRate`, these methods stuck below 1.

Confidence

The confidence property of the results is presented in Figure 6.10. It is displayed by box plot, grouped by the classifier, linkage method, and threshold. The goal function is to maximize the confidence values.



(a) Weighted (b) Average

(c) Single (d) Complete

FIGURE 6.10: Confidences of classifiers tested

As seen in Figure 6.10, the linkage method has a slight impact on the confidence values. Weighted, average and single linkage methods resulted in the same statistics of the result set in terms of the confidence property. Compared to the other linkage methods,

the complete linkage method has a few hardly noticeable differences. For example, the minimum confidence values using 9–NN and 9–NNW has decreased in case of 0.6 threshold compared to the others. In this setup, the first quartile is also decreased, while there is no outlier detected. However, the 5–NNW and the 13–NNW developed higher first quartile with the complete linkage method, while no outlier is detected. With 0.7 threshold, 11–NN and 11–NNW achieved a significantly lower first quartile using the complete linkage method, and the minimum of the 11–NNW slightly decreased. In the rest of the thresholds, the difference lies only in the outlier data.

In terms of the classifiers, it can be said that besides the obvious 1–NN and 1–NNW confidence values, the Naive Bayes resulted also 1 confidence with only one outlier, which is only rounded to 1. The third quartile and the maximum value is 1 regardless of the classifier, the linkage method and the threshold. 9–NN and 9–NNW achieved the significantly higher first quartile and minimum using 0.6 threshold. It can be also observed, that the 3–NN and 3–NNW has the first quartile in the 1 value with 0.7 threshold. However, with 0.8 threshold, 5–NN achieved the equality of minimum and first quartile, while there is no outlier data. Some classifiers resulted in the first quartile as 1, however, the number of outliers fairly increased. Most classifiers has all their box plot values as 1 using 0.9 threshold, however, the number of outliers is still relevant.

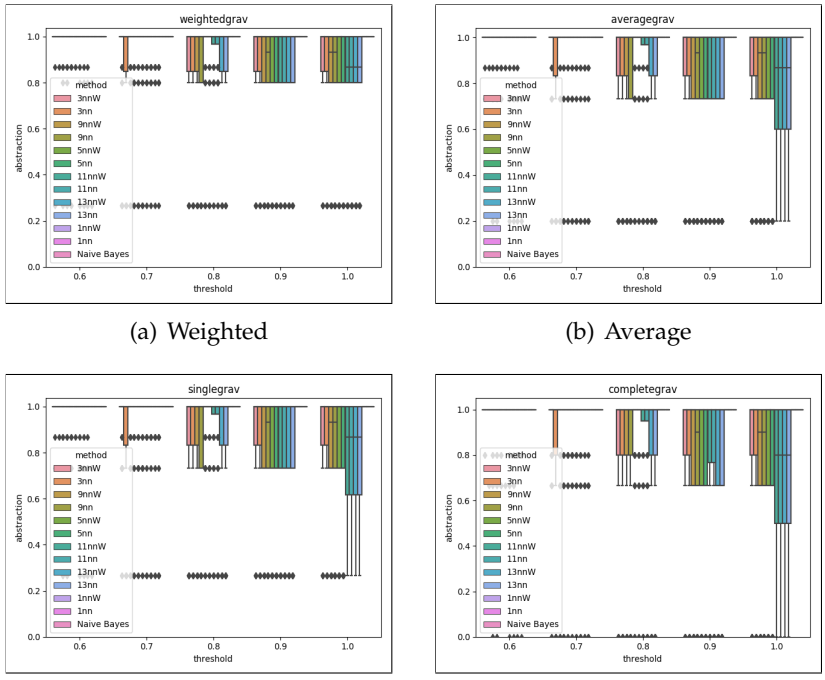
Abstraction

To minimize the size of the resulted list, the abstraction feature is introduced. However, to be consistent with the goal functions of the hitRate and the confidence, the goal for the abstraction should also be maximization. To eliminate the number of rooms from the property, the level of abstraction is designed to be a real number in the $[0, 1]$ range.

$$\hat{a} = 1 - \frac{a - 1}{n - 1} \tag{6.1}$$

Equation 6.1 shows the calculation of abstraction level based on the set size, where a is the set size, n is the number of classes and $\hat{a}$ is the normalized abstraction level. In case the set size is 1, the

abstraction level is 1, while the highest possible set size results 0 as abstraction level.



(a) Weighted (b) Average

(c) Single (d) Complete

FIGURE 6.11: Abstraction of classifiers tested

Figure 6.11 shows the abstraction levels of the classifier, linkage method and threshold setups. As it can be seen, linkage method has a high impact on the abstraction feature. From the point of view of minimal abstraction value, the complete linkage method behaves diverse. It shows that some classifiers have cases when the list of all rooms is the prediction results. Weighted linkage method only treats cases as outlier below 0.8 abstraction with every threshold tested. Moreover, compared to the others, the weighted linkage method does not let the minimum abstraction below 0.8, even with 1 threshold. However, average and single linkage methods mainly differ in the minimal level of abstraction.

From the point of view of the classifiers, the 1–NN, 1–NNW and Naive Bayes have constant abstraction level with 1. However, using 0.6 as the threshold, other classifiers behave alike, except those have outliers. Only the 3–NN has a minimum lower than 1 in case

of 0.7 threshold regardless of the linkage method. With 0.8 threshold, the classifiers that not lowered their minimum are 5–NN and 5–NNW. Moreover, the amount of change in the case of 11-NN and 11–NNW are also low. The other classifiers took the minimum value to the second row of outlier in the graph. Using 0.9 threshold, most of the classifiers took the minimum and first quartile values to the second row of outlier. With the increment of the threshold to 1, 11–NN, 11–NNW, 13–NN and 13–NNW dropped their minimum value last row of outliers, except with weighted linkage.

6.2.3 Tuning

When the accuracy is the main goal, the concept can return all the rooms as the result producing a low abstraction level. Moreover, when the level of abstraction is aimed to be as low as possible, the performance of the classification can be poor. For example, Figure 6.11 shows that the level of abstraction is the best using 0.6 threshold, the confidence of the classifiers, shown in Figure 6.10, is weak, and the accuracy is below potential values.

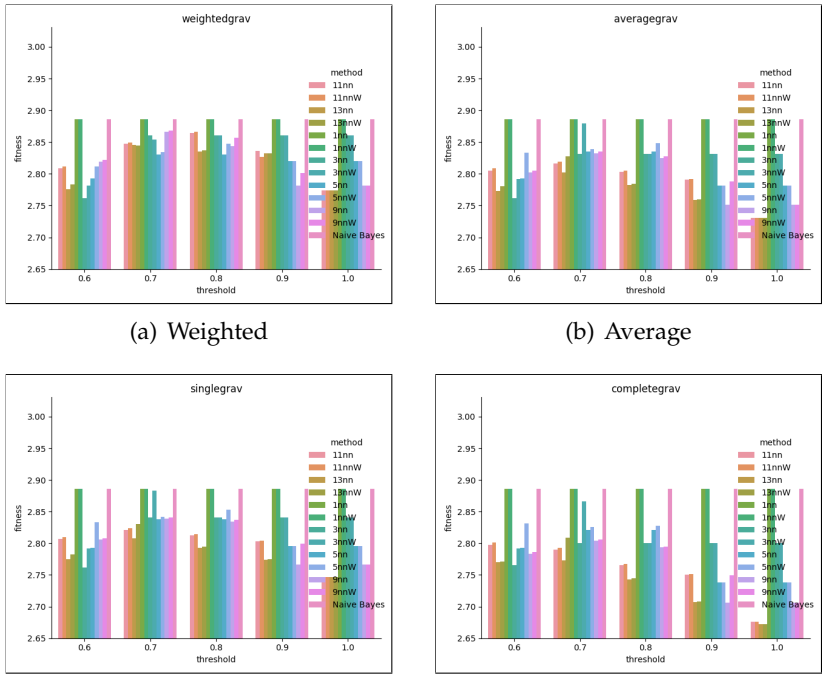
$$fitness = w_h \cdot hitRate + w_c \cdot \overline{confidence} + w_a \cdot \overline{abstraction} \quad (6.2)$$

Therefore, the threshold and the linkage can not be based on only one of these features. To help find the balance, a fitness function is introduced using these features. The introduced fitness function can be seen in Equation 6.2, where w denotes the weight of the given property. The aim is to maximize the fitness value.

The equally weighted fitness value of the tested setups can be seen in Figure 6.12. As can be seen, the 1–NN, the 1–NNW and Naive Bayes did not change their fitness values, however, the difference to 3 fitness value is only the accuracy. The different linkage methods resulted in noticeable differences.

Complete linkage method with 1.0 threshold using 13–NN and 13–NN resulted in the lowest fitness values in the experiment. Compared to the other linkage methods, complete linkage has major differences in fitness values. However, the characteristics of the fitness values is alike to the average and the single linkage methods.

Average and single linkage methods have the fewest differences. The dissimilarity between these two linkage methods grows by the incremental of the threshold. The single linkage method could



(a) Weighted (b) Average

(c) Single (d) Complete

FIGURE 6.12: Fitness of classifiers tested

reach the same or slightly higher fitness values with every classifier.

Weighted linkage method achieved slightly higher fitness values, than single linkage. One of the main differences is in the case of 0.8 threshold using 11–NN, 11–NNW, 13–NN and 13–NNW. The other notable difference appeared using 13–NN and 13–NNW with 0.9 threshold. Weighted linkage method differs partly to the other methods tested. It can be seen, that using 0.7 threshold, the 3–NNW has lower fitness, than the 3–NN, opposite to the results of other methods.

The 3–NNW classifier achieves the highest fitness value besides the 1–NN, 1–NNW and Naive Bayes. As it can be seen, using either average, single or complete linkage method and 0.7 threshold, it is slightly lower, than the local maximum. The incremental of the threshold eventually reduces its fitness values.

Discussion

The increment of the threshold does not necessarily improve the classification properties in every case. There is a value, which in case of further increment, does not have any effect, or even reduces the property value. For example, the abstraction is the most reasonable in the case of 0.8 or 0.9 as threshold.

The variety of linkage methods does not have an impact on the hit rates, and has a low effect on the confidence property. However, the level of abstraction highly depends on this parameter. For example, the complete linkage method resulted all the available rooms in some cases, which resulted in a 0 abstraction level. While the weighted, average and single linkage resulted in at least 0.2 abstraction. Based on the fitness value, the complete linkage method performed the worst in the experiment. The average and single linkage methods are similar in characteristic to the complete linkage, but both performed better. The difference between the average and single linkage methods slightly favors to the single linkage method. However, the weighted linkage method could achieve slightly higher fitness values, than single linkage. Hence, the weighted and the single linkage methods seem to be advisable.

The 3–NNW classifier seems to be the best candidate in the perspective of confidence and abstraction using at least 0.7 as threshold. Moreover, the 3–NNW achieved the highest fitness value, almost reaching the maximal fitness value in the setup.

Naive Bayes classifier was tested on this environment, however, none of its cases used the concept. Therefore the examination in larger scope is admissible.

6.3 Real Life Scenario

In the experiment, the k–NN and the Naive Bayes classifiers are used to the available functionality to return the class probabilities. The threshold is noted as TH, and $TH \in \{0.6, 0.7, 0.8, 0.9, 1\}$. In this experiment, each combination of linkage method and distance function is performed for each classifier and threshold. The linkage methods in the experiment are average, complete, single and weighted. The distance function is selected to be the commonly-used Euclidean distance, and the dissimilarity value of gravitational force-based approach detailed in Section 5.3.

6.3.1 Results

The experimental results are examined in three view-of-point. The first is the improvement of the classification accuracy, previously called `hitRate`. The second is the confidence of the enhancement process. The third is the abstraction level of the process.

Percentage of Enhancement Usage

The percentage of enhancement usage can be expected to be as high as possible in case of low confidence level. Table 6.4 shows the averages of enhancement usage percentage in each classifier tested. The average enhancement usage percentage is 20.53. The 1–NN and the 1–NNW always reach the threshold, hence the enhancement is not applied to these classifiers. The 9–NN, 13–NN and 13–NNW methods used the enhancement method in nearly a third of the cases on average. However the Naive Bayes classifier only adopts the enhancement method in average 2% of its cases. The lowest percentage of enhancement usage is 0.65 in the case of Naive Bayes. The highest value of the minimal percentages is 14.84 in the case of 9–NNW classifier. The average of the minimal percentages is 7.25. The highest percentage of usage is 57.42 in case of 13–NN and 13–NNW classifiers, hence most of half of its cases used the enhancement method. The lowest value of the maximal percentages is 7.74 besides the 1–NN and 1–NNW. The average value of the maximal percentage for each classifier is 34.04.

From the setups, 16 cases resulted in the highest, 57.42 percentage of enhancement usage. All these cases use 13–NN or 13–NNW classifiers with 1 threshold and resulted a 100% accuracy. In most of the cases, the Max Set Size is 71, which is its highest possible value in this experiment. However, the lowest value of Average Set Size 13.6, which resulted by single linkage and gravitational distance with both 13–NN and 13–NNW.

Accuracy

The accuracies are compared to the classification results presented in Table 4.5. With this comparison, the effect of the enhancement can be measured to the classification.

Table 6.5 shows the used classifiers in the experiment, and their accuracy in percentage. In every variant of classifiers, the enhancement could increase its accuracy. On average, a 8 % increase can be

TABLE 6.4: Average percentages of enhancement usage both with Euclidean and Gravitational distance

	Percentage of Enhancement		
	Average	Min	Max
1NN	0.00	0.00	0.00
1NNW	0.00	0.00	0.00
3NN	18.97	1.94	23.23
3NNW	17.68	2.58	23.23
5NN	21.29	3.23	34.84
5NNW	23.35	9.03	34.84
9NN	31.23	14.19	46.45
9NNW	29.03	14.84	46.45
11NN	29.42	12.90	55.48
11NNW	29.68	13.55	55.48
13NN	32.39	10.32	57.42
13NNW	31.74	10.97	57.42
Naive Bayes	2.06	0.65	7.74
Total Result	20.53	0.00	57.42

TABLE 6.5: Comparison of accuracies based on the usage of the enhancement

	Without enhancement	With enhancement
1NN	89.68	91.61
1NNW	89.68	91.61
3NN	92.26	98.71
3NNW	92.26	98.71
5NN	89.03	100
5NNW	90.32	100
9NN	90.97	100
9NNW	92.26	100
11NN	90.32	100
11NNW	90.32	100
13NN	90.32	100
13NNW	90.32	100
Naive Bayes	87.1	90.3

observed. While the highest increment is 12 in the case of 5–NN,

the lowest increment happens with both 1NN and 1–NNW methods. The accuracies can be investigated based on the distance function to test the benefit of the gravitational force-based approach.

The accuracy results based on the distance function are shown in Table 6.6 and Table 6.7. In both Tables, the minimal, the maximal and the average accuracy are shown according to the linkage method and the classifier.

TABLE 6.6: Mean of average accuracies using euclidean distance

	average			complete			single			weighted		
	min	max	avg	min	max	avg	min	max	avg	min	max	avg
1NN	91.61	91.61	91.61	91.61	91.61	91.61	91.61	91.61	91.61	91.61	91.61	91.61
1NNW	91.61	91.61	91.61	91.61	91.61	91.61	91.61	91.61	91.61	91.61	91.61	91.61
3NN	91.61	98.71	97.29	91.61	98.71	97.29	91.61	98.71	97.29	91.61	98.71	97.29
3NNW	92.90	98.71	97.55	92.90	98.71	97.55	92.90	98.71	97.55	92.90	98.71	97.55
5NN	94.84	100	97.94	94.84	99.35	97.42	94.19	99.35	97.55	94.84	99.35	97.42
5NNW	95.48	100	98.45	96.13	99.35	98.06	96.13	99.35	98.19	96.13	99.35	98.06
9NN	96.13	100	98.71	95.48	100	98.45	96.13	100	98.84	96.13	100	98.71
9NNW	96.13	100	98.71	95.48	100	98.45	96.13	100	98.84	96.13	100	98.71
11NN	94.19	100	97.94	93.55	100	97.42	94.19	100	97.81	94.19	100	97.68
11NNW	94.84	100	98.06	94.19	100	97.55	94.84	100	97.94	94.84	100	97.81
13NN	94.84	100	98.32	94.19	100	97.94	94.84	100	98.19	94.84	100	98.06
13NNW	94.84	100	98.32	94.19	100	97.94	94.84	100	98.19	94.84	100	98.06
Naive Bayes	86.45	89.68	87.1	86.45	89.68	87.1	86.45	89.68	87.1	86.45	89.68	87.1
Total	86.45	100	96.28	86.45	100	96.03	86.45	100	96.21	86.45	100	96.13

The statistic of average accuracies using Euclidean distance can be seen in Table 6.6. The lowest value of minimum accuracy 86.45 occurred using Naive Bayes with any linkage method tested. The highest minimum accuracy is 96.13, which was achieved by 5–NNW using complete, single and weighted linkage methods, and by 9–NN and 9–NNW using average, single and weighted linkage. The lowest maximal accuracy is obtained by Naive Bayes classifier with any linkage method tested. The maximal accuracy is 100 in every linkage method, 9–NN, 9–NW, 11–NN, 11–NNW, 13–NN and 13–NNW classifiers reached this value with every linkage method. However, using average linkage method, the 5–NN and 5–NNW classifiers could also achieve 100 as maximal value. The average accuracy is slightly different in each linkage method, but the highest value is 96.28 in the case of average linkage method.

The statistic of average accuracies using gravitational distance can be seen in Table 6.7. The lowest value of minimum accuracy is 89.45 in the case of Naive Bayes classifier with any linkage method

TABLE 6.7: Mean of average accuracies using gravitational distance

	average			complete			single			weighted		
	min	max	avg	min	max	avg	min	max	avg	min	max	avg
1NN	91.61	91.61	91.61	91.61	91.61	91.61	91.61	91.61	91.61	91.61	91.61	91.61
1NNW	91.61	91.61	91.61	91.61	91.61	91.61	91.61	91.61	91.61	91.61	91.61	91.61
3NN	91.61	98.71	97.29	91.61	98.71	97.29	91.61	98.71	97.29	91.61	98.71	97.29
3NNW	92.90	98.71	97.55	92.90	98.71	97.55	92.90	98.71	97.55	92.90	98.71	97.55
5NN	94.84	99.35	97.42	94.84	99.35	97.42	94.84	100	97.94	94.84	99.35	97.42
5NNW	96.13	99.35	98.06	96.13	99.35	98.06	95.48	100	98.45	96.13	99.35	98.06
9NN	96.13	100	98.71	96.13	100	98.71	96.77	100	98.97	96.13	100	98.71
9NNW	96.13	100	98.71	96.13	100	98.71	96.77	100	98.97	96.13	100	98.71
11NN	94.19	100	97.68	94.84	100	97.81	95.48	100	98.19	94.19	100	97.68
11NNW	94.84	100	97.81	95.48	100	97.94	96.13	100	98.32	94.84	100	97.81
13NN	94.84	100	98.06	94.84	100	98.06	96.13	100	98.58	94.84	100	98.06
13NNW	94.84	100	98.06	95.48	100	98.19	96.13	100	98.58	94.84	100	98.06
Naive Bayes	86.45	89.68	87.10	86.45	90.32	87.23	86.45	90.32	87.23	86.45	89.68	87.10
Total	86.45	100	96.13	86.45	100	96.17	86.45	100	96.41	86.45	100	96.13

tested. The highest minimal accuracy is 96.77, which was achieved by using 9–NN and 9–NNW classifiers with single linkage. The lowest maximal accuracy is 89.68 in the case of Naive Bayes classifier and both average and weighted linkage methods. The maximal accuracy is 100 with every linkage method, 9–NN, 9–NNW, 11–NN, 11–NNW, 13–NN and 13–NNW classifiers achieved this accuracy using any linkage method tested. However, the 5–NN and 5–NNW classifiers also resulted in at least one case with 100 accuracy using single linkage method. The average accuracy is similar in each linkage method, although the single linkage method performed the highest average accuracy with 96.41.

Abstraction

The level of abstraction can is represented by the average set size and the max set size. The aim for both parameters is minimization. The max set size can be seen in Table 6.8.

Table 6.8 shows the mean of Max Set Size values in case of linkage and classifier methods using euclidean distance. The lowest average value in this Table is 21 in case of Naive Bayes with weighted linkage, besides the 1–NN and 1–NNW. The average values of other combinations are significantly above this value, as the second lowest average value is 61 with 3–NNW and weighted linkage. The highest average of Max Set Size value is 71 in multiple

TABLE 6.8: Average of Max Set Size

	Euclidean				Gravitational			
	average	complete	single	weighted	average	complete	single	weighted
1NN	1	1	1	1	1	1	1	1
1NNW	1	1	1	1	1	1	1	1
3NN	71	71	70	70.4	55	62.4	43	69
3NNW	71	71	70	61	55	62.4	39.4	69
5NN	71	71	70.4	70.4	46.6	62.4	37	69
5NNW	71	71	71	71	55	69	37	69
9NN	71	71	71	71	55	69	37	69
9NNW	71	71	71	71	55	69	37	69
11NN	71	71	71	71	55	69	37	69
11NNW	71	71	71	71	55	69	37	69
13NN	71	71	71	71	55	69	36.4	69
13NNW	71	71	71	71	55	69	36.4	69
Naive Bayes	71	71	62.8	21	55	42.6	15.4	69
Total Result	60.23	60.23	59.4	55.52	46.05	55	30.35	57.86

cases. Regardless of the classifier, the lowest average of Max Set Size could be achieved by weighted linkage method with 55.52.

The mean of Max Set Size values in case of linkage and classifier methods using gravitational distance can be also seen in Table 6.8. The lowest average value in this Table is 15.4 in the case of Naive Bayes with single linkage, besides the 1–NN and 1–NNW. The average values of other combinations are significantly above this value, as the second lowest average value is 36.4 with 13–NN and 13–NNW using single linkage. The highest average of Max Set Size value is 69 in multiple cases. Regardless of the classifier, the lowest average of Max Set Size could be achieved by single linkage method with 30.35.

Average Set Size The Average Set Size can be expected to be as low as possible. This condition can be seen in Table 6.9.

Table 6.9 shows the mean values of the Set Size Averages using euclidean and gravitational distance. The lowest average set size 1 in the case of 1–NN and 1–NNW, however, the following lowest value occurred using Naive Bayes as 1.26 with Euclidean and 1.24 with gravitational distance. The highest average set size value is 21.34 in case of euclidean distance by using single linkage method and 13–NN classifier. In the case of gravitational distance, the highest value 13.62 occurred using 13–NN and weighted linkage method. The lowest mean of averages for the linkage methods is 8.1 using Euclidean distance with complete linkage, and 5 using gravitational distance with single linkage method.

TABLE 6.9: Mean of Set Size Averages in the Viewpoint of Classifier and Linkage Methods

	Euclidean				Gravitational			
	average	complete	single	weighted	average	complete	single	weighted
1NN	1	1	1	1	1	1	1	1
1NNW	1	1	1	1	1	1	1	1
3NN	9.72	8.22	13.40	8.59	7.21	7.67	4.62	8.92
3NNW	9.01	7.75	12.55	7.84	6.81	7.20	4.38	8.38
5NN	10.15	8.06	14.83	9.67	7.31	8.61	4.99	9.19
5NNW	11.09	8.93	16.27	10.62	8.08	9.47	5.44	10.20
9NN	13.92	11.61	21.17	14.58	10.41	12.83	7.27	13.34
9NNW	13.14	10.75	19.88	13.54	9.78	11.90	6.67	12.46
11NN	12.87	11.10	19.48	13.14	9.68	11.89	6.81	12.42
11NNW	13.14	11.19	19.73	13.25	9.76	11.99	6.86	12.51
13NN	13.94	12.05	21.34	14.59	10.62	13.09	7.40	13.62
13NNW	13.80	11.75	21.06	14.36	10.33	12.92	7.26	13.25
Naive Bayes	1.77	1.94	1.95	1.26	1.62	1.65	1.24	1.77
Total Result	9.58	8.1	14.13	9.49	7.2	8.56	5	9.08

Confidence

The confidence is represented in this experiment by the Average Probability. The Average Probability can be expected to be as high as possible.

The means of average probability values using Euclidean distance are calculated. The highest mean value of average probability is 1 in the case of Naive Bayes with any linkage method tested. The lowest mean value is 0.95 with 11–NN classifier using complete, single or weighted linkage methods. The mean of minimal average probabilities is 0.94, while the lowest minimal average probability is 0.89 in the case of 13–NN and complete linkage method. The highest minimal average value is 1 and it occurs with Naive Bayes classifier with any linkage method. Every classifier resulted in 1 maximal average probability.

The means of average probabilities using Gravitational distance for each classifier and linkage method combination are examined. The highest mean value of average probability is 1 using Naive Bayes classifier with any linkage method examined. The lowest mean value of average probability is 0.95 with 11–NN classifier in every case and 13–NN with complete linkage method. The mean of minimal average probabilities is 0.94, and the highest minimal value is 1 using Naive Bayes. The lowest minimal value is 0.89 using 13–NNW with complete linkage. Every classifier resulted in 1 maximal average probability.

6.3.2 Cases

We can determine combinations, which perform the best according to different criteria.

The Average Set Size is examined based on Table 6.9. The $(1, 10]$ range had been selected for further investigation.

The average accuracy and the percentage of the usage are jointly examined. Some configuration case resulted in 0.65 percentage of usage with 86.45 average accuracy using Naive Bayes classifier. In addition, 3–NN configurations could achieve 1.94 percentage of usage with average accuracy between 92.26 and 92.69. However, the usage of the enhancement method not only could not increase the accuracy in these cases, but it also caused a slight decrease in the accuracy compared to previous work presented in Table 6.5. The other cases did not lead to accuracy decrement.

Cases with the maximal possible value in the experiment as Max Set Size is eliminated. The aim of the enhancement to classify with a given probability without returning with all the possible classes.

In the following, the data set is examined separately based on the distance function.

Euclidean distance

Table 6.10 shows the 6 cases left in the narrowed data set using euclidean distance.

TABLE 6.10: Classifier results of the euclidean distance in the reduced dataset, where TH is the threshold, and ACC is the accuracy

Method name	Linkage	TH	Distance function	ACC	Average probability	% of usage	Average Set Size	Max Set Size
3NNW	single	0.6	euclidean	92.90	0.93	2.58	2.63	66
3NNW	weighted	0.6	euclidean	92.90	0.93	2.58	1.37	21
5NN	weighted	0.6	euclidean	94.84	0.90	3.23	1.70	68
5NN	single	0.6	euclidean	94.19	0.90	3.23	2.59	68
Naive Bayes	single	1	euclidean	89.68	1.00	7.74	4.17	66
Naive Bayes	weighted	1	euclidean	89.68	1.00	7.74	1.76	21

Three classifiers are presented, the 3–NNW, the 5–NN and the Naive Bayes. Each classifier uses both single and weighted linkage and the same thresholds. Four cases use 0.6 threshold, while the others use 1 threshold. The accuracy ranges between 89.68 and 94.84, and the average accuracy is 92.37. Only the 5–NN resulted in different accuracy while using different linkage methods. The average probability is 0.9 in 3–NNW, 0.93 in 5–NN and 1 in Naive Bayes, and the mean value for the six cases is 0.94. The average percentage of usage is 4.5, the average max set size is 51.67. However, two cases resulted in only 21 as max set size, marked with green in the Table.

Gravitational distance

In the narrowed data set, 126 cases is presented, that uses gravitational distance. Based on the properties, different cases can be highlighted.

The accuracy is selected to be used as a filtering property. We include the cases, when the accuracy is the highest occurred value in the dataset. Hence, 13 cases are highlighted, where the lowest average set size is 6.06 and the lowest max set size is 31. It contains mainly setups with single linkage method, only two average linkage method cases appear in these cases. The number of cases with 0.8 threshold is 4, with 0.9 is 7 and with 1 is 2.

Based on the average set size property, the cases with lower than 2 are presented in the highlighting. The total number of these cases is 8, 1 with 1 as threshold and 7 with 0.6 as threshold. All of the four linkage methods are presented, 3 cases with single linkage, 2-2 cases with complete and average, and 1 case with weighted linkage method. However, the average accuracy is 93.55, while the lowest max set size is 13.

Using the max set size as the highlighting property, 3 cases are presented in Table 6.11. The selection is based on the lowest max set size value. Other cases are accepted when their max size value does not exceed the double of the minimal value.

Table 6.11 shows, that the 5–NN classifier is presented twice in this highlighting, while Naive Bayes is presented once. The complete linkage method is omitted, while the average, weighted and single linkage methods are shown. The average accuracy is 93.33, while the average value of the average set size is 1.48. The best

TABLE 6.11: Classifier results of the gravitational distance in the reduced dataset, where TH is the threshold, and ACC is the accuracy

Method name	Linkage	TH	Distance function	ACC	Average probability	% of usage	Average Set Size	Max Set Size
5NN	average	0.6	gravitational	94.84	0.90	3.23	1.24	13
5NN	weighted	0.6	gravitational	94.84	0.90	3.22	1.35	25
Naive Bayes	single	1	gravitational	90.32	1	7.74	1.88	25

case in the view of max set size is the 5–NN with average linkage method using 0.6 threshold.

6.3.3 Discussion

Based on Table 6.4 and Table 6.5, the following observations can be made. The 1–NN and 1–NNW methods do not use the enhancement due to the characteristic of the method. However, it resulted in different accuracy than the previous study due to different implementation. On average, the enhancement is used in 20%, and all the classifiers, except 1–NN and 1–NNW could clearly improve their accuracies.

Table 6.6 and Table 6.7 implies that the gravitational distance could reach or slightly increase the average accuracy in every linkage method tested. The accuracy of Naive Bayes classifier did not vary using euclidean distance in regard to the linkage methods. However, using gravitational distance, the complete and linkage methods show minor improvements. Moreover, the highest minimal accuracy is increased from 96.13 to 96.77 by changing euclidean distance to gravitational distance.

The level of abstraction was presented using Table 6.8 and 6.9. The average of max set size is the lowest in case of gravitational distance using single linkage method. It can be seen in Table 6.8, that gravitational distance resulted in overall lower maximal set sizes than euclidean distance. In addition, the weighted linkage performed the lowest average of max set size with euclidean, but using gravitational distance, the weighted linkage reached significantly worse results than the best performing one. However, in Table

6.9, gravitational distance also performed better in the viewpoint of the set size averages. Besides, single linkage method achieved the worst in case of euclidean distance, but performed the best in gravitational distance.

There is no significant difference between the classifiers, the linkage methods or either the distance functions from the point of view of confidence.

Table 6.10 shows the best cases using Euclidean distance function. The highlighted cases show the options for setups, which use can be suggested based on the priority. When the accuracy has the highest priority, the 5–NN classifier, with weighted linkage method and 0.6 threshold is suggested to be used in symbolic indoor positioning problems. This case can perform decent accuracy, and the average set size is close enough to one, but the maximal size of result set is almost the highest possible.

The usage of gravitational distance function increases the number of possible setups, that fulfills the narrowing criteria. Based on the priority of the properties, different cases can be suggested. In the case of minimization the max set size, the 5–NN classifier with average linkage method and 0.6 threshold can be suggested. This case can perform a decent accuracy, and the average set size is fairly close to one, while the maximal size of result set is acceptably low.

6.4 Conclusions

The proposed concept uses topological information to handle uncertainty in symbolic indoor positioning. The enhancement uses hierarchical grouping in case of a low confidence level.

Thesis 3.

I designed a classification-based symbolic indoor positioning method enhanced by hierarchical clustering, which considers the topology of the building based on the confidence of the classification.

Related Papers: [5], [12]

Chapter 7

Summary

This research is related to positioning or location-awareness in an indoor environment and Smart Environments. Positioning is essential for applications like navigation or tracking. Global Positioning System (GPS) is the most popular positioning system, but it can not be used in an indoor environment due to its unique properties. Indoor Positioning Systems is an active research field since the $1990's$ and a hot topic these days. Although different indoor positioning systems can be found, there is no accepted standard for the problem.

Indoor Positioning Systems (IPS) are based on various technologies, but Hybrid Indoor Positioning has emerged in the $2010's$, that uses multiple technologies and sensors to determine the position. The Miskolc IIS (Institute of Information Science) Hybrid IPS Dataset was constructed to provide a static context for the evaluation of the classification-based symbolic indoor positioning methods. It contains measurements of multiple built-in sensors of mobile phones, such as Bluetooth, WiFi and Magnetometer. The measurements were recorded in a three-story building of the University of Miskolc. This dataset is available in the UCI Machine Learning Repository. This data set was useful for the scientific community which is demonstrated by the high number of downloads.

Symbolic positioning was considered as a classification task, where the classes are the positions and the attributes are the measured values. The experiment was focused on the performance of well-known classification methods such as k–NN, Naive Bayes, Decision Tree, Rule Induction and Artificial Neural Network (ANN) using RapidMiner and Weka. The k–NN algorithm is recommended because it could achieve high accuracy, and does not require modification when the training set changes. The ANN method is also recommended due to its high accuracy; however, its high training

time could limit its applicability in frequently changing environments.

Initially, the evaluation was based on the CRISP approach, which is a classic way to determine the accuracy of a classifier. Experimental results show that a more accurate classification method predicts further the symbolic positions from the original location when it is misclassified, than a less accurate classifier. Furthermore, less accurate classifiers often predict the neighboring, and nearby symbolic position. Hence, the CRISP approach is not sufficient to evaluate classifiers for indoor positioning purposes, because it does not take into account the topology.

As a major contribution of my research, a new, gravitational force-based approach is presented, which considers the topology of the building in the classification error. The error values based on the new approach should be proportional to the sizes of the Zones, reflect the layout of the Zones and be asymmetrical in case of size differences. It has three parameters, which are the capacity function, the reference points and the distance function. Applicability of the presented method is demonstrated with two experiments in the two-dimensional space. The topology of the environment is modeled with IndoorGML (Indoor Geographic Markup Language). Based on the experimental results, the presented approach fulfills the requirements for considering the topology.

The proposed method was compared to the traditional CRISP approach in a classification task. The comparison was performed over a data set and a map. The IndoorGML standard can help to calculate the distance of rooms, both in coordinate system and graphs. Therefore, three variants of the proposed method were selected based on the capabilities of the IndoorGML.

Analysis of the presented evaluation methods consisted of two points of view. Firstly, the classifiers were ranked based on the classification error values. Secondly, the distribution of the proportion of classifiers was determined by dividing the error values by the best value. Based on the experimental results, the usage of the proposed gravitational force-based approach in the case of an indoor positioning application is recommended instead of the CRISP approach.

To improve the classification for symbolic indoor positioning purposes, a novel method was required. Some well-known classifier accepts classes as a prediction based on confidence values. But when the confidence for each class is relatively small, the accuracy

of the classifier vary in a moderate range. To overcome this occurrence, the concept of hierarchical grouping enhanced classification can be introduced. The classification enhancement requires three parameters, namely classifier, threshold, and dendrogram. Hierarchical clustering algorithms can be applied using a topological description to generate the grouping. The different setups were compared by the entanglement value to identify those with similar behaviors.

Different features can be introduced to evaluate the enhanced classification process. The accuracy is represented by the hit rate, the confidence is used to record the accepted value, while the abstraction shows the goodness of the size of the predicted classes. Since the features are contradictory, the tuning of these features is examined in a narrow environment to find the best combination of dendrogram, threshold and classifier using gravitational force-based distance.

The test in a real life scenario is executed to observe the behavior of the enhanced classification. The percentage of usage, the accuracy, the abstraction and the confidence is analysed. Due to the accuracy feature, the concept is also compared to the traditional classification process. The enhanced classification seems to be advantageous based on our experimental results in the case of indoor positioning purposes.

7.1 Contributions

To the best of our knowledge, there was no data set that allowed the comparison of various indoor positioning methods using multiple technologies at the time. Hence, the Miskolc IIS Hybrid IPS data set was constructed, which allows the comparison of indoor positioning algorithm based on multiple sensors. The created data set was used as a base for my further works. The contribution of this data set to the research field is shown by independent citations.

Thesis 1.

Room-level indoor positioning can be considered as a classification problem. I created a data set, which allows benchmarking of classification-based symbolic indoor positioning methods.
Related Publications: [6], [9], [7], [4].
Citations: [CGOC18], [CGOC17], [MS+19], [Sat18], [MS+18], [Bog17],

[Fen+20], [YZZ20], [SS+20], [NMN20], [AM20], [Elg+20], [Ara+20], [MGT18]

As far as we know, there was no application-specific approach to the evaluation of classifiers for indoor positioning purposes. A novel method was presented for classification error calculation for symbolic indoor positioning purposes. This approach uses the topology of the building as domain-specific knowledge, and uses the gravitational force as the base idea. The proposed method was compared to the traditional CRISP approach, and results imply that gravitational force-based approach can be beneficial.

Thesis 2.

I proposed a topology-based evaluation method for classification-based indoor positioning algorithms which allows a more detailed evaluation.
Related Publications:[9], [11], [8], [10] ,[1], [2].
Citations: [HHAR19],[Wan+20][YZZ20],[Fen+20],[Yan+21]

A novel method was required to improve the classification for symbolic indoor positioning purposes, which considers the topology. Some well-known classifier accepts classes as a prediction based on confidence values. But when the confidence for each class is relatively small, the accuracy of the classifier varies in a moderate range. To overcome this occurrence, the concept of hierarchical grouping enhanced classification can be introduced. Experimental results show that the enhanced classification could found the balance between accuracy and the abstraction level.

Thesis 3.

I designed a classification-based symbolic indoor positioning method enhanced by hierarchical clustering, which considers the topology of the building based on the confidence of the classification.
Related Papers: [5], [12]
Citations: –

Összegzés

Ez a kutatás a pozicionálás vagy helyzetmeghatározás beltéri környezetben és az intelligens környezet témaköréhez kapcsolódik. A pozicionálás elengedhetetlen az olyan alkalmazásokhoz, mint a navigáció vagy a nyomon követés A globális helymeghatározó rendszer (GPS) a legnépszerűbb pozicionáló rendszer, viszont a beltér egyedi tulajdonságai miatt nem használható. A beltéri helyzetmeghatározó rendszerek az 1990-es évek óta aktív kutatási területnek számít, és manapság is kutatott téma. Habár léteznek különböző beltéri helyzetmeghatározó rendszerek, nincs elfogadott szabvány a kérdéskör megoldására.

Beltéri helyzetmeghatározó rendszerek különböző technológiákon alapulnak, de az elmúlt évtizedben kialakult a hibrid beltéri helyzetmeghatározás, amely több technológiát használ a pozíció meghatározásra. A Miskolc IIS Hibrid IPS adathalmaz statikus kontextus biztosítására lett létrehozva az osztályozási módszerek értékelésére. A mobiltelefon több beépített szenzorainak, például Bluetooth, Wifi és magnetométer méréseit tartalmazza. A mérések a Miskolci Egyetem három emeletes épületében lettek elvégezve. Ez az adathalmaz elérhető a UCI Machine Learning Repositoryban. Az adathalmaz alapjául szolgált a további munkáimnak.

A szimbolikus pozicionálást osztályozási feladatnak lehet tekinteni, ahol az osztályok a pozíciók és az attribútumok a mért értékek. A kísérlet olyan ismert osztályozási módszerekre összpontosított, mint például a k–NN, a Naív Bayes, a döntési fa, a szabályindukció, és a mesterséges neurális hálózat, a RapidMiner és a Weka alkalmazásával. A k–NN osztályozó ajánlott, mivel nagy pontosságot tud elérni, és nem igényel módosítást, amikor a tanítóhalmaz megváltozik. A mesterséges neurális hálózat szintén ajánlott a nagy pontossága miatt, a magas tanítási költség viszont korlátozhatják az alkalmazhatóságát gyakran változó környezetben.

Kezdetben az értékelés a CRISP megközelítésen alapult, amely klasszikus módja az osztályozó pontosságának meghatározására.

A kísérleti eredmények azt mutatják, hogy egy pontosabb osztályozó tévesztés esetén távolabbi szimbolikus pozíciót ad eredményül, mint egy kevésbé pontos osztályozó. Ezenkívül a kevésbé pontos osztályozó gyakran szomszédos vagy közeli szimbolikus pozíciót határoz meg, amikor téveszt. Tehát a CRISP megközelítés nem elegendő a beltéri helyzetmeghatározási célú osztályozók értékeléséhez, mivel nem veszi figyelembe a környezet topológiáját.

Kutatásom jelentős hozzájárulásaként bemutatom az új, tömegvonzáson alapuló megközelítést, amely figyelembe veszi az épület topológiáját az osztályozási hibában. Az új megközelítésen alapuló hibaértékeknek arányosnak kell lenniük a zónák méreteivel, tükrözniük kell a zónák elhelyezkedését és aszimmetrikusnak kell lenniük méretbeli különbségek esetén. Három paraméterrel rendelkezik, amely a kapacitásfüggvény, a referencia pontok és a távolság függvény. A bemutatott módszer alkalmazhatóságát két kísérlettel demonstráljuk a kétdimenziós térben. A környezet topológiáját az IndoorGML segítségével írjuk le. A kísérleti eredmények alapján a bemutatott megközelítés teljesíti a topológia figyelembevételének kapcsán támasztott követelményeit.

A javasolt módszert összehasonlítottuk a klasszikus CRIPS megközelítéssel egy osztályozási folyamat során. Az összehasonlítást egy adathalmaz és egy térkép segítségével végeztük. Az IndoorGML szabvány segít kiszámítani a szobák távolságát, mind a koordináta rendszerben, mind gráfokban. Ezért a javasolt módszer három változatát választottuk az IndoorGML képességei alapján.

A bemutatott értékelési módszerek elemzése két szempontból állt. Először az osztályozókat az osztályozási hibaértékek alapján rangsoroltam. Másodszor, az osztályozók arányának eloszlását vizsgáljuk, ahol a legjobb értékkel skálázzuk. A kísérleti eredmények alapján a javasolt tömegvonzás alapú megközelítés használata a beltéri helyzetmeghatározási alkalmazás esetén a CRISP megközelítés helyett ajánlott.

A szimbolikus beltéri helyzetmeghatározási célú osztályozás javításához új módszerre volt szükség. Néhány osztályozó konfidencia érték alapján ad előrejelzést. Abban az esetben, amikor a konfidencia érték minden osztályra viszonylag alacsony, az osztályozási pontosság nagy mértékben ingadozik. Ennek a kiküszöbölésére bevezethető a hierarchikus csoportosítással javított osztályozás fogalma. Az osztályozás ily módon való fejlesztésének három paraméterre van szüksége, nevezetesen az osztályozó, a küszöbérték és

a dendrogram. A hierarchikus klaszterezési algoritmusok topológia leírás alapján alkalmazhatóak a csoportosítás létrehozására. A különféle beállításokat összehasonlítottuk összeakadás érték alapján, hogy azonosítsuk a hasonlóan viselkedő beállításokat.

Különböző jellemzőket lehet bevezetni a javított osztályozási folyamat értékeléséhez. A pontosságot a találati arány képviseli, a megbízhatóságot használják az elfogadott érték rögzítésére, míg az absztrakció a jósolt osztályok méretének jóságát mutatja. Mivel a bevezetett jellemzők ellentmondanak egymásnak, ezért ezen jellemzők hangolását szűk környezetben vizsgáljuk meg. Így megtaláljuk a hierarchikus klaszterezés kapcsolási módszerének, a küszöbértéknek és az osztályozónak a legjobb kombinációját tömegvonzás alapú távolság felhasználásával.

A javított osztályozási viselkedésének megfigyelésére tesztet végeztünk el valós életbeli forgatókönyv alapján. Elemezzük a használati százalékot, a pontosságot, az absztrakciót és a megbízhatóságot. A pontosság miatt a koncepciót össze lehet hasonlítani a hagyományos osztályozási folyamattal is. Kísérleti eredményeink alapján a javított osztályozás előnyösnek tűnik beltéri helyzetmeghatározási alkalmazás esetén.

Tudományos Eredmények

Legjobb tudomásunk szerint nem állt rendelkezésre olyan adathalmaz, amely lehetővé tenné a különféle beltéri helymeghatározási módszerek összehasonlítását egyszerre több technológia felhasználásával. Ezért elkészült a Miskolc IIS Hibrid IPS adathalmaz, amely lehetővé teszi beltéri helyzetmeghatározó algoritmusok összehasonlítását több szenzor alapján. Az adathalmaznak a kutatási területhez való hozzájárulását jól mutatják a független idézetek.

1. Tézis

A szobaszintű beltéri helyzetmeghatározás osztályozási problémának tekinthető. Készítettem egy adathalmazt, amely lehetővé teszi az osztályozás alapú szobaszintű beltéri helyzetmeghatározási módszerek teljesítményértékelését.
Kapcsolódó cikkek: [6], [9], [7], [4].
Idézések: [CGOC18],[CGOC17],[MS+19],[Sat18], [MS+18],[Bog17], [Fen+20], [YZZ20], [SS+20], [NMN20],[AM20], [Elg+20], [Ara+20], [MGT18]

Tudomásunk szerint nem volt alkalmazásspecifikus megközelítés a beltéri helyzetmeghatározási célú osztályozók értékeléséhez. Új módszer került bemutatásra az osztályozási hiba kiszámításához szimbolikus beltéri pozicionálás céljára. Ez a megközelítés az épület topológiáját használja témaspecifikus tudásként, és alapötletként a tömegvonzást használja. A javasolt módszert összehasonlítottuk a hagyományos CRISP megközelítéssel, és az eredmények azt mutatják, hogy a tömegvonzás alapú megközelítés előnyösnek tűnik.

2. Tézis

Definiáltam egy topológia alapú értékelési módszert az osztályozáson alapuló beltéri helyzetmeghatározási algoritmusokhoz, amely részletesebb értékelést tesz lehetővé.
Kapcsolódó cikkek:[9], [11], [8], [10] ,[1], [2].
Idézések: [HHAR19],[Wan+20],[YZZ20],[Fen+20], [Yan+21]

A szimbolikus beltéri helyzetmeghatározási célú osztályozás javításához új módszerre volt szükség, amely figyelembe veszi a topológiát. Néhány ismert osztályozó a konfidenciaérték alapján ad előrejelzést. Abban az esetben, amikor minden osztály konfidenciaértéke alacsony, az osztályozási pontosság ingadozik. Ennek az előfordulásnak a kiküszöbölésére bevezethető a hierarchikus csoportosítás által javított osztályozás fogalma. A kísérleti eredmények azt mutatják, hogy a javított osztályozás megtalálhatja az egyensúlyt a pontosság és az absztrakciós szint között.

3. Tézis

Hierarchikus klaszterezéssel kiegészítettem osztályozáson alapuló szimbolikus beltér helyzetmeghatározási módszereket, amelyek így figyelembe veszik az épület topológiáját az osztályozás megbízhatósága alapján
Kapcsolódó cikkek: [5], [12]

Idézések: –

Bibliography

[Abb+19] Moustafa Abbas et al. "WiDeep: WiFi-based accurate and robust indoor localization system using deep learning". In: *2019 IEEE International Conference on Pervasive Computing and Communications (PerCom*. IEEE. 2019, pp. 1–10.

[AM20] Fahad Alhomayani and Mohammad H. Mahoor. "Deep learning methods for fingerprint-based indoor positioning: a review". In: *Journal of Location Based Services* 14.3 (2020), pp. 129–200. DOI: 10.1080/17489725.2020.1817582. eprint: https://doi.org/10.1080/17489725.2020.1817582. URL: https://doi.org/10.1080/17489725.2020.1817582.

[Ara+20] Fernando J. Aranda, Felipe Parralejo, Fernando J. Álvarez, and Joaquín Torres-Sospedra. "Multi-Slot BLE Raw Database for Accurate Positioning in Mixed Indoor/Outdoor Environments". In: *Data* 5.3 (2020). ISSN: 2306-5729. DOI: 10.3390/data5030067. URL: https://www.mdpi.com/2306-5729/5/3/67.

[BA78] Roger K Blashfield and Mark S Aldenderfer. "The literature on cluster analysis". In: *Multivariate Behavioral Research* 13.3 (1978), pp. 271–295.

[Bar+16] P. Barsocchi, Antonino Crivello, D. Rosa, and Filippo Palumbo. "A multisource and multivariate dataset for indoor

localization methods based on WLAN and geo-magnetic field fingerprinting". In: *2016 International Conference on Indoor Positioning and Indoor Navigation (IPIN)* (2016), pp. 1–8.

[BD05] Christian Becker and Frank Dürr. "On location models for ubiquitous computing". In: *Personal and Ubiquitous Computing* 9.1 (2005), pp. 20–31.

[Bel58] Richard Bellman. "On a routing problem". In: *Quarterly of Applied Mathematics* 16.1 (1958), pp. 87–90.

[Bog17] Bence Bogdándy. "Comparison and Implementation of WiFi RSSI Filtering Methods". In: (2017). Student Research.

[Bor+05] Gaetano Borriello et al. "WALRUS: wireless acoustic location with room-level resolution using ultrasound". In: *Proceedings of the 3rd international conference on Mobile systems, applications, and services*. ACM. 2005, pp. 191–203.

[BP00] Paramvir Bahl and Venkata N Padmanabhan. "RADAR: An in-building RF-based user location and tracking system". In: *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*. Vol. 2. IEEE. Tel Aviv,Israel, 2000, pp. 775–784.

[Bra+14] Ferenc Brachmann et al. "The effects of hardware and software-based signal distortion in multi-platform indoor positioning systems". In: *Proceedings of the 11th International Symposium on Location-Based Services, ISBN*. Vienna, Austria, 2014, pp. 978–1.

[Bra86] Bart Braden. "The Surveyor's Area Formula". In: *The College Mathematics Journal* 17.4 (1986), pp. 326–337. ISSN: 07468342, 19311346. URL: http://www.jstor.org/stable/2686282.

[BS93] Alison K Brown and Mark A Sturza. *Vehicle tracking system employing global positioning system (gps) satellites*. US Patent 5,225,842. 1993.

[Car+18] Ulises Carrasco, Pedro Daniel Urbina Coronado, Mahmoud Parto, and Thomas Kurfess. "Indoor location service in support of a smart manufacturing facility". In: *Computers in Industry* 103 (2018), pp. 132 –140. ISSN: 0166-3615. DOI: https://doi.org/10.1016/j.compind.2018.09.009. URL: http://www.sciencedirect.com/science/article/pii/S0166361517306966.

[CD07] Padraig Cunningham and Sarah Jane Delany. "k-Nearest neighbour classifiers". In: *Multiple Classifier Systems* 34 (2007), pp. 1–17.

[CGOC17] Edwin Cabrera-Goyes and Diego Ordóñez-Camacho. "Towards a Bluetooth Indoor Positioning System with Android Consumer Devices". In: *Information Systems and Computer Science (INCISCOS), 2017 International Conference on*. IEEE. 2017, pp. 56–59.

[CGOC18] Edwin Cabrera-Goyes and Diego Ordóñez-Camacho. "Posicionamiento en espacios interiores con Android, Bluetooth y RSSI". In: *Enfoque UTE* 9.1 (2018), pp. 118–126.

[Con] *Confusion Matrix*. http://aimotion.blogspot.hu/2010/08/tools-for-machine-learning-performance.html. [Online; accessed 5-September-2016].

[Dav+12] Jim Davis et al. "Smart manufacturing, manufacturing intelligence and demand-dynamic performance". In: *Computers & Chemical Engineering* 47 (2012). FOCAPO 2012, pp. 145 –156. ISSN: 0098-1354. DOI: https://doi.org/10.1016/j.compchemeng.2012.06.037. URL: http://www.sciencedirect.com/science/article/pii/S0098135412002219.

[Elg+20] Kevin Elgui, Pascal Bianchi, François Portier, and Olivier Isson. "Learning methods for RSSI-based geolocation: A comparative study". In: *Pervasive and Mobile Computing* 67 (2020), p. 101199. ISSN: 1574-1192. DOI: https://doi.org/10.1016/j.pmcj.2020.101199. URL: http://www.sciencedirect.com/science/article/pii/S1574119220300687.

[Fel+03] Silke Feldmann, Kyandoghere Kyamakya, Ana Zapater, and Zighuo Lue. "An Indoor Bluetooth-Based Positioning System: Concept, Implementation and Experimental Evaluation." In: *International Conference on Wireless Networks*. 2003, pp. 109–113.

[Fen+20] Yanxiao Feng, Julian Wang, Howard Fan, and Ce Gao. "BIMIL: Automatic Generation of BIM-Based Indoor Localization User Interface for Emergency Response". In: *HCI International 2020 – Late Breaking Posters*. Ed. by Constantine Stephanidis, Margherita Antona, and Stavroula Ntoa. Cham: Springer International Publishing, 2020, pp. 184–192. ISBN: 978-3-030-60700-5.

[Flo62] Robert W Floyd. "Algorithm 97: shortest path". In: *Communications of the ACM* 5.6 (1962), p. 345.

[FM04] Giles M Foody and Ajay Mathur. "A relative evaluation of multiclass image classification by support vector machines". In: *IEEE Transactions on Geoscience and Remote Sensing* 42.6 (2004), pp. 1335–1343.

[GB05] Jerzy W Grzymala-Busse. "Rule induction". In: *Data Mining and Knowledge Discovery Handbook*. Springer, 2005, pp. 277–294.

[Goo] *Google Indoor Maps*. `https://www.google.com/intl/en/maps/about/partners/indoormaps/`. [Online; accessed 02-Nov-2015].

[Gro+11] Robert M Groves et al. *Survey methodology*. Vol. 561. John Wiley & Sons, 2011.

[Gro+19] Bernhard Groβwindhager et al. "SnapLoc: An ultra-fast UWB-based indoor localization system for an unlimited number of tags". In: *2019 18th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*. IEEE. 2019, pp. 61–72.

[Gua+20] Weipeng Guan et al. "Indoor Localization System of ROS mobile robot based on Visible Light Communication". In: *arXiv preprint arXiv:2001.01888* (2020).

[Hal+09] Mark Hall et al. "The WEKA data mining software: an update". In: *ACM SIGKDD explorations newsletter* 11.1 (2009), pp. 10–18.

[Han+17] David Hanley et al. "MagPIE: A dataset for indoor positioning with magnetic anomalies". In: *2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. IEEE. 2017, pp. 1–8.

[HC08] Christopher J Hegarty and Eric Chatre. "Evolution of the global navigation satellitesystem (gnss)". In: *Proceedings of the IEEE* 96.12 (2008), pp. 1902–1917.

[HC10] Kyuwon Han and Sung Ho Cho. "Advanced LANDMARC with adaptive k-nearest algorithm for RFID location system". In: *Proceedings of the 2nd IEEE International Conference on Network Infrastructure and Digital Content (ICNIDC'10), Sep 24–26, 2010, Beijing, China*. IEEE Piscataway, NJ, USA. 2010, pp. 595–598.

[HHAR19] M. N. Hashim, M. I. Hassan, and A. Abdul Rahman. "MOBILE INDOOR LASER SCANNING FOR 3D STRATA REGISTRATION PURPOSES BASED ON INDOORGML". In: *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* XLII-4/W16 (2019), pp. 241–245. DOI: 10.5194/isprs-archives-XLII-4-W16-241-2019. URL: https://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XLII-4-W16/241/2019/.

[Hig+01] Jeffrey Hightower, Chris Vakili, Gaetano Borriello, and Roy Want. "Design and calibration of the spoton ad-hoc location sensing system". In: *unpublished, August* (2001).

[HK00] Eui-Hong Sam Han and George Karypis. "Centroid-based document classification: Analysis and experimental results". In: *European conference on principles of data mining and knowledge discovery*. Springer. 2000, pp. 424–431.

[HK13] Markus Hofmann and Ralf Klinkenberg. *RapidMiner: Data Mining Use Cases and Business Analytics Applications*. Chapman & Hall/CRC, 2013. ISBN: 1482205491, 9781482205497.

[Hoe+19] Fabian Hoeflinger et al. "Passive indoor-localization using echoes of ultrasound signals". In: *2019 International Conference on Advanced Information Technologies (ICAIT)*. IEEE. 2019, pp. 60–65.

[Hss+14] Badr Hssina, Abdelkarim Merbouha, Hanane Ezzikouri, and Mohammed Erritali. "A comparative study of decision tree ID3 and C4. 5". In: *International Journal of Advanced Computer Science and Applications* 4.2 (2014), pp. 0–0.

[HWB00] Jeffrey Hightower, Roy Want, and Gaetano Borriello. "SpotON: An indoor 3D location sensing technology based on RF signal strength". In: *UW CSE 00-02-02, University of Washington, Department of Computer Science and Engineering, Seattle, WA* 1 (2000).

[JHL97] Biing-Hwang Juang, Wu Hou, and Chin-Hui Lee. "Minimum classification error rate methods for speech recognition". In: *IEEE Transactions on Speech and Audio Processing* 5.3 (1997), pp. 257–265.

[JL95] George H. John and Pat Langley. "Estimating Continuous Distributions in Bayesian Classifiers". In: *Eleventh Conference on Uncertainty in Artificial Intelligence*. San Mateo: Morgan Kaufmann, 1995, pp. 338–345.

[JMF99] Anil K Jain, M Narasimha Murty, and Patrick J Flynn. "Data clustering: a review". In: *ACM computing surveys (CSUR)* 31.3 (1999), pp. 264–323.

[KEP99] RK-S Kwan, Alan C Evans, and G Bruce Pike. "MRI simulation-based evaluation of image-processing and classification methods". In: *IEEE Transactions on Medical Imaging* 18.11 (1999), pp. 1085–1097.

[KK04] Kamol Kaemarungsi and Prashant Krishnamurthy. "Modeling of indoor positioning systems based on location fingerprinting". In: *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*. Vol. 2. IEEE. Hong Kong, China, 2004, pp. 1012–1022.

[KT14] Szabolcs Karsai and Zsolt Tóth. "Comparison of WiFi-based indoor positioning techniques". In: *Proceedings of the 1st International Conference on Future RFID Technologies* (Eger, Hungary). Nov. 2014.

[KY10] Hakan Koyuncu and Shuang Hua Yang. "A survey of indoor positioning and object locating systems". In: *IJCSNS International Journal of Computer Science and Network Security* 10.5 (2010), pp. 121–128.

[Las+14] Heiner Lasi et al. "Industry 4.0". In: *Business & information systems engineering* 6.4 (2014), pp. 239–242.

[Lee+14] J Lee et al. "OGC® indoorgml". In: *Open Geospatial Consortium standard* (2014).

[Lei+01] Charles Eric Leiserson, Ronald L Rivest, Thomas H Cormen, and Clifford Stein. *Introduction to algorithms*. Vol. 6. MIT press Cambridge, MA, 2001.

[Li+12] Binghao Li, Thomas Gallagher, Andrew G Dempster, and Chris Rizos. "How feasible is the use of magnetic field alone for indoor positioning?" In: *Indoor Positioning and Indoor Navigation (IPIN), 2012 International Conference on*. IEEE. 2012, pp. 1–9.

[Li+14] Liqun Li et al. "Epsilon: A Visible Light Based Positioning System." In: *NSDI*. 2014, pp. 331–343.

[Lic13] M. Lichman. *UCI Machine Learning Repository*. 2013. URL: http://archive.ics.uci.edu/ml.

[Liu+07] Hui Liu, Houshang Darabi, Pat Banerjee, and Jing Liu. "Survey of wireless indoor positioning techniques and systems". In: *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on* 37.6 (2007), pp. 1067–1080.

[Loh+17] Elena Simona Lohan et al. "Wi-Fi crowdsourced fingerprinting dataset for indoor positioning". In: *Data* 2.4 (2017), p. 32.

[Lu+17] Shaoping Lu, Chen Xu, Ray Y. Zhong, and Lihui Wang. "A RFID-enabled positioning system in automated guided vehicle for smart factories". In: *Journal of Manufacturing Systems* 44 (2017), pp. 179 –190. ISSN: 0278-6125. DOI: https://doi.org/10.1016/j.jmsy.2017.03.009. URL: http://www.sciencedirect.com/science/article/pii/S0278612517300390.

[McD+07] Erik McDermott et al. "Discriminative training for large-vocabulary speech recognition using minimum classification error". In: *IEEE Transactions on Audio, Speech, and Language Processing* 15.1 (2007), pp. 203–223.

[MGT18] Kostadin Mishev, Ana Gjorgjevikj, and Dimitar Trajanov. "Probabilistic approach by using N-peak Gaussian distribution for indoor positioning". In: *ICT Innovations 2018, Web Proceedings ISSN 1857-7288* (2018), pp. 61–74.

[MS+18] Germán Martín Mendoza-Silva et al. "Long-Term WiFi Fingerprinting Dataset for Research on Robust Indoor Positioning". In: *Data* 3.1 (2018), p. 3.

[MS+19] Germán Martín Mendoza-Silva, Miguel Matey-Sanz, Joaquín Torres-Sospedra, and Joaquín Huerta. "BLE RSS Measurements Dataset for Research on Accurate Indoor Positioning". In: *Data* 4.1 (2019), p. 12.

[MY01] Larry M Manevitz and Malik Yousef. "One-class SVMs for document classification". In: *Journal of Machine Learning Research* 2.Dec (2001), pp. 139–154.

[New14] Nic Newman. "Apple iBeacon technology briefing". In: *Journal of Direct, Data and Digital Marketing Practice* 15.3 (2014), pp. 222–225.

[New99] Isaac Newton. *The Principia: mathematical principles of natural philosophy*. Univ of California Press, 1999.

[Ni+04] Lionel M Ni, Yunhao Liu, Yiu Cho Lau, and Abhishek P Patil. "LANDMARC: indoor location sensing using active RFID". In: *Wireless networks* 10.6 (2004), pp. 701–710.

[NMN20] A. Narzullaev, Z. Muminov, and M. Narzullaev. "Contact Tracing of Infectious Diseases Using Wi-Fi Signals and Machine Learning Classification". In: *2020 IEEE 2nd International Conference on Artificial Intelligence in Engineering and Technology (IICAIET)*. 2020, pp. 1–

5. DOI: 10.1109/IICAIET49801.2020.9257812.

[Ogc] *OGC IndoorGML-with Corrigendum*. http://docs.opengeospatial.org/is/14-005r4/14-005r4.html. [Online; accessed 13-October-2017].

[PM92] Sankar K Pal and Sushmita Mitra. "Multilayer perceptron, fuzzy sets, and classification". In: *IEEE Transactions on neural networks* 3.5 (1992), pp. 683–697.

[Pow11] David Martin Powers. "Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation". In: (2011).

[Riz19a] Hamada Rizk. "Device-invariant cellular-based indoor localization system using deep learning". In: *The ACM MobiSys 2019 on Rising Stars Forum*. 2019, pp. 19–23.

[Riz19b] Hamada Rizk. "Solocell: Efficient indoor localization based on limited cell network information and minimal fingerprinting". In: *Proceedings of the 27th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. 2019, pp. 604–605.

[Ros01] Udo Rossbach. *Positioning and navigation using the Russian satellite system GLONASS*. Univ. d. Bundeswehr München, Fak. f. Bauingenieur-u. Vermessungswesen, Studiengang Geodäsie und Geoinformation, 2001.

[Sär+15] Simo Särkkä, Ville Tolvanen, Juho Kannala, and Esa Rahtu. "Adaptive Kalman filtering and smoothing for gravitation tracking in mobile systems". In: (2015), pp. 1–7.

[Sat18] Adam Satan. "Bluetooth-based indoor navigation mobile system". In: *2018 19th International Carpathian Control Conference (ICCC)*. IEEE. 2018, pp. 332–337.

[SH05] Barbara Elwood Schlatter and Amy R. Hurd. "Geocaching: 21st-century Hide-and-Seek". In: *Journal of Physical Education, Recreation & Dance* 76.7 (2005), pp. 28–32. DOI: 10.1080/07303084.2005.10609309. eprint: https://doi.org/10.1080/07303084.2005.10609309. URL: https://doi.org/10.1080/07303084.2005.10609309.

[SS+20] Emilio Sansano-Sansano, Fernando J. Aranda, Raúl Montoliu, and Fernando J. Álvarez. "BLE-GSpeed: A New BLE-Based Dataset to Estimate User Gait Speed". In: *Data* 5.4 (2020). ISSN: 2306-5729. DOI: 10.3390/data5040115. URL: https://www.mdpi.com/2306-5729/5/4/115.

[Taj14] Tibor Tajti. "Indoor localization with mobile phone". In: *Proceedings of the 1st International Conference on Future RFID Technologies* (Eger, Hungary). Nov. 2014.

[Tan] *schlegelp/tanglegram*. https://github.com/schlegelp/tanglegram. [Online; accessed 13-August-2019].

[Tot16] Zsolt Toth. "ILONA: indoor localization and navigation system". In: *Journal of Location Based Services* 10.4 (2016), pp. 285–302. DOI: 10.1080/17489725.2017.1283453. eprint: http://dx.doi.org/10.1080/17489725.2017.1283453. URL: http://dx.doi.org/10.1080/17489725.2017.1283453.

[TS+14] Joaquın Torres-Sospedra et al. "UJIIndoorLoc: A new multi-building and multi-floor database for WLAN fingerprint-based indoor localization problems". In: *Proceedings of the fifth conference on indoor positioning and indoor navigation*. 2014.

[TS+15] Joaquin Torres-Sospedra et al. "UJIIndoorLoc-Mag: A new database for magnetic field-based localization problems". In: *Indoor Positioning and Indoor Navigation (IPIN), 2015 International Conference on*. IEEE. 2015, pp. 1–10.

[Wan+20] X H Wang, X Gao, K Zang, and H J Meng. "Ontology Based Semantic Understanding for 3D Indoor Scenes". In: *Journal of Physics: Conference Series* 1607 (2020), p. 012103. DOI: 10.1088/1742-6596/1607/1/012103. URL: https://doi.org/10.1088/1742-6596/1607/1/012103.

[Wei04] Zeev Weissman. "Indoor location". In: *White paper, Tadlys Ltd* (2004).

[WGM00] SS Wang, Marilynn Green, and M Malkawa. "E-911 location standards and location commercial services". In: *Emerging Technologies Symposium: Broadband, Wireless Internet Access, 2000 IEEE*. IEEE. Richardson,TX, USA, 2000, 5–pp.

[WH92] Roy Want and Andy Hopper. "Active badges and personal interactive computing objects". In: *Consumer Electronics, IEEE Transactions on* 38.1 (1992), pp. 10–20.

[WJH97] Andy Ward, Alan Jones, and Andy Hopper. "A new location technique for the active office". In: *Personal Communications, IEEE* 4.5 (1997), pp. 42–47.

[Wu+13] Chenshu Wu, Zheng Yang, Yunhao Liu, and Wei Xi. "WILL: Wireless indoor localization without site survey". In: *Parallel and Distributed Systems, IEEE Transactions on* 24.4 (2013), pp. 839–848.

[Wwr] *IEEE 802.11 RSSI Documentation*. https://msdn.microsoft.com/en-us/library/cc234011.aspx. [Online; accessed 5-September-2016]. 2007.

[Xu+17] He Xu et al. "An RFID indoor positioning algorithm based on Bayesian probability and K-nearest neighbor". In: *Sensors* 17.8 (2017), p. 1806.

[YA04] Moustafa Youssef and Ashok Agrawala. "Handling samples correlation in the horus system". In: *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*. Vol. 2. IEEE. 2004, pp. 1023–1031.

[YA05] Moustafa Youssef and Ashok Agrawala. "The Horus WLAN location determination system". In: *Proceedings of the 3rd international conference on Mobile systems, applications, and services*. ACM. Seattle, WA, USA, 2005, pp. 205–218.

[Yan+21] Juntao Yang et al. "Semantics-guided reconstruction of indoor navigation elements from 3D colorized points". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 173 (2021), pp. 238 – 261. ISSN: 0924-2716. DOI: https://doi.org/10.1016/j.isprsjprs.2021.01.013. URL: http://www.sciencedirect.com/science/article/pii/S0924271621000137.

[YZZ20] Zhengwu Yuan, Xupeng Zha, and Xiaojian Zhang. "Adaptive Multi-Type Fingerprint Indoor Positioning and Localization Method Based on Multi-Task Learning and Weight Coefficients K-Nearest Neighbor". In: *Sensors* 20.18 (2020). ISSN: 1424-8220. DOI: 10.3390/s20185416. URL: https://www.mdpi.com/1424-8220/20/18/5416.

Author's publication

[1] Krisztian Ilku and **Tamas, Judit**. "IndoorGML Modeling: A Case Study". In: *Carpathian Control Conference (ICCC), 2018 19th International*. IEEE. 2018, pp. 633–638.

[2] Krisztian Ilku and **Tamas, Judit**. "Topology-based Classification Error Calculation based on IndoorGML Document". In: *THE 11TH CONFERENCE OF PHD STUDENTS IN COMPUTER SCIENCE*. Institute of Informatics of the University of Szeged. 2018, pp. 101–105.

[3] *Miskolc IIS Hybrid IPS Data Set*. `http://archive.ics.uci.edu/ml/datasets/Miskolc+IIS+Hybrid+IPS`. [Online; Date donated 04-July-2016].

[4] **Tamás Judit**, Tóth Zsolt. "Osztályozáson alapuló pozicionálási módszerek vizsgálata a miskolci informatikai épület hibrid adathalmaza alapján". In: *Müszaki tudomány az Észak-Kelet Magyarországi régióban 2016*. 2016, pp. 664–668.

[5] **Tamas, Judit**. "Hierarchical Clustering based on IndoorGML Document". In: *2019 IEEE 15th International Scientific Conference on Informatics (INFORMATICS 2019)*. IEEE. 2019, pp. 411–416.

[6] Zsolt Tóth, Péter Magnucz, Richárd Németh, and **Tamás, Judit**. "Data model for hybrid indoor positioning systems". In: *Production Systems and Information Engineering* 7.1 (2015), pp. 67–80.

[7] Zsolt Toth and **Judit Tamas**. "Miskolc IIS hybrid IPS: Dataset for hybrid indoor positioning". In: *2016 26th International Conference Radioelektronika (RADIOELEKTRONIKA)*. IEEE. Kosice, Slovakia, 2016, pp. 408–412.

[8] **Judit Tamas** and Zsolt Toth. "Limitation of CRISP accuracy for evaluation of room-level indoor positioning methods".

In: *2018 IEEE International Conference on Future IoT Technologies (Future IoT)*. 2018, pp. 1–6. DOI: 10.1109/FIOT.2018.8325585.

[9] **Tamas, Judit** and Zsolt Toth. "Classification-based symbolic indoor positioning over the Miskolc IIS Data-set". In: *Journal of Location Based Services* 12.1 (2018), pp. 2–18.

[10] **Tamas, Judit** and Zsolt Toth. "Topology-based Classification Error Calculation for Symbolic Indoor Positioning". In: *Carpathian Control Conference (ICCC), 2018 19th International*. IEEE. 2018, pp. 643–648.

[11] **J. Tamas** and Z. Toth. "Topology-based Evaluation for Symbolic Indoor Positioning Algorithms". In: *IEEE Transactions on Industry Applications* (2019), pp. 1–1. ISSN: 0093-9994. DOI: 10.1109/TIA.2019.2928489.

[12] **Tamas, Judit** and Zsolt Toth. "Classification Refinement with Category Hierarchy". In: *The 11th International Conference on Applied Informatics (ICAI 2020)*. Submitted.