DAILY LIFE ORIENTED INDOOR LOCALIZATION BY FUSION OF SMARTPHONE SENSORS AND Wi-Fi

by

Ayse Vildan Nurdag

B.S., Computer Engineering, Yildiz Technical University, 2013

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Computer Engineering
Boğaziçi University
2018

DAILY LIFE ORIENTED INDOOR LOCALIZATION BY FUSION OF SMARTPHONE SENSORS AND Wi-Fi

APPROVED BY:

Prof. Arda YURDAKUL (Thesis Supervisor)	....
Dr. Bert ARNRICH (Thesis Co-supervisor)	....
Prof. Cem ERSOY	....
Assist. Prof. Bahri Atay ÖZGÖVDE	....

DATE OF APPROVAL: 17.09.2018

ACKNOWLEDGEMENTS

I would like to thank my thesis advisors Dr. Bert Arnrich and Prof. Arda Yurdakul of the Computer Engineering Department at Boğaziçi University.

Dr. Arnrich has always motivated me to focus, proceed and progress with my research. Prof. Yurdakul has encouraged me to do superior work. She was communicative and enlightening. Her instructions expanded my perspective.

I thank Prof. Cem Ersoy and Dr. Atay Özgövde for participating in my jury. Their constructive criticism helped me improve my work and enriched my point of view for the future.

I thank my parents Nimet and Veysi Nurdağ for they heartened and supported me throughout the process and made our home available for data collection.

My dear husband, Mehmet Hoşoğlu, who was there all along, touched every sentence during the thesis writing and accompanied and encouraged me in the conference with my presentations. I would like to thank him for endless support.

I also would like to thank Kübra Hoşoğlu and Ebru Barut for their help on editing my thesis.

ABSTRACT

DAILY LIFE ORIENTED INDOOR LOCALIZATION BY FUSION OF SMARTPHONE SENSORS AND Wi-Fi

Smartphones are leading among the fastest-growing technologies. With their numerous features, smartphones are the best assistants to users in their lives on several counts. However, a smartphone still requires an extensive configuration to assist every user efficiently and effectively. In this thesis, we are motivated to develop a system that makes a smartphone self-configure automatically depending on its place. This has been well established for outdoor environments with contributions of GPS (Global Positioning System). However, GPS does not provide accurate data in indoor environments. Hence, in this thesis, we aim to determine the exact place of a smartphone in a room by exploiting on-device sensors and Wi-Fi services. The key point of our study is that it entirely works on the smartphone. In accordance with our motivation, sensors data and Wi-Fi RSSI values were collected from fixed places via Data Collection Application which we developed on an Android smartphone. A fusion fingerprint database was created. Five supervised machine learning algorithms were evaluated on the fingerprint database in terms of classification accuracy and process time. The best performance was obtained from Decision Tree Classifier with 98% accuracy rate on 20% of training samples. Predictive power of used features were studied to specify which sensors are more meaningful for distinguishing indoor places from each other. Depending on model evaluation results, a Data Classification Application was developed on the same Android smartphone to generate a dedicated decision tree for each different room. Tests were carried out in three different rooms to show that more than 80% accuracy was achieved in finding the correct place in each room.

ÖZET

AKILLI TELEFON ALGILAYICILARI VE KABLOSUZ BAĞLANTI BİRLEŞİMİ İLE GÜNLÜK YAŞAM ODAKLI İÇ MEKAN KONUMLANDIRMASI

Akıllı telefonlar en hızlı gelişen teknolojilerin başında geliyor. Sayısız özellikleriyle, akıllı telefonlar insanların pek çok konuda en büyük yardımcısıdır. Öte yandan, insanlara daha etkin ve verimli bir şekilde destek sunabilmeleri için yapılandırılmaları gerekiyor. Bu çalışmayı yapmamız için bizi teşvik eden, bulundukları konumlara göre akıllı telefonların otomatik olarak yapılandığı sistemdir. Bu özellik, Küresel Konumlandırma Sistemi'nin (KKS) katkılarıyla dış mekanlarda mümkündür. Fakat KKS verilerinin iç mekanlarda yeterli doğruluk sağlayamamasından dolayı, bu tezde, cihazın kendi sensörleri ve Wi-Fi kullanılarak akıllı telefonun bir oda içerisindeki yerinin tam olarak belirlenmesi amaçlanmıştır. Çalışmamızın kilit noktası, tamamen akıllı telefon üzerinden çalışılıyor olmasıdır. Motivasyonumuza uygun olarak iç mekanda belirli yerlerden, sensör verileri ile Wi-Fi sinyal verileri, bir Android akıllı telefonda geliştirdiğimiz Veri Toplama Uygulaması kullanılarak toplandı. Bu uygulama ile bir tümleşik veri tabanı yaratıldı. Beş farklı denetimli makine öğrenme algoritması oluşturulan veri tabanına uygulanarak doğruluk ve işlem süresi kriterlerine göre değerlendirildi. Öğrenme setinin %20'sinde %98 doğruluk sağlayan Karar Ağacı Sınıflandırıcısı en başarılı sınıflandırıcı olduğu belirlendi. Kullanılan sensör verilerinin hangisinin konumları birbirinden ayırmakta daha anlamlı olduğunu bulmak için, özelliklerin tahminleme gücü araştırıldı. Model değerlendirme sonuçları baz alınarak, her odaya özgü karar ağacı oluşturmak amacıyla yine aynı Android akıllı telefon üzerinde Veri Sınıflandırma uygulaması geliştirildi. Üç farklı odada gerçekleştirlen testlerde, her bir odadaki doğru noktayı saptama başarısının yüzde 80'den fazla olduğu görüldü.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS iii
ABSTRACT iv
ÖZET v
LIST OF FIGURES viii
LIST OF TABLES xiii
LIST OF SYMBOLS xiv
LIST OF ABBREVIATIONS xv
1. INTRODUCTION 1
1.1. Thesis Outline 5
2. RELATED WORKS 6
2.1. Wi-Fi-Based Localization Systems 7
2.2. Magnetic Field-Based Localization 9
2.3. Ambient Sensors-Based Localization 10
2.4. Fusion Technologies in Indoor Localization 11
2.5. Machine Learning in Localization 12
2.6. Daily-life Oriented Applications Based on Smartphone Sensors 13
3. OVERVIEW 16
4. DATA COLLECTION 19
4.1. Device Specification 19
4.2. Environmental Conditions 20
4.3. Data Collection Application 22
4.4. Data Exploratory Visualization 25
5. MACHINE LEARNING MODELS EVALUATION 51
5.1. Creating Fusion Dataset 51
5.2. Algorithms and Techniques 53
5.2.1. Decision Tree Classifier 54
5.2.2. K-Nearest Neighbor 54
5.2.3. Ada Boost Classifier 55
5.2.4. Gaussian Naïve Bayes 55

5.2.5. Support Vector Machines 56
5.3. Benchmark Model 56
5.4. Our Evaluation Methodology 57
5.4.1. Evaluation Metrics 57
5.4.2. Data Pre-processing 58
5.4.3. Shuffling and Splitting Data 59
5.4.4. Model Evaluation 59
5.5. Refinement of the Best Model 68
5.5.1. Improving Results with Choosing the Best Model 68
5.5.2. Feature Importance 68
6. DATA CLASSIFICATION APPLICATION 71
6.1. Weka and J48 71
6.2. Cross Validation Computation on Smartphone 76
6.3. Effect of Collected Dataset Amount to Accuracy 82
7. CONCLUSIONS 85
REFERENCES 88
APPENDIX A: RADARCHARTS FOR LIVING ROOM AND OFFICE . . . 93

LIST OF FIGURES

Figure 4.1. Floor plan of the home used in experiments 21

Figure 4.2. Data Collection Application interface 22

Figure 4.3. Chosen places in lounge 23

Figure 4.4. Chosen places in living room 24

Figure 4.5. Proximity sensor data from lounge 29

Figure 4.6. Lux sensor data from lounge 29

Figure 4.7. Magnetic field data on X-axis from lounge 30

Figure 4.8. Magnetic field data on Y-axis from lounge 30

Figure 4.9. Magnetic field data on Z-axis from lounge 31

Figure 4.10. RSSI values of 1.Wi-Fi AP from lounge 31

Figure 4.11. RSSI values of 2.Wi-Fi AP from lounge 32

Figure 4.12. RSSI values of 3.Wi-Fi AP from lounge 32

Figure 4.13. RSSI values of 4.Wi-Fi AP from lounge 33

Figure 4.14. Proximity sensor data from living room 34

Figure 4.15. Lux sensor data from living room 34

Figure 4.16. Magnetic field data on X-axis from living room 35

Figure 4.17. Magnetic field data on Y-axis from living room 35

Figure 4.18. Magnetic field data on Z-axis from living room 36

Figure 4.19. RSSI values of 1.Wi-Fi AP from living room 36

Figure 4.20. RSSI values of 2.Wi-Fi AP from living room 37

Figure 4.21. RSSI values of 3.Wi-Fi AP from living room 37

Figure 4.22. RSSI values of 4.Wi-Fi AP from living room 38

Figure 4.23. Proximity sensor data from office 39

Figure 4.24. Lux sensor data from office 39

Figure 4.25. Magnetic field data on X-axis from office 40

Figure 4.26. Magnetic field data on Y-axis from office 40

Figure 4.27. Magnetic field data on Z-axis from office 41

Figure 4.28. RSSI values of 1.Wi-Fi AP from office 42

Figure 4.29. RSSI values of 2.Wi-Fi AP from office 42

Figure 4.30. Radarchart of Place1 in lounge at various times 44

Figure 4.31. Radarchart of Place2 in lounge at various times 45

Figure 4.32. Radarchart of Place3 in lounge at various times 46

Figure 4.33. Radarchart of Place4 in lounge at various times 47

Figure 4.34. Radarchart of Place5 in lounge at various times 48

Figure 5.1. Model Evaluation with lounge data 61

Figure 5.2. Model Evaluation with living room data 62

Figure 5.3. Model Evaluation with office data 63

Figure 5.4. Learning curves for lounge data 65

Figure 5.5. Learning curves for living room data 66

Figure 5.6. Learning curves for office data 67

Figure 5.7. Normalized weights for features of lounge data 69

Figure 5.8. Normalized weights for features of living room data 69

Figure 5.9. Normalized weights for features of office data 70

Figure 6.1. User interface of classification application 72

Figure 6.2. CSV files in local memory of phone which were created by Data Collection Application 72

Figure 6.3. Sample screen-shots of a trained model and some test results for chosen train and test files 73

Figure 6.4. Trained model for lounge 74

Figure 6.5. Trained model for living room 75

Figure 6.6. Trained model for office 76

Figure 6.7. 10-fold Cross Validation for lounge 77

Figure 6.8. 10-fold Cross Validation for living room 78

Figure 6.9. 10-fold Cross Validation for office 78

Figure 6.10. Classification results for the different size of datasets 84

Figure A.1. Radarchart of Place1 in living room at various times 94

Figure A.2. Radarchart of Place2 in living room at various times 95

Figure A.3. Radarchart of Place3 in living room at various times 96

Figure A.4. Radarchart of Place4 in living room at various times 97

Figure A.5. Radarchart of Place5 in living room at various times 98

Figure A.6. Radarchart of Place1 in office at various times 99

Figure A.7. Radarchart of Place2 in office at various times 100

Figure A.8. Radarchart of Place3 in office at various times 101

Figure A.9. Radarchart of Place4 in office at various times 102

Figure A.10. Radarchart of Place5 in office at various times 103

LIST OF TABLES

Table 2.1. Comparison of existing wireless technology for indoor localization . 7

Table 4.1. Technical features of Samsung Galaxy Note II 20

Table 4.2. Sensor Models of Samsung Galaxy Note II 20

Table 4.3. Data collection times for the lounge 26

Table 4.4. Data collection times for the living room 27

Table 4.5. Data collection times for the office 28

Table 6.1. Results of iterative cross-validation for lounge 80

Table 6.2. Results of iterative cross-validation for living room 81

Table 6.3. Results of iterative cross-validation for office 82

Table 6.4. Times of data collection 83

LIST OF SYMBOLS

dBm	Decibel-Milliwatt
rad/s	Radian per second
T	Tesla
V	Volt
μ	Micro

LIST OF ABBREVIATIONS

AP	Access Point
API	Application Program Interface
BLE	Bluetooth Low Energy
BSSID	Basic Service Set Identifier
CMOS	Complementary Metal Oxide Semiconductor
CS	Color Sensor
FM	Frequency Modulation
GPS	Global Positioning System
GSM	Global System for Mobile
I2C	Inter-Integrated Circuit Protocol
IR	Infrared
LSB	Least Significant Bit
NN	Nearest Neighbour
PDR	Pedestrian Dead Reckoning
PS	Proximity Sensor
RGBW	Red,Green,Blue,White
RSS	Received Signal Strength
RSSI	Received Signal Strength Indicator
SMO	Sequential Minimal Optimization
SSID	Service Set Identifier
OS	Operating System
Wi-Fi	Wireless Fidelity

1. INTRODUCTION

This has happened to everyone at least once: You fall asleep on the couch and there is no one to wake you up and tell you to go to your bed. Imagine your phone, which is right next to you, thinking about you in a situation like this: *he is sleeping on the couch right now and it is the middle of the night. I should set an alarm for him and make sure he does not fall asleep here.* Imagine your phone taking itself into do not disturb mode so that you do not get distracted when you sit down to study. It could turn on night mode on when on your night stand. It could be in studying mode when in your office and in silent mode while in the meeting room. It could remind you to adjust your posture when you are at the study desk for too long. And even imagine it playing soothing music for you when put on dining table. All of these will be possible if we can achieve reliable indoor localization.

Indoor localization is a system that locates and navigates people in buildings. The existence of the huge and complicated buildings in modern life has created a need for distinguishing indoor locations and navigating people who spend most of their time inside. The acquisition of physical location is the fundamental basis for Location-Based Systems (LBS). Global Positioning System (GPS), the prevalent technology for outdoor localization, does not work well in indoor environments due to the blocking of signals by walls, floors and ceilings. Therefore, indoor localization has been an active research field. To acquire high-accuracy localization in indoor environments, many techniques have been developed.

With the rapid development in sensor technology, the variety of sensors on the phone has increased. Modern smartphones are equipped with many different sensors such as accelerometer, gyroscope, lux, proximity, magnetic field and temperature sensors. Hence, they can already hear, see and sense the environment. They have high computational performance and programmable capabilities. Most wireless technologies such as Bluetooth and Wireless Fidelity (Wi-Fi) are also accessible by smartphones. With wide usage, the smartphone has become the first truly pervasive computer [1].

Thus, the majority of the indoor positioning studies have been based on the mobile platform.

Inside buildings, Wi-Fi has been seen as a good alternative to GPS. The RSSI is used to calculate the distance measurements required for Wi-Fi-based location estimation. Received Signal Strength Indicator (RSSI) is a measure to indicate relative quality of a signal which is received from a client device. Localization techniques based on the RSSI can be divided into two main categories: The signal propagation model and fingerprinting [2]. For the first method, multiple Wi-Fi transmitters must be exactly located in an indoor location. These transmitters are then used as reference points. Knowing exact locations of transmitters makes possible to calculate relative positions of the receivers. Distance can be estimated by using received signal information. If we briefly summarize the method, the strength of the signal emitted from the receiver is known. Also, the strength of the received signal is known. The difference between the first strength and last strength is described as signal attenuation due to the path loss. Path loss can be calculated by measuring signal attenuation between the receiver and transmitter when they are located at a known distance from each other. However, the signal propagation model is not sufficiently applicable since there are many components affecting signals such as walls, doors and movable furniture (even people). The signal strength reduces when one of said objects is between the transmitter and the receiver due to its physical resistance. Hence, the path loss value cannot reflect the real attenuation. In this manner the location of the receiver can be erroneously estimated as far away from its real location. The next method widely used for Wi-Fi technology is fingerprinting. This is a more common method than the previous one in recent studies [3]. The main idea of fingerprinting is recording RSSI values from Wi-Fi Access Points (APs) located around for the determined locations. We will explain the fingerprinting approach in detail in the next sections. Briefly, in this method the receiver is routed in different locations. RSSI from Wi-Fi APs around must be systematically recorded to a database. The information of the determined location and which transmitter sends the signals are also recorded to the database. The database is also referred to as fingerprint map. Thus, fingerprinting requires data collection. By using the Wi-Fi fingerprint database, estimation can be done for any location with a common similarity algorithm, such as Euclidian distance. On the other hand, signals can be affected by physical obstacles such as people inside, and RSSI values can be changed at different times. This may causes differences between current RSSI value and recorded RSSI value on fingerprint database. Even so, it is still currently the most used technique in recent studies given its low cost and high accuracy [4].

Besides, the features of smartphones are not limited to receiving Wi-Fi signals. Beyond signal-based technologies, there are many sensor technologies on smartphones. Basically, sensors can be considered under two groups: motion sensors and ambient sensors. The gyroscope and accelerometer are well known sensors as motion sensors. The gyroscope measures the rate of rotation of a device's x, y and z axes in rad/s. It is an electronic circuit that provides information about the orientation of the smartphone in three-dimensional space. Most applications use the data to perform functions on the smartphone, such as rotating the screen. The accelerometer measures acceleration in each of the three dimensions, as its name implies. Both accelerometer and gyroscope are hardware-based. Some other software-based sensors like step detector, step counter, sleep tracker and motion type detector (walking, running, etc.) can derive their data both from the accelerometer and from the gyroscope. They are also used in indoor navigation studies to track users' by their movements. This is referred to as pedestrian dead-reckoning [5]. The dead-reckoning technique does not directly work as a localization system. It provides only relative position of the pedestrian. The method is that a pedestrian is tracked from a determined start point by using step count, step length and heading angle. The accelerometer makes it possible to measure step count information by providing acceleration data. The gyroscope is used to measure angular heading.

Moreover, another fingerprinting approach has been proposed by using a magnetometer in smartphones. A magnetometer is an electronic compass technology that measures magnetic fields of environment. However magnetic fields may fluctuate at certain places, which may cause some big localization errors. Nevertheless, magnetic fingerprinting has become a widely used technique in recent studies [7]. Ambient sensors, such as barometer, proximity sensor, ambient light sensor, thermometer and humidity sensor, are frequently used sensors. They can be called environmental sensors, since they are used to sense physical specification of an environment. Due to the growing sensor technology and their high-sensitivity measurements, ambient sensors have been used to distinguish different environments. Some have started to employ these technologies in indoor localization. Barometers have been used for [8] detecting floor level and elevators in interior places by monitoring variations of atmospheric pressure. Ambient light sensors are also used as ambient sensors for indoor localization but suffer from time variation during a day and from seasons in the year, due to the daylight variations and usage of florescence in indoor places [9].

In addition, smartphone cameras and microphones also have been proposed as supported technologies. Even though energy consumption of cameras is significantly high [10] and they reduce the battery life, some recent works use cameras and image processing to detect interior places in indoor localization systems. In a similar manner, microphones have been used to detect ambient sounds [9]. However, instead of a single technology-based indoor localization approach, the majority of studies combine multiple technologies and data sources to increase efficiency of the systems.

The physical specifications of an interior place are another considered issue in indoor localization studies. Indoor maps have been published by Google for airports, malls, stadiums and other terminals of public transportation [11]. On the otherhand, there are many studies that have focused on more individual places such as home and offices [12].

Everyone has a daily routine, and people spend their time in certain places generally during a day. In this study, we aimed to use indoor localization to support mobile phone applications as a location-based service. We do not attempt to navigate the user and do not track the phone. We focus on improving the ability of the most ubiquitous technology of recent times, the smartphone. Our motivation has become to make smartphone more self-configured by considering users location in indoor environments. Our main target is to make it possible to determine whether a smartphone is on study desk or on the nightstand, or whether in front of the window or on the top of dinner table in a same room. Thereby, our daily settings can be automated by the smartphone itself. For example, if a user puts his phone on the bedside table, night mode can be turned on. If the user puts his phone on a study desk, do not disturb mode can be activated automatically. On the kitchen table, some dinner music can be played.

Generally, applications have focused on distinguishing different rooms, corridors and such places that are obviously separated from each other. On the other hand, they are tracking the phone. In this thesis, we are not tracking a smartphone or not navigating a user indoor. We construct a system based on where smartphones are placed during a day. Thus, we have done our experiments in certain locations in home and office environments. We have collected magnetic field, ambient light and proximity sensor and Wi-Fi data from three different indoor locations. All sensor data and Wi-Fi data have been used to create a fingerprint database. The produced database becomes an input for evaluation of supervised machine learning algorithms. This is giving an advantage to analyze the ambient specification of the environment. A successful classification based on ambient features of indoors to distinguish different corners from each other in a room can be used in lots of smartphone applications supported technology. Mobile devices can take automatic actions according to its current position. It can be also increase the precision of tracking systems in indoor environments.

1.1. Thesis Outline

The rest of the thesis is organized as follows: In Chapter 2, we present the related work in indoor localization, Wi-Fi fingerprinting, magnetic fingerprinting, ambient sensing and data classification methods for sensor data. In Chapter 3, we provide an overview for a better understanding of the thesis. In Chapter 4, we present a data collection approach and methodologies. In Chapter 5, we provide an extensive comparison of machine learning models on our fusion dataset. In Chapter 6, we introduce our data classification application and its algorithms, our tests and obtained results. In Chapter 7, we conclude the thesis and discuss the future work.

2. RELATED WORKS

Indoor localization has been an attractive research field due to its wide application domain. The infrastructure requirement may be the first basis for classification among current indoor localization systems. A special infrastructure must be deployed to create indoor localization systems with some technologies, such as Bluetooth Low Energy (BLE).

Verhaevert *et al.* [13] propose a room level localization system motivated by elderly and needy people who need to be followed or for protecting valuable objects in a house against the theft. They deployed BLE nodes on the objects and people wanted to be located. In addition, a BLE sensor is placed in the approximate center of each room. There is no need to fingerprint or measure RSSI values, and the statistical triangulation methods are not used. The central BLE sensor sends advertising packets, and the RSSI values are calculated for each receiver nodes by measuring the difference between the transmitted and received power. They are calculating an average of the measured RSSI values over a total of five values. By this manner they reached the most accurate results with a correct localization more than 90%. Even if battery replacement is not required on a regular basis, the system requires extra infrastructure installation in the home environment. The motivation of the study is similar to ours; we have used the home environment also. Nevertheless, we posit that for the indoor locations such as rooms in home, the infrastructure installation is not practical, while there are various alternative and ubiquitous technologies.

As to infrastructure-less technologies, we can investigate recent studies under the following categories: Wi-Fi-based, inertial sensor-based, magnetic field-based, ambient sensors-based and fusion technologies-based systems.

2.1. Wi-Fi-Based Localization Systems

Before looking through Wi-Fi-based systems, it is helpful to survey wireless-based localization systems briefly. Liu classifies wireless-based technologies according to their coverage area distances. It deals with FM and Global System for Mobile (GSM) under the long distance, Wi-Fi and ZigBee under the medium distance, and Bluetooth and Radio-frequency identification (RFID) under short distance wireless technologies. Doiphode *et al.* [4] present a classification for wireless technologies in indoor localization studies similar to that of Liu *et al.* [3]. Table 2.1 combines their comparison tables [3,4].

Table 2.1: Comparison of existing wireless technology for indoor localization

Technology	**Range**	**Accuracy**	**Dedicated Infrastructure**	**Disadvantages**
GSM	100 m $\sim$ 10 km	50-500m	No	Highly patented
Wi-Fi	35 m (indoor)	1-5m	No(for most places)	High variance signal
Bluetooth	10 m	Connectivity Range	Yes	Cover range is limited
RFID	1m	Connectivity Range	Yes	Cover range is limited
ZigBee	30 $\sim$ 60 m	Connectivity Range	Yes	Need dedicated infrastructure

However, the mathematical methods to analyze wireless signals are definite and common for all of those technologies. They can be split into proximity, triangulation, and fingerprint. Proximity is the simplest method. Generally GSM-based localization systems use this method. If a user node can connect to a base station, the location of the user can be estimated approximately. Naturally, there is high variance in this method. Therefore it has not been featured in recent localization studies.

Triangulation is the geometric calculation of a user based on at least three known stations. There are two different types of triangulation. The first one is angle-based (AOA). In this method, the directional antenna technology is used in both user and anchor nodes. The angle of the received signal is retrieved by nodes. A later version is time-based triangulation (TOA or TDOA). This is based on distance calculated from travel time of the signals. Since the speed of signal is known, the distance can be calculated by using time information. However, triangulation methods can suffer from wall penetration and resistance to obstacles in indoor locations.

On the other hand, fingerprint technology is a prominent term in recent localization literature. This is commonly used with signal-based technologies. It can be explained as the characteristic or feature of signals briefly. Received Signal Strength (RSS) leaves a fingerprint. If we assumed that there are specific wireless sources such as Wi-Fi access points in an indoor environment, for some specific locations, the features of RSS will be different from each other. Thus, it can be handled as fingerprint data for those locations. If the fingerprint information of each location can be obtained, the current location can be estimated. However, this method requires an extensive training phase.

Basically, there are two different fingerprinting methods: radio-map-based and map-free fingerprinting. The radio-map-based fingerprinting has two phases: offline phase and online phase. A site survey in an environment is required in the offline phase. The location information, which may be strict coordinates or labels, and RSS values from base stations/access points are being collected during site survey. Hence, a map is obtained at the end of offline phase. Then in the online phase, current RSS information is collected by the user node to apply localization technique in order to estimate location. Liu *et al.* [14] describe the main challenge in fingerprinting as the RSS could be influenced from diffraction, reflection and scattering in the propagation.

In the map-free fingerprinting method, RSS data is not collected from each location in the indoor environment. Some locations are chosen as landmarks. The landmark represents a known place that can be used to calibrate indoor navigation. It is generally used with the pedestrian dead-reckoning method, which will be investigated later in this section.

2.2. Magnetic Field-Based Localization

Fingerprinting method is not only used with Wi-Fi but also used with magnetic field technology as well. Chunk *et al.* [7] propose an indoor localization system based only on geomagnetism. They create magnetic field fingerprints map by using an electronic compass. As usual in the fingerprinting method, there are two different phases (training and estimating). Both are realized with a single device they developed. They have worked in a lab building that houses machine shops, machine rooms for servers and desktops. The structure of the building includes steel and concrete as usual. According to the result of their investigations, magnetic field features of indoor locations are characteristic in most cases. 70% of the predicted data had errors of less than 2 meters. However, while magnetic field data is being collected, the orientation of the electronic compass must be taken in the account. Thus, they used their fingerprint map to predict orientation as well. The advantage of the magnetic fingerprint is that it does not require any fixed local references, like the Wi-Fi AP. The disadvantage of this work in itself is that they must be working with their own production because it requires a cost. Accordingly, their systems are working in server-client architecture. That is, the electronic compass reads the data and transfers them to the computer.

The other magnetic-field-base localization system, IndoorAtlas [15] takes advantage of the modern smartphone. It is a cloud-based location service. Under the IndoorAtlas, there are three different applications; IndoorAtlas Floor PlansTM (web application to build floor plans) IndoorAtlas Map CreatorTM (mobile application to collect magnetic field data) and IndoorAtlas API (to use the location service). The API is used for sending processed sensor data to the location service. The location service estimates the current location and returns the estimated locations to the API's event listener. They claimed that they can estimate the location inside a building within 0.1-2.0 meters.

The magnetic field may vary within a single room due to some metal objects in the room, the construction features, or the electronic items (television, desktop computer, etc.) that are static objects in the room. For this reason, in the present study, we also want to observe whether the magnetic field is characteristic in a fixed position in the determined locations in a single room.

2.3. Ambient Sensors-Based Localization

Mazilu *et al.* [12], unlike prior work, use only ambient sensors to assess their effectiveness for indoor localization. Since they are not tracking users or objects, and they only aim to label users location directly, their study is similar to ours. They claimed that ambient properties of different rooms differ due to the environmental variations. For example, a room with south frontage of a home may receive more day light than rooms with north frontage. In addition, artificial light may differ from between rooms owing to light source. The heater size may affect the room temperatures, while humidity level may be affected from user activities such as cooking and taking showers. They used light, humidity and temperature sensors together. As a result of their works, combinations of ambient sensors allow to distinguish rooms in a home with an accuracy between 0.72 and 0.81 metres, obtained from experiments using 132 hours of data collected from 3 residences. As to their methodology, it can be said that a fingerprinting approach is used for ambient sensors. They have collected data from different rooms in the training phase then applied some machine learning approach such as C4.5 to make prediction in the test phase. According to their inference at the end of their experiments, artificial light is the most informative variable, but it is not available during the day in a home. In addition, while temperature and humidity tend to have constant values during the day for each location, light information is directly related to the time of the day (i.e., natural or artificial light). Two of the prominent advantages of using ambient sensors in indoor localization is that it does not require additional infrastructure deployment and has low power consumption. On the other hand, ambient-sensors-based localization has some limitations: for example, ambient data is affected by events such as opening windows or starting the air-conditioning unit (for temperature and humidity sensors).

Based on the work of the Mazilu *et al.* [12], we also aimed to measure how informative the ambient features of different locations are in only one room. Depending on the physical laws, temperature and humidity can radiate quickly in a room, and measured values do not vary within the room, whereas the light value may vary at different locations of a room. For example, a table in front of a window may receive much larger amount of daylight, while an interior seat may have a lower light value. Hence, we used ambient light sensor data in our study.

2.4. Fusion Technologies in Indoor Localization

Beyond the single technology-based systems, the fusion technologies have started to be studied by researchers in indoor localization studies. Tejada *et al.* [9] propose a system that uses smartphone magnetometer, microphone and ambient light sensor in a fusion approach. They took sessional changes in light patterns into consideration by doing experiments in an office environment during summer and winter. To make clear the fusion concept, it is useful to explain that the information sources are not used sequentially or separately in the localization system, but rather all data sources are used in a same fingerprint database. Still, in a fusion fingerprint database the components may have different weight from each other. As the results of the study of Tejada *et al.* [9] since magnetic fields have smaller variations than other sources, such as indoor light intensity or environmental audio, the magnetic field signal source has more weight in the estimation model of the user location.

Chen *et al.* [16] propose a sensor fusion framework for combining Wi-Fi, Pedestrian Dead Reckoning (PDR) and landmarks. They have developed an Android application for real time localization and navigation. PDR, discussed above, is another widely-used localization technique. It determines the current position based on the previous position of user, estimated step length and walking direction of the pedestrian.

Localization accuracy of the PDR approach can be affected by determined initial point, step detection, step length and walking direction. In particular, initial point is very important since PDR can extract relative location according to the initial point. Hence, this information is provided by the determined initial position (referred to as a landmark). A Wi-Fi positioning system is used to detect landmarks in order to make a start point for the PDR algorithm and to maintain calibration during navigation. Moreover, magnetometer sensor data is combined in the system to detect walking direction, and ambient pressure sensor data is used for floor changes.

Wi-Fi has not only been combined with PDR, it has been also started to be combined with various technologies. In recent studies, WAIPO, proposed by Gu *et al.* [17], uses Bluetooth and Wi-Fi interfaces, camera, magnetometer, gyroscope and accelerometer sensors on the smartphone for localization. In [17] RSS-based localization is improved by spatio-temporal co-occurrence, user location preferences and the magnetic calibration. This is a smartphone-based localization system in a server–client architecture. They build the photo fingerprint by capturing the photos of each room. Photos are taken by smartphone automatically. Thus, the quality of photos is poor and cannot be controlled by user. Besides, this causes obvious privacy issues. Using cameras in localization systems increases computational effort while decreasing power usage significantly compared to other ambient-sensor-based technologies.

2.5. Machine Learning in Localization

Machine learning is an inevitable part of fingerprinting method. Therefore, one of the most important factors in accuracy of localization is deriving the appropriate machine learning algorithm. Bozkurt *et al.* [18] present a competitive study about machine learning algorithms for indoor positioning. Evaluation of algorithms according to performance of the classification for indoor positioning was the objective of their study. For this purpose, Nearest Neighbour (NN), Sequential Minimal Optimization (SMO), J48, Nave Bayes and BayesNet algorithms were comparatively tested. Experiments were performed by using WEKA library. They used UJIIndoorLoc database, which can be downloaded from UCI Machine Learning Repository. According to their test results, by using the whole dataset, J48 gives the best accuracy (99.89%). Compared with the above systems, our approach, instead of positioning between rooms, focuses on positioning between different locations within the same room. It uses Wi-Fi data, magnetometer, ambient light and proximity sensor data by combining all of them in a single fingerprint database. Thus, it does not require any infrastructure deployment to the environment. Nowadays, Wi- Fi access points are everywhere. Our approach works based on any Wi-Fi APs, not specific ones. Moreover, other used ambient sensors exist on almost every modern smartphone. The proposed system works only on Android Platform and there is no need to send data to any other platform. Users can collect data by smartphone, can keep them in the smartphone and can classify them in the smartphone. This brings the advantage of being able to see data and classification results in real time with mobile applications. Users can conduct experiments more easily in this manner. It also keeps energy consumption at a balanced level. In indoor environments, such as homes, university buildings and offices, Wi-Fi services are generally used in active mode.

2.6. Daily-life Oriented Applications Based on Smartphone Sensors

Ambient light and proximity sensors are generally integrated with each other and they exist in almost all smartphone models. They can be used in a variety of different applications. For example, screen of a smartphone can be turned off by using the proximity sensor when user puts the phone on his/her ear. The light sensor is used for adjusting display light of smartphones [26]. Infra-red-based proximity sensors can detect objects that are up to 200 cm meters away and they are used in most up to date smartphone models.

In recent studies, these sensors are used to get information from an environment user has been. Kim *et al.* [29] propose a system that collects data from ambient light sensors and extracts information about environment luminance from different light sources. Hariadi *et al.* [27] provide an application that gives information to the user about convenience of the lighting condition in terms of room types. It is expected that user can save energy on lighting by using ambient light sensor on the smartphone.

Fahim *et al.* [28] propose an activity recognition system that uses ambient light and proximity sensors for classifying user activities. They use proximity sensor to find the place of the smartphone that might be at the hand or in the pocket of users.

A person spending most of the time at indoor locations usually places the smartphone at a fixed place nearby, such as on the study desk he is sitting in, on the table she is dining, instead of carrying the phone in the pockets. At this kind of places, there might be static objects around the smartphone. For example, there might be a bookshelf on a table where user put smartphone on. In addition, there might be a stable night lamp on the nightstand and user can puts his/her phone while he/she is sleeping. Hence, proximity sensor of the smartphone can detect stationary objects at these locations and distance measure from these objects provide information about the location of the smartphone.

The ambient luminance value may also be different in various locations in a room. For example, a table in front of a window gets more illumination under the daylight than a table placed a faraway location. Different lighting conditions in different places make the ambient light sensor produce different values. Thus, it provides information about the location of the smartphone. Likewise, magnetic field features of indoors comprise of refraction of the geo-magnetic field by steel structures, and fixed large objects. Kim *et al.* [29] has created a magnetic field map for an interior of 250 square meters by using multiple magnetic field sensor in a mobile robot, and navigated robot by using this map. Their proposed navigation system based on magnetic field maps obtains the mean distance error which is less than 0.1 m.

As mentioned Section 2.1, it is attempted to enhance the performance of Wi-Fi based positioning, which is commonly used technology in indoor localization by using supporting technologies such as magnetic field based localization. Contribution of the ambiance characteristics of indoors to seperate different locations has been investigated in this thesis. Magnetic field sensor, ambient light and proximity sensors which can be found in most of smartphones has been chosen for investigation. Our main goal is to extract the ambient character of certain positions rather than providing a navigation system.

In recent studies, various machine learning approaches are applied on Wi-Fi fingerprint databases for indoor localization [32, 33, 34]. Cheng *et al.* [34] propose an enhance indoor localization scheme with machine-learning to enhance accuracy in noisy environments by integrating AP selection and the proposed signal strength reconstruction. Zhang *et al.* [30] apply deep learning algorithms on a fusion dataset which combines Wi-Fi and magnetic field data. They improve effectiveness of smartphone indoor localization compared to existing approaches based on Wi-Fi only.

3. OVERVIEW

In this thesis, we aim to develop a methodology to identify user-selected places of a smartphone in an indoor environment. This is the necessary step to develop various place-oriented user-assistance applications on smartphones as exemplified in the Introduction section.

Perspective of a typical smartphone user simply puts our methodology. Assume that the user wants to identify different places in an indoor location, say in a lounge. Assume also that she wants the smartphone to switch to silent mode automatically on study desk or set an alarm on sofa. This example shows that she needs to select these places firstly. For this specific example, these places are sofa and study desk. Obviously, selected places need to be as dissimilar from each other as possible so that smartphone sensors and Wi-Fi can differentiate them. Our user wants her smartphone to be aware of these places whenever she puts it. Hence, the the phone needs to learn these places by itself. As a result, a supervised learning method has to exist in the phone. A supervised learning method requires a substantial amount of labelled data for correct identification of a place. Hence, our user needs to collect data from all selected places at different times in various days. While collecting data, she has to label the data with the names of the selected places. So, she needs our application for labelled data collection. After sufficient amount of data is collected, the machine learning algorithm that we implemented on the smartphone can be executed to identify the places. If the user is not happy with the identification results, then she can continue data collection until she is happy with the results. She can use our application for different indoor locations, such as her lounge, sitting room and her office. Location and place change can be easily sensed by the smartphone after executing the necessary software that we developed throughout this thesis.

The following chapters of this thesis further details this methodology as described below:

(i) Since data collection is done with smartphones, we have developed an application to collect sensor and Wi-Fi data of a smartphone. When the developed application is initiated by user, the application perceives lux, proximity, magnetic field sensor data and Wi-Fi signals around. Received values are recorded to the local memory of phone with the timestamp. Details of this process will be given in the next chapter, “Data Collection”. To demonstrate that our application can differentiate different locations and places in those locations, data collection environments are chosen as two rooms in a house and an office. In the same application, we developed a user interface to allow the user to collect data for five different places for each chosen environment. During the data collection, smartphone is located on close points in the same place. As an example, during the data collection from the table in the middle of lounge, the smartphone can take place at any corner of the table. On the other hand, the smartphone is always put on the table with the same orientation at every single data collection instant so as to fully exploit magnetic field sensor data from its three different axes. At the last part of Chapter 4, raw forms of sensor data are visualised to observe ambient features of locations where data was collected from. Visualisation part is carried out on a regular personal computer, not on the smartphone. This part is essential to demonstrate that smartphone sensors behave discriminatively in different environments and places.

(ii) We want the smartphone to learn the places by itself. Therefore, we need a machine learning algorithm executing on the phone whenever the user wants to improve the identification of the places. Training time should not take too long to make the user get bored and quit the training process. To start with, we developed a shifting-window based mechanism for data cleansing. This part is implemented on the phone because it is also used during place identification. We determined the best machine learning algorithm offline. To do this, we used five well-known machine learning algorithms from the literature, exercised them on our data set and compared their performance in terms of execution time and accuracy. The complete model selection process is explained in “Machine Learning Models Evaluation” chapter.

(iii) Though we determined the learning algorithm offline, the phone has to generate the model for each room by itself, because each indoor location has its own constraints due to the placement of the furniture, construction characteristics and the usual behaviour of its inhabitants. Hence, the best learning model for one room is usually different from the other environments. To achieve this, we deployed Weka library on Android to implement "Data Classification Application" which is explained in Chapter 6. By using this application, the user can now both train the model and execute it to identify selected places with her smartphone. We demonstrated that a unique model has been generated for every room. Classification accuracy and time are directly correlated with the training set size. So the user might also wonder how many times she has to visit her selected places so that she can get a reliable place identification. To answer this question, we carried out a number experiments to present them in the same chapter.

Hence, in this thesis we achieved an end-to-end user experience on the smart phone for collecting and cleansing data, using this data for training a model, exploiting the same model for place identification, improving the model accuracy by collecting more data whenever desired. Within our knowledge, this thesis is the first study of this kind in the literature.

4. DATA COLLECTION

In accordance with our motivation, we have developed a data collection system that allows us to gather data from the daily life locations of users. Since we aim to distinguish pre-determined minimal places in a room or in an office from each other, and focus on making this study a part of a daily routine of a user, a simple, handy Android application has been developed to collect ambient sensors data and Wi-Fi signals, and record them in the memory of smartphone.

Nowadays, most smartphone producers are in tight competition, and smartphones are well-equipped with high-tech sensor technology. Recent technology has reached the level of taking selfie photos by squeezing phone [19]. There are many brands, models and operating systems actively used in market.

Since the Android OS has the highest share in the market and most brands such as Samsung, Sony, HTC, LG etc. use Android OS, many developers prefer to use Android [20]. Especially, when an application works with sensors, usable device range becomes important. While some smartphone models may have temperature, heart rate, and even squeez sensors, some others may have only light sensors and accelerometers. Producers organize popular technologies in smartphones in order to provide a balanced price range. Hence, Android is the best OS to obtain a product range for developers. For these reasons, in this thesis, Android OS is used to develop applications. We have worked with the Samsung Galaxy Note II. Note II was released in November 2012 [21].

4.1. Device Specification

The device specifications, which are given in Table 4.1 and Table 4.2, support our aims in terms of sensor technologies and computational power. Samsung Galaxy Note II (GT-N7100) is powered by a 1.6GHz quad-core and it comes with 2GB of Random Access Memory (RAM).

Table 4.1: Technical features of Samsung Galaxy Note II

Feature	**Value**
Processor	1.6GHz Quad-Core
RAM	2 GB
Internal Storage	16 GB
Expandable Storage(up to)	64 GB
Operating System	Android 4.1

Table 4.2: Sensor Models of Samsung Galaxy Note II

Sensor Name	**Model**
Compass/ Magnetometer	AK8963C Magnetic Field Sensor
Proximity sensor	CM36651 Color&Proximity Sensor
Ambient light sensor	CM36651 Color&Proximity Sensor

4.2. Environmental Conditions

Unlike many other indoor localization studies, the data has been collected only from determined locations. We have determined five locations in each selected room. In a home environment, the experiments have been done in 2 different rooms (1 and 2 in Figure 4.1). The floor plan of the home and the room sizes are given below in Figure 4.1, which is a representative floor provided by the builder.

Figure 4.1: Floor plan of the home used in experiments

- 1 Lounge : 25.00 m^2
- 2 Living Room : 15.15 m^2
- 3 Kitchen : 13.70 m2

In addition, we have done experiments in an open office environment where there are divided rooms at different floors. We have determined five tables at locations that differ within but not between floors to collect data. Due to the fact that there is only one Wi-Fi AP around the office building, we aimed to measure the effect of the numbers of Wi-Fi AP's in indoor localization. For each determined location, the smartphone was approximately placed to the same place with the same position and direction during the data collection process. To make our tests under stable environmental conditions, we have paid attention to keep objects, especially electronics and metal items, in the places they have always been. However, the light conditions change according to the daily routine. During the day, normal daylight has been used as expected in every home, while after the sunset artificial light has been put into use.

4.3. Data Collection Application

In this section, we explain our raw sensor data collecting process.

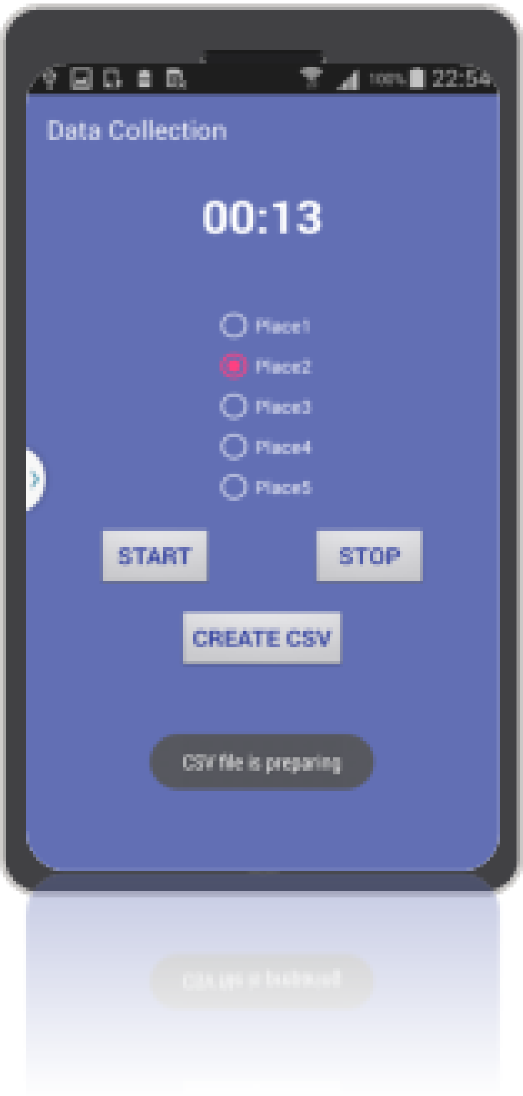

Figure 4.2: Data Collection Application interface

The application can be used only with start and stop buttons to collect data. They are shown in Figure 4.2. While user activate the record by start button, the counter on the screen shows seconds. The user places the smartphone in a fixed position of his choice. In Figure 4.3 and Figure 4.4, smartphone placements and chosen locations are shown for the lounge and living room.

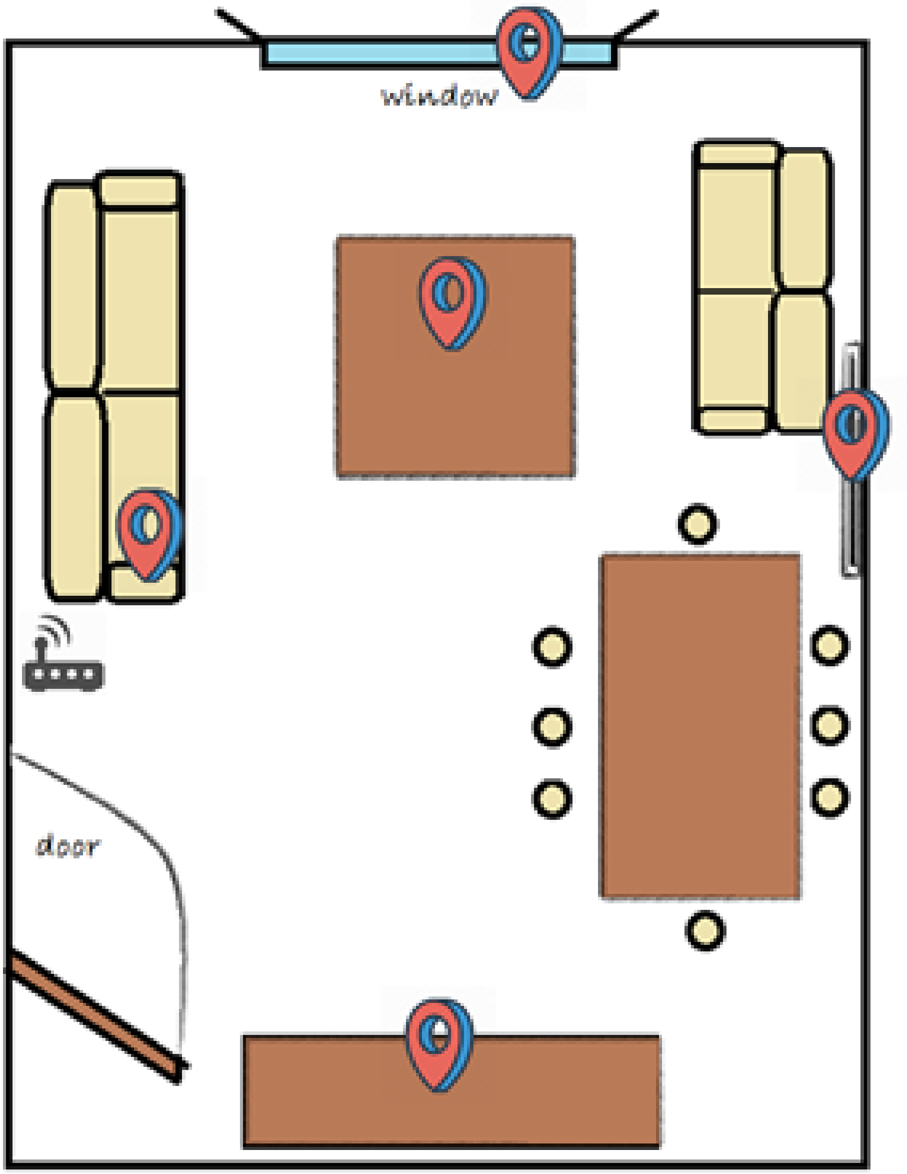

Figure 4.3: Chosen places in lounge

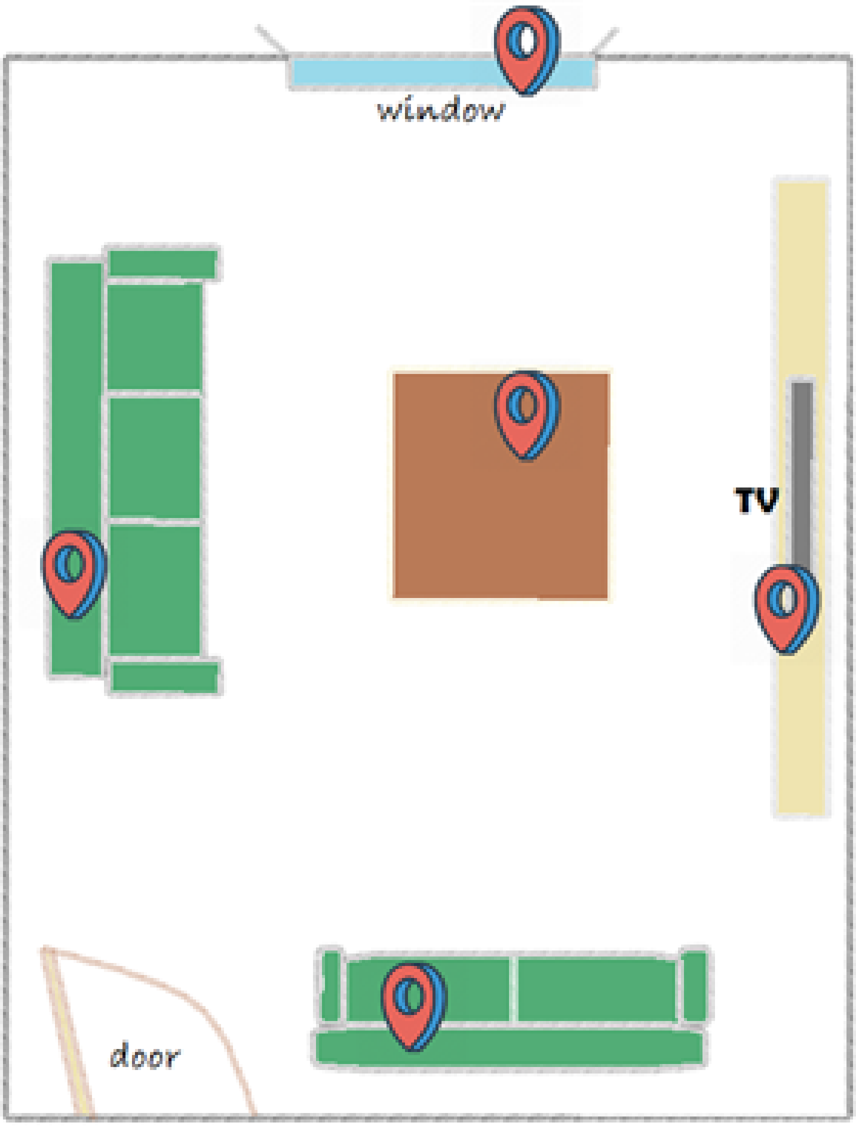

Figure 4.4: Chosen places in living room

At the determined locations, the smartphone is placed and kept at the same position for approximately one minute. The data collection method is activated by the start button being pressed manually by the user after placement. Then, the stop button terminates the data collection. During the time between start and stop button pressing, the received sensor data is recorded on the local memory of the phone without processing. A timer is displayed to control the collection time by users. In this manner, we have collected data at different times in a day. In Android OS, the SensorEventListener interface allows developers to read sensor data by easy methods.

The data reading frequency is adjustable. The predefined constant values in Android are as follow:

- SensorManager.`SENSOR_DELAY_NORMAL` (delay of 200000 microseconds) (default value)
- SensorManager.`SENSOR_DELAY_GAME` (delay of 20000 microseconds)
- SensorManager.`SENSOR_DELAY_UI` (delay of 60000 microseconds)
- SensorManager.`SENSOR_DELAY_FASTEST` (delay of 0 microseconds)

We used SensorManager.`SENSOR_DELAY_NORMAL` value for delay equal to 0.2 seconds.

To access Wi-Fi connection, the Wi-FiManager class has been used. This class provides the primary Application Program Interface (API) for managing all aspects of Wi-Fi connectivity. We have created a class from the BroadcastReceiver abstract class. This class has been used to activate the Wi-Fi scanner method of Wi-FiManager. The BroadcastReceiver can receive data depending on Wi-FiManager configurations. We have designed an asynchronized process to run BroadcastReceiver, and scan Wi-Fi signals at least 8 times in a minute. This method provides Service Set Identifier (SSID), Basic Service Set Identifier (BSSID), and RSSI information of the Wi-Fi AP located nearby.

4.4. Data Exploratory Visualization

It is an important step to discover the data before machine learning algorithms are applied. In this way, the size of the data, the attributes of the data, and the data types of those attributes are understood.

In the given raw data graphics of lounge, living room, and office environments, it can be observed that approximately 60–150 seconds' data were collected at different times. Times of the collection activities for lounge are shown in Table 4.3, the living room in Table 4.4, and the office in Table 4.5.

Table 4.3: Data collection times for the lounge

Room	**Date**	**Time**
lounge	25.6.2017	15:00
lounge	26.6.2017	00:50
lounge	27.6.2017	13:46
lounge	27.6.2017	17:20
lounge	27.6.2017	19:10
lounge	28.6.2017	10:40
lounge	28.6.2017	11:55
lounge	28.6.2017	15:40
lounge	27.6.2017	15:32
lounge	3.7.2017	18:00
lounge	8.7.2017	11:15
lounge	9.7.2017	13:42
lounge	9.7.2017	17:13
lounge	9.7.2017	19:16
lounge	10.7.2017	20:35
lounge	22.7.2017	09:42
lounge	23.7.2017	09:25

Table 4.4: Data collection times for the living room

Room	Date	Time
living room	25.6.2017	14:49
living room	26.6.2017	00:40
living room	27.6.2017	14:08
living room	27.6.2017	15:33
living room	27.6.2017	17:35
living room	27.6.2017	19:20
living room	27.6.2017	21:10
living room	28.6.2017	11:45
living room	28.6.2017	11:35
living room	28.6.2017	15:50
living room	29.6.2017	10:25
living room	3.7.2017	18:20
living room	9.7.2017	17:00
living room	9.7.2017	18:55
living room	10.7.2017	20:45
living room	22.7.2017	09:50
living room	23.7.2017	09:35

Table 4.5: Data collection times for the office

Room	**Date**	**Time**
Office	28.4.2017	10:52
Office	28.4.2017	16:00
Office	28.4.2017	17:14
Office	05.05.2017	12:15
Office	12.5.2017	14:32
Office	22.5.2017	14:24
Office	22.5.2017	15:15
Office	23.5.2017	15:24
Office	23.5.2017	17:44
Office	29.5.2017	13:25
Office	29.5.2017	15:51
Office	31.5.2017	15:52
Office	5.5.2017	12:15

We have labelled chosen places for each room as Place1, Place2, Place3, Place4 and Place5.

Place1 in the lounge is located in front of the window, on the floor. There is a flowerpot right next to it. Place2 is located on a radiator that is on the right side. Place3 is on a coffee table, the furthest spot from the window. Place4 is on the sofa's side furthest from the window, to the right next to the Wi-Fi modem. Place5 is on the console opposite the wall with the window.

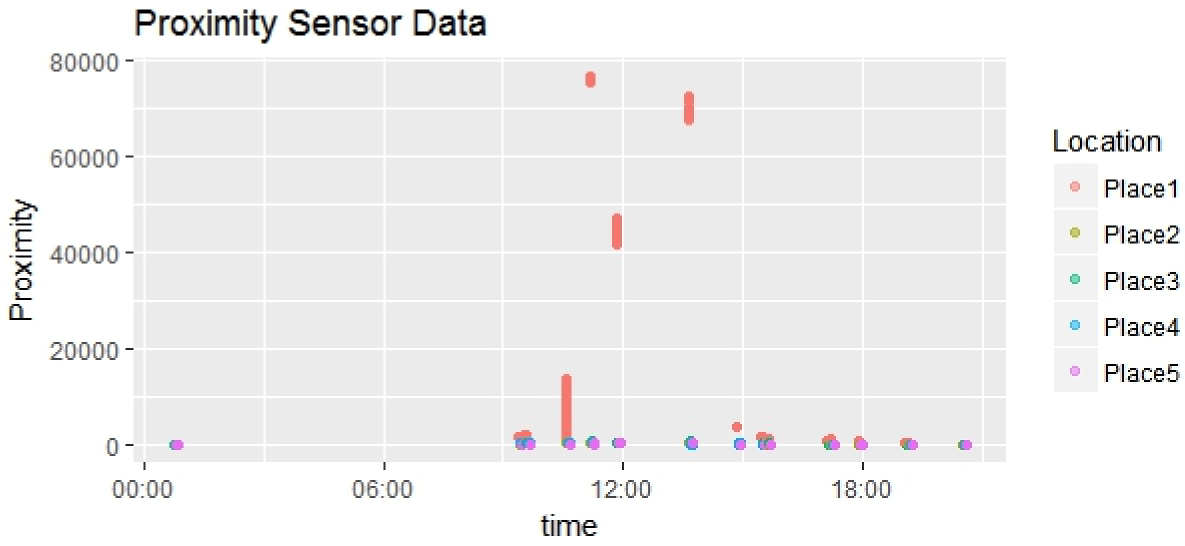

Figure 4.5: Proximity sensor data from lounge

As it is seen in Figure 4.5, proximity sensor fluctuates for Place1 during the day, while showing stable values for other locations. The position of the flowerpot changes according to the window being open or closed. Similarly, it is thought that whether the curtains are blinded or not affects the variation of the proximity value.

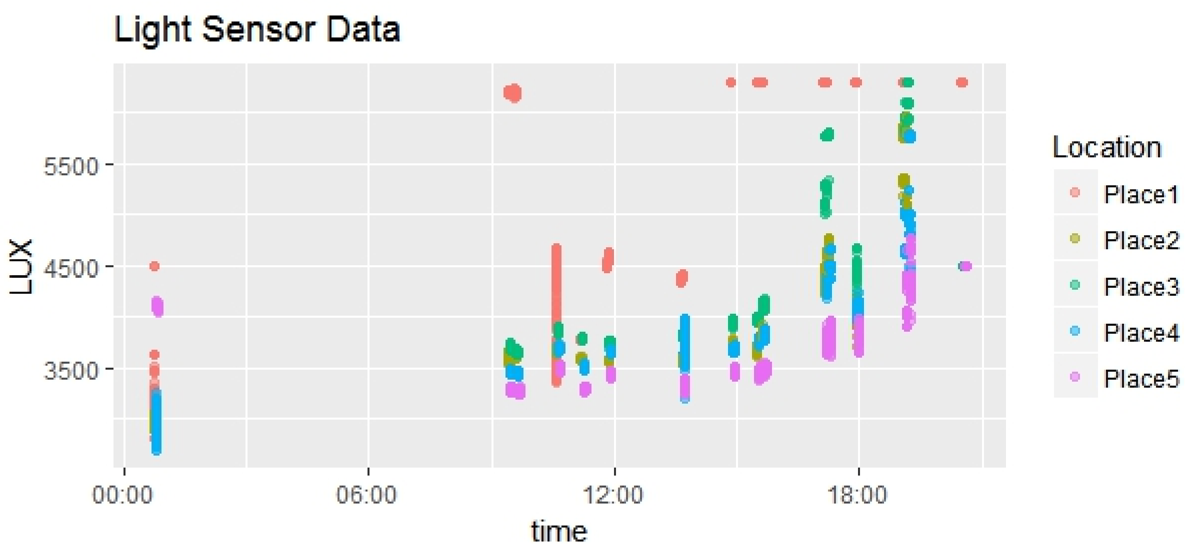

Figure 4.6: Lux sensor data from lounge

Figure 4.6 shows that, light sensor varies for all locations at different times a day. However, lux value varies differently at different days even at similar times of day. While, the lowest lux value is at the furthest position to the window (Place5) when there is day light, the highest lux value is at Place1, which is right next to the window. After sunset, this alignment changes under fluorescence light.

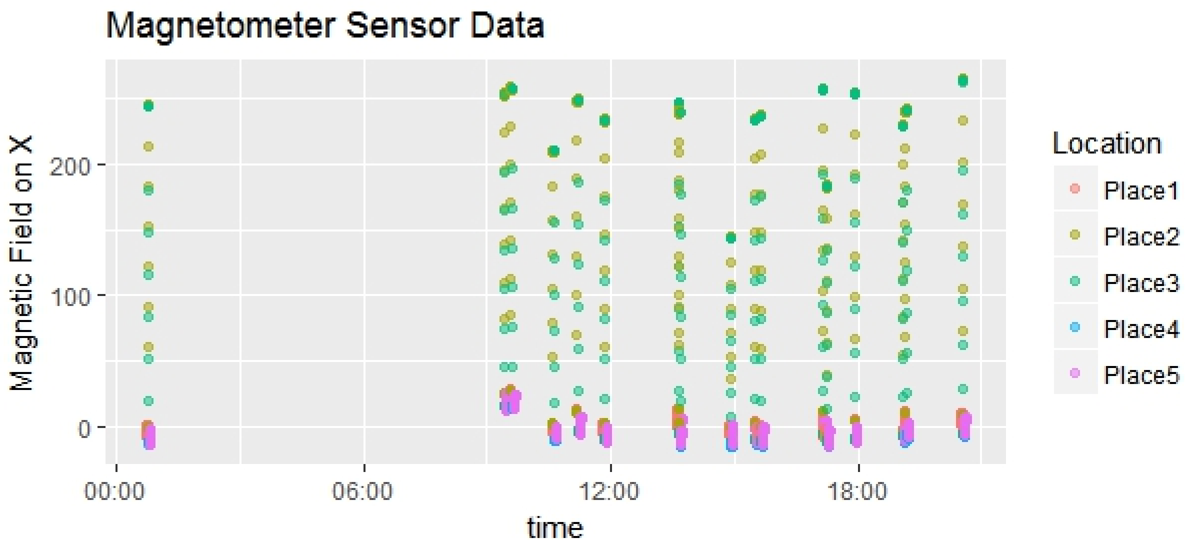

Figure 4.7: Magnetic field data on X-axis from lounge

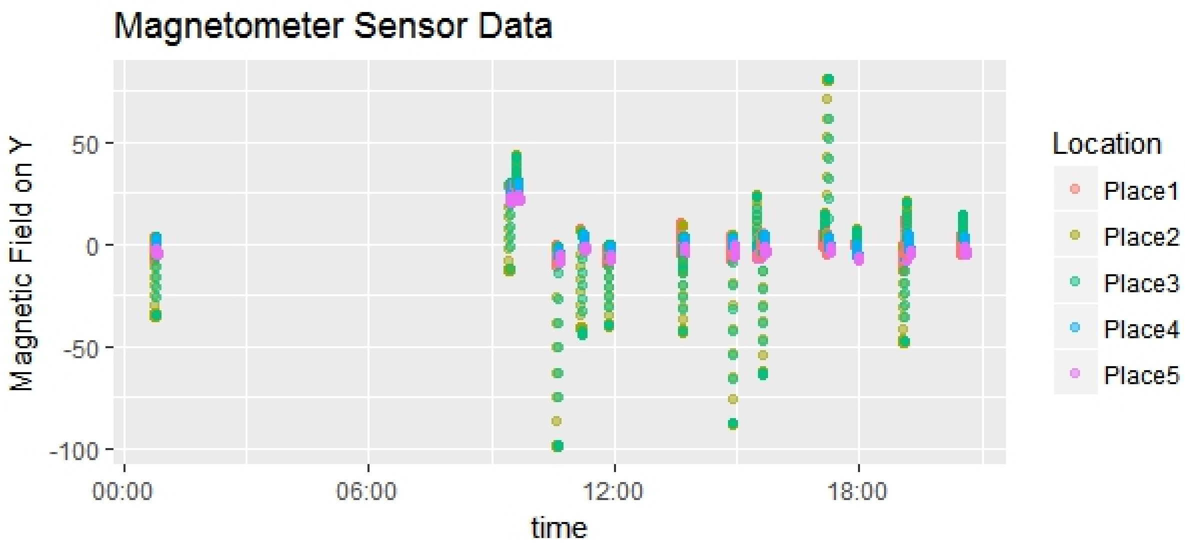

Figure 4.8: Magnetic field data on Y-axis from lounge

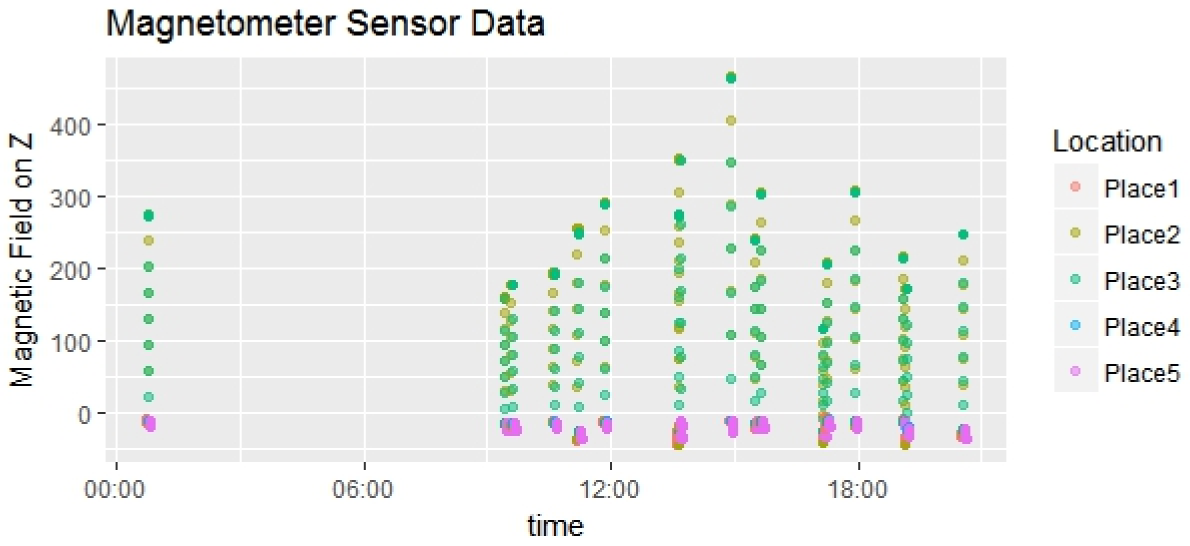

Figure 4.9: Magnetic field data on Z-axis from lounge

It can be seen in Figure 4.7, Figure 4.8 and Figure 4.9, while magnetic field shows similar values for all locations at different days and hours, it considerably fluctuates for Place2 and Place3 at X, Y and Z axes.

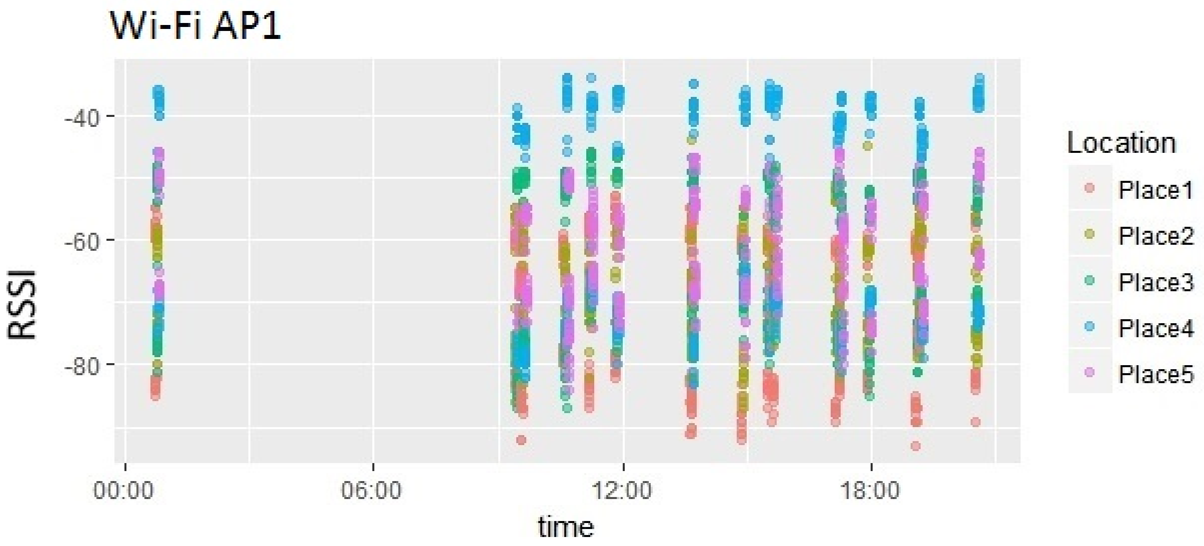

Figure 4.10: RSSI values of 1.Wi-Fi AP from lounge

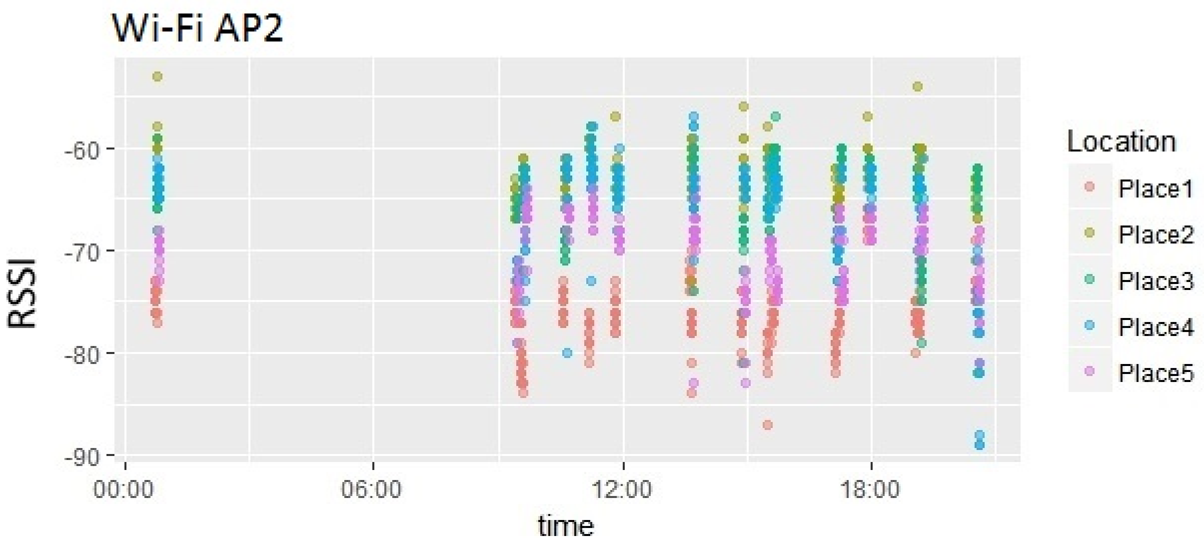

Figure 4.11: RSSI values of 2.Wi-Fi AP from lounge

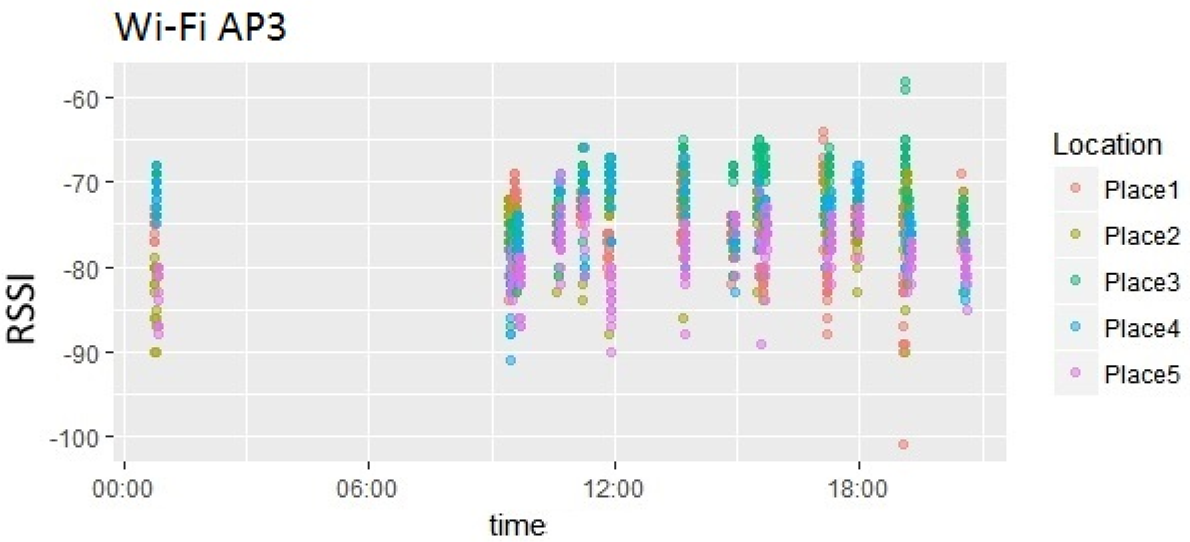

Figure 4.12: RSSI values of 3.Wi-Fi AP from lounge

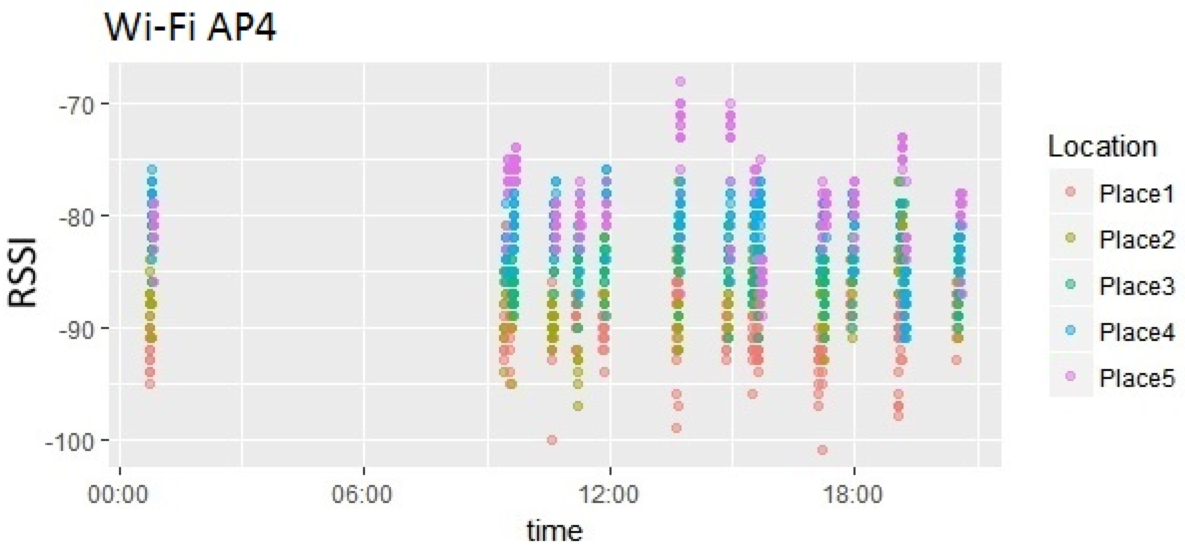

Figure 4.13: RSSI values of 4.Wi-Fi AP from lounge

When it comes to Wi-Fi values, there is a Wi-Fi access point in this room, as mentioned before. Graphic that is shown in Figure 5.6, belongs to this access point. Other access points (Wi-Fi AP2, Wi-Fi AP3 and Wi-Fi AP4) are the APs of which the signals are available for all locations and the ones that give the most results from Wi-Fi scanning and their RSSI values are shown in Figure 4.10, in Figure 4.11, in Figure 4.12 and in Figure 4.13. The values of Wi-Fi AP1 fluctuate more than those of the others. In the process of gathering data, movement inside the room was inevitable. The user placing the phone and then taking it back when the time is up was a handicap for Wi-Fi signals inside the room. Values taken from other APs are mostly in a certain interval for each location. On the other hand, RSSI values are scaled between -40 to -80 for Wi-Fi AP1, showing that RSSI values are higher for AP1 in this room than the others (AP2, AP3, AP4).

For the living room, Place1 is located on radiator in front of the window. Place2 is next to the TV inside the bookshelf. Place3 is on the floor in the middle of the room. Place4 is on the big sofa, located on the left side of the room. Place5 is on the little sofa, which is the furthest spot from the window.

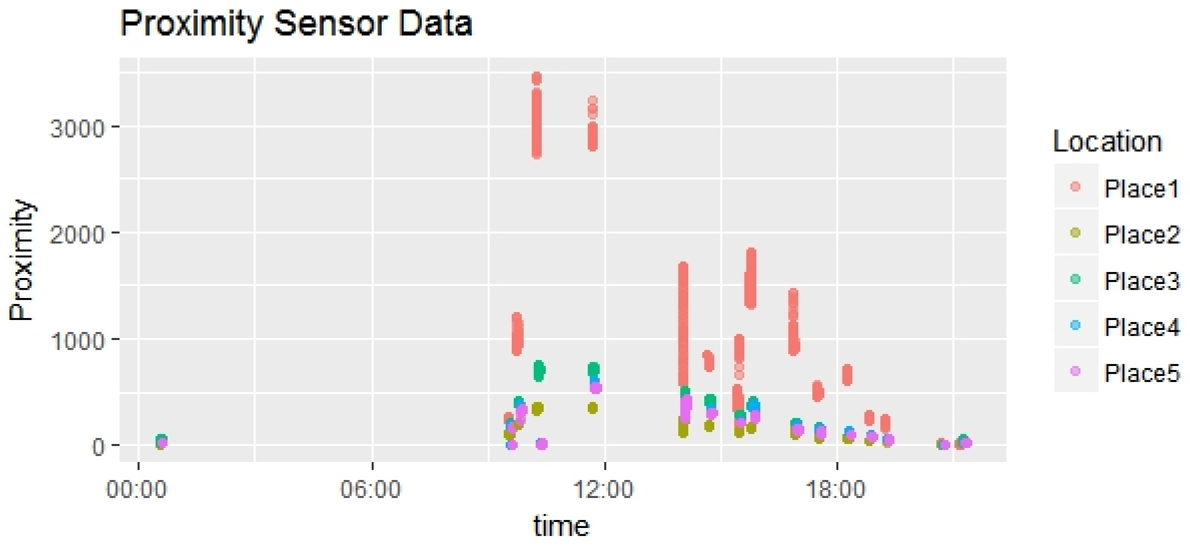

Figure 4.14: Proximity sensor data from living room

As it is seen in Figure 4.14, proximity sensor data, as in the other room, fluctuates at Place1, which is closest location to the window. It is stable for other locations. The curtain at Place1 is open or closed from time to time, changing in accordance with the situation of window.

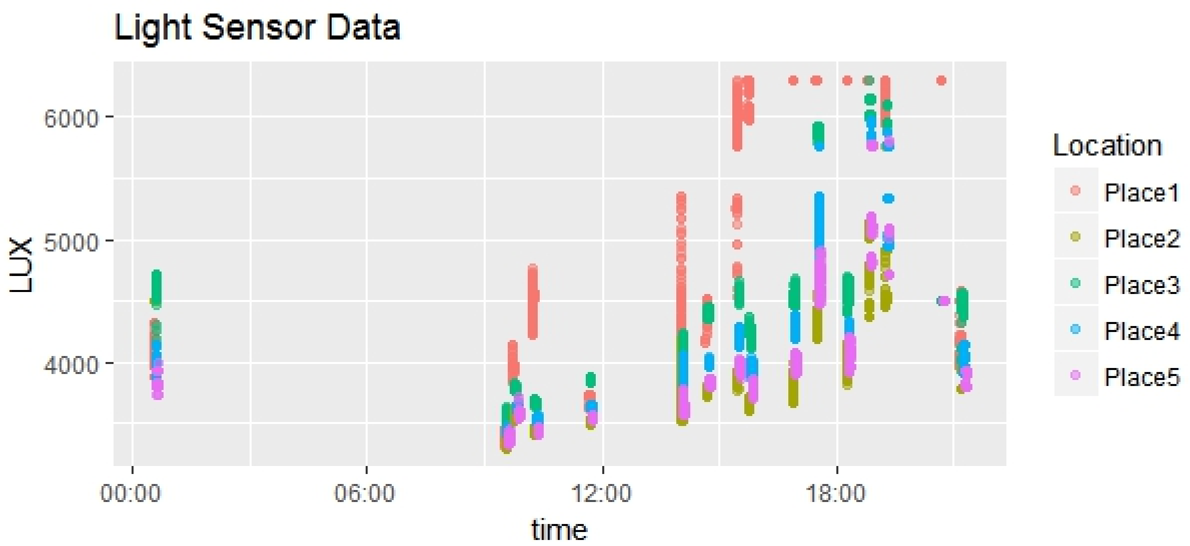

Figure 4.15: Lux sensor data from living room

Figure 4.15 shows that, under the daylight conditions, light sensor gives the highest value at the window side, while giving the lowest value at Place2. Since Place2 is located on the bookshelf, it predicted to receive less light than other locations. Data gathered after 21:00 and 00:00 were gathered under fluorescence light. In this case, values are totally different from the values taken under the day light. However, lux values of all locations under the fluorescence light are regular for all examples.

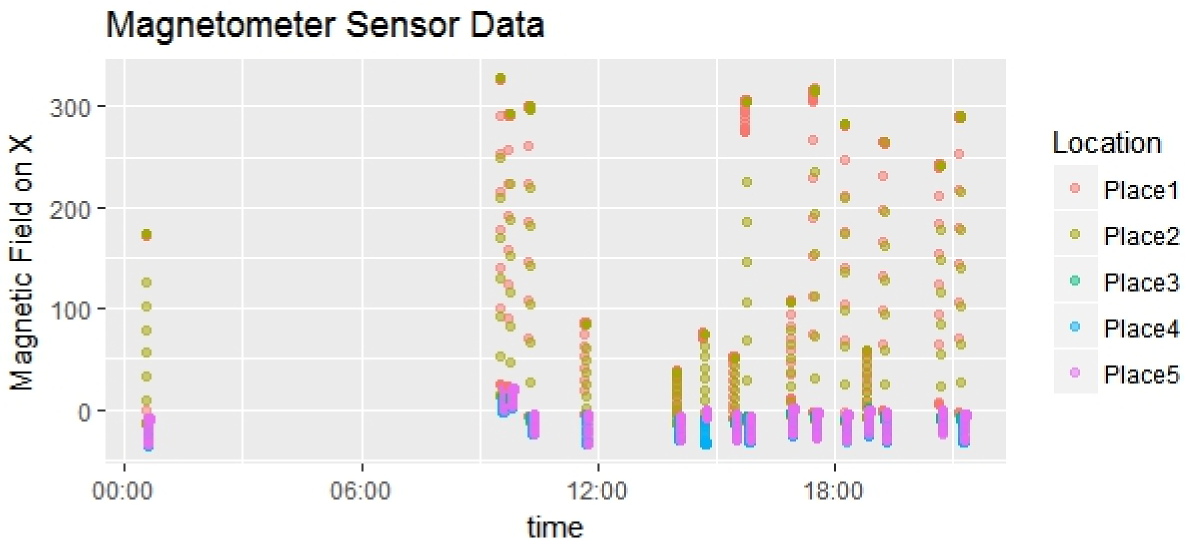

Figure 4.16: Magnetic field data on X-axis from living room

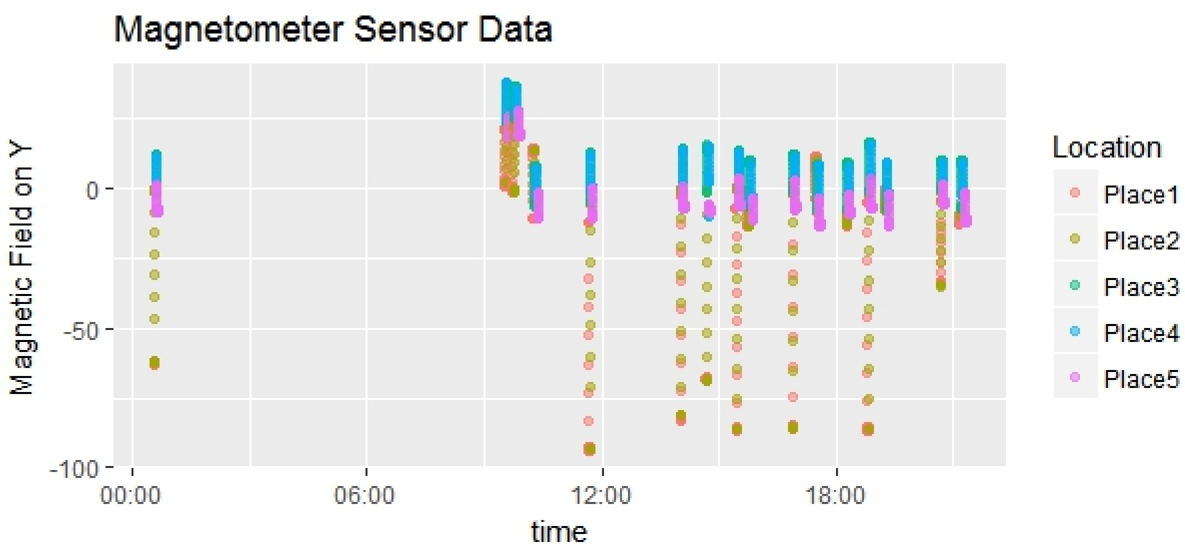

Figure 4.17: Magnetic field data on Y-axis from living room

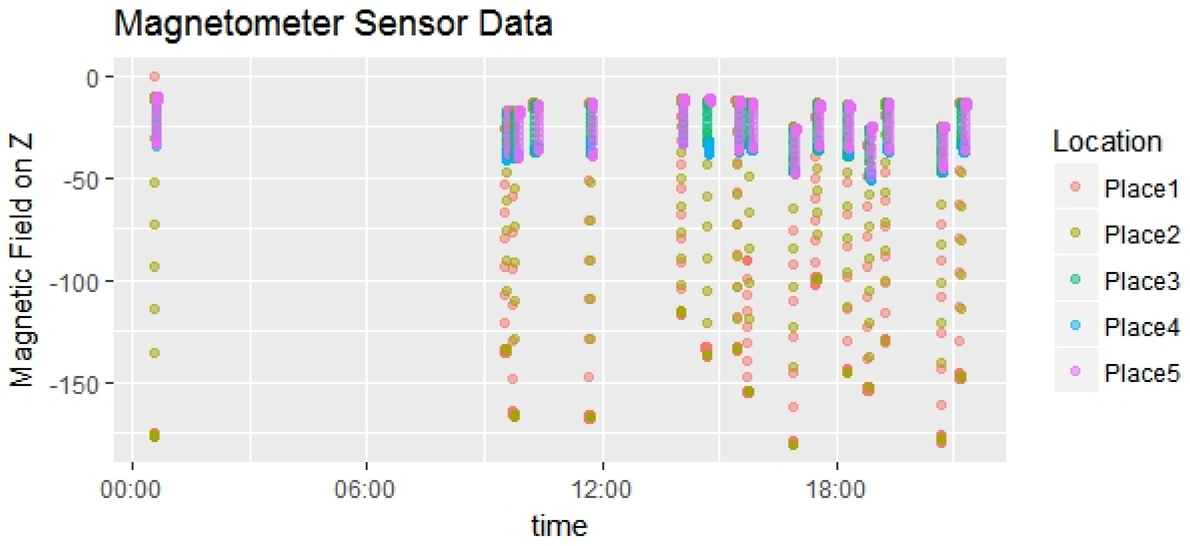

Figure 4.18: Magnetic field data on Z-axis from living room

When Figure 4.16, 4.17 and 4.18 is observed, it is clear that magnetic field data, while giving regular values for all locations, varies on a large scale for Place2, which is right next to the TV.

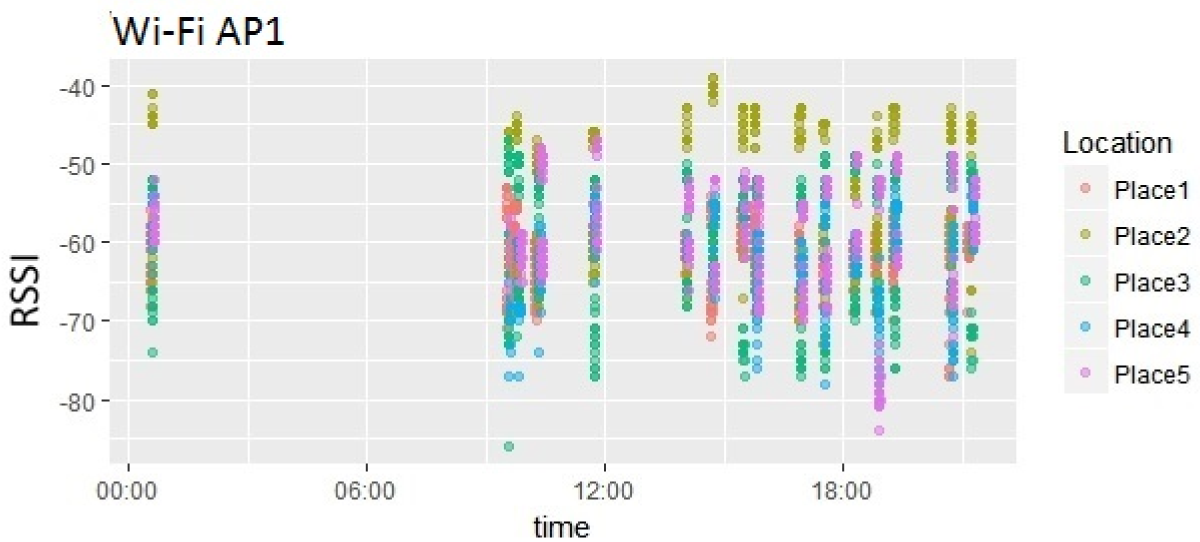

Figure 4.19: RSSI values of 1.Wi-Fi AP from living room

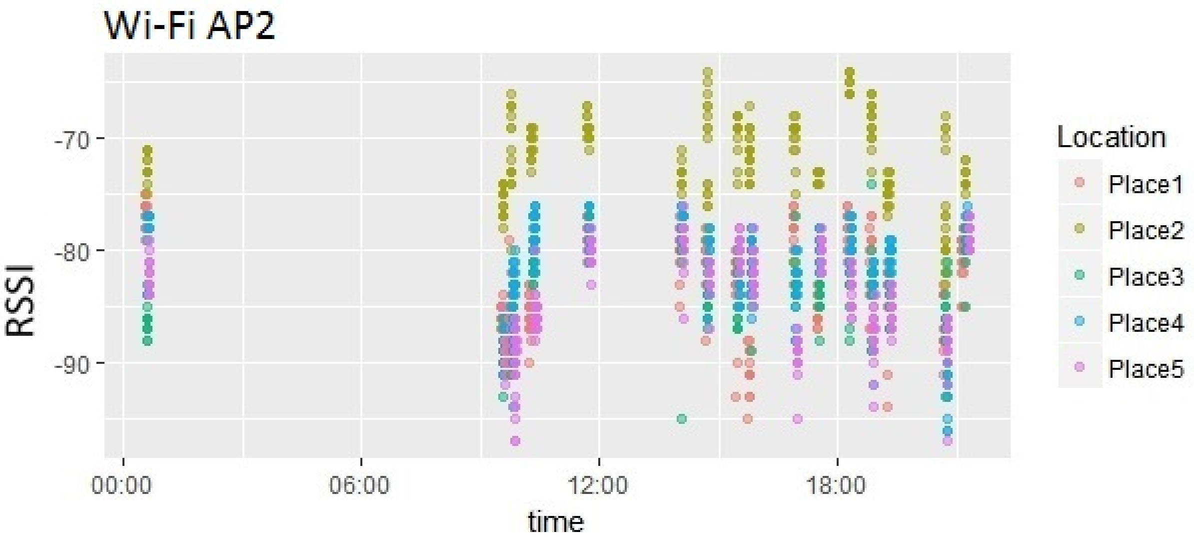

Figure 4.20: RSSI values of 2.Wi-Fi AP from living room

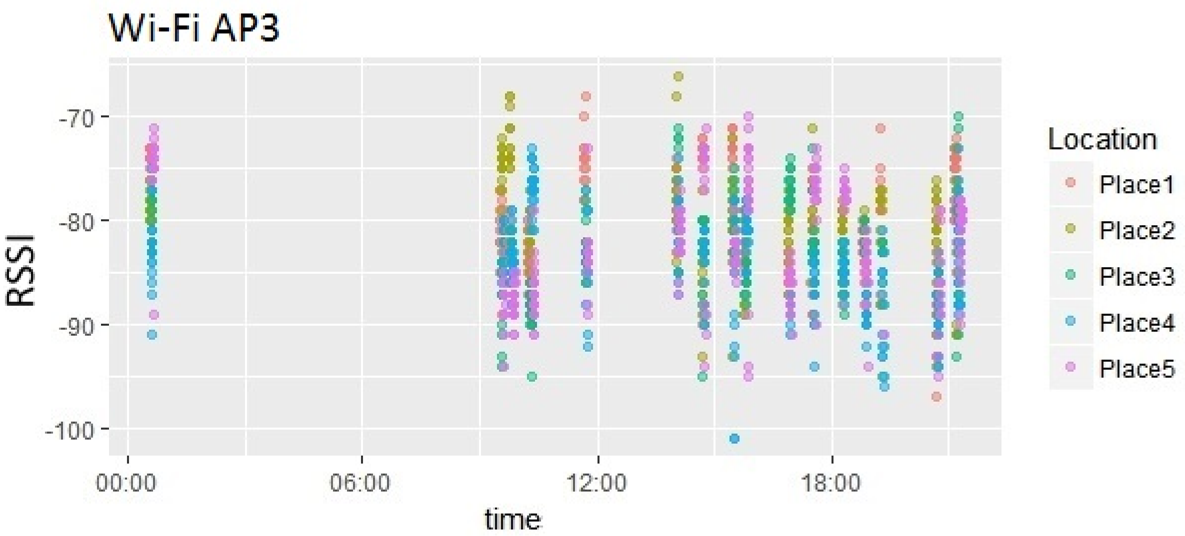

Figure 4.21: RSSI values of 3.Wi-Fi AP from living room

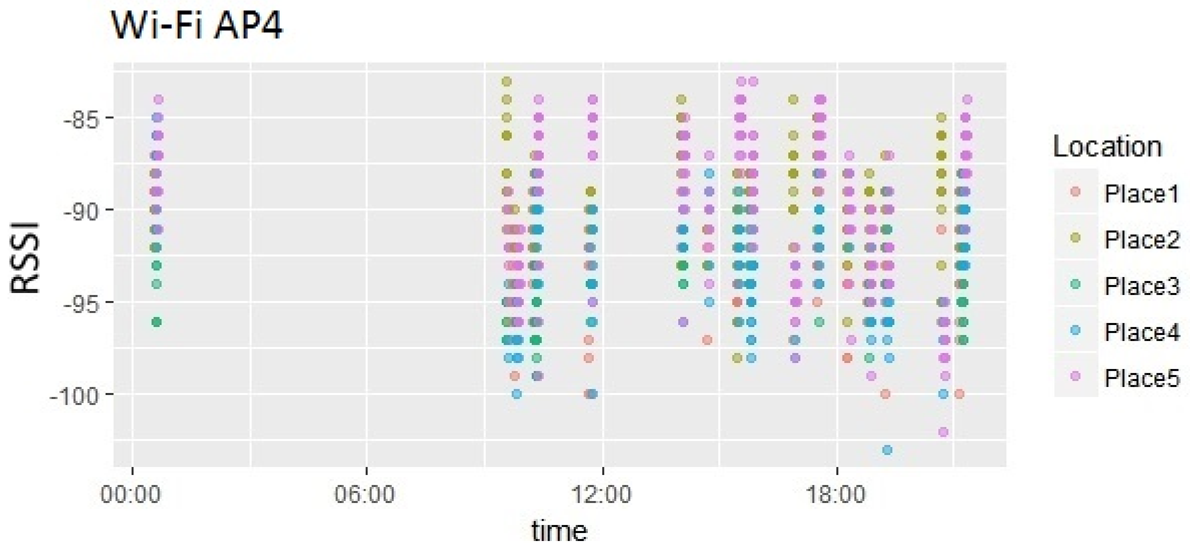

Figure 4.22: RSSI values of 4.Wi-Fi AP from living room

While scanning for Wi-Fi, it might not be possible to get signals from all Wi-Fi access points in every scanning process. Signals can be interrupted due to various physical reasons. From Figures 4.19, 4.20, 4.21 and 4.22, it has been observed that signals coming from Wi-Fi AP4 are weaker than the others. On the other hand, since Wi-Fi AP1 (in the lounge) is in the same house, the highest RSSI values are gathered from this access point.

In the office environment, tables that data gathered from are different only with regard to their positions. There is a laptop on each table. Any phones or electronic devices were not interfered with while gathering data. Data was gathered only during the work hours. As it is seen in Figure 4.23, on the contrary to the house environment, proximity sensor data fluctuates in the office. People in the office kept working on their tables in the process of data gathering. Therefore, proximity sensors sense different objects nearby.

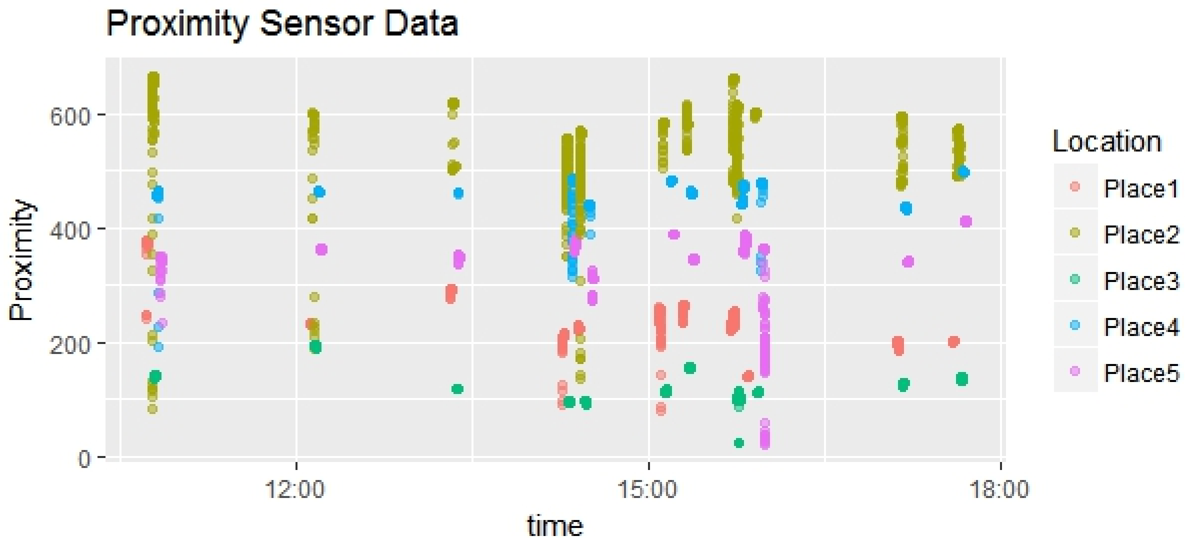

Figure 4.23: Proximity sensor data from office

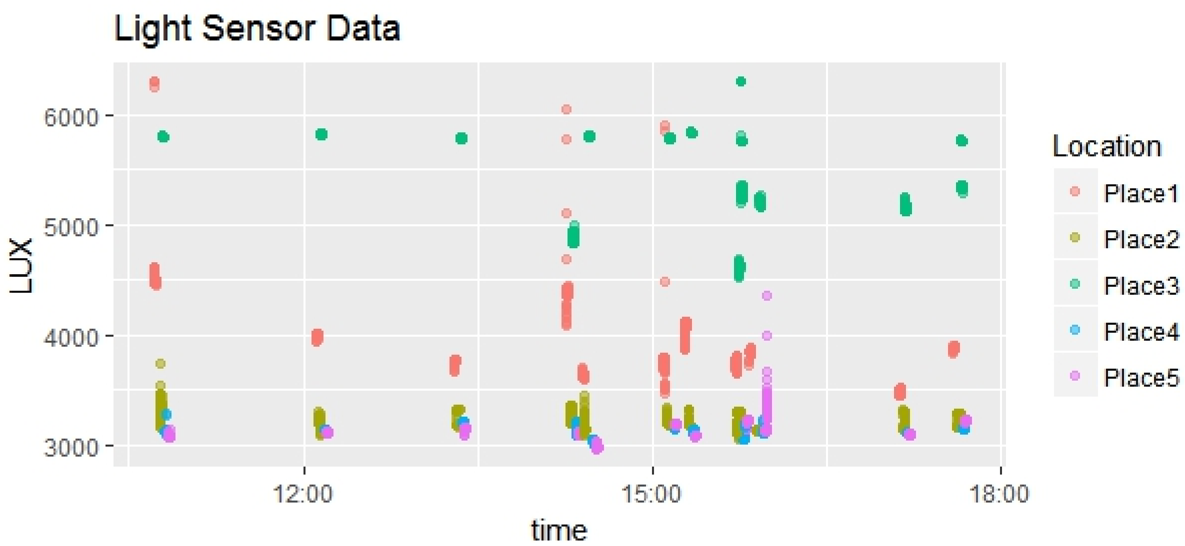

Figure 4.24: Lux sensor data from office

Since fluorescence light is available at all times during the day in the office, Figure 4.24 shows that the daylight effect was felt less than the home environment. It has been observed that light has different values for different positions in the environment.

As it is seen in Figure 4.25, 4.26 and 4.27, magnetic field variations have similar values for all locations, even though there are possible variations for each of the three axes in the office.

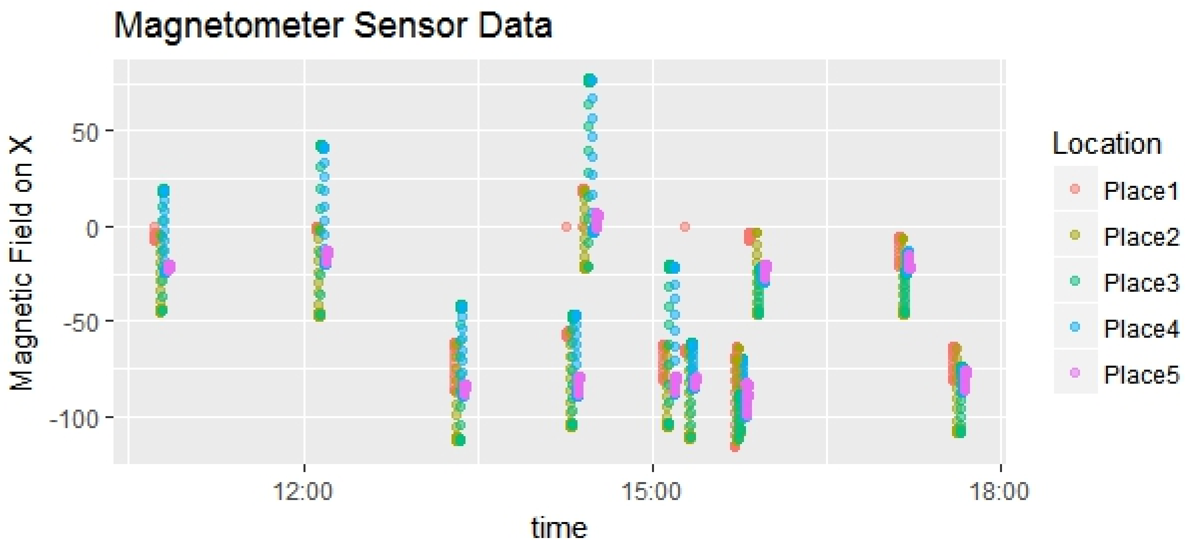

Figure 4.25: Magnetic field data on X-axis from office

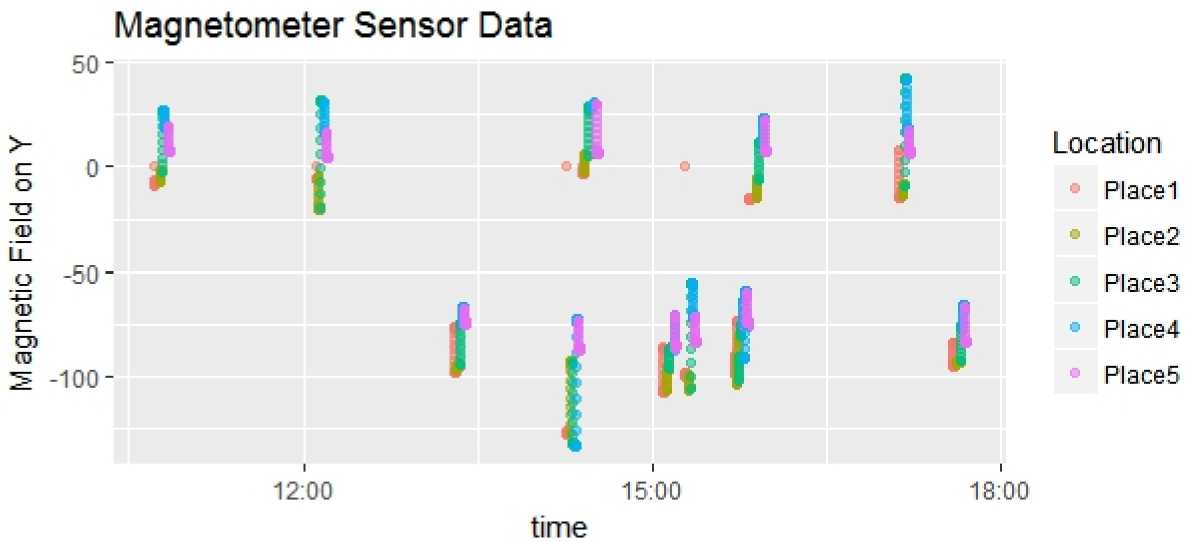

Figure 4.26: Magnetic field data on Y-axis from office

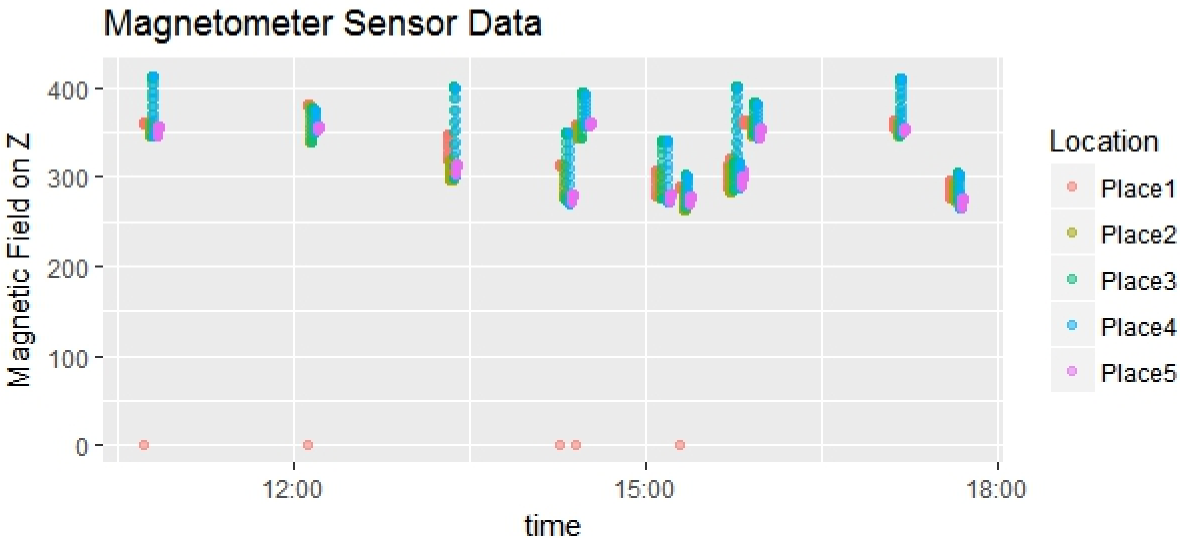

Figure 4.27: Magnetic field data on Z-axis from office

The office is a building with 19 floors. The entire building has been provided with Wi-Fi service from a single access point for each floor. Wi-Fi AP1 is the Wi-Fi service provided by the company. Figure 4.28 shows that, RSSI values of Wi-Fi AP1 vary for all locations on a large scale. Wi-Fi AP2 is a connection that is not possible to access from every location. As it is seen in Figure 4.29, RSSI values of Wi-Fi AP2 are distinctive in this regard. The connection is not available at the tables facing north, while it is available at the tables facing south.

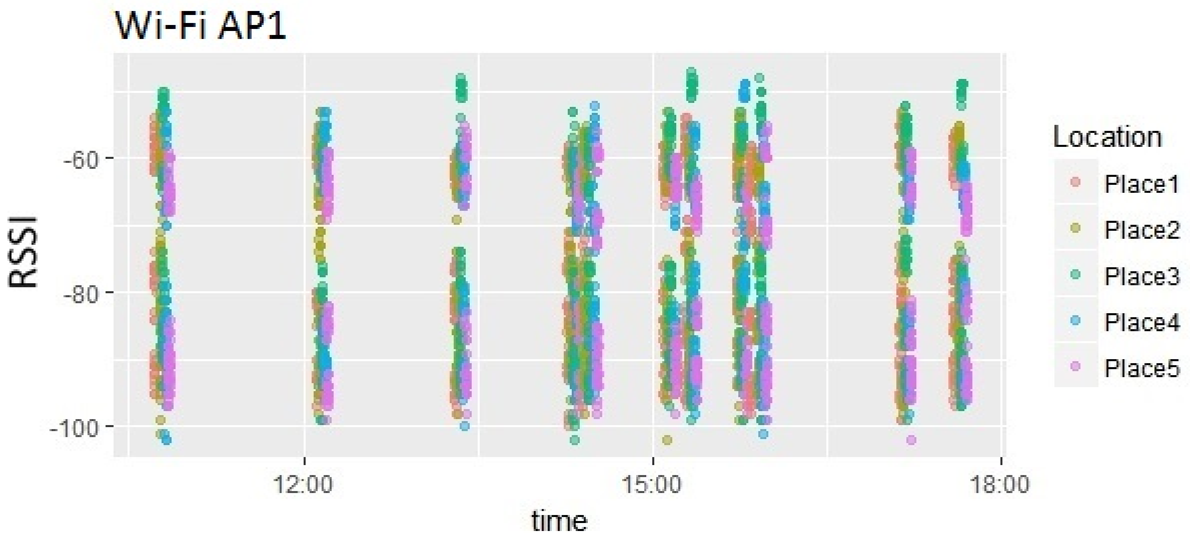

Figure 4.28: RSSI values of 1.Wi-Fi AP from office

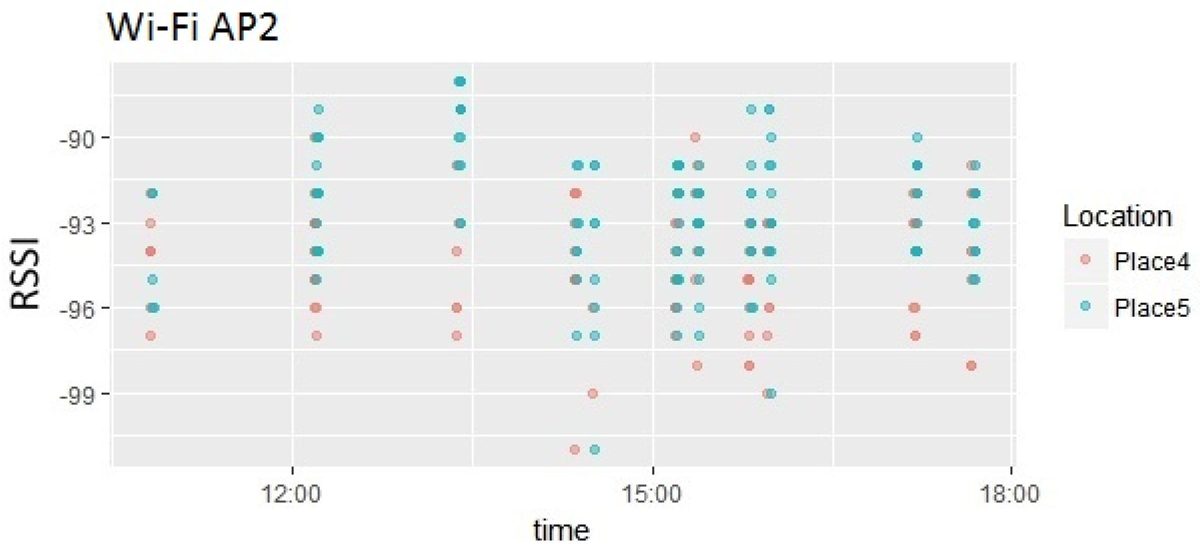

Figure 4.29: RSSI values of 2.Wi-Fi AP from office

To visualize the characteristics of the sensor data for different locations, radarcharts have been created. The radarchart, also referred to as a spider graph, is an effective visualization method to compare multiple quantitative variables. This kind of radarchart is beneficial for observing that which variables have similar values or if there are any extremes for each variable. For collected dataset, these charts provide a chance to make comparison between data which are collected from at the same times of different days.

In addition, to visualize our dataset on radarcharts, we scaled all variables by using z-score. A z-score, which is also known as a standard score, can be put on a normal distribution curve. Z-scores of variables are calculated by subtracting the mean of all data points from each individual data point, then dividing those points by the standard deviation of all points. Thus, z-scores can expressed as number of standard deviations from their means.

$$z = \frac{X - \mu}{\sigma} \tag{4.1}$$

where X is the value of a variable, μ is mean, and μ is standart deviation. As we explained in the data collection part, we have datasets from different times of different days. Each sample for each place is approximately 1 minute long. We have created radarcharts for these 1-minute samples separately. To put a dot on radarchart for each variable, we have calculated median values for each variable.

In the radarcharts, variables are represented as;

- L : Lux
- P : Proximity
- X : Magnetic field on X-axis of smartphone
- Y : Magnetic field on Y-axis of smartphone
- Z : Magnetic field on Z-axis of smartphone
- R1 : RSSI of Wi-Fi AP1
- R2 : RSSI of Wi-Fi AP2
- R3 : RSSI of Wi-Fi AP3
- R4 : RSSI of Wi-Fi AP4

In the lounge, when we look at Figure 4.30 we can observe that, for Place1, lux value almost reaches maximum at any time but evening times without florescence. In

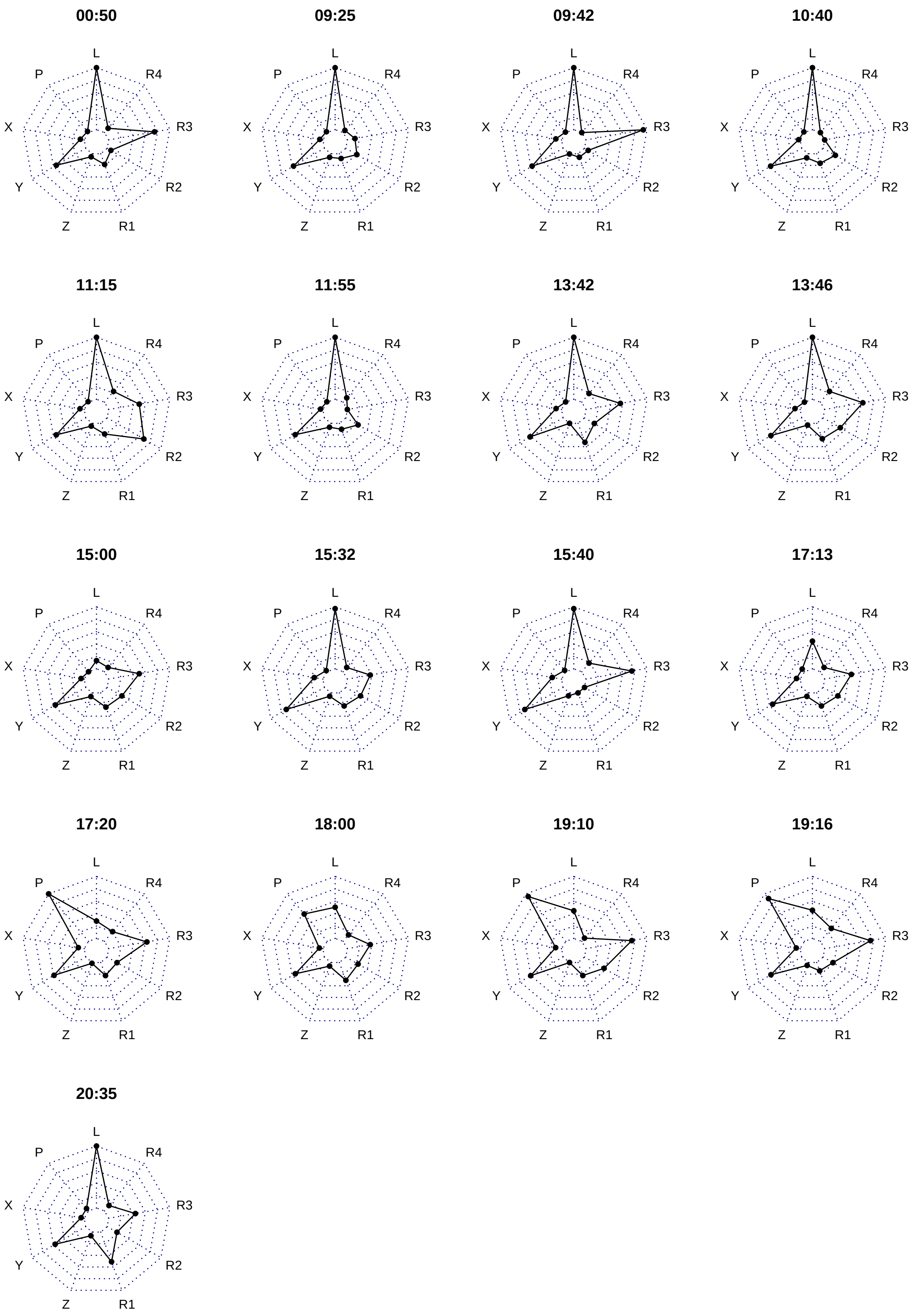

Figure 4.30: Radarchart of Place1 in lounge at various times

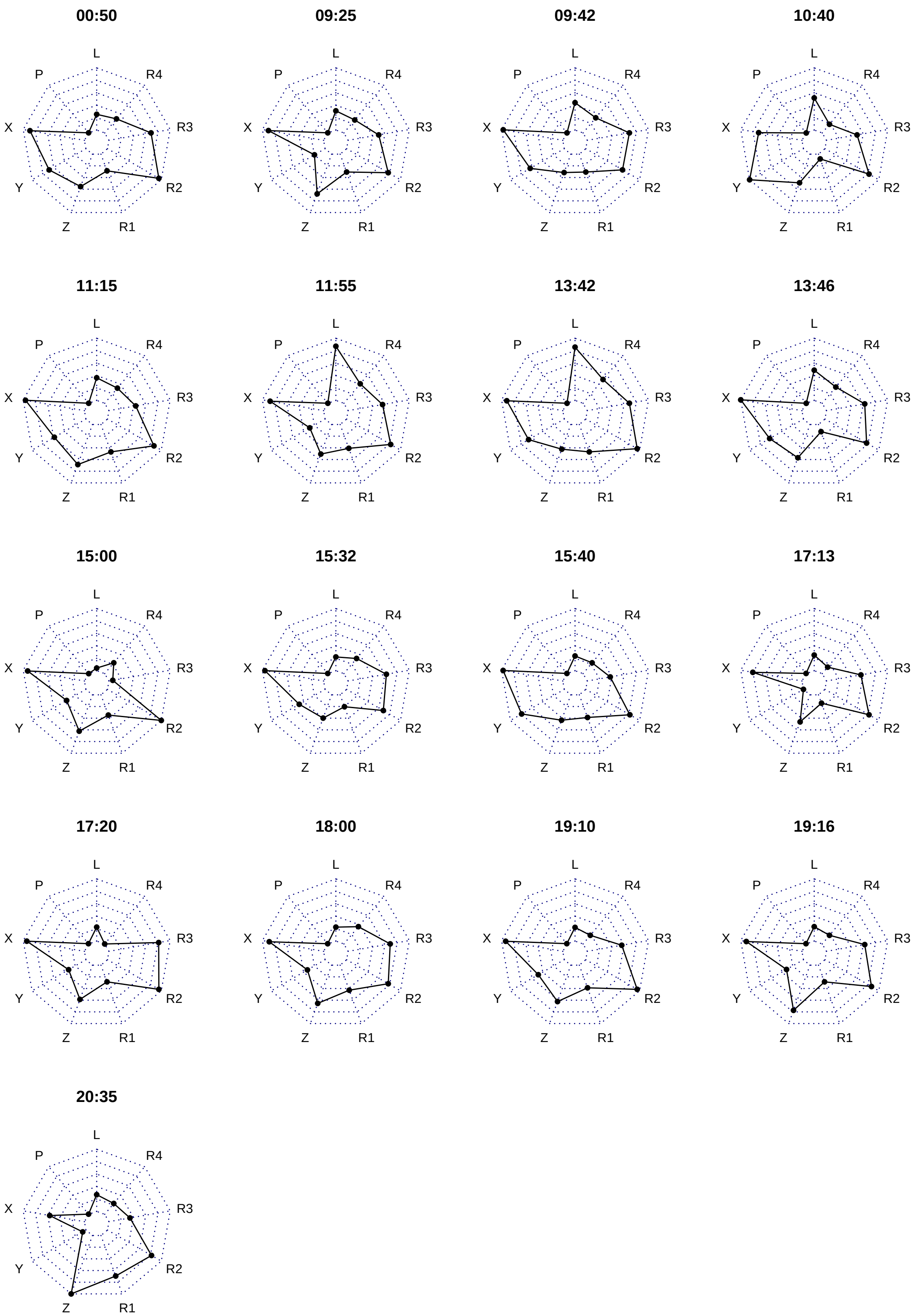

Figure 4.31: Radarchart of Place2 in lounge at various times

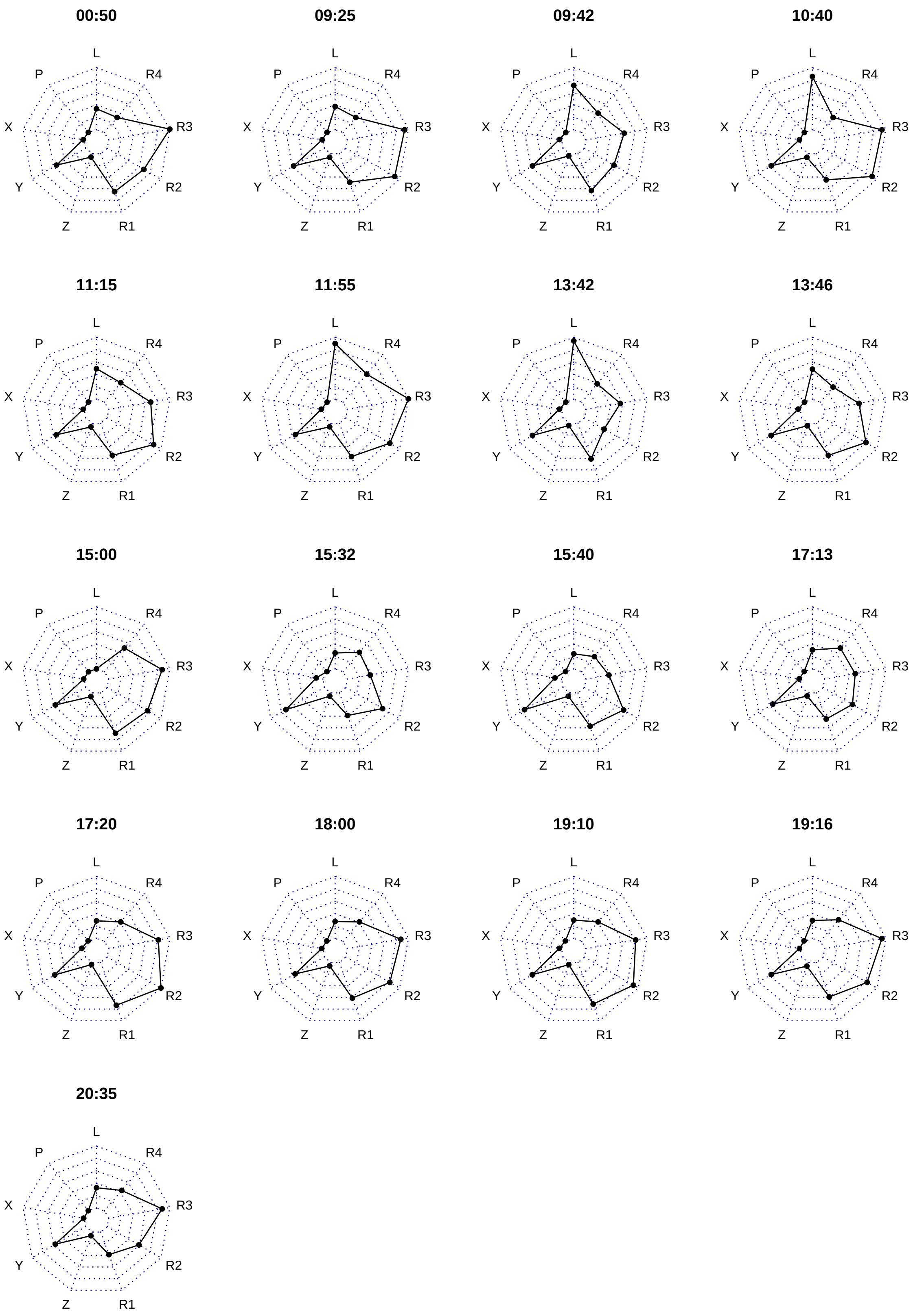

Figure 4.32: Radarchart of Place3 in lounge at various times

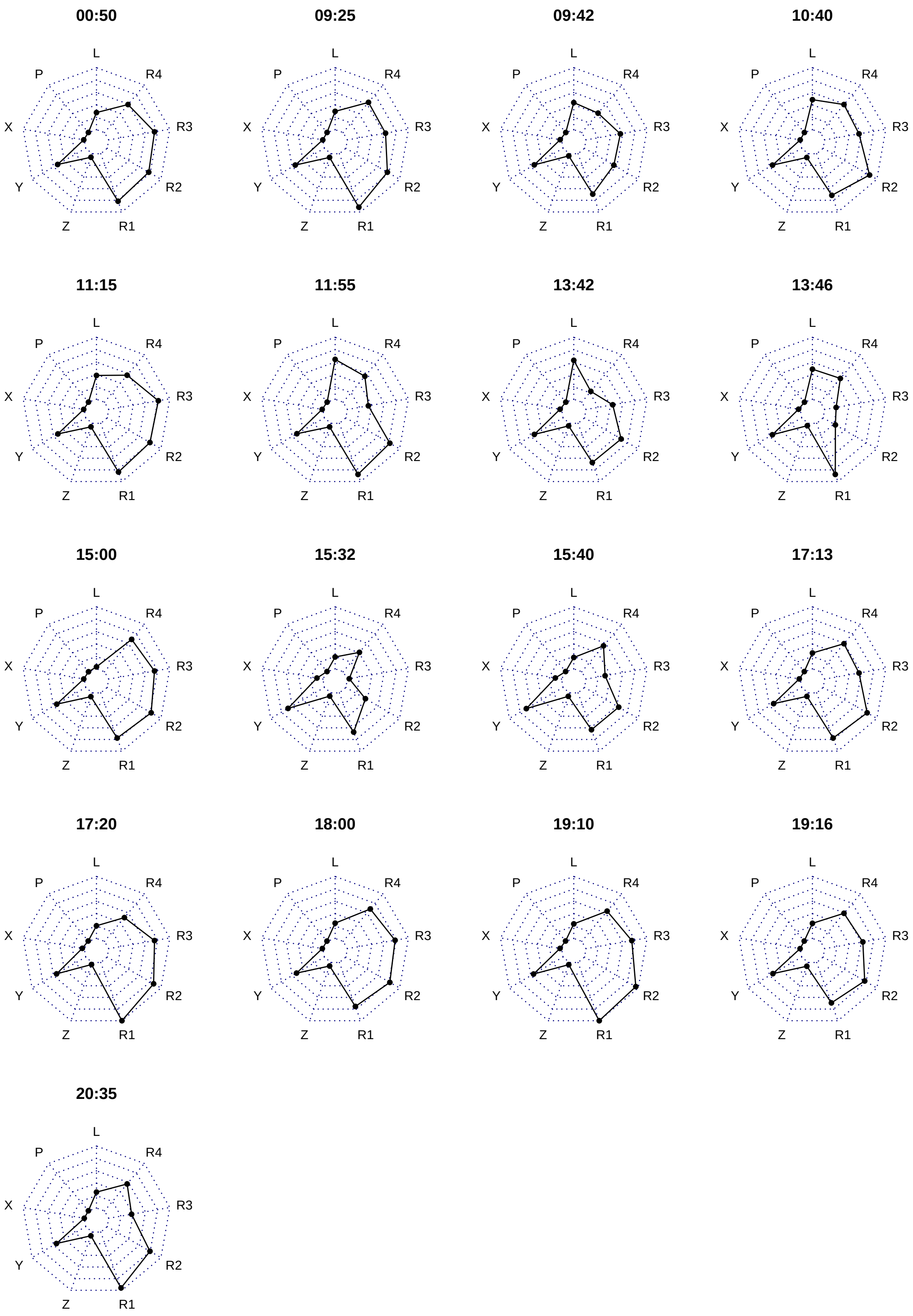

Figure 4.33: Radarchart of Place4 in lounge at various times

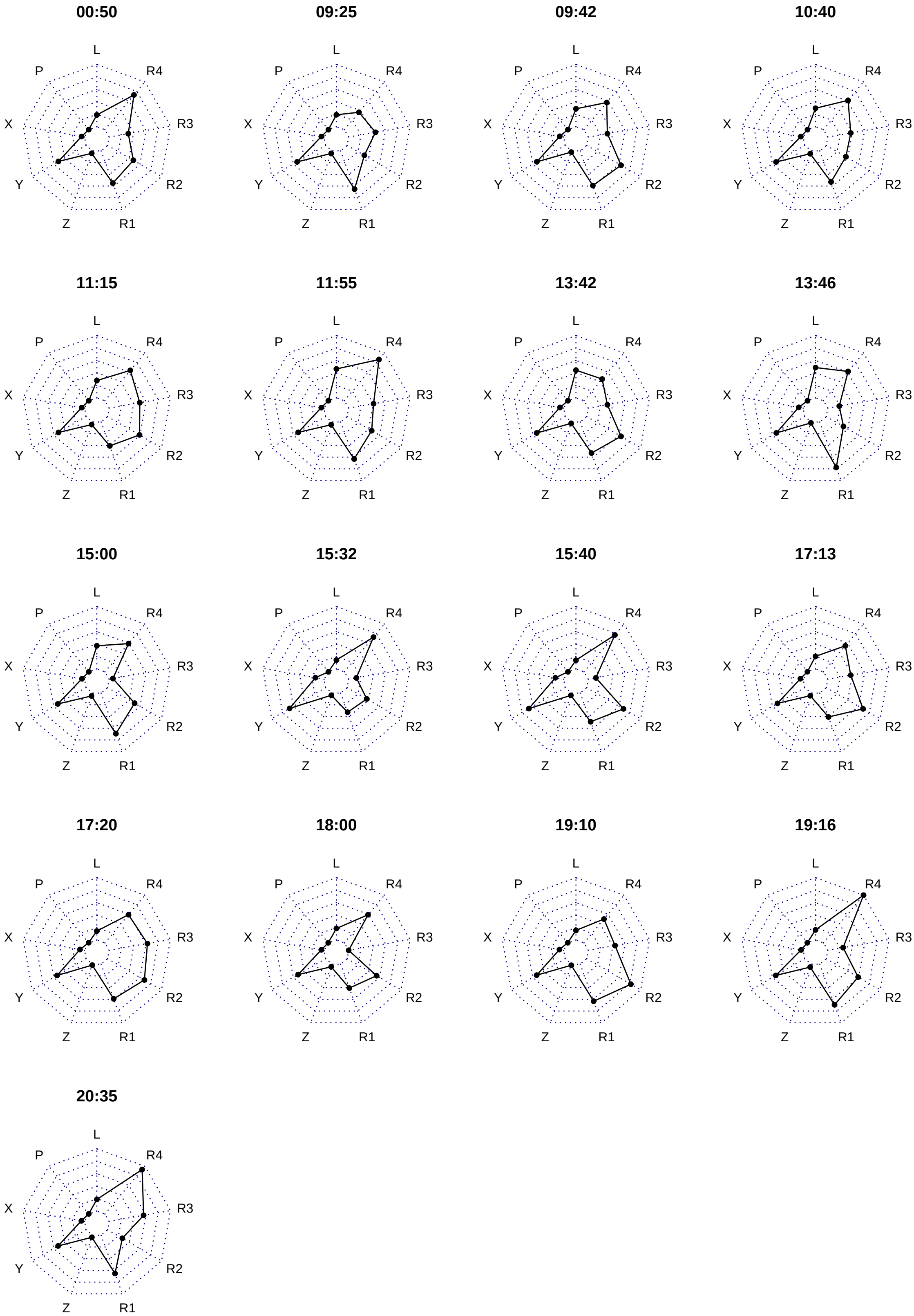

Figure 4.34: Radarchart of Place5 in lounge at various times

this place, which is the location nearest to the window, lux value is always the highest in the room. Magnetic field is a stable component for each axis. The Wi-Fi RSSI values do not show similar characteristic during the day for all places.

As it is seen in Figure 4.31, in Place2 at the lounge, it can be said that the polygons of close times look like similar, although they have been collected on different days. For example, the polygon at 19:10 and the polygon at 19:16 are almost the same, with slight difference in R1 and Z. Another notable point for Place2 is that magnetic field on X-axis mostly reaches to maximum value at different hours. And the average values of R2 component during all day is remarkably higher than the average values of R2 component of other places.

Figure 4.32 shows that in Place3, from 5:00 p.m. to morning hours, light component proportionally shows stable values. And the values of magnetic fields are almost same at any time of day.

When we look at Figure 4.33, in Place4, magnetic field values on X, Y, Z axes do not change much at different times of the day. Apart from this, it can be seen that R1 value is relatively higher due to the proximity of Place4 to Wi-Fi AP in the same room.

As it is seen in Figure 4.34, in Place5, values of light sensor are lower than values of light sensor of other places at almost each time of the day because Place4 is the furthest location to window in the lounge.

It is noteworthy to emphasize that, while the values of proximity sensor are almost zero in all places, in Place1, it shows an increase in certain hours of the day. The reason behind this increase stems from the changing position of curtain which tends to be brought down during evening hours.
For living room and office we have also created radarcharts. They can be found in Appendix A.

For living room, it is deduced that Place1 is the only location that proximity sensor gives high value since Place1 is located next to the window and movement of curtain triggers the proximity sensor. At the morning hours, light sensor reaches maximum value at this location for this room.

Values of the magnetic field on X,Y,Z axes are constant during all day and Value of light sensor is close to minimum except the hours between 09.50 a.m. and 03.30 p.m. at Place2. In addition to that, R1 and R2 reach the highest value at this location for this room. On the other hand, R3 reaches the highest value at Place5.

At Place3 in living room, radarcharts are providing similar values at close hours even though related data were collected on different days. For instance, radarcharts of 05.35 p.m. and 06.55 p.m. show similar polygons.

In the office environment, Wi-Fi signals originate from only one source. Thus, polygons created for the office are hexagons. Radarcharts for office show that most of the time, RSSI values does not change for each place at different hours. Proximity sensor data and magnetic field values change location by location. Place4 and Place5 are two tables at one meter distance from each other. Similarity of polygons of radarcharts for those places overlap with their physical proximity.

As it is seen from radarcharts, sensor data show various characteristics in different places at different times. Hence, it could be said that, every sensor variable can be used as an attribute to distinguish places from each other. Time is also a distinctive parameter for sensor data classification.

5. MACHINE LEARNING MODELS EVALUATION

With the concepts of machine learning, different systems have become understandable. There are two main approaches in machine learning: supervised learning and unsupervised learning. For using supervised learning models, given training dataset must include target output for each sample in the dataset. Supervised learning models correlates outputs with the given samples. In unsupervised learning models, there are not known outputs for the given dataset. It correlates samples of given dataset with each other and makes extractions based on the similarities of samples with each other.

The main target of this study is classifying different locations in a room by using ambient features, time and Wi-Fi RSSI signals around the smartphone. It is important to use the correct learning algorithm while performing this classification, in terms of classification accuracy and processing performance. In this chapter, performance of the different machine learning algorithms are evaluated for the created fingerprint database. Since, the data has been collected from known places, each record in the dataset includes place label in itself. Therefore, our smartphone sensor and Wi-Fi dataset is appropriate to classify with supervised machine learning models.

5.1. Creating Fusion Dataset

The data collection application provides a fusion dataset as well as raw data. Raw data contains sensor data with different lengths. For example, for the 30-second-long dataset, there are approximately

- 300.000 records for light sensor data,
- 300.000 records for proximity sensor data,
- 70.000 records for magnotemeter data,
- 300 rows for Wi-Fi data.

The differences between the data record counts are caused by sampling a period of sensors.

As we can see in the raw data graphics, these records may include outliers. To eliminate these outliers, we applied a sliding window median function to raw data. This function provides equal-length sensor data for all sensors. It means, if we take w as window size, the median of the first w values of raw data records is calculated and added to the new array. Then, if we take s as shift size, the median value of the next w records is calculated by beginning from the (s+1)th record. This calculation needs to be maintained until the end of the raw data records. We divided raw data records in 1-second-length windows, and applied a 0.5-second length segments as shift size. Duration of the data recording for a place is the same for each sensor. They all are activated by start button and deactivated by stop button at the same time. Hence, the duration (measured in seconds) is common. The given formulas are used for calculating window size and shift size, respectively.

$$WindowSize = \frac{Number\ of\ raw\ data\ records}{Duration\ in\ seconds} \tag{5.1}$$

$$ShiftSize = \frac{Number\ of\ raw\ data\ records}{Duration\ in\ seconds * 2} \tag{5.2}$$

Preprocessed data can be acquired by applying this function. This function is operated by data collection application after the process of gathering raw data. A preprocessed data table is saved on local memory of a phone. The preprocessed data table is used in data classification, which will be explained in detail below.

As to Wi-Fi, it has been processed differently due to its different characteristics.

While other sensors produce only numeric data, Wi-Fi data includes more than one component in different data types for scanning. It includes SSID, BSSID and RSSI values at least. As we mentioned above, we did not install any specific access point and used only existing ones. During our data collection phase, it was observed that the owners of the access points, such as neighbours, may change the settings of their access points. SSID is the most changeable setting, of course. Someone may want to change the visible name of their access points frequently. On the other hand, the existing access points may vary day after day. If specific Wi-Fi AP's are not used in this type of localization, Wi-Fi becomes depended on many changeable situations. They can be deactivated by their users at any time. Thus, in the home environment, we create Wi-Fi fingerprint by the following methodology:

During a data collection period, each received Wi-Fi signal is recorded in cache memory with their BSSID, SSID and RSSI information for each location. At the end of the collection phase –via stop click– received signals are sorted according to their power in descending order. Then, BSSID's of Wi-Fi access points that have the strongest signals are appended on each other. Obtained strings are assigned to places as Wi-Fi labels.

In the office environment, since there are only Wi-Fi APs mostly, Wi-Fi data is handled as numeric data. We assigned RSSI values with the methodology of other numeric sensor data.

5.2. Algorithms and Techniques

To find the most appropriate supervised machine learning algorithm, AdaBoost-Classifier, DecisionTreeClassifier, SVC, GaussianNB, and KNeighborsClassifier algortihms were used to train our dataset. In the following section, these algorithms will be briefly explained.

5.2.1. Decision Tree Classifier

It is a predictive machine learning model. The components of a decision tree model can be similar to those of a real tree. A root, branches, internal nodes and terminal nodes are the essential components. Each node of decision tree demonstrates an attribute. The branches show the probabilistic values of the attributes in the observed samples. The terminal nodes represent the dependent variable.

The dependent variable can also be referred to as the class attribute, which is the attribute to be predicted. Since its value is attached to the values of all of the other attributes, it is named as dependent. Other attributes supporting the predicted value of the dependent variable are the independent variables in the dataset.

Among the data classiffication models, the tree is much simpler, comprehensible model [36]. It is constructed based on the probabilities of variables, thus the tree models may vary from each other in terms of nodes placement. If the internal node and terminal nodes look very similar on each side of the root, then the tree is referred to as a balanced tree. Generally, a balanced tree is a good model. If the subtrees in the decision tree only have one solution, then all of the subtrees are degraded to simple solutions and the construction process may be improved without any change in the result. In literature, there are remarkable studies by Ross Quinlan in this field [37] The prominent algorithms of Quinlan are ID3 and C4.5 [38]. The ID3 algorithm was first created, and the C4.5 is the naturally improved version of the previous algorithm [39]. The principle idea behind the both algorithms is information theory.

5.2.2. K-Nearest Neighbor

According to this algorithm, feature extraction during classification is used to find the closeness of the new sample to be classified according to k of the previous samples [43].

For example, to find an accurate class for a new sample with k = 3, the nearest three of the previously classified samples are taken. If these k samples belong to a class, the new sample also belongs to the same class. To find the closeness between different samples, the Euclidian distance can be used.

It is a simple machine learning algorithm. However, it is preferred in small-scale learning processes because it performs the operations on large-scale data with a long time basis.

5.2.3. Ada Boost Classifier

In a boosting classifier, a lot of weak learners are combined to form one powerful learner. In this method, predictors are trained sequentially [42].

In the Adaboost (Adaptive Boosting) algorithm, an estimator takes underfit training situation into account, which is performed by another estimator previously [41].

To build an Adaboost classifier following steps are applied: the first classifier is trained on the training set and makes predictions. Then, relative weight of incorrectly categorized training data is increased. The second classifier is trained with the increased relative weight of incorrectly categorized training data. After the second classifier predicts, the weights are updated again. This process continues until all classifiers in the Ada-Boost Classifier are trained.

5.2.4. Gaussian Naïve Bayes

This algorithm is based on Bayes Theorem. It is a widely used supervised learning model since it is powerful on noisy data and easy to implement [44]. It can be performed on large and more complicated databases. Despite of its plain design and simplified assumptions, Naive Bayes classifier gives much better results than expected in real world situations.

5.2.5. Support Vector Machines

They are supervised learning models and widely used for classification and regression analysis.

The SVM learning algorithm creates a model that assigns new samples to a category by applying a non-probabilistic binary linear classification method.

SVMs can efficiently perform both linear and non-linear classification with a kernel function [40].

5.3. Benchmark Model

In the literature, several comparative study investigates performance of different machine learning methods for indoor positioning.

First, Bozkurt *et al.* [18] provides an evaluation of algorithms according to performance of the classiffication for indoor positioning. For this purpose, Nearest Neighbour (NN), Sequential Minimal Optimization (SMO), J48, Nave Bayes and BayesNet algorithms were comparatively tested. Experiments were performed by using WEKA library. They used UJIIndoorLoc database, which can be downloaded from UCI Machine Learning Repository. According to their test results, by using the whole dataset, J48 gives the best accuracy (99.89%). UJIIndoorLoc dataset only includes Wi-Fi RSSI data as an attribute.

Zhang *et al.* [33] proposes combining a grid search based kernel support vector machine with principle component analysis. Principle component analysis reduces dimension of measurements. They used grid search for fine tuning the support vector machine algorithm. They compared K-Nearest Neighbour, Back Propagation Neural Network and Support Vector Machine based methods on their RSSI-based dataset for doing indoor localization. They have collected RSSI samples from 16 different Aps. They performed a comparison by using iteratively 30, 50, 70 samples as training dataset and remain samples as test dataset. Localization performance is evaluated in terms of computational efficiency and accuracy. They are compared average location errors in meters. Their proposed Support Vector Machine algorithm with principal component analysis is better than K-Nearest Neighbour, Back Propagation Neural Network and classical Support Vector Machine algorithms.

Tariq *et al.* [35] compares different machine learning classifiers performance on capacitive sensor-based indoor localization system. They have done their tests in a 3m $\times$ 3m room. Their performance metrics are localization accuracy, average distance error, precision and recall. In addition to that, they assess the contribution of training data size on localization accuracy with different algorithms. Bayes Net, K-Nearest Neighbours, Support Vector Machine, Random Forest and Adaptive Boosting algorithms were evaluated.

The common point of these studies is the use of data accuracy as an evaluation metric in each case. The performance of algorithms varies for different size of training samples. Contrary to our approach, their studies are only one technology-based localization. Fodero *et al.* [30] combines two different technologies, Wi-Fi and magnetic field. They collected data with a smartphone from indoor locations. They used Deep Neural Network algorithm for the classification of indoor location dataset.

5.4. Our Evaluation Methodology

Given machine learning algorithms were performed on each dataset separately for three different indoor areas where data collected from (lounge, living room, office) since we have focused on positioning in a room.

5.4.1. Evaluation Metrics

Machine learning models evaluation must be performed according to the characteristic of the dataset. In this study, our dataset includes ambient sensors and Wi-Fi data as attributes and it is a multiclass dataset since there are five places in each dataset for different rooms. In accordance with a multi-class dataset, accuracy was used for performance indicator to find the best supervised machine learning model among evaluated methods. Accuracy can be measured with metrics such as True Positive (TP), False Positive (FP), False Negative (FN) and True Negative (TN). These values are obtained from a confusion matrix, which shows the number of correctly or incorrectly predicted data points. Accuracy is the possibility of the classifier can correctly predict positive and negative samples.

Comparison of different supervised machine learning models has been performed based on given evaluation metrics in below:

- Classification accuracy on training dataset
- Classification accuracy on test dataset
- Process time of data training and testing.

A free software machine learning library Scikit-learn [45] was used for implementation by Python programming language.

5.4.2. Data Pre-processing

At the beginning, each numerical feature have been normalized by using Min-Max Scaler. This function transforms features by scaling each feature individually to a given range.

Then categorical variables are converted to numerical variables by using the one-hot encoding scheme. Generally, learning models takes numerical inputs as features. Using one-hot encoding scheme for transforming categorical variables to numerical variables is common process. For a possible value in a categorical feature, a dummy variable is created by one-hot encoding.

5.4.3. Shuffling and Splitting Data

After the data pre-processing, dataset was shuffled and splitting data to obtain training data and test datasets. Each dataset was shuffled for for each room separately. 80% of the data is used for training and 20% for testing by random sampling for each each room separately.

5.4.4. Model Evaluation

At this part, a prediction function was implemented. This function;

- Trains the chosen learner on the training data and calculates the process time,
- Performs prediction on the test dataset,
- Records the total prediction time,
- Calculates the accuracy score for both the training subset and testing subset.

This function is called for 5 different learning models by using Sklearn library which are given below;

- AdaBoostClassifier(random state=10)(ABC)
- DecisionTreeClassifier(random state=10)(DTC)
- SupportVectorClassifier(random state=10)(SVC)
- GaussianNaiveBayes()(GNB)
- KNeighborsClassifier(number of neighbors=5)(KNN)

Random state parameters are used as a seed in random number generator. Using a same number as random state while a classifier is running guarantees that produces outputs are same for each classification. *Number of neighbors* parameter is used as for determining how many neighbor is used in comparison of distances to current point at each iteration.

These classifiers were evaluated by changing the learning sample sizes. Learning data size was calculated the number of records equal to 20%, 50%, and 100% of the training data, and used each training set separately in the prediction process to observe their effects to model performance. With this method, we would like to show that how much training data is required to obtain the highest accuracy. As to, training and predicting time comparison between different algorithms, it could be said that, SVC spends the longest time during training and prediction. Increasing the number of samples for each algorithm increases the processing time. ABC runs longer on the training set than on the test set. KNN spends more time on prediction than on training.

The initial results are given below in Figure 5.1, Figure 5.2, Figure 5.3 for each room.

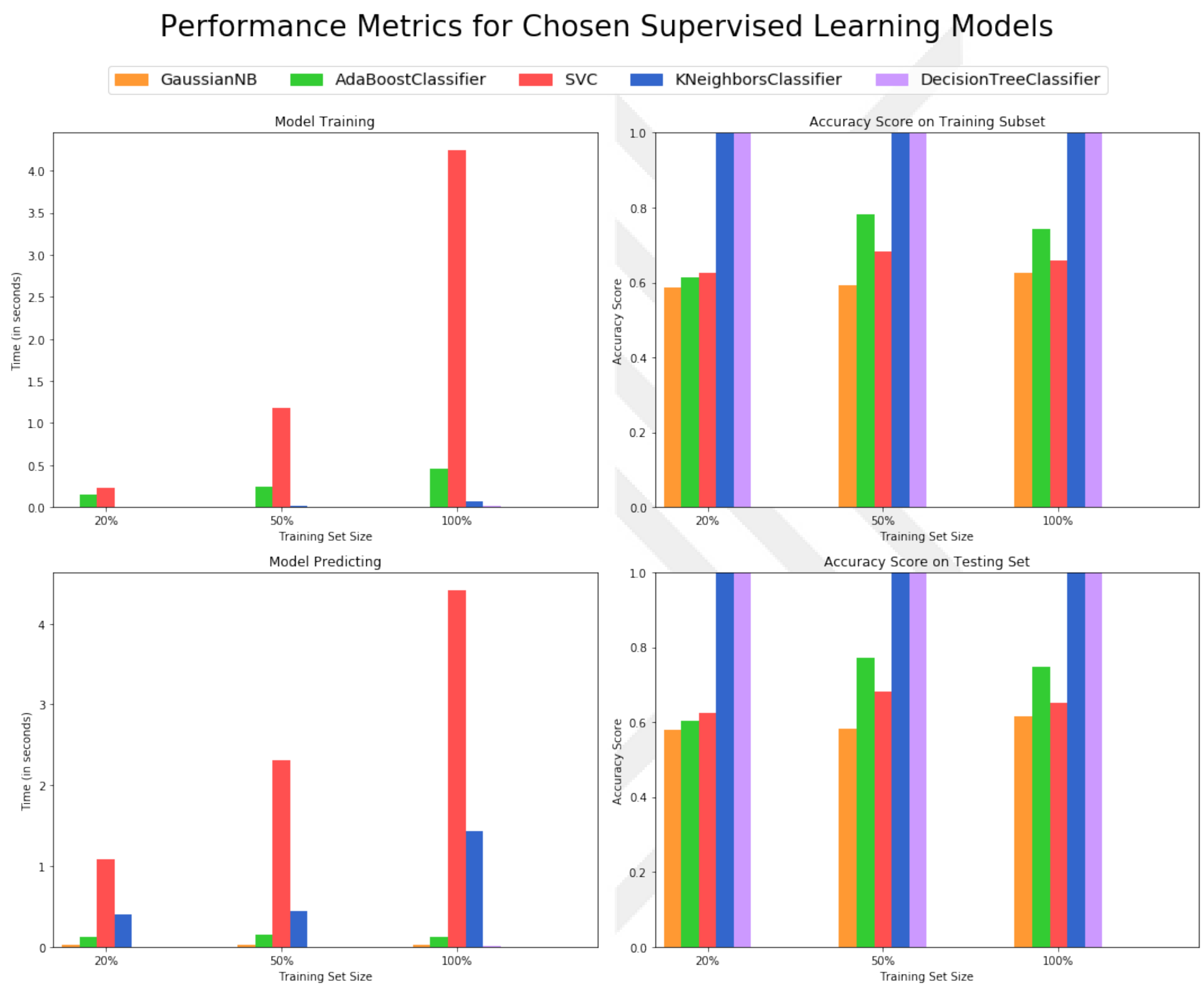

Figure 5.1: Model Evaluation with lounge data

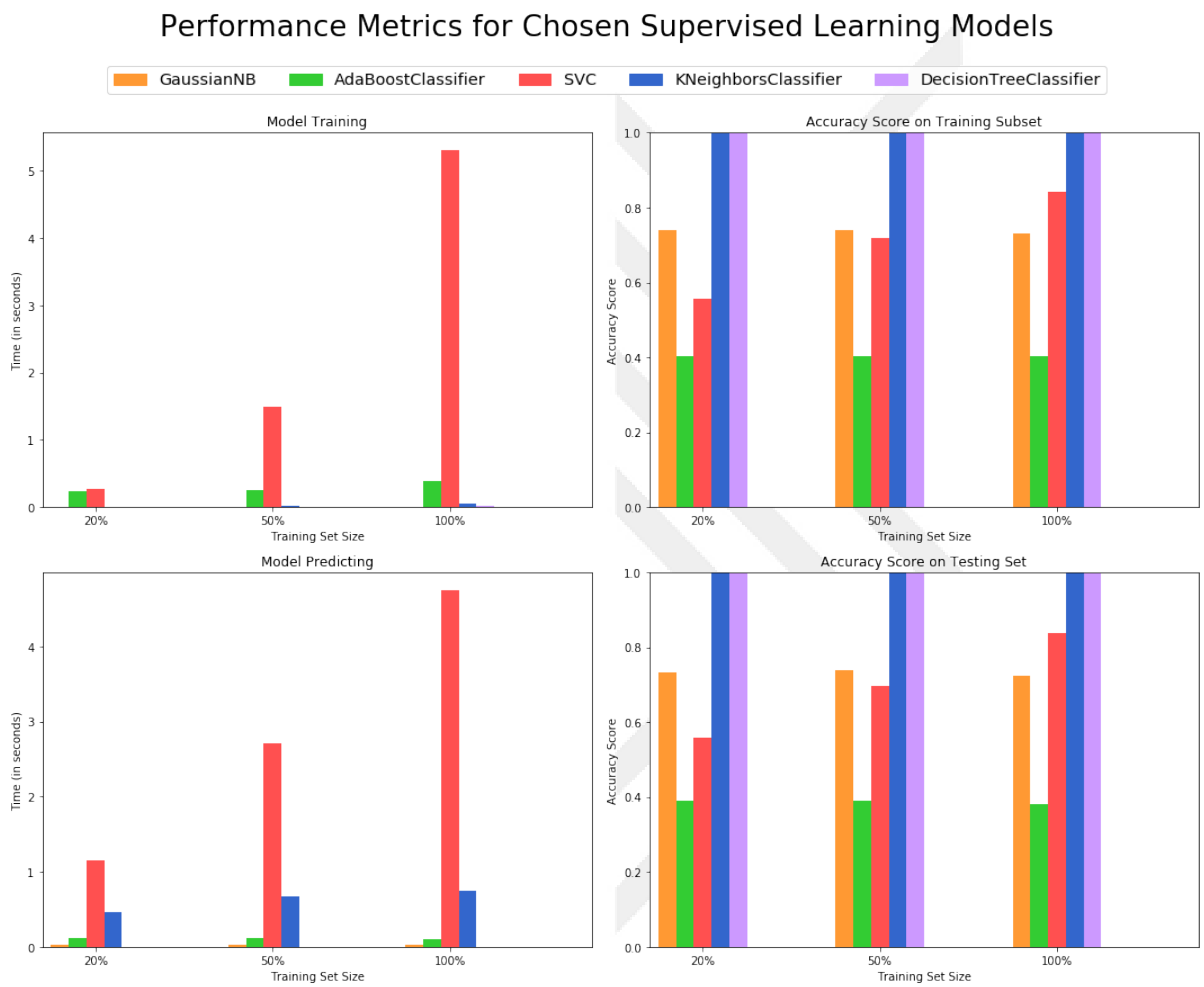

Figure 5.2: Model Evaluation with living room data

Figure 5.3: Model Evaluation with office data

The given figures show that, for the Decision Tree Classifier and KNN, training dataset volume does not affect the accuracy metric after 50% of the data. Also we can say that, enhancing training dataset increase the accuracy except the AdaBoostClassifier.

In addition, learning curves are created for each model to observe training score and cross-validation score depending on the training size. It can be seen in Figures 5.4, 5.5, 5.6 that, the validation score could be increased with more training samples in all models except AdaBoostClassifier. In addition that, DesicionTreeClassifier gives the highest validation score with the lowest training examples.

According to given figures above, it can bee seen that, Decision Tree Classifier is the best according to accuracy and processing time. Hence, it has been chosen model for indoor positioning with room level data.

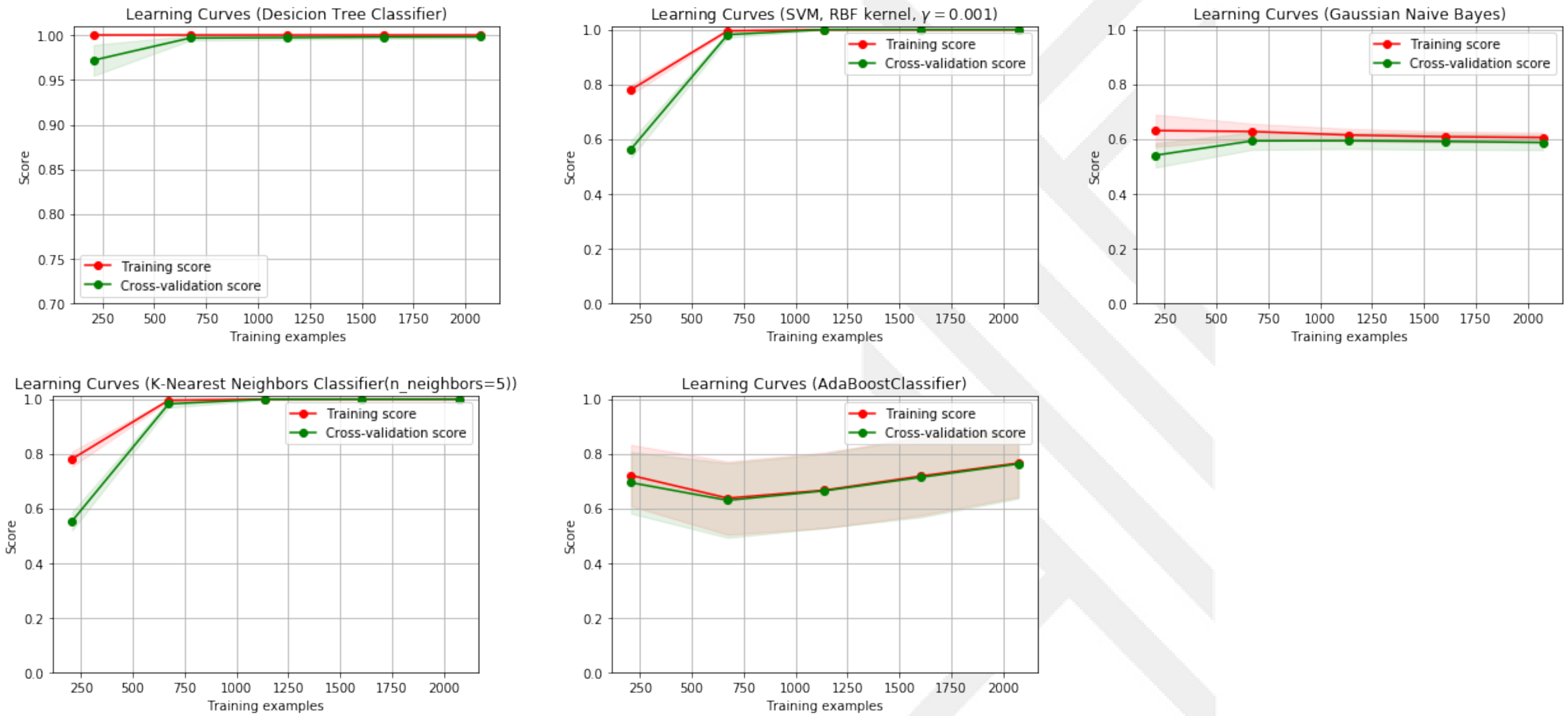

Figure 5.4: Learning curves for lounge data

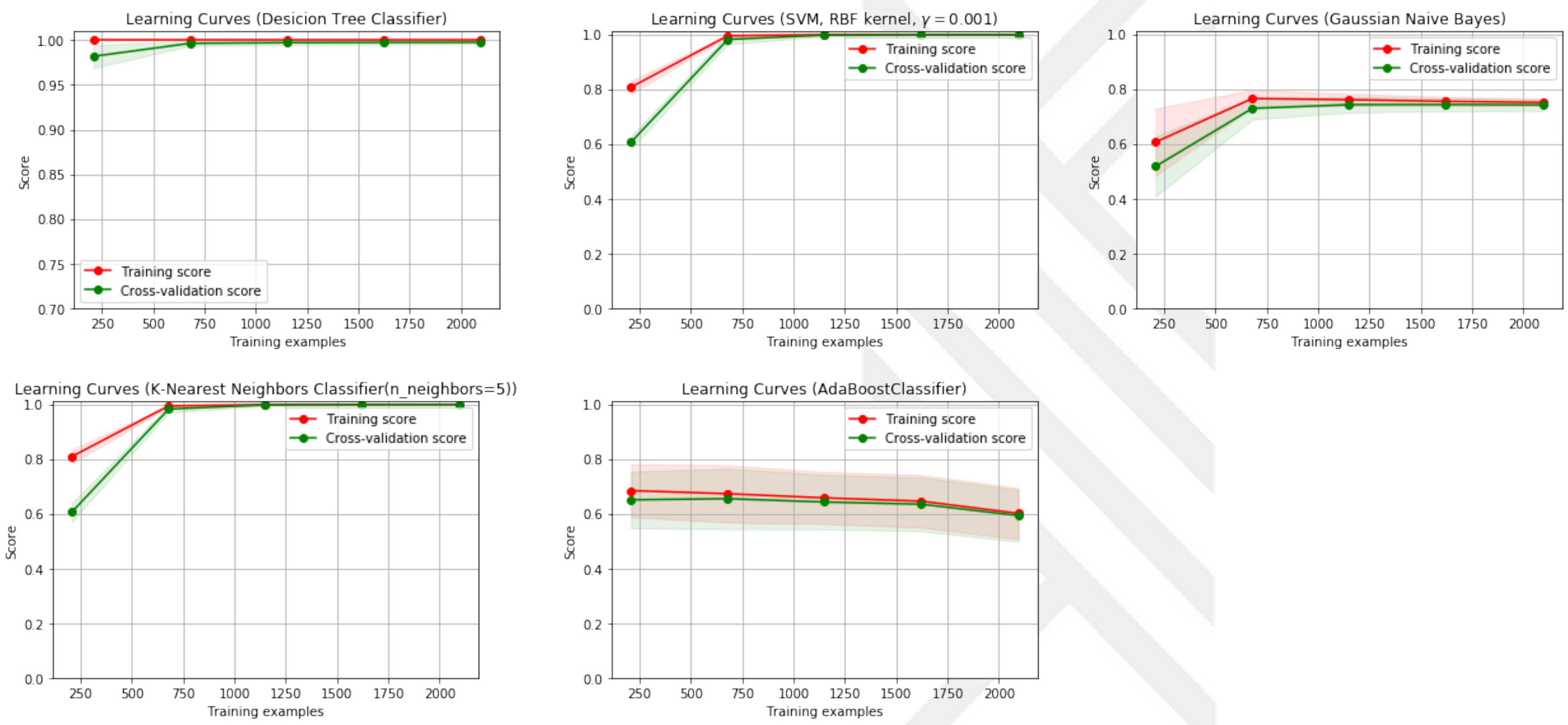

Figure 5.5: Learning curves for living room data

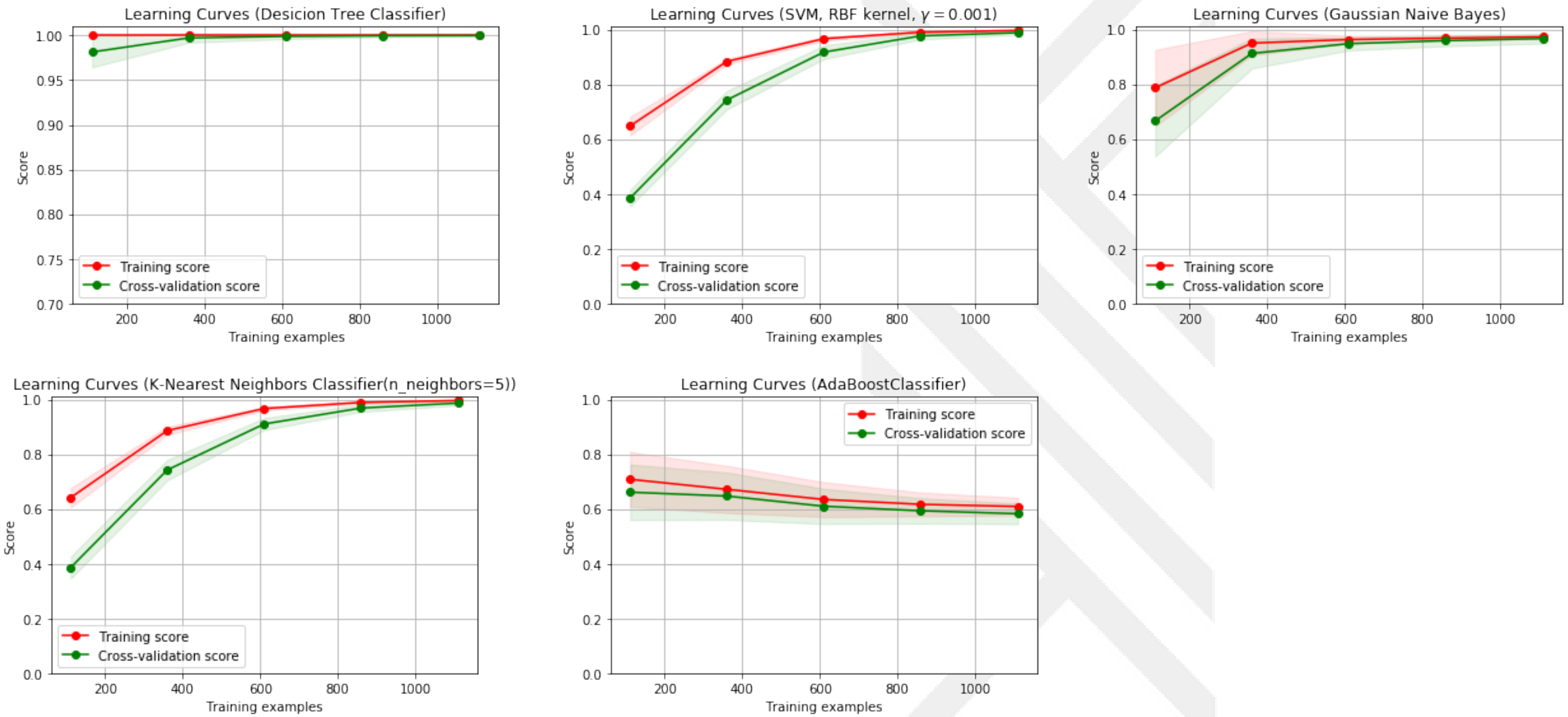

Figure 5.6: Learning curves for office data

5.5. Refinement of the Best Model

Finding optimal parameters and optimizing model is an inevitable part of supervised learning. Thus, fine tuning on parameters and feature importance analysis was performed on our best model, namely Desicion Tree Classifier.

5.5.1. Improving Results with Choosing the Best Model

We have implemented a grid search algorithm to optimize the parameters of decision tree classifier in order to improve the localization accuracy. We used GridSearchCv algorithms in ScikitLearn [45]. For this aim, the best parameter was searched for maximum depth of tree model.

From 3 to 10 (possible depth level numbers for the decision tree model) values were given for maximum depth parameter in grid search algorithm. Finding the most appropriate maximum depth parameters protects model from overfitting. Grid search algorithm has returned the maximum depth value equals to 6 for the best model. To compare accuracy scores of optimized model and unoptimized model, prediction process was done with the best classifier model by using fine tuned maximum depth parameter. In grid search algorithm we used all training and test set without any data reduce. A comparison is made between predictions by using initial and final models, and it has been observed that the accuracy scores did not change in both cases.

5.5.2. Feature Importance

In this part, it has been aimed to determine which features provide the most predictive power which is the percentage of information in the class variable that can be explained by the features in the model. This is an important part to specify which sensors are more meaningful for distinguishing indoor locations from each other. By determining feature importance, Data Collection Application may be optimized by removing less important sensor data collection, then the power consumption of the application may be reduced.

As a result of this analysis, it is observed that the most predictive features for each room is different. Normalized weights for each room are shown in Figures 5.7, 5.8, 5.9.

Normalized feature weights were changed for each room due to difference between environmental conditions of the rooms where data is collected from. Cumulative feature weights represents cumulative sum of the features weights.

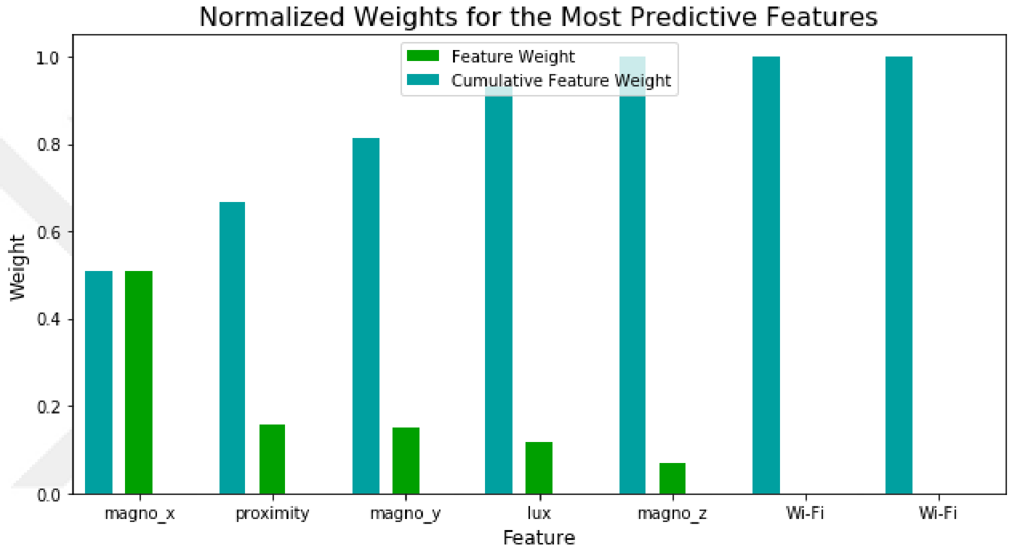

Figure 5.7: Normalized weights for features of lounge data

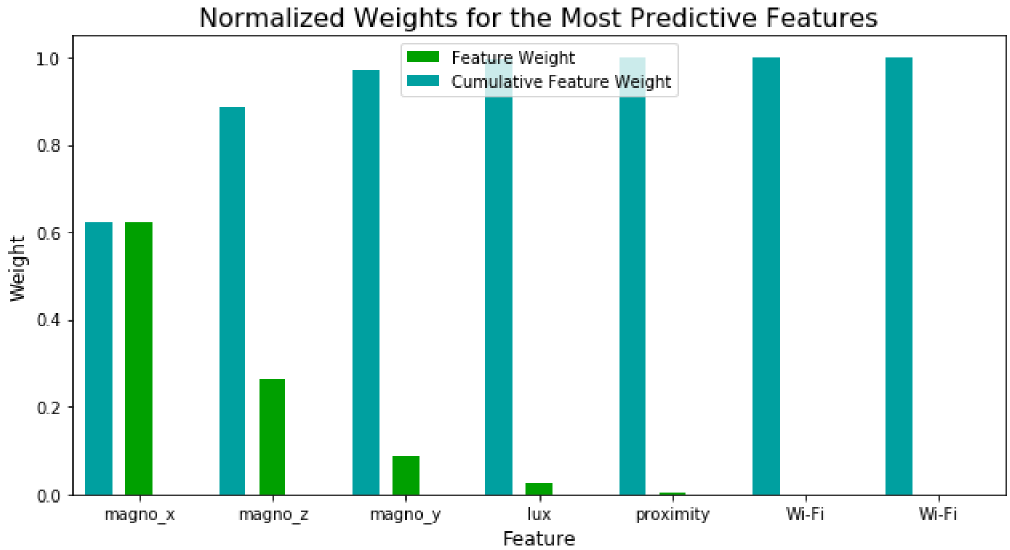

Figure 5.8: Normalized weights for features of living room data

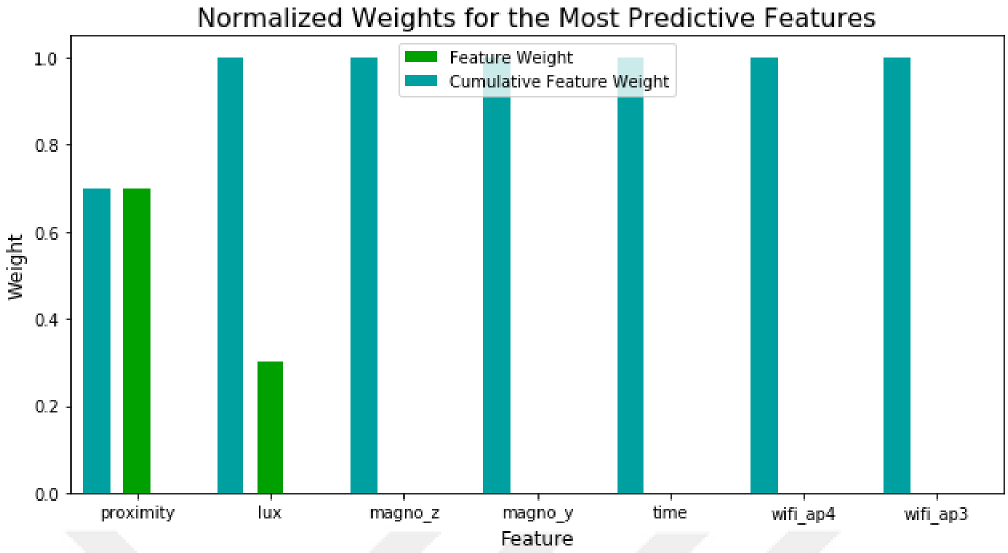

Figure 5.9: Normalized weights for features of office data

6. DATA CLASSIFICATION APPLICATION

After the analysis of machine learning models, an Android application is implemented to run J48 algorithm on the sensor data to show smartphones computational availability to realize classification.

6.1. Weka and J48

Weka is an open source machine-learning tool that includes a notable collection of algorithms to apply on different datasets given by users. It is a commonly used tool for preprocessing data, classification, and clustering since it is also called from independent Java codes.

Recently, since mobile devices have processors that can compete with computers (and their computational power reached considerably high levels), data mining on mobile devices became applicable. Hence, Weka moved onto the mobile platform especially on Android devices, which allow developers to implement Java codes. This progress makes it easier to make a classification, gathering, and it also helps rule mining functions on Android platforms.

In this thesis, a Weka library is deployed on an Android application to classify collected sensor data. The developed methods are explained in the following sections.

Interface of the application is shown in Figure 6.1.

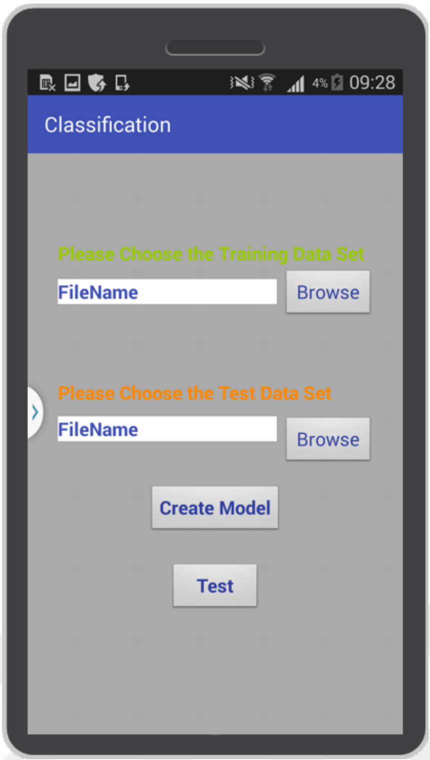

Figure 6.1: User interface of classification application

As it can be seen in Figure 6.2, by using this application, the user can choose the training dataset from the local memory of the phone.

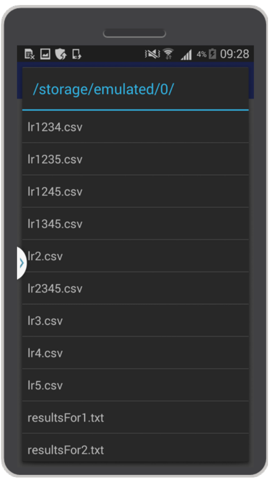

Figure 6.2: CSV files in local memory of phone which were created by Data Collection Application

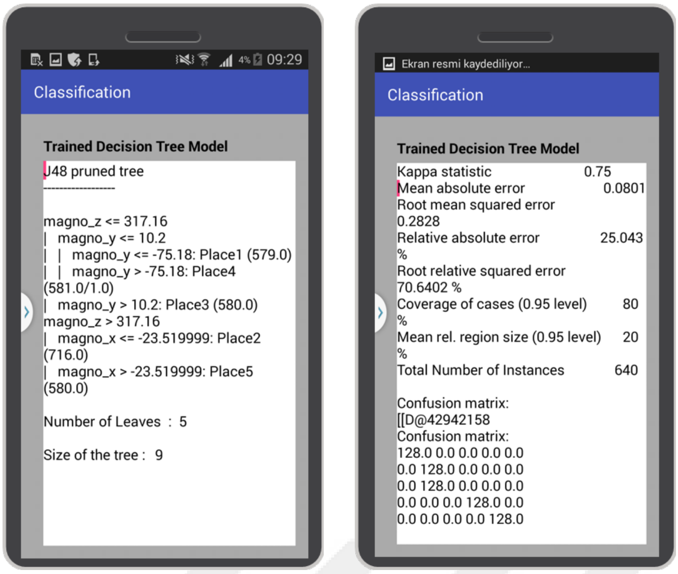

Figure 6.3: Sample screen-shots of a trained model and some test results for chosen train and test files

With this application, users can take the opportunity to classify data collected by the same phone cumulatively. Users can display the decision tree model and classification results in the screen as it is shown in Figure 6.3.

Trained decision tree models created by Data Classification Application are given in Figure 6.4, 6.5 and 6.6 for each location:

```
magno_z <= 0
|   proximity <= 620
|   |   magno_x <= -4.89
|   |   |   magno_y <= 0.42
|   |   |   |   magno_x <= -10.139999: Place4 (1637.0)
|   |   |   |   magno_x > -10.139999
|   |   |   |   |   magno_z <= -18.359999
|   |   |   |   |   |   lux <= 3428: Place5 (16.0)
|   |   |   |   |   |   lux > 3428: Place4 (715.0)
|   |   |   |   |   magno_z > -18.359999: Place3 (765.0)
|   |   |   magno_y > 0.42: Place3 (862.0)
|   |   magno_x > -4.89
|   |   |   magno_z <= -17.699999
|   |   |   |   lux <= 4978: Place5 (2541.0)
|   |   |   |   lux > 4978
|   |   |   |   |   magno_z <= -26.88: Place1 (288.0)
|   |   |   |   |   magno_z > -26.88: Place3 (129.0)
|   |   |   magno_z > -17.699999
|   |   |   |   lux <= 3685: Place4 (360.0/5.0)
|   |   |   |   lux > 3685
|   |   |   |   |   time <= 1503232320000: Place3 (355.0)
|   |   |   |   |   time > 1503232320000
|   |   |   |   |   |   proximity <= 10: Place1 (124.0)
|   |   |   |   |   |   proximity > 10
|   |   |   |   |   |   |   proximity <= 57: Place5 (3.0)
|   |   |   |   |   |   |   proximity > 57: Place2 (4.0/1.0)
|   proximity > 620
|   |   lux <= 4088
|   |   |   lux <= 3795: Place1 (164.0)
|   |   |   lux > 3795: Place3 (450.0)
|   |   lux > 4088: Place1 (1985.0)
magno_z > 0: Place2 (2558.0)
```

Figure 6.4: Trained model for lounge

As it is seen in Figure 6.4, decision tree model of lounge is based on magnetic field, lux, proximity and time attributes. Wi-Fi is not used as an attribute for classification in this room.

```
magno_x <= 20.16
|   magno_z <= -30.9
|   |   magno_y <= 4.62: Place4 (2475.0)
|   |   magno_y > 4.62
|   |   |   lux <= 3556: Place4 (141.0)
|   |   |   lux > 3556: Place3 (2614.0)
|   magno_z > -30.9
|   |   magno_x <= -8.22
|   |   |   magno_y <= -7.14
|   |   |   |   time <= 1503006000000: Place5 (14.0)
|   |   |   |   time > 1503006000000: Place2 (50.0)
|   |   |   magno_y > -7.14
|   |   |   |   magno_x <= -8.43: Place2 (1518.0/1.0)
|   |   |   |   magno_x > -8.43
|   |   |   |   |   time <= 1503060600000: Place5 (4.0)
|   |   |   |   |   time > 1503060600000: Place2 (115.0)
|   |   magno_x > -8.22
|   |   |   magno_y <= -3.06
|   |   |   |   magno_x <= -7.92
|   |   |   |   |   time <= 1503060600000: Place5 (209.0)
|   |   |   |   |   time > 1503060600000: Place2 (128.0)
|   |   |   |   magno_x > -7.92: Place5 (2247.0)
|   |   |   magno_y > -3.06
|   |   |   |   lux <= 3411: Place5 (142.0)
|   |   |   |   lux > 3411
|   |   |   |   |   magno_z <= -17.82: Place2 (774.0)
|   |   |   |   |   magno_z > -17.82
|   |   |   |   |   |   magno_x <= 0.66
|   |   |   |   |   |   |   SSID_key = 84:16:f9:bf:b5:2f18:28:61:
                                      36:04:0be4:18:6b:a4:30:d2: Place1 (1.0)
|   |   |   |   |   |   |   SSID_key = 84:16:f9:bf:b5:2f18:28:61:
                                      36:04:0b28:28:5d:18:5f:c1: Place2 (4.0/2.0)
|   |   |   |   |   |   |   SSID_key = 18:28:61:36:04:0b28:28:5d:
                                      18:5f:c184:16:f9:bf:b5:2f: Place1 (3.0/2.0)
|   |   |   |   |   |   |   SSID_key = 84:16:f9:bf:b5:2f28:28:5d:
                                      18:5f:c118:28:61:36:04:0b: Place3 (3.0/1.0)
|   |   |   |   |   |   |   SSID_key = 84:16:f9:bf:b5:2f18:28:61:
                                      36:04:0bc4:6e:1f:9e:f9:88: Place1 (0.0)
|   |   |   |   |   |   |   SSID_key = 18:28:61:36:04:0b84:16:f9:
                                      bf:b5:2fe4:18:6b:a4:30:d2: Place1 (1.0)
|   |   |   |   |   |   |   SSID_key = 18:28:61:36:04:0b84:16:f9:
                                      bf:b5:2f28:28:5d:18:5f:c1: Place1 (1.0)
|   |   |   |   |   |   magno_x > 0.66: Place2 (32.0)
magno_x > 20.16: Place1 (2614.0)
```

Figure 6.5: Trained model for living room

Figure 6.5 shows that, on contrary to the decision tree model of lounge, Wi-Fi is used as an attribute in decision tree model of living room while proximity does not take place in this model. Also, magnetic field, lux and time attributes are used for this room.

```
proximity <= 390
|   lux <= 3461: Place5 (1391.0/3.0)
|   lux > 3461
|   |   proximity <= 197
|   |   |   lux <= 4116: Place1 (129.0)
|   |   |   lux > 4116: Place3 (1387.0)
|   |   proximity > 197: Place1 (1258.0)
proximity > 390
|   proximity <= 487
|   |   lux <= 3229: Place4 (1388.0)
|   |   lux > 3229: Place2 (26.0)
|   proximity > 487: Place2 (1361.0)
```

Figure 6.6: Trained model for office

As it is seen in Figure 6.7, for the office, decision tree model is less complicated than the models of other rooms. Simplicity of the decision tree model of office coincides with our observation in the radarchart graphics given in Section 4.4.

Moreover, obtained decision tree models by Data Classification Application, completely overlapping with the feature importances of each room which are given in Section 5.6.2. The most important features are magnetic field, lux and proximity sensors data for decision trees of lounge and living room while there are only lux and proximity sensor data in the decision tree model of office.

6.2. Cross Validation Computation on Smartphone

When the training set includes the test set, the error rate of the prediction decreases defectively. Not using the entire dataset during training is a way out to handle with this problem. Some data can be excluded to use as test data. In this manner, the real performance of the trained model on new datasets can be assessed. This is the basic concept of the cross validation.

Here we used 10 fold cross validation, which is the most common kind of cross validation. In 10-fold cross validation, the dataset is divided into 10 subsets. One of them is used as the test set iteratively. In each iteration, except for the chosen subset, the other subsets are used together as a training set. Classification algorithm is run for the chosen subset and the prediction is made based on the training set. Then it repeats 9 more times by changing the test set with next the subset. Consequently, the average error rate is calculated according to predictions. The 10-fold cross validation method has been applied on each consolidated dataset to measure how successful a prediction could be made according to those models.

```
Results
======

Correctly Classified Instances      12942              99.8919 %
Incorrectly Classified Instances       14               0.1081 %
Kappa statistic                         0.9986
Mean absolute error                     0.0005
Root mean squared error                 0.0199
Relative absolute error                 0.1717 %
Root relative squared error             4.9714 %
Coverage of cases (0.95 level)         99.8919 %
Mean rel. region size (0.95 level)     20.0201 %
Total Number of Instances           12956
Confusion matrix:
[[D@430d9918
Confusion matrix:
2561.0 0.0 0.0 1.0 0.0
2.0 2558.0 2.0 0.0 0.0
0.0 0.0 2560.0 0.0 2.0
0.0 0.0 2.0 2704.0 2.0
0.0 1.0 0.0 2.0 2559.0
```

Figure 6.7: 10-fold Cross Validation for lounge

```
Results
======

Correctly Classified Instances       13070               99.8701 %
Incorrectly Classified Instances        17                0.1299 %
Kappa statistic                          0.9984
Mean absolute error                      0.0006
Root mean squared error                  0.0197
Relative absolute error                  0.1749 %
Root relative squared error              4.9342 %
Coverage of cases (0.95 level)          99.9236 %
Mean rel. region size (0.95 level)      20.0627 %
Total Number of Instances            13087
Confusion matrix:
[[D@42700e50
Confusion matrix:
2615.0 0.0 3.0 0.0 0.0
2.0 2616.0 0.0 0.0 0.0
3.0 0.0 2614.0 1.0 0.0
1.0 1.0 0.0 2616.0 0.0
0.0 5.0 0.0 1.0 2609.0
```

Figure 6.8: 10-fold Cross Validation for living room

```
Results
======

Correctly Classified Instances        6937               99.9568 %
Incorrectly Classified Instances         3                0.0432 %
Kappa statistic                          0.9995
Mean absolute error                      0.0004
Root mean squared error                  0.0132
Relative absolute error                  0.1207 %
Root relative squared error              3.2911 %
Coverage of cases (0.95 level)          99.9568 %
Mean rel. region size (0.95 level)      20      %
Total Number of Instances             6940
Confusion matrix:
[[D@4287dec0
Confusion matrix:
1387.0 0.0 0.0 0.0 1.0
0.0 1387.0 0.0 0.0 1.0
0.0 0.0 1387.0 0.0 1.0
0.0 0.0 0.0 1388.0 0.0
0.0 0.0 0.0 0.0 1388.0
```

Figure 6.9: 10-fold Cross Validation for office

As it is shown in Figure 6.7, 6.8 and 6.9, cross validation achieves almost 100% accuracy for each location. The reason behind this high accuracy rate is that although the cross-validation is performed within the dataset, there is still data collected from the same time from the same place.

To improve the reliability of cross-validation, we distinguished the learning set from the data of the test set. We used 1.,2.,3.,...(n-1). datasets to build a training model and applied it on n datasets for classification. This process was iterated n times until all datasets were once in the testing and computed the overall confusion matrix. All computations have been made on a smartphone. Results for all three rooms are given in Table 6.1, Table 6.2, and Table 6.3:

Table 6.1: Results of iterative cross-validation for lounge

Dataset #	**Correctly Classified**	**Incorrectly Classified**
1	79,6774	20,3226
2	100	0
3	80	20
4	80,1441	19,8559
5	0	100
6	79,6774	20,3226
7	80	20
8	80	20
9	99,7015	0,2985
10	99,8387	0,1613
11	99,8858	0,1142
12	99,8592	0,1408
13	79,875	20,125
14	99,8387	0,1613
15	59,8611	40,1389
16	68,5915	31,4085
17	80	20
Average	**80,4088**	**19,5912**

Table 6.2: Results of iterative cross-validation for living room

Dataset #	**Correctly Classified**	**Incorrectly Classified**
1	62,0635	37,9365
2	20	80
3	100	0
4	80,4348	19,5652
5	99,8387	0,1613
6	79,8611	20,1389
7	100	0
8	100	0
9	99,8864	0,1136
10	99,8889	0,1111
11	81,3415	18,6585
12	84,5588	15,4412
13	100	0
14	99,7222	0,2778
15	99,8507	0,1493
16	100	0
Average	**87,9654**	**12,0346**

Table 6.3: Results of iterative cross-validation for office

Dataset #	Correctly Classified	Incorrectly Classified
1	99,8361	0,1639
2	80	20
3	100	0
4	60,6349	39,3651
5	98,5714	1,4286
6	80,4412	19,5588
7	100	0
8	100	0
9	100	0
10	58,9063	41,0938
11	99,1935	0,8065
Average	**88,8712**	**11,1288**

As it can be seen from the results, since the data received at close intervals show similar characters, the accuracy is higher when this data is tested. Hence, enhancing collected dataset by getting samples from different times of a day is going to increase the accuracy of classification. With the current dataset, the classification method reaches 85,7485% accuracy.

6.3. Effect of Collected Dataset Amount to Accuracy

To evaluate the effect of enhancing the training set amount on classification accuracy, a different experimental test is applied.

In this test, training dataset has been collected from five chosen places in a room at the evening times of different days. Collection times of the data are given in Table 6.4.

Table 6.4: Times of data collection

Time of Data Collection	**dataset**
20.12.2017 23:05	D1
22.12.2017 23:58	D2
24.12.2017 00:17	D3
24.12.2017 03:37	D4
24.12.2017 21:15	D5
24.12.2017 23:25	D6
26.12.2017 21:25	D7
26.12.2017 22:31	D8
27.12.2017 00:19	D9
27.12.2017 19:41	D10
27.12.2017 21:30	D11

The dataset which is named as D11 was used as test data. And rest of datasets were used for training. Classification application was run iteratively. For the first iteration, only D1 data was used as training data. In the following steps, the next dataset has been appended on previous data. For example, in the second iteration our training data was D1+D2 and, in the third iteration our data training data was D1+D2+D3. In each iteration, our dataset enhanced cumulatively with the collected datasets and the classification application run for classifying test data (D11). We recorded the correctly classified rate for all classification. With this method, we aimed to demonstrate how much we have to train to reach the maximum accuracy for classification.

Since the room conditions were stable during the data collection, the values read by sensors in datasets are similar. Therefore, correctly classified rate is getting higher while enhancing the training data cumulatively. The results are given in Figure 6.10. We reached almost 100% accuracy at end of the 8th iteration.

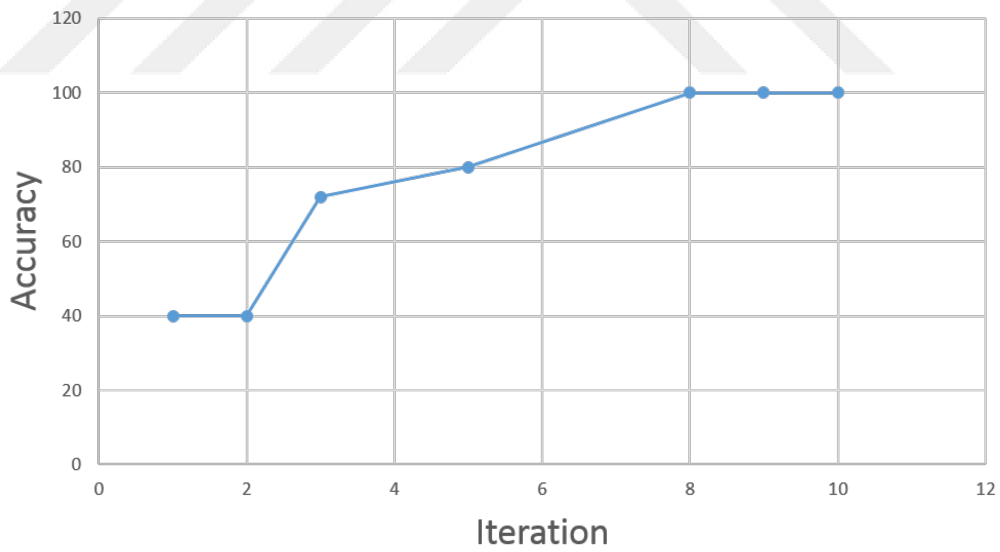

Figure 6.10: Classification results for the different size of datasets

Results of the test shows that, under the stable conditions with regards to level of light and position of furniture around, when data is collected from the same location for eight times , the classification accuracy can reach almost 100% accuracy.

7. CONCLUSIONS

Beyond the indoor localization, we presented a study on investigation of environmental features to distinguish different places from each other in a room. We used ambient light and proximity sensors since indoor places have different lumination at different corners and includes lots of stable objects around places where a user places his/her smartphone. In addition, like lots of studies in literature, we have collected magnetic field sensor data and Wi-Fi data to combine them with ambient sensors data. We have marked five different points for three different locations (two houses, one office). By using different ambient sensors, we tested whether the devices could distinguish the characteristics of these locations.

In this thesis, we first offer a sensor data collection application that can be used in everyday life. With this application running on the Android platform, we aimed to identify place of devices in closed and physically small spaces, like our homes and business locations. We have collected data from five pre-determined places in three different rooms. On different days, we collected 1-minute-long data at certain hours of the day from these points. When collecting data, we have worked in the ordinary circumstances of the places. That is, we have not made any special changes in the position of the objects in the rooms. The data collection application we have developed presents the raw and processed state of the collected sensor data.

At the second part, we have visualised collected raw data to explore sensor data and Wi-Fi. We made observations on raw data to see different values produces from different locations according to their environmental differences. The graphs we obtained show that the sensor data has a certain characteristic for each position, depending on the time the data was collected. Since the ambient light changes at different times of the day, we decided to take the time as a component in addition to the sensor data.

In the data collection application, raw data is also processed to filter out the noise. In this process we have implemented the median function on the one-second window. We combined all the sensor data in a fusion and obtained a fusion database. We have evaluated our fusion database by using machine learning concept which is a hot research topic. We thereby created data sets that we can analyze with several supervised machine learning algorithms.

At the evaluation of the machine learning models part, we measured the performance of five different machine learning algorithms for each room individually. We determined the best classifier as the Decision Tree Classifier in terms of accuracy and process time. In this part, firstly we pre-processed our dataset by normalizing numerical features and converting categorical features to numerical values with one-hot encoding scheme. Then we shuffled dataset and splitted into test and training datasets. We performed Decision Tree Classifier, Ada Boost Classifier, K-Nearest Neighbor, Support Vector Machine, Gaussian Naïve Bayes algorithms on separately 20% 50% 100% on training dataset to create model. Then we compared accuracy on training set, accuracy on test set, and process times of each algorithms. In addition to that, we created learning curves for each algorithms. The best performance was obtained with the Decision Tree Classifier with 98% accuracy rate on 20% of training samples. To optimize the best classifier we applied some fine tuning methods. By using the grid search algorithm, we found the best parameters for the maximum depth in the tree model. We performed the Decision Tree Classifier with the optimal maximum depth value and compared the accuracy scores with the unoptimized model. Accuracy score did not change in both models. Predictive power of used features were also investigated to specify which sensors are more meaningful for distinguishing indoor locations from each other. Weighted feature importances are changed by environmental conditions of indoor locations where data collected from. According to the results we obtained, proximity and light sensor data in an office environment were the most decisive features, while magnetic field data and Wi-Fi were the most decisive features in home environment. Since the most predictive feature changes from a place to another place, we decided to use each feature in our proposed system. We have shown that by combining different ambient sensors and Wi-Fi, it is possible to distinguish certain places where users can usually place their phones in daily life in a room.

Unlike the studies in the literature, we used light and proximity sensors in addition to Wi-Fi and magnetic field sensor, which are the most commonly used data sources in indoor positioning. We used these sensors to investigate whether or not the interior spaces could be characterized. Algorithms we performed for classification provide remarkable accuracy. However, predictive power of used features did not show a stable performance in every room. Thus, to prove contribution of ambient light, proximity and magnetic field sensor data, more experiments are required in more rooms with different characteristics.

At the last part, we developed an application to create a decision tree model by using Weka library on Android. By using this application, a decision tree model was created for each location. To test the model success, firstly, 10-fold cross validation was applied for each dataset. As a result of these tests, we have seen that the models are close to 100% accuracy. However, since there might be samples from the test set in the learning set, we have applied an iterative test. We left a certain data set, collected at a time from the learning set, and built a decision tree model with the rest of data. We then tested the remaining dataset. This method was iteratively applied on each set of data from the learning set. As a result of this iterative test, we achieved an average accuracy of 85%. In addition, we demonstrate that, by enhancing size of collected data we can reach almost 100% classification accuracy.

In this study, data collection, pre-processing, building the decision-tree model and making classification according to the created model are all done on the mobile device. In this way, we provide a basis for smartphones to support users in their daily lives by taking automatic actions, which is our main motivation in this thesis. For future work, we aim to enable smartphones to automatically perform certain operations at certain times and at certain locations according to the habits of their users.

REFERENCES

1. Ballagas, Rafael, *et al.* “The smart phone: a ubiquitous input device.” *IEEE Pervasive Computing 5.1: 70-77*, 2006.

2. Salamah, Ahmed H., *et al.* “An enhanced Wi-Fi indoor localization system based on machine learning.” *Indoor Positioning and Indoor Navigation (IPIN), International Conference on. IEEE*, 2016.

3. Liu, Junjie. “Survey of wireless based indoor localization technologies.” *Department of Science Engineering, Washington University*, 2014.

4. Doiphode, Siddhesh, J. W. Bakal, and Madhuri Gedam. “Survey of Indoor Positioning Measurements, Methods and Techniques.” *International Journal of Computer Applications 140.7*, 2016.

5. Pratama, Azkario Rizky, and Risanuri Hidayat. “Smartphone-based pedestrian dead reckoning as an indoor positioning system.” *System Engineering and Technology (ICSET), 2012 International Conference on. IEEE*, 2012.

6. Lim, Chin-Heng, *et al.* “A real-time indoor Wi-Fi localization system utilizing smart antennas.” *IEEE Transactions on Consumer Electronics 53.2*, 2007.

7. Chung, Jaewoo, *et al.* ”Indoor location sensing using geo-magnetism.” *Proceedings of the 9th international conference on Mobile systems, applications, and services. ACM*, 2011.

8. Xia, Hao, *et al.* “Using multiple barometers to detect the floor location of smart phones with built-in barometric sensors for indoor positioning.” *Sensors 15.4: 7857-7877*, 2015.

9. Galván-Tejada, Carlos E., *et al.* "Infrastructure-less indoor localization using the microphone, magnetometer and light sensor of a smartphone." Sensors 15.8 (2015): 20355-20372.

10. Mulloni, Alessandro, *et al.* "Indoor positioning and navigation with camera phones." *IEEE Pervasive Computing 8.2*, 2009.

11. Google Inc., *Google Maps.* 2017. https://www.google.com/maps/about/partners/, accessed at July 2018.

12. Mazilu, Sinziana, and G. Troster. "A study on using ambient sensors from smartphones for indoor location detection." *Proceedings of 12th Workshop On positioning, navigation and communication (WPNC). IEEE.*, 2015.

13. Verhaevert, Jo, and Dries Van Baelen. "Localization at room resolution with bluegiga bluetooth low energy." *Antennas Propagation Conference (LAPC), 2016 Loughborough. IEEE*, 2016.

14. Liu, Hui, *et al.* "Survey of wireless indoor positioning techniques and systems." *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 37.6: 1067-1080*,2007.

15. Indoor Atlas Ltd. *Indoor Atlas.* 2017. http://www.indooratlas.com, accessed at July 2018.

16. Chen, Zhenghua, *et al.* "Fusion of Wi-Fi, smartphone sensors and landmarks using the Kalman filter for indoor localization." *Sensors 15.1: 715-732*, 2015.

17. Gu, Fei, Jianwei Niu, and Lingjie Duan. "WAIPO: A Fusion-Based Collaborative Indoor Localization System on Smartphones." *IEEE/ACM Transactions on Networking*, 2017.

18. S. Bozkurt, G. Elibol, S. Gunal and U. Yayan, "A comparative study on machine learning algorithms for indoor positioning," *2015 International Symposium on In-*

novations in Intelligent SysTems and Applications (INISTA), Madrid, 2015.

19. HTC Corporation. *HTC*. 2017. http://www.htc.com/us/smartphones/htc-u11/, accessed at July 2017.

20. IDC Research, Inc. *IDC*. 2017. http://www.idc.com/promo/smartphone-market-share/os accessed at July 2017.

21. NDTV Convergence Limited. *NDTV*. 2017. http://gadgets.ndtv.com/samsung-galaxy-note-ii-538, accessed at July 2017.

22. Capella Microsystems Inc. *Capella Microsystems*. 2017. http://www.capellamicro.com.tw/, accessed at July 2017.

23. Hssina, Badr, *et al.* "A comparative study of decision tree ID3 and C4. 5." *International Journal of Advanced Computer Science and Applications 4.2: 13-19*, 2014.

24. Ishwaran, H. Rao, J. S. Decision tree: introduction. In Kattan (ed.), *Encyclopedia of Medical Decision Making. Sage Publications: 323-328*, 2009.

25. Rebecca, Severino. *Radar Chart*. 2017. http://www.datavizcatalogue.com/, accessed at August 2017.

26. Chang, Tom, and John Canfield. "Integrated ambient light sensor and distance sensor." *U.S. Patent No. 8,705,015*, 2014.

27. Hariadi, Tony K., Achmad Khoirul Habib Juwito, and Anna Nur Nazilah Chamim. "Smartphone-based lux meter with decision support system." *Control System, Computing and Engineering (ICCSCE), 2017 7th IEEE International Conference on. IEEE*, 2017.

28. Fahim, Muhammad, *et al.* "Physical activity recognizer based on multimodal sen-

sors in smartphone for ubiquitous-lifecare services." *AFRICON, 2017 IEEE. IEEE*, 2017.

29. Kim, Han-Sol, Woojin Seo, and Kwang-Ryul Baek. "Indoor positioning system using magnetic field map navigation and an encoder system." *Sensors 17.3: 651*, 2017.

30. Zhang, Wei, *et al.* "DeepPositioning: Intelligent Fusion of Pervasive Magnetic Field and WiFi Fingerprinting for Smartphone Indoor Localization via Deep Learning." *Machine Learning and Applications (ICMLA), 2017 16th IEEE International Conference on. IEEE*, 2017.

31. Zhang, Jie, *et al.* "Large-scale WiFi indoor localization via extreme learning machine." *Control Conference (CCC), 2017 36th Chinese. IEEE*, 2017.

32. L. Zhang, N. Xiao, J. Li and W. Yang, "Heterogeneous Feature Machine Learning for Performance-Enhancing Indoor Localization," *IEEE 87th Vehicular Technology Conference (VTC Spring), Porto, Portugal. IEEE*, 2018.

33. Zhang, Lingwen, *et al.* "An efficient machine learning approach for indoor localization." *China Communications 14.11: 141-150*, 2017.

34. Cheng, Yen-Kai, Hsin-Jui Chou, and Ronald Y. Chang. "Machine-Learning Indoor Localization with Access Point Selection and Signal Strength Reconstruction." *Vehicular Technology Conference (VTC Spring), 2016 IEEE 83rd. IEEE*, 2016.

35. Tariq, Osama Bin, *et al.* "Performance of Machine Learning Classifiers for Indoor Person Localization With Capacitive Sensors." *IEEE Access 5: 12913-12926*, 2017.

36. Gharibian, Farnaz, and Ali A. Ghorbani. "Comparative study of supervised machine learning techniques for intrusion detection." *Communication Networks and Services Research, 2007. CNSR'07. Fifth Annual Conference on. IEEE*, 2007.

37. Quinlan, J. Ross. "Induction of decision trees." *Machine learning 1.1: 81-106*, 1986.

38. Quinlan, J. Ross. "Bagging, boosting, and C4. 5." *AAAI/IAAI, Vol. 1.*, 1996.

39. Quinlan, Ross. "Data mining tools See5 and C5. 0.", 2004.

40. Lu, Shu-Xia, Jie Meng, and Gui-En Cao. "Support vector machine based on a new reduced samples method." *Machine Learning and Cybernetics (ICMLC), 2010 International Conference on. Vol. 3. IEEE*, 2010.

41. Wu, Shuqiong, and Hiroshi Nagahashi. "Parameterized adaboost: introducing a parameter to speed up the training of real adaboost." *IEEE Signal Processing Letters 21.6: 687-691*, 2014.

42. Friedman, Jerome, Trevor Hastie, and Robert Tibshirani. "Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors)." *The annals of statistics 28.2: 337-407*, 2000.

43. Peterson, Leif E. "K-nearest neighbor." *Scholarpedia 4.2: 1883*, 2009.

44. John, George H., and Pat Langley. "Estimating continuous distributions in Bayesian classifiers." *Proceedings of the Eleventh conference on Uncertainty in artificial intelligence. Morgan Kaufmann Publishers Inc.*, 1995.

45. Pedregosa, Fabian, *et al.* "Scikit-learn: Machine learning in Python." *Journal of machine learning research 12.Oct: 2825-2830*, 2011.

APPENDIX A: RADARCHARTS FOR LIVING ROOM AND OFFICE

Created radarcharts for five chosen places in living room and office are given in next pages.

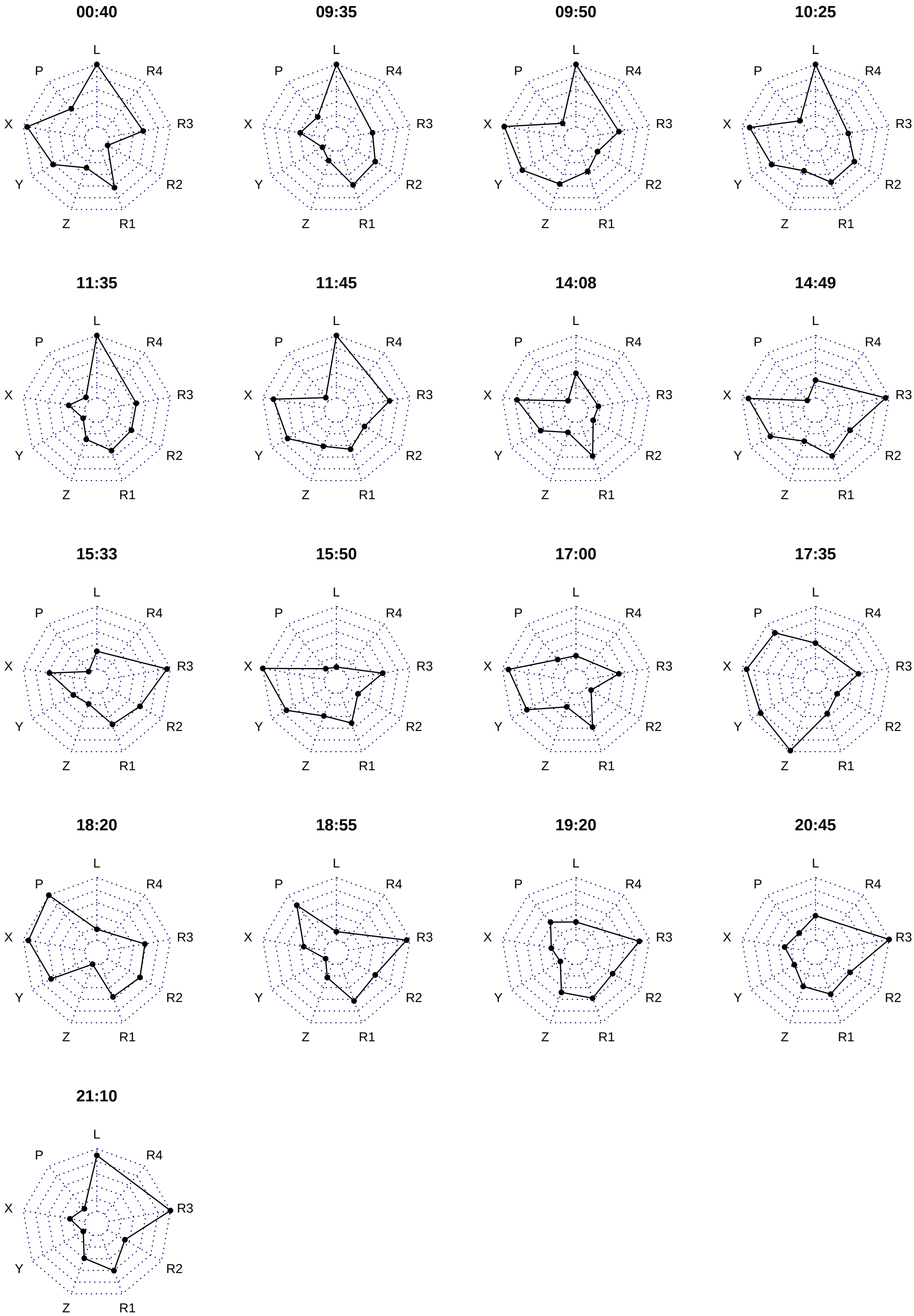

Figure A.1: Radarchart of Place1 in living room at various times

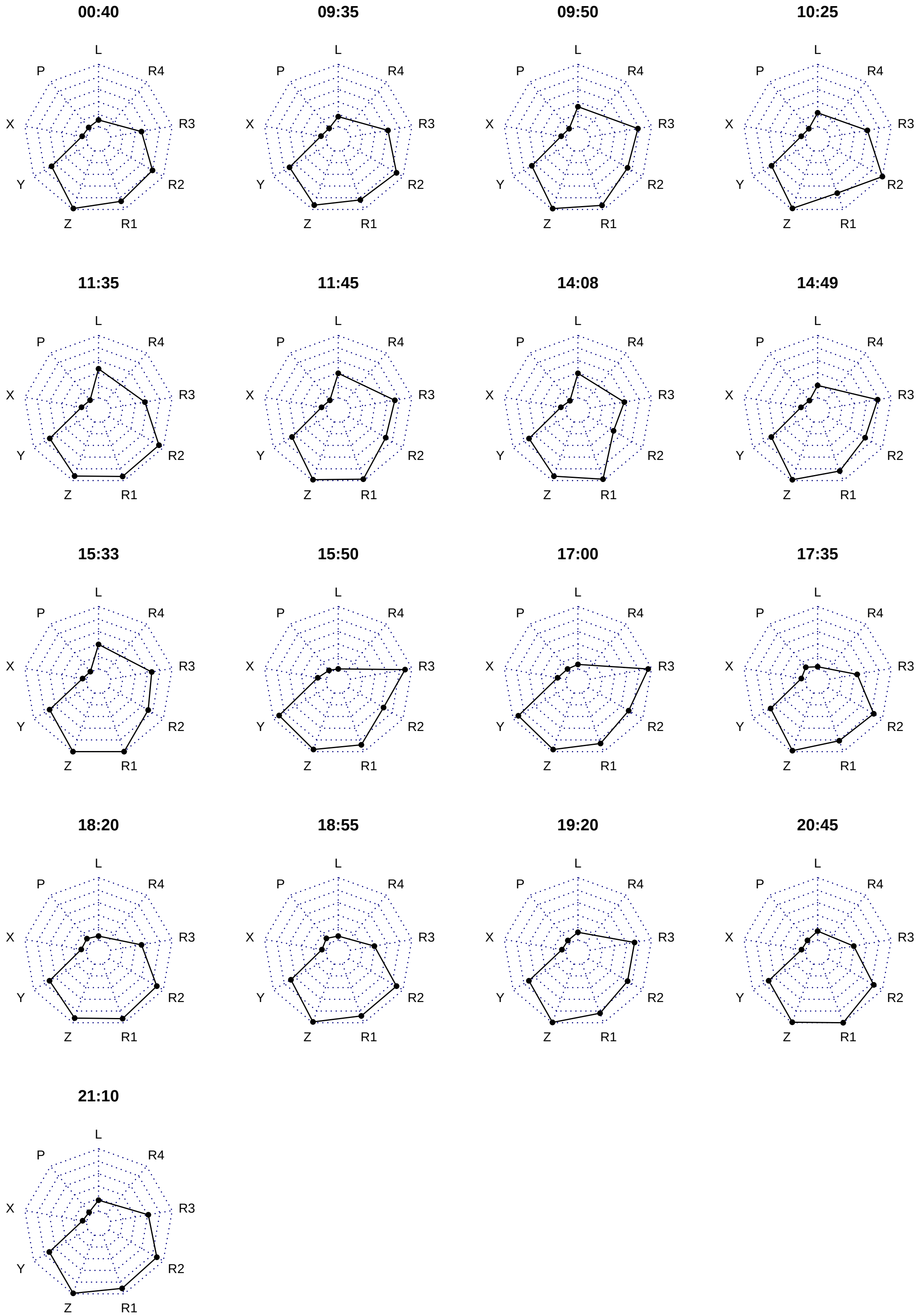

Figure A.2: Radarchart of Place2 in living room at various times

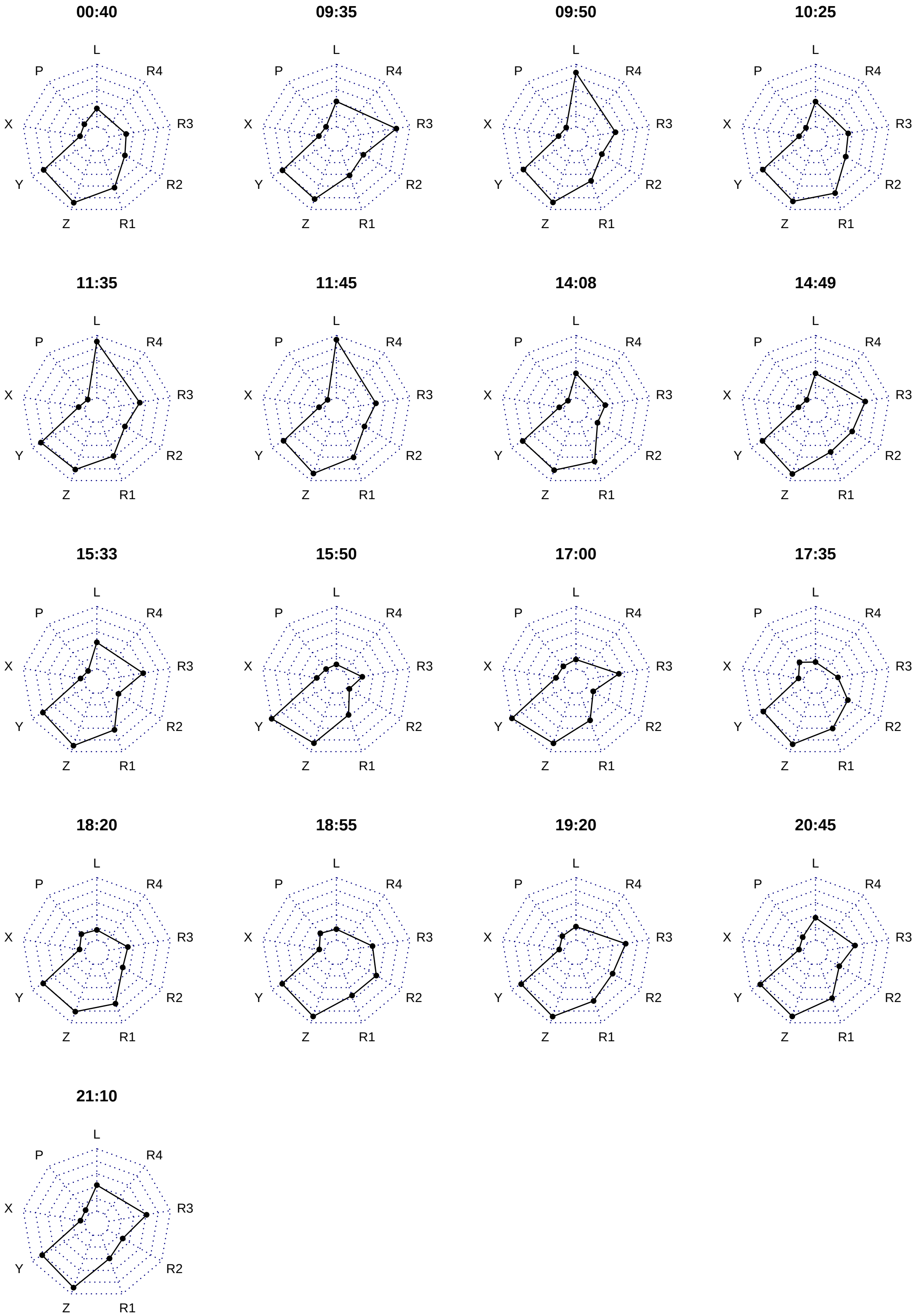

Figure A.3: Radarchart of Place3 in living room at various times

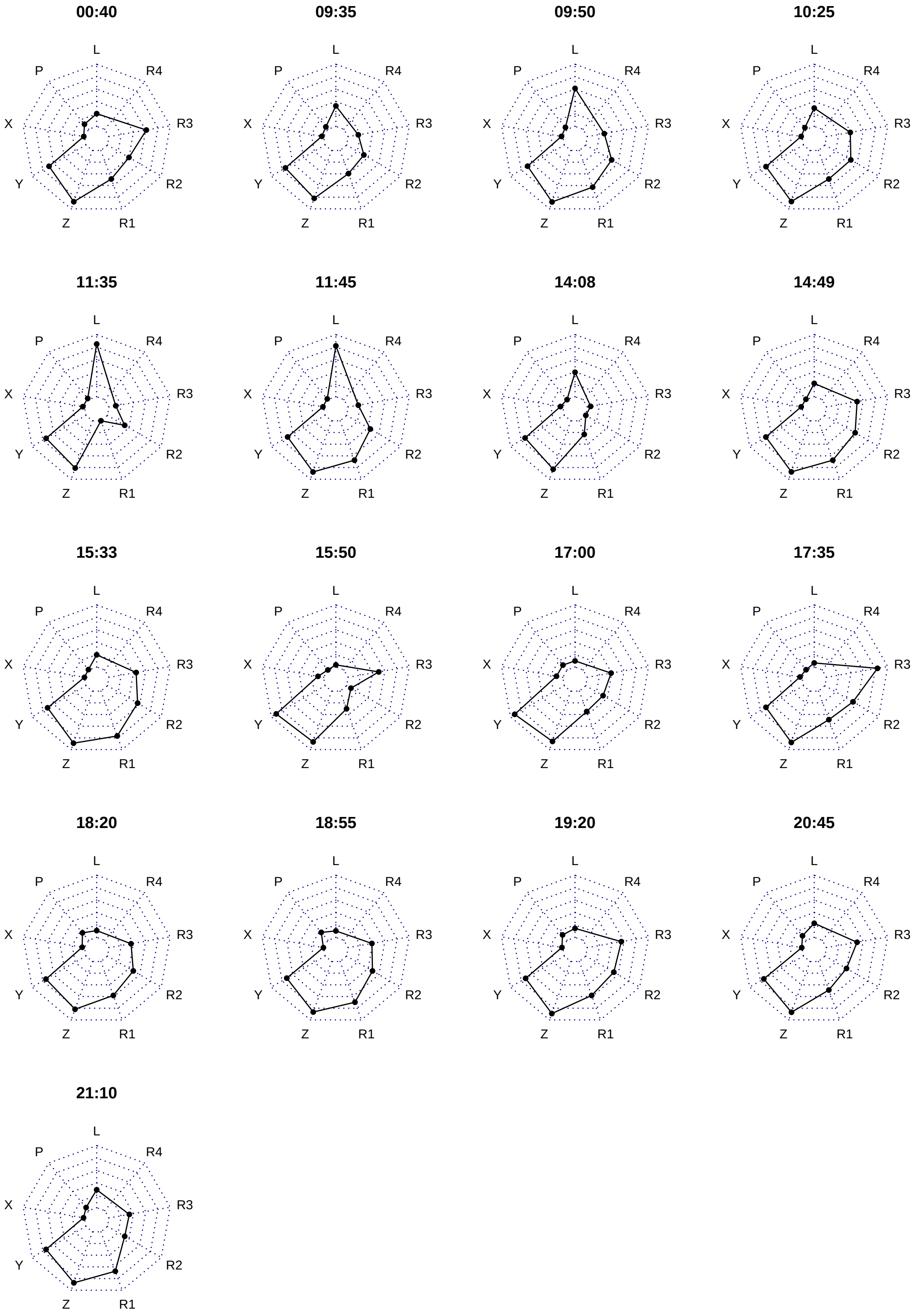

Figure A.4: Radarchart of Place4 in living room at various times

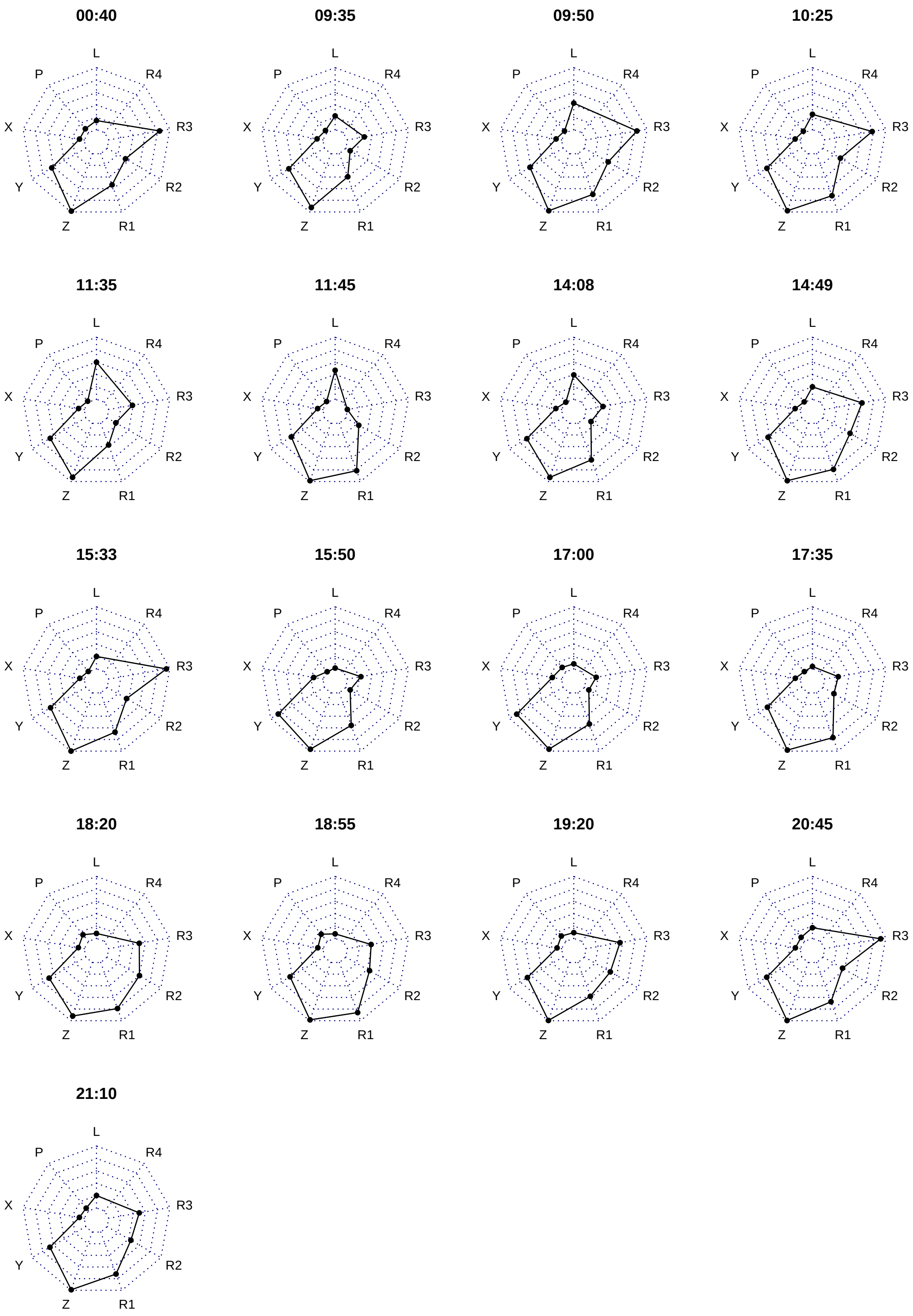

Figure A.5: Radarchart of Place5 in living room at various times

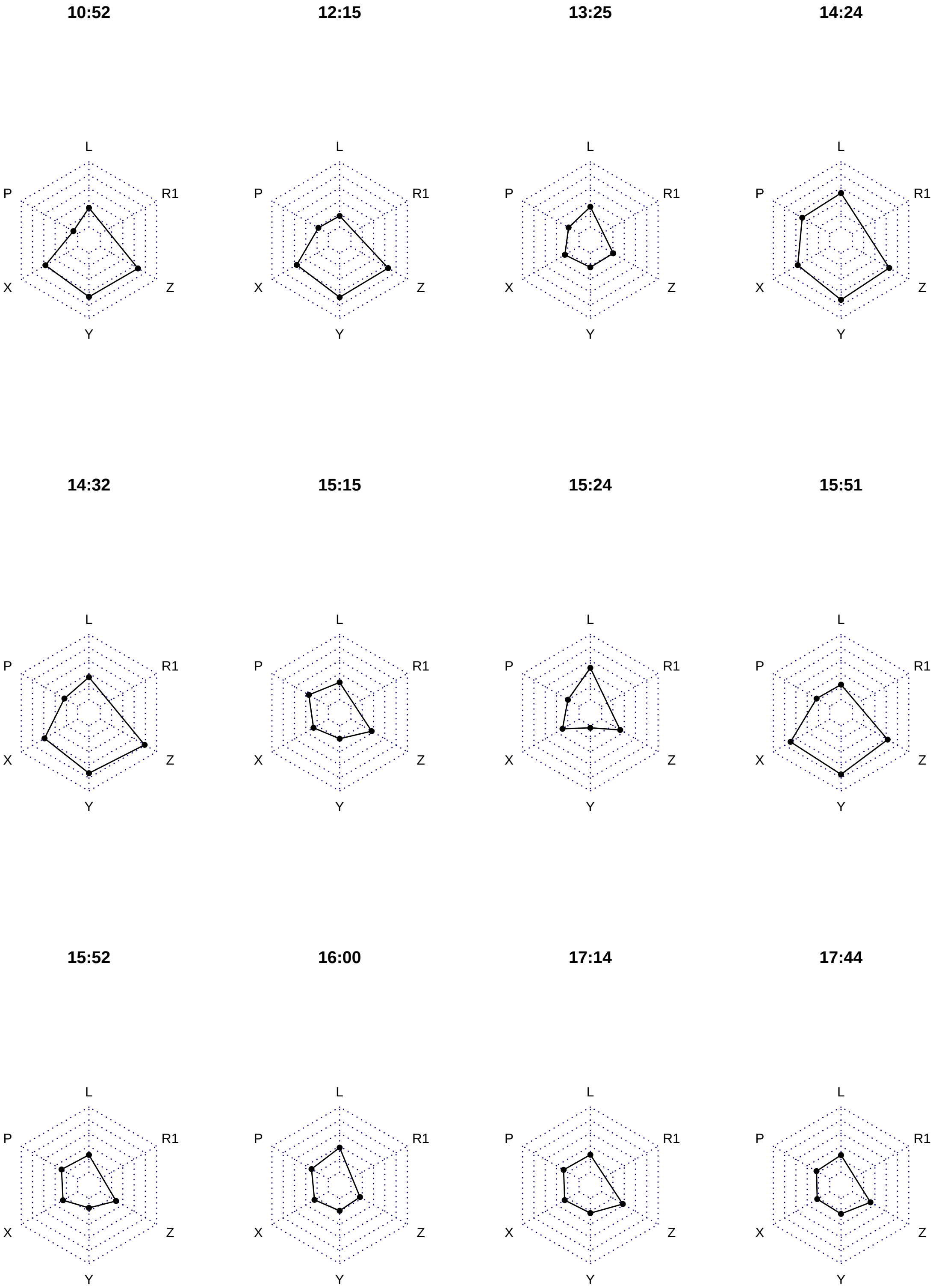

Figure A.6: Radarchart of Place1 in office at various times

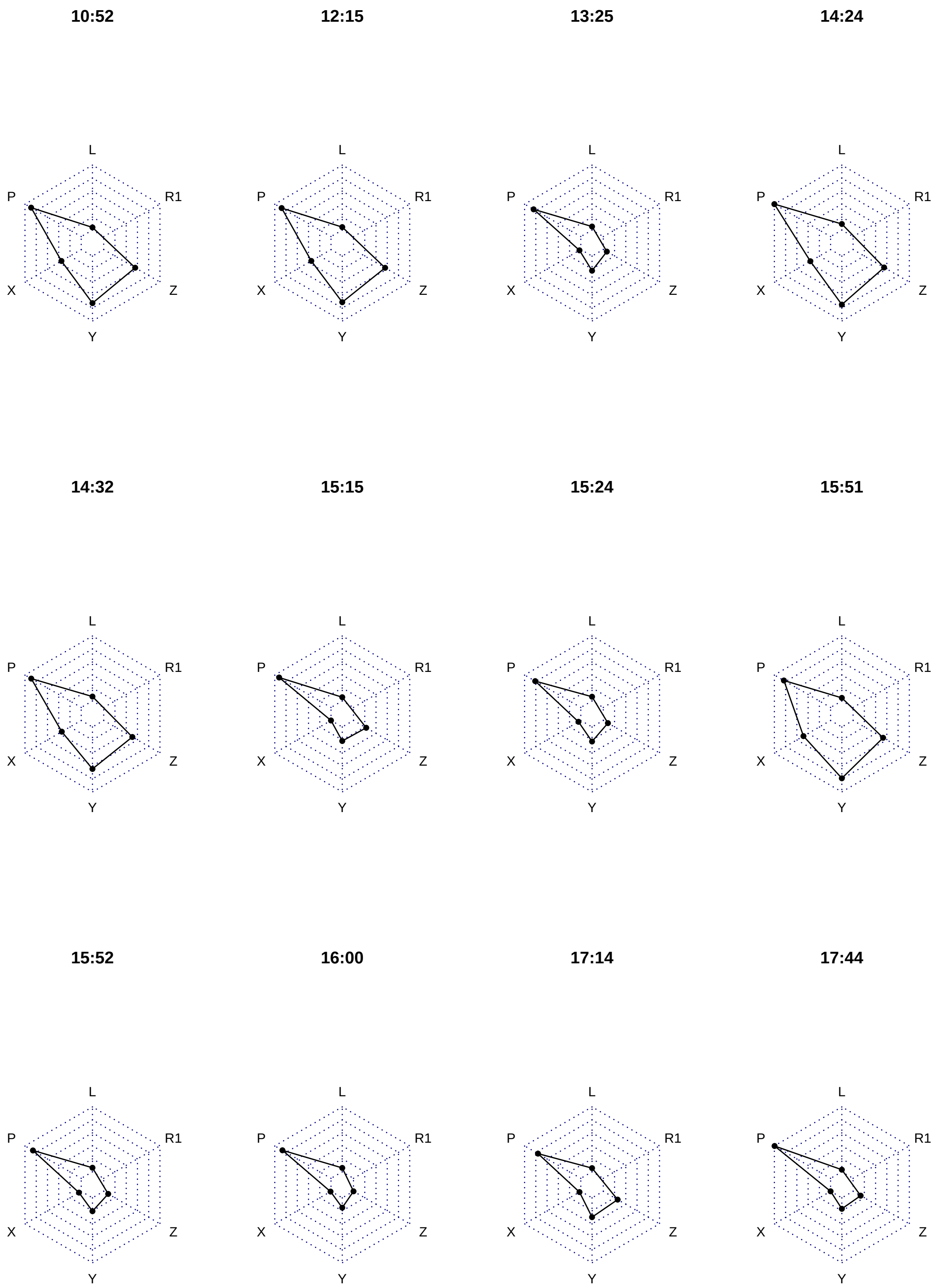

Figure A.7: Radarchart of Place2 in office at various times

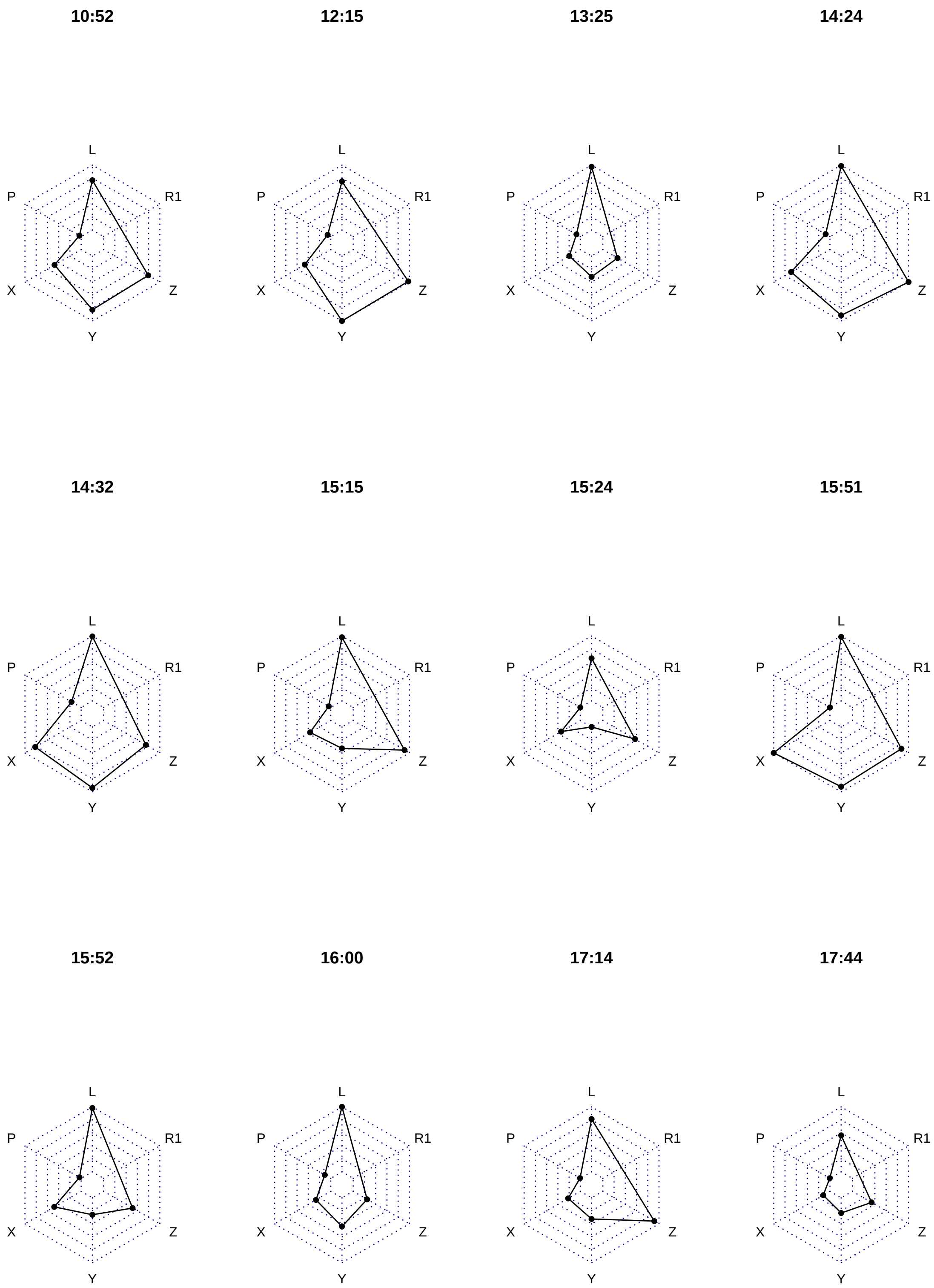

Figure A.8: Radarchart of Place3 in office at various times

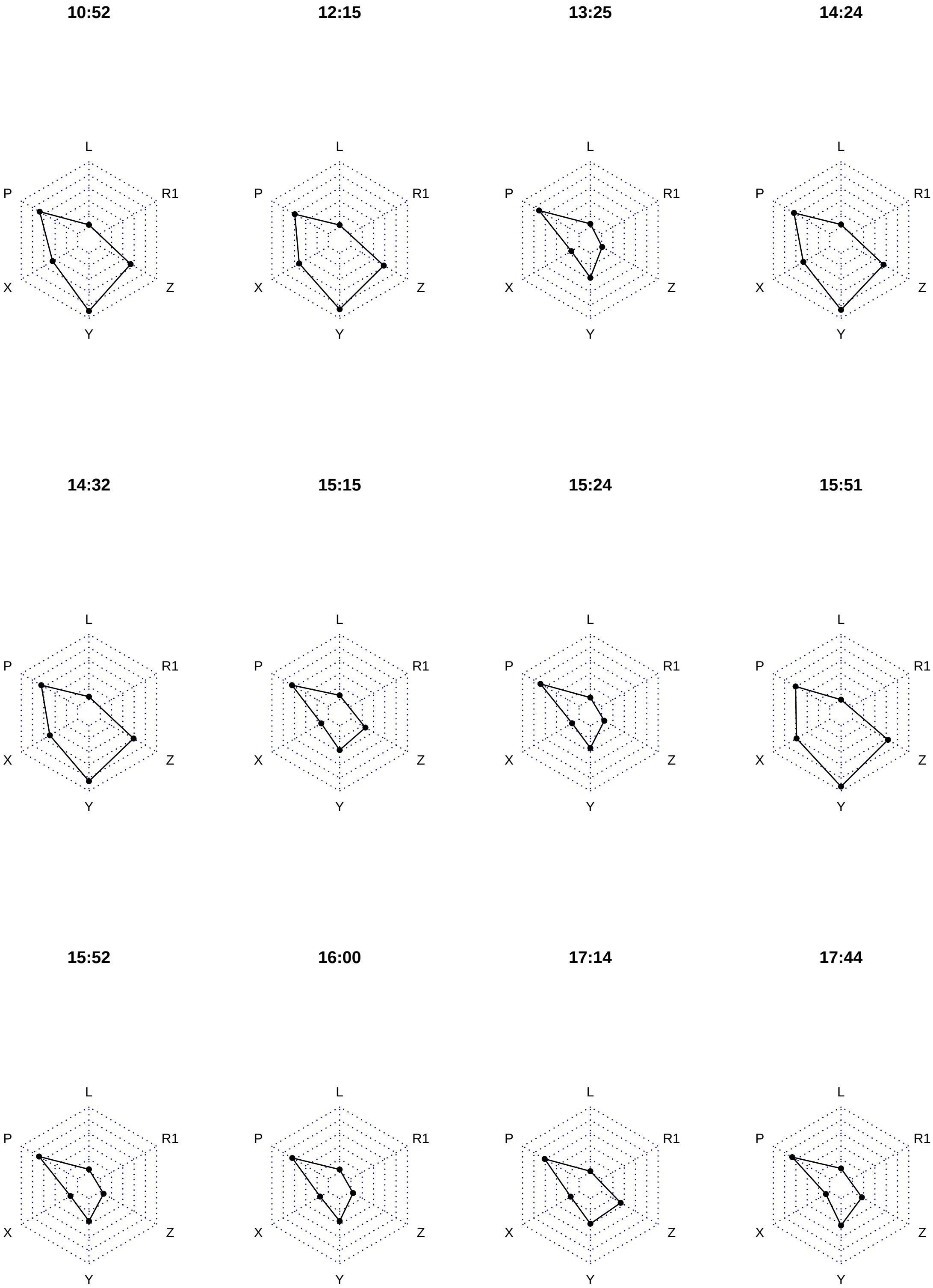

Figure A.9: Radarchart of Place4 in office at various times

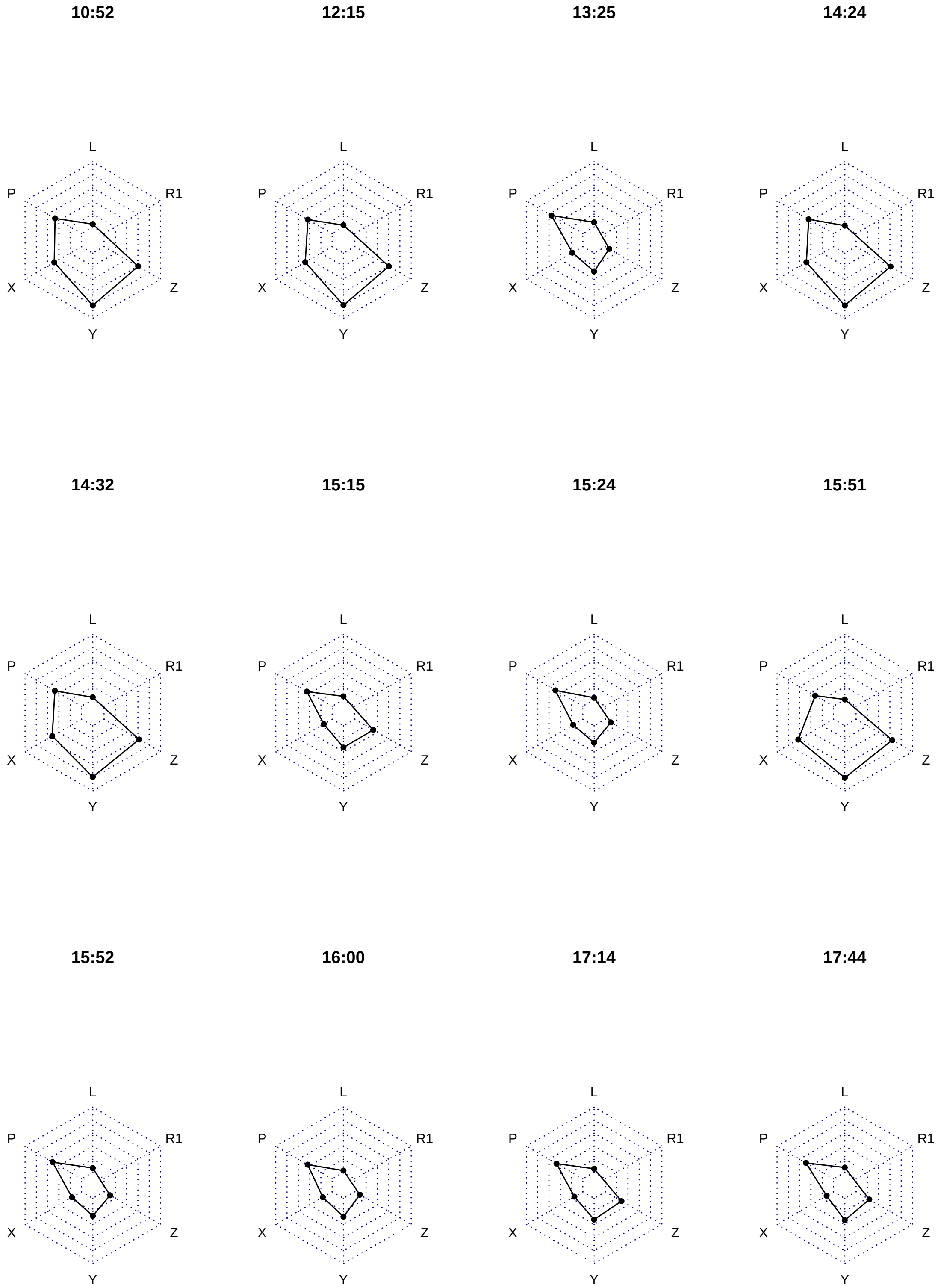

Figure A.10: Radarchart of Place5 in office at various times