Master thesis

Refining user context detection on smartphones

N.R. de Jong

Student: 851409351
Date: 10/15/2019

Open Universiteit
www.ou.nl

MASTER THESIS

REFINING USER CONTEXT DETECTION ON SMARTPHONES

by

N.R. de Jong

in partial fulfillment of the requirements for the degree of

Master of Science
in Software Engineering

Open University of the Netherlands, Faculty of Management, Science & Technology
Master Software Engineering
to be defended publicly on Tuesday October 15, 2019 at 15:00.

Student number:	851409351	
Course code:	IM9906	
Thesis committee:	Dr. Greg Alpar (chairman),	Open University
	Dr. Ir. Hugo Jonker (supervisor),	Open University
	Dr. Arjen Hommersom (2nd supervisor),	Open University

An electronic version of this thesis is available at `http://dspace.ou.nl/`.

Open Universiteit
www.ou.nl

CONTENTS

List of Figures **iv**

List of Tables **v**

Summary **vii**

1 Introduction **1**
1.1 Objectives and research questions 1
1.2 Contributions 2

I Phase 1 **4**

2 Literature survey **5**
2.1 Context definition 5
2.2 Context recognition 7
2.2.1 Personal context type 7
2.2.2 Social/environment context type 7
2.2.3 Location context type 7
2.2.4 Activity context type 8
2.2.5 Behavior context type 11
2.2.6 Context recognition using external data sources 11
2.3 Context classification process 12
2.3.1 Algorithms and efficiency 12
2.3.2 Frameworks and tools 12
2.3.3 Data sets 13

3 Conclusions of Phase 1 **14**
3.1 Refining context 14
3.1.1 Multi label classification 15
3.1.2 About labels for context 15
3.2 Definition of Phase 2 17
3.2.1 Data source importance 17
3.2.2 Multiple levels of detail 18
3.2.3 Combining context categories 18
3.3 Research continuation 19

II Phase 2 **20**

4 Methodology **21**
4.1 Scope 21
4.2 Collecting data 21
4.3 Analysis 22
4.3.1 Data source importance (SRQ2.1) 22

4.3.2 Multiple levels of detail (SRQ2.2 & SRQ2.3) 23
4.3.3 Combining context categories (SRQ2.4 & SRQ2.5) 23
4.4 Validation 24

5 Collecting data **26**
5.1 Capturing sensor data 26
5.1.1 Which data is captured 26
5.2 Labeling context 30
5.3 Processing and labeling captures 31

6 Analysis **32**
6.1 Data source importance (SRQ2.1) 32
6.1.1 Feature ranking for building-level location recognition 33
6.1.2 Feature ranking for room-level location recognition 35
6.1.3 Feature ranking for activity recognition 36
6.1.4 Most important data sources 36
6.2 Multiple levels of detail (SRQ2.2 & SRQ2.3) 38
6.3 Combining context categories (SRQ2.4 & SRQ2.5) 41
6.3.1 Improving activity recognition using location information 41
6.3.2 Association between location and activity 42

7 Conclusions and future work **44**
7.1 Conclusions 44
7.2 Discussion 46
7.3 Future work 46

A Appendix Feature association data **48**
A.1 Feature vs Location association (Building-level) 48
A.2 Feature vs Location association (Room-level) 51
A.3 Feature vs Activity association (Building-level) 56
A.4 Feature vs Activity association (Room-level) 61

B Appendix Data source importance data **65**

Bibliography **69**
References 69

Glossary **78**

List of Figures

3.1 Example for an activity label tree 16

List of Tables

2.1 Activities supported by the Activity recognition API (copy from [Goob]) 9

5.1 Data sources captured on the smartphone 27
5.2 Activity context labels captured on the smartphone 30
5.3 Location context labels captured on the smartphone 31

6.1 An overview of the features with the highest association with the Building-level locations for the Building-level dataset. Features with an association value (η for correlation ratio and V for Cramer's V, for which the features are marked with a star) below 0.3 are not included because they are considered uncorrelated. Consecutive features in this ordered list for the RSSI value of WiFi access points are taken together as the specific access points are irrelevant here. 34
6.2 The mean GPS speed calculated for each building-level location of the building-level dataset 35
6.3 The frequency distribution of the feature ranks per data source. For both the building- and room level data sets, the association between each feature and the activity and location classes is classified in the association value ranges shown in this table. Per data source the frequencies of feature classes are summed for the building- and room-level datasets and for activity and location recognition. 37
6.4 Confusion matrix for location prediction using the building-level dataset. Summed over 100 training and test sequences. 39
6.5 Confusion matrix for location prediction using the room-level dataset. Summed over 100 training and test sequences. 39
6.6 Decision tree metrics for location prediction using the building- and room-level datasets 40
6.7 Confusion matrix for activity prediction using the room-level dataset. Summed over 100 training and test sequences. 42
6.8 Confusion matrix for activity prediction using the building-level dataset. Summed over 100 training and test sequences. 42
6.9 Decision tree metrics for activity prediction comparison with and without inclusion of location information, using the building-level dataset 43
6.10 Decision tree metrics for activity prediction comparison with and without inclusion of location information, using the room-level dataset 43

A.1 The association between features and the Building-level locations for the Building-level dataset. Only features with an association value (η for correlation ratio and V for Cramer's V, for which the features are marked with a star) >= 0.3 are shown. 50

A.2 The association between features and the Room-level locations for the Room-level dataset. Only features with an association value (η for correlation ratio and V for Cramer's V, for which the features are marked with a star) >= 0.3 are shown. 55
A.3 The association between features and the activities for the Building-level dataset. Only features with an association value (η for correlation ratio and V for Cramer's V, for which the features are marked with a star) >= 0.3 are shown. 60
A.4 The association between features and the activities for the Room-level dataset. Only features with an association value (η for correlation ratio and V for Cramer's V, for which the features are marked with a star) >= 0.3 are shown. 64

B.1 The frequency distribution of the feature ranks per data source. For the building-level data set, the association between each feature and the activity classes is classified in the association value ranges shown in this table. Per data source the frequencies of feature classes are summed for the activity classes from the building-level dataset 65
B.2 The frequency distribution of the feature ranks per data source. For the room-level data set, the association between each feature and the activity classes is classified in the association value ranges shown in this table. Per data source the frequencies of feature classes are summed for the activity classes from the room-level dataset 66
B.3 The frequency distribution of the feature ranks per data source. For the building-level data set, the association between each feature and the location classes is classified in the association value ranges shown in this table. Per data source the frequencies of feature classes are summed for location recognition on the building-level dataset. 67
B.4 The frequency distribution of the feature ranks per data source. For the room-level data set, the association between each feature and the location classes is classified in the association value ranges shown in this table. Per data source the frequencies of feature classes are summed for the location classes from the room-level dataset 68

SUMMARY

Smartphones are used numerous tasks and are carried with us the whole day. Apps are also becoming smarter and the next step of smart apps is to make them adapt to the users environment. The underlying concept for such adaption is context recognition. Currently, context recognition applications are presented in literature, but the majority is limited to recognize only a select number of classes from a single category of context.

The objective of this research is to refine context recognition on a smartphone, which means more contextual information is added. In order to clarify what the term 'user context' means, what it contains, how it can be categorized, what has been studied with regard to context recognition and to understand in what ways context can be refined, a literature survey is presented. From that survey a new taxonomy of context is derived, which is based on a combination of several other taxonomies, and has a new category called 'behavior'.

Using the results of the literature survey, a new proposal for refining context is presented which uses a set of labels to describe context instead of a single one. This is useful because it allows for multiple aspects of context to be recognized at once. Multi-label classification can be used to train a multi-label classifier with variable size label sets and to predict such sets. The results of the literature survey are also used to define experiments for the second part of this research, which is defined based on the results of the first part. First different smartphone data sources such as motion sensors and connectivity interfaces are investigated for their importance in activity and location recognition. Second the extent to which two different detail levels for location context can be recognized is investigated. Finally, the effect of combining context categories is analyzed by means of investigating to what extent activity recognition can be improved when adding (room level) location information as input to the activity recognition classifier.

In order to perform these experiments, data is required to analyze. No existing dataset could be found which contains data from various smartphone data sources, two levels of detail and labeled with activities and locations. Therefor real life smartphone context data is collected about every fifteen minutes for a ten week period using a custom android app. This app gathers data from various data sources such as sensors (e.g. motion, environment, position sensors), WiFi and GPS. During this period of data gathering contexts are manually labeled using another custom android app on a separate smartphone. The collected data is transformed into two datasets for two levels of location detail: the building-level dataset and the room-level dataset. Because data from only one person is collected, the results will not hold generally, but suffices for the explorative nature of this research.

For the data sources investigated, the associations between the features and the classes to predict are calculated, where features with a high association with a class is considered important for context recognition. The most important smartphone data sources for activity and location recognition are found to be WiFi, GPS, battery information, accelerometer, current time, linear acceleration, gravity, gyroscope and game rotation vector. For the datasets considered, a decision tree is trained for both the room-level and building-level location. The calculated accuracy, precision, recall and F1 score are all around 92% for the room-level and 98% for the building-level. Though it has to be noted that the decision tree likely overfits the data because data originates from a single user, these high scores indicate the potential for recognizing a more detailed level of (location) context.

To investigate the added value of combining context categories for context recognition, the association between the location and activity classes are determined. For the room-level dataset a high association is found; for the building-level dataset the association for activity to predict the location is very high, but for the location to predict the activity it is much lower. A decision tree then trained for the activity classes and the results are compared with the same decision tree trained for a dataset where the location is included. Including the building-level location does not seem to have an effect, but including the room-level location shows an improvement of five to six percent.

The global conclusion is that context can be refined by combining different context categories and that, at least for categories Location and Activity, a category can be used to improve the score of the classifier for another category. Predicting a more detailed (location) context look promising, especially when data sources such as WiFi are used as input to the classifier.

Future work can focus on a more general application by using data from a large group of persons. Also other data sources can be investigated for their importance within context recognition. Higher level data sources such as human behavior patterns can potentially add important information to the concept of context, thereby allowing for substantial refinement. Future work can also be dedicated to multi-label context recognition, which is a promising development for a practical application of recognizing multiple aspects of context.

1

Introduction

For years, smartphones have gained an increasing role in our personal and business lives. They are used throughout the day for numerous tasks en are probably one of the most essential tools in our daily lives. As such, they are carried with us most of the day. Applications running on these smartphones are becoming smarter and next steps in improving applications intelligence are to make them adapt to the users environment to make them smarter.

The underlying concept to adapt such applications to the users environment is *context recognition*, where the smartphone 'recognizes' certain contexts and allows the applications to change their appearance or functionality to offer more appropriate functionality for the recognized context. An example of such an adaption to context is prioritizing business or private contacts according to whether the user is at the office or at home. Another example is to provide a simpler user interface when the user is walking or running.

Office and home are examples of location context information, walking and running are activities. Although such context information could be very helpful to adapt to different contexts, more detailed context information could enrich the context and gives even more opportunities for applications to become smarter.

1.1. Objectives and research questions

This research focuses on recognizing more detailed context information. How this can be done is the objective of this research and therefor the following research question is defined:

RQ *In what manner can the concept of 'user context' be refined using smartphone sensors and data sources?*

In order to investigate in what ways context can be refined, this main research question is to be refined by a set of sub research questions. To be able to do this, first the concept of context has to be clear. Secondly, different aspects of context recognition have to be explored to be able to select those subjects within context recognition that are candidates to be important for refining context and will be further investigated within this research.

For these reasons, the research is split into the following two phases:

- **Phase 1** - Explores the research field of context recognition, with focus on what context is and how it is studied, and defines research for Phase 2 based on results and conclusions of Phase 1. An additional set of sub research questions is defined for the research of Phase 2.
- **Phase 2** - Research as defined at the end of Phase 1 and focuses on how 'user context' can be refined. This phase will answer the sub research questions for Phase 2, as defined in Phase 1.

To give direction to Phase 1, two sub research questions are defined which will be answered in this phase to be able to precisely define research for Phase 2:

SRQ1.1 *What is user context and what does it consist of?*

SRQ1.2 *How can this research be organized to contribute to refining context?*

The first sub research question of the first part (SRQ1.1) is answered using a literature survey, which is presented in Chapter 2, and discusses amongst others the definition of (user) context and how context information and recognition can be categorized.

When it is clear what context is and what it includes, steps are taken to investigate what ways this research can contribute to refining context, which is covered by the second sub research question of the first part (SRQ1.2). For this sub research question it is important to know what is already studied with respect to context recognition on smartphones. The literature survey from Chapter 2 is therefor expanded to include related work for smartphone user context recognition and tools and algorithms that can be used for this.

Chapter 3 presents conclusions of the literature survey and discusses subjects which are considered potentially important for refining user context. Using the conclusions for the literature survey, subjects within the field of context recognition are selected for further investigation. They will be discussed at the end of Chapter 3 and will be accompanied by a new set of sub research questions for Phase 2.

1.2. CONTRIBUTIONS

A lot of scientific work is done concerning detecting context on smartphones. But as wide as the concept of *context* is, the subjects within this research area have a large variety. Many work is brought together within the literature survey presented in Chapter 2. Within this literature survey, a new taxonomy is presented for the different categories of context. Using the conclusions from the literature, a proposal for a multi-label classifier for context recognition is discussed in Chapter 3.

For context recognition a dataset is created using real-life data captured with a custom made Android app on a smartphone for about every fifteen minutes during a ten week period. Within this period, context is manually labeled using another custom made Android app on a separate smartphone. A large number of parameters from different smartphone data sources, for example (motion) sensors and network interfaces, are collected. It is investigated which features, the parameters from the data sources captured, are most important for detecting activities or locations.

For adding more detail to context, an exploration is executed in which multiple levels of location context are classified using raw data from a large number of smartphone data sources. The building level location classification is compared with room level classification.

Finally, associations between location and activity contexts are explored and the added value of using location information in activity recognition is investigated.

I

PHASE 1

2

LITERATURE SURVEY

Context is a diverse and wide subject of study, with various directions. This chapter presents an exploration of the different fields of study and gives an overview of work done related to user context recognition, on smartphones in particular. The main subjects discussed within this chapter are what context is, how context recognition is performed in other studies, and which tools and algorithms can be used to accomplish context recognition.

Conclusions about this chapter are presented in Chapter 3, as well as answers to the sub research questions for the first phase based on these conclusions. Also, research for phase two is defined in that chapter, accompanied by a new set of sub research questions.

2.1. CONTEXT DEFINITION

In literature, several different definitions of *context* exist. Abowd et al. [Abo+99] provided a definition for context which is widely accepted:

> *Context is any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves.* [Abo+99]

According to this definition, context is a broad concept and can contain anything imaginable as long as it is relevant to the situation. Within this research, the context entity is the user of the smartphone, and context contains anything relevant to the current situation of the smartphone, and thus the user. Context related to the user of a device is often called *'user context'*.

Since context as a research area has gained more interest, several categorizations are proposed for organizing context information. Abowd et al. [Abo+99; DA00; DAS01] presented four primary categories of context, based on four of the five W's (Who, Where, What, When):

- **Identity** - a unique identifier of an entity such as a phone number or e-mail address
- **Location** - information such as position, orientation, and elevation
- **Status/activity** - characteristics of the environment, such as temperature, light intensity, humidity etc.

- **Time** - time related information, such as history, timestamps, timespans, duration, and periods

Another categorization is proposed by Goker et al. [GM02], which identified five parts of user context:

- **Environment** - entities of the environment, for example objects, temperature, light intensity, humidity, persons, and media such as movies and sound
- **Personal** - physiological context such as blood pressure and weight, and the mental context such as mood and happiness
- **Task** - what a person is doing within the context, such as activities, tasks, and events. Can also include other persons that are relevant to the context
- **Social** - social aspects of the current user context, such as friends, enemies, neighbors, and relatives
- **Spatio-temporal** - time-related aspects such as time, location, and speed

Hoyos et al. [HGB13] proposed the following taxonomy of context types:

- **Physical** - physical information about the environment such as time, speed, temperature and light intensity
- **Environment** - information about people and objects in the user's environment, such as distances and locations
- **Computational (system)** - software and hardware related information such as network traffic, hardware status, and accessed information
- **Personal** - personal information such as preferences, age, gender, and psychological state
- **Social** - social aspects of a user, region or place, such as laws, friends, enemies, neighbors, relatives, and the user's role
- **Task** - information about the activities the user can or has to do, or is doing

When comparing these categorizations, it can be noticed that categorizations often have comparable categories, for example *Time* and *Spatio-temporal*, or *Status* and *Environment*. For this research, these different categorizations will be combined into one categorization, which is described below, to make references to context categories unambiguous, but also to introduce a new category. The category that is new is *Behavior*, which covers the human behavior patterns and context information related to this subject. Behavior patterns are thought to be important for reasoning about context and predicting upcoming contexts .

- **Environment** - physical information such as temperature, humidity and light intensity, as well as people and objects

- **Location** - information such as position, orientation, and elevation
- **Time** - aspects of time, such as the current time, history, duration, periods, and seasons
- **Personal** - personal information such as name, home address, phone number, e-mail address, age, gender, weight, preferences, and psychological state
- **System** - software and hardware related information such as network traffic, hardware status, accessed information, usage information, and search behavior
- **Social** - social aspects of a user, region or place, such as laws, friends, enemies, neighbors, relatives, and the user's role
- **Activity** - the activities a user or other person relevant to the user context can or has to do or is doing
- **Behavior** - behavioral patterns in (partial) context, familiarity with a context

2.2. CONTEXT RECOGNITION

2.2.1. PERSONAL CONTEXT TYPE

LiKamWa et al. used a supervised learning approach on phone usage data to classify the smartphone users' daily mood into four major mood types [LiK+11]. They continued with this research to improve the accuracy by using a personal training method [LiK+13]. Mathur et al. used random forest classifiers and SVM to classify user engagement out of smartphone usage logs. Bogomolov et al. also used smartphone usage data but also weather conditions and personality traits to classify the smartphone users stress level [Bog+14] and happiness [BLP13]. Lee et al. used a Bayesian Network classifier to classify a smartphone users emotions in seven classes [Lee+12].

2.2.2. SOCIAL/ENVIRONMENT CONTEXT TYPE

Chittaranjan et al. used supervised learning to automatically infer the personality type of the user of a smartphone based on phone usage data. Statistics from calls, SMS, Bluetooth, and application usage are collected and analyzed on a monthly basis [CBG11]. Chen et al. used Bluetooth to classify the ambient social context of its user [Che+14] [Che+15]. They used a method to continuously and incrementally construct the social context to be able to deal with newly appeared classes.

2.2.3. LOCATION CONTEXT TYPE

Chon et al. used crowdsourcing to automatically characterize places using media created on a smartphone and location information from GPS and WiFi [Cho+12]. Montoliu et al. used clustering on both time and location to discover places of interest for the user of a smartphone [MBG13].

Chon and Cha proposed a system which automatically identifies points of interest using smartphone sensors such as GPS, accelerometer, and WiFi [CC11]. They did not only focus on *Location* context, but also on *Activity* and *Environment*. A framework for activity recognition, based on Relative Markov Networks , is proposed by Liao et al. [LFK05].

They performed experiments using this framework for location-based activity recognition, in which activities are for example "At home", "At work", "Dining", or "Visiting", which are a combination of *Location* and *Activity* context types. Later they used Conditional Random Fields (CRF) to generate a hierarchical model of the smartphone users activities and places [LFK07].

2.2.4. ACTIVITY CONTEXT TYPE

Activity recognition is extensively studied for various aspects, as shown by the amount of related work presented in this section. Detecting activities a person is doing, using (sensor) data from a device such as a smartphone or wearable is often called Human Activity Recognition (HAR). The term Activities of Daily Living (ADL) is used to denote activities which are generally performed on a daily basis by the targeted group of people. It originates from healthcare and is used in activity recognition literature due to the application of activity recognition for monitoring elderly people.

Incel et al. reviewed existing activity recognition systems in 2013 [IKE13]. They explained typical steps within the activity recognition systems and architectural choices of classification. Shoaib et al. also reviewed existing activity recognition systems in 2015, but focused solely on online solutions [Sho+15]. With online activity recognition they mean that the complete system, from data collection to classification, is performed on the smartphone itself.

One such system is presented by Siirtola and Röning [SR12]. They used supervised learning to perform real-time physical activity recognition on a smartphone using the build-in accelerometer. They used K Nearest Neighbors (knn) and Quadratic Discriminant Analysis (QDA) to classify five everyday activities. All data processing except the training algorithm was performed on the smartphone. Guinness also presented an activity recognition system using accelerometer data and machine learning, but they also used GPS and 3rd party geospatial information such as train stations and bus stops [Gui15]. The system automatically and continuously detects the user's smartphone current activity. A wide range of machine learning classification algorithms are compared for their accuracy on this particular system. Conti et al. used a mechanism which eavesdroppes a smartphones network activity to recognize activities performed on the smartphone, such as apps used or specific actions performed such as sending an e-mail [Con+15]. Their focus is more privacy related, because activity detection is not performed on the smartphone, but on a separate device (which may be malicious or not).

Reyes-Ortiz wrote a complete book about activity recognition on smartphones, which covers amongst others dataset generation and hardware friendly online and offline HAR [Rey14; Rey15]. In 2016, he and others proposed a system for activity recognition on smartphones using its accelerometer and gyroscope sensors [Rey+16]. They focused on activity transition recognition, which can be useful when an application needs to adapt when context changes.. Siirtola and Röning performed activity recognition using sensor fusion, combining data from different sensors, to improve recognition reliability when the accelerometer approach is not reliable enough [SR16]. Their system can determine the reliability of the accelerometer approach and can decide to use the sensor fusion approach when reliability is too low. This solution is chosen because the sensor fusion approach is much more battery intensive [SR16] compared to the accelerometer approach.

As part of a PhD project, Pires et al. compared different Artificial Neural Networks for

IN_VEHICLE	The device is in a vehicle, such as a car.
ON_BICYCLE	The device is on a bicycle.
ON_FOOT	The device is on a user who is walking or running.
RUNNING	The device is on a user who is running.
STILL	The device is still (not moving).
TILTING	The device angle relative to gravity changed significantly.
UNKNOWN	Unable to detect the current activity.
WALKING	The device is on a user who is walking.

Table 2.1: Activities supported by the Activity recognition API (copy from [Goob])

ADL recognition on smartphones using motion sensors and behavior patterns [Pir+17d]. They also applied sensor fusion with motion sensors and magnetic sensors [Pir+18b; Pir+17c], acoustic sensor (e.g. microphone) [Pir+17b] and even combined these sensors with the user calendar [Pir+17a]. For using acoustic smartphone sensors to recognize ADL, Pires et al. reviewed related work about using audio fingerprints [Pir+18d].

A specific activity that is subject of research, often in wellbeing applications, is sleep detection. Chen et al. used motion sensors, light sensor, microphone and smartphone usage data to automatically, without intervention from its user, detect sleep duration [Che+13]. Lane et al. used sleep duration detection in combination with physical activity duration and social environment detection to give the smartphone user feedback about its wellbeing. They used the smartphone accelerometer to detect physical activities and the microphone to detect conversations, where the total duration of conversations on a day denotes the social environment. [Lan+14].

Another specific activity subject of research is eating and drinking behavior. Biel et al. presented a system which is able to recognize when the user is eating, and if so differentiate between a full meal or a snack [Bie+18]. Santani et al. used data from various smartphone sensors (location information, accelerometer, WiFi, Bluetooth, battery status, screen status, and app usage) to recognize whether the user drank alcohol [San+18]. Bae et al. used a Random Forest model to classify the alcohol drinking behavior of a smartphone user at a certain time [Bae+18]. Their system is able to classify alcohol drinking behavior into three classes from non-drinking to drinking large amounts of alcohol.

Google made the Activity Recognition API available in the Google Play services, which can be used by Android apps [Gooa]. The supported activities for this API are shown in Table 2.1 [Goob].

The Activity Recognition API also supports an interface for subscribing to activity transitions. For this interface only a subset of the activities (IN_VEHICLE, ON_BICYCLE, RUNNING, STILL, and WALKING) are supported [Andd]. The inner working of the Activity Recognition API is IP information from Google and is therefor not published.

OPTIMIZING ACTIVITY RECOGNITION

For performing activity recognition on a smartphone, the place where the smartphone is held during activities can have significant influence on classification.

Miluzzo et al. presented a system which is able to automatically detect where a smartphone is held, for example in the users pocket, hand, or inside a backpack [Mil+10]. To be able to deal with differences in the places where smartphones are held, Martin et al. used an architecture where first the smartphone place is detected, and then, using this information, the activity is detected [Mar+13]. Khan et al. presented an activity recognition algorithm which is able to classify fifteen different activities, where the smartphone can be held on different body parts [Kha+14]. They implemented an efficient HAR using sensor fusion and Support Vector Machines (SVMs) on data from the accelerometer, pressure sensor, and microphone. Siirtola and Röning succeeded in making an activity recognition system which works user and body position independent and can handle variation in hardware or uncalibrated sensors well [SR13].

Since everyone is unique, a generalization of human activity recognition will probably not have an outstanding precision. For this, the last few years solutions for this are studied to improve precision by utilizing a more personalized approach. Siirtola et al. presented a method for performing user-dependent HAR on an smartphone using sensor fusion for data labeling and single sensor data for classification [SKR16]. Later they used incremental learning to personalize activity recognition [SKR]. For this, the system initially uses a user-independent model and when personal data becomes available, the model is updated with this new information. Saha et al. used accelerometer and gyroscope data to perform fine-grained HAR with taking into account different configurations in terms of device and user variation [Sah+18].

Lane et al. proposed Community Similarity Networks (CSNs) to find 'similar' users to be able to improve HAR [Lan+11]. A CSN is able to measure inter-person similarities and using data from these 'similar' users a more personalized model could be generated. Later Lane described how crowd sourcing, the process of gathering data from a large number of participants, could be used to gather data for the CSN approach for HAR [Lan12]. Abdullah et al. described how such a crowd sourcing framework could be developed in a scalable way, so that CSNs could be efficiency applied on a large scale system which supports many different types of users [ALC12].

Koskimäki and Siirtola described how human-independent and personal models for HAR could be combined to improve HAR precision [KS16]. Their system is able to combine these two models efficiently, so they can be used real time and on the smartphone itself. Lane and Georgiev used deep learning, low-power Deep Neural Networks (DNNs), to perform HAR [LG15].

ACTIVITY RECOGNITION ON WEARABLES

Not only smartphones are used for activity recognition, but using wearables for this classification is studied. Tapia et al. used five external accelerometers and a heart rate monitor to recognize a smartphone users' physical activity [Tap+07]. They used the WEKA toolkit [Uni; FHW16] with the C4.5 decision tree and Naive Bayes classifier to recognize several physical activities typically performed in a gym.

Lester et al. used an external board with multiple different sensors to generate features for their Hidden Markov Model (HMM) classifier, which is able to to detect more general physical activities such as riding a bike, sitting or walking [Les+05]. Instead of using multiple different sensors, Siirtola et al. only used a wrist accelerometer to detect various physical sport activities. The used a Periodic Quick Test (PQT) classifier to detect such activities, which are long-term in words of multiple minutes [SKR11]. Castro et al. used features from a wearable which captures several physiological variables such as heart and respiration rate, as well as accelerometer data, to recognize four basic physical activities. Their "wearable assisted HAR" [Cas+17] used the C4.5 decision tree and Naive Bayes classifier to classify a patients health status. An IOT system is integrated to be able to, amongst others, remote monitor the health status of patients [Cas+17].

Since a few years more and more smartwatches are used, and with their variety of sensors such as a heart rate sensor and an accelerometer, they could provide relevant information for HAR. Bhattacharya and Lane used Restricted Boltzmann Machines (RBM) to detect physical activities (walking, running, standing, and on motorized transport) on a smartwatch [BL16]. They also experimented detecting gestures and whether an activity of a user is performed indoor or outdoor.

Kim et al. used accelerometer and microphone data from smartwatches to classify physical activities (eating, shower, sleeping, watching TV and vacuuming). Instead of combining data from these two data sources into one classifier, they used one classifier for each data source and combined the results of these classifiers into an activity classifier using a mapping table [Kim+16].

2.2.5. BEHAVIOR CONTEXT TYPE

Apart from the context types described earlier, human behavior patterns are also important within the context. It can describe whether the context is normal or exceptional, or be used to add more detail to the context by predicting the upcoming context. Thereby human behavior patterns can add a reason to a context, or relate different contexts with each other.

Xu et al. developed an app usage prediction model which can be used to predict which apps will be used, to be able to optimize the performance of a smartphone [Xu+13]. Do and Gatica-Perez studied how human mobility and location visiting patterns could be used to automatically label different locations [DG14]. They used location data gathered on a large amount of smartphones over one and half year to predict locations. Later they presented a framework for predicting the next location and app usage by using smartphone sensors. Eagle et al. used the MIT Reality mining set [EP06] to detect human behavior patterns based on the most important locations (home, office, etc.) [EP09]. They extracted daily patterns from groups of people and used this to predict behavior for the next hours. Like this study, Phithakkitnukoon et al. also investigated how human behavior patterns can be detected using the locations of groups of people [Phi+10]. They used millions of locations collected from the cellular network, where locations were captured during calls and text messages, to detect daily activity patterns. Also on a large scale and using data from the telecommunications industry, Paraskevopoulos et al. studied call activity and mobility patterns, anomalous behavior, and the effect of large events on these patterns [Par+13].

Horanont et al. investigated the effects of weather conditions on human behavior patterns [Hor+13]. GPS location traces are collected on smartphones and were geocoded into addresses using the GPS coordinates. These addresses were then grouped into different categories of activities related to the locations.

2.2.6. CONTEXT RECOGNITION USING EXTERNAL DATA SOURCES

Apart from using wearables to detect user context, Saha et al. used electricity meter data in combination with WiFi and microphone data from a smartphone to detect activities which use electricity [Sah+14]. Their system detects which home appliances are used, when they are used and by whom. Within the training phase, the user has to visit each room for a few minutes and has to sequentially turn on every appliance to include in the classifier.

Aran et al. used smart home sensors to detect behavior patterns of elderly [Ara+16]. They investigated whether it is possible to detect changes in these patterns using anomaly detection, with the goal to determine whether building a system which keeps an eye on elderly people is feasible.

2.3. CONTEXT CLASSIFICATION PROCESS

2.3.1. ALGORITHMS AND EFFICIENCY

Smartphones are becoming more and more powerful, but still hardware is limited in terms of computation power and energy use when running traditional machine learning algorithms. The limitations for ADL recognition on smartphones are explored by Pires et al. [Pir+18a]. Anguita et al. proposed a change to the SVM to use fixed-point arithmetic instead of floating-point arithmetic, thereby making this algorithm more computational efficient while maintaining similar accuracy [Ang+12a; Ang+12b].

Lane et al. investigated the amount of resources required to run deep learning algorithms on smartphones [Lan+15]. Later, Lane et al. proposed a software accelerator for executing deep learning algorithms on smartphones, which lowers the required resources such as computing power and memory usage with regard to the original deep learning algorithms [Lan+16].

Martin et al. compared three lightweight classifiers (Naive Bayes, decision tables and C4.5 decision trees) by their accuracy and computational resources when ran for activity recognition on smartphones [Mar+13]. They aim at building an activity recognition system which runs in the background continuously.

Different data sources on smartphones such as sensors or radio interfaces have variations in the manner how and when data is sampled or captured. Combining such different data sources (called *'sensor fusion'*) into one classifier is studied by Radu et al. [Rad+16]. They used deep learning methods to overcome problems with the different sampling techniques, and their results show that deep learning methods can outperform previous activity recognition solutions. Later, Radu et al. studied how multimodel deep learning techniques can be applied to improve context and in particular activity recognition [Rad+18]. Sensor fusion techniques that can be applied on sensors in mobile devices, with taking into account the limitations of these mobile devices, are explored by Pires et al. [Pir+16].

Apart from the techniques that can be used for context recognition, Yurur et al. proposed a generic architecture for context-aware applications. The middleware, which can be used by context-aware applications to request context information, consists of layers such as data acquisition, interpretation and reasoning [Yür+16].

2.3.2. FRAMEWORKS AND TOOLS

At the end of the previous millennium, Salber, Dey and Abowd proposed '*The Context Toolkit*': a middleware layer between the 'sensors' and applications to provide a uniform API for context-aware applications [SDA99]. They focused on detecting context on a location such as a room within an office, where the context information comes from PC's within such a room. Their implementation is able to provide information about the activity level of a room and who are active (who are using a PC). Riahi and Moussa presented a typical architecture used for context-aware applications and consists of the following layers [RM15]:

1. **Context acquisition** - Collection of data from sensors and other data sources

2. **Context interpretation** - Analysis and processing of the captured data and conversion into a higher level

3. **Context storage** - Storing the interpreted data for later use

4. **Context diffusion** - Servicing the application layer with data and notifications when the context changes

5. **Application layer** - Uses the context information to adopt to changing environments

A framework specifically for recognizing ADL is proposed by Pires et al. [Pir+18c]. Within this framework, each sensor has a processing pipeline for data acquisition, optimization and feature extraction, leading to a data fusion component which combines all features from the sensor pipelines. Identification of ADL is then done using pattern recognition and machine learning techniques, with data from the data fusion component.

Most context recognition applications discussed earlier used machine learning techniques to achieve this. Of course, these techniques can be applied using a wide variety of tools and programming languages, but many tools and libraries are available to support easier use of these machine learning techniques. The Weka toolkit [Uni; FHW16] is a set of tools and algorithms which can be used in all phases of machine learning development. It is used by Lee et al. for context recognition explicitly [Lee+12]. Popular machine learning frameworks of the last years are TensorFlow [Ten] (developed by Google), PyTorch [PyT] (developed by Facebook), Keras [Ker] and Scikit-learn [Sci]. Many other popular frameworks exist and new ones are added quickly.

2.3.3. DATA SETS

In 2004, Eagle and Pentland collected data on 100 smartphones over 9 months and published an anonymous version called the 'MIT Reality mining set' [EP06]. The captured data contains call logs, visible Bluetooth devices, cell tower IDs, application usage en phone status. Anguita et al. captured sensor data from 30 smartphones where volunteers were performing 6 selected ADL [Ang+13]. Siirtola et al. compared 15 different open data sets for activity recognition and they also made their data set from an earlier research publicly available [SKR18].

3

CONCLUSIONS OF PHASE 1

The majority of work done for context recognition focuses on Activity recognition. Where the target context recognition category is Location, often also the Activity category is considered. Most related work in context recognition use a small set of labels, and as such the labels do not represent a detailed context. More levels of detail within context recognition are not considered yet. Typically not only a small set of labels is used, but also the amount of data sources used for context recognition is limited. For activity recognition, the accelerometer is used is most of the discussed work.

Refining or enriching context can be achieved by broadening context (adding information from other categories) or deepening (adding more detailed information for already available context information). Combining context categories can both broaden and deepen at the same time.

When combining context aspects from different context categories, or when using multiple levels of detail, the context can no longer be described using a single label. The following section zooms in at the implications of this in further detail. Thereafter, using the conclusions of this research performed so far, subsequent investigation will be defined.

3.1. REFINING CONTEXT

When refining context, new information is added from different categories of context, or from another detail level. Then it seems not logical but also impractical to represent context by a single label. Context-aware applications may also be interested in different levels of detail. For one application the context 'shopping' might for example be of interest, while for another application the context 'walking' might be more relevant. Although during 'shopping' the user might also be 'walking', such overlap in context labels cannot be accommodated straightforward using a single context label.

When combining context categories, at a certain extent such context labels can be combined also into a single label, for example "sleeping-home" or "lunch-work", but this has some serious practical implications. First, combining context information into a single label makes training a context recognition classifier impossible when the number of contexts to recognize increases, due to the rapidly growing combinations of context aspects. Second, combining context information into a single label does not allow for partial context updates, for example when room-level location changes, but building-level location does not.

To overcome these disadvantages, a multi-label context could be used, where context is not represented as a single label, but as a set of labels. This allows for describing a context such as:

<sitting on couch>, <watching tv>, <with friends>, <in living room>, <at home>, <in Emmen>, <in The Netherlands>, <since 20:00>, <for 30 minutes>

3.1.1. MULTI LABEL CLASSIFICATION

Traditionally, classifiers using machine learning are trained to predict a single label, where the labels are mutually exclusive by definition. For the situation described above this does not suffice. A technique that is explicitly invented for such situations is called *Multi-label classification*.

According to Tsoumakas and Katakis, methods for multi-label classification can be subdivided into *problem transformation* and *algorithm adaption* methods [TK07]. The first transforms the multi-label problem into one or more single-label problems, while the latter changes an existing method so that it directly support multiple labels. Two problem transformation problems are the following [TK07]:

- For each label l in the dataset use a binary classifier with the complete dataset transformed so that each example is labeled l if the l is contained in the original example label set, otherwise the example is labeled as $\neg l$.
- The dataset is transformed so that each example is added for each label of the original example label set. Then a *distribution classifier* is used to calculate the probabilities of each label, and using a simple or more sophisticated threshold the labels for the label set are chosen.

Several algorithm adaption methods for multi-label classification are proposed, which are for example based on k-nearest neighbor [ZZ05], decision trees [Ven+08; CK01], or neural networks [Zha09; Wan+16].

3.1.2. ABOUT LABELS FOR CONTEXT

Refining context is discussed before, but in what directions context can be refined, and how this works for specific types of context will be presented next.

LOCATION CONTEXT LABELS

A subdivision of different aspects of Location context is given below. As anything relevant to the user can be included in the concept of context, this list is not a complete overview, but categorizes the most important aspects of Location context.

- Physical location
  - Altitude
  - Speed
  - Position
- Location type, such as landform (Mountain, Valley, Plains, Beach, etc.)
- Familiarity with location

The physical position is probably one of the most important aspects of location, but can be described at different levels of abstraction:

1. **Coordinates** - Geographic coordinate system: latitude and longitude
2. **Address** - Address information such as city, street, street number and postal code
3. **POI** - Description of a location by its function, for example supermarket, home or office

While *coordinates* and *address* level describe the 'where' of a location, the *POI* level describes 'what' the location is, but also relates it to the physical address. Numerous location services or libraries are available which can transform a *coordinates* position into an *address* and vice versa. To get *POI* information, location services exist which can be used to find public POI descriptions. Non public POI's, private or personal places, such as 'home' and 'office' have to be added by the user, or being automatically recognized on a smartphone.

Not only can a physical position be described at different levels of abstraction, but also at different levels of detail or covered area size. For example see the following physical position refinement from continent to room level:

> Continent > Country > State/Province > City > Neighborhood > Building > Floor/Wing > Room

Note that any of these could be included in a location address, but coordinates are useful mainly from *continent* up to the *building* level. The *POI* level is presumed mainly of interest for the *building* to *room* levels.

ACTIVITY CONTEXT LABELS

Activity context labels can also be used at different levels of detail, but decreasing the detail level can be seen as generalizing an activity, as can be seen in Figure 3.1. The same situation can be identified for label 'food and drinks', which is a generalization of for example 'breakfast', 'lunch', 'dinner', and 'snack'.

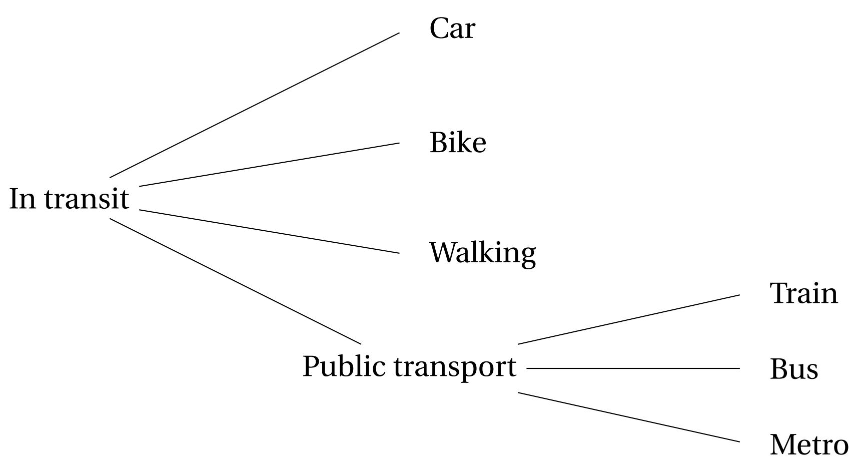

Figure 3.1: Example for an activity label tree

Although activity labels can be hierarchical, the depth of the hierarchy where labels still make sense is often limited to two or three. This is in contrast with the location labels, where a larger amount of detail levels can still be useful.

OTHER CONTEXT LABELS

As with location and activity context, other categories of context can also be represented by sets of labels. For example the current time, which can be described using the exact time, by a part of the day, the current season, etc.. This research however focuses on location and activity context, and therefor other context categories are not further discussed.

3.2. DEFINITION OF PHASE 2

For the first phase, sub research question one (SRQ1.1) is answered in the literature survey presented in Chapter 2. Sub research question two of the first phase (SRQ1.2) can be answered using the results from Chapter 2.

Multi-label context recognition is a promising technique for recognizing multiple aspects of context. It can be seen as a tool which can be used for refining context, but does not refine context itself. Since this research is about refining context, this technique will not be further investigated within the research of phase two. Within the rest of this research, context recognition will only predict one label.

There are two important possible directions for further research: improving or extending context recognition. Extending context has a lot of possible research subjects to offer and due to the limited experience with machine learning techniques, improving existing context recognition algorithms is expected to be unrealistic within the limited time for this research. Therefor, this research focuses on exploring the possibilities for refining context by adding more information, which can be done by adding more data (sources) and recognizing multiple categories and levels of context. Another possible way to refine context is to combine multiple categories of context, because different categories are expected to provide valuable information for other categories. For example when the activity recognized is *walking* and the location is *supermarket*, the activity could be extended with *shopping*.

From this, the following three subjects are distilled:

- Which data sources are important for recognizing different types of context categories and can thus possibly be used to extend the context information?
- Context can be refined by adding more detailed context information. But how well can such detailed context level be recognized?
- Combining context categories is expected to refine the context. To what extent can one context category improve context recognition for another category?

These subjects will be investigated in phase two of this research and the following sections discuss these subjects. For each subject of research one or two new sub research questions are defined, which will be answered in phase two of this research.

For this research, Location is subdivided into two levels of detail: the *basic level* and the *detailed level*. The basic level locations are the main locations visited, for example *home* and *office*. The detailed level location is typically a 'room level' description such as bathroom or kitchen.

3.2.1. DATA SOURCE IMPORTANCE

For context recognition within this research, a large amount of smartphone data sources are used. As the set of data sources used to detect context will be increased, it is important to know which data sources have the greatest influence on the accuracy of context detection. Including data sources not relevant for detecting a context category can add noise to the classifier, which results in a decreased accuracy. Such data sources can better be left out of the feature set for the classifier. Sub research question one of the second part (SRQ2.1) is used to investigate which data sources have the biggest impact on the context categories Location and Activity. The absence of impact from a data source on the context detection does not mean the data source is useless for context detection, since it could have impact on context categories not considered, or specific contexts not taken into account.

SRQ2.1 *Which data sources have the largest impact on the detecting results for context categories Location and Activity?*

3.2.2. MULTIPLE LEVELS OF DETAIL

A context refinement could be established when a more detailed level of context will be recognized. For this subject, the possibilities will be explored to detect a more detailed level of context for the Location category. Of course, context recognition does not have to be limited to two levels of detail, but only two levels are chosen to gain insight in how well a detailed level location can be predicted.

The goal is to explore the possibilities and not to build a perfect solution, and for that reason the input for the context classifier does not have to be optimized. Therefor, only raw smartphone data sources are used. Sub research question 2.2 (SQR2.2) is defined to be able to compare basic level with detailed level Location classification:

SRQ2.2 *To what extent can basic level user context be determined for context category Location, using raw smartphone data sources?*

As the baseline is set for detecting basic level Location user context, detailed level user context can be determined and compared to the results for the basic level detection. The sub research question for investigating the possibilities of detecting the detailed level user context is formulated as follows:

SRQ2.3 *To what extent can detailed level user context be determined for context category Location, using raw smartphone data sources?*

3.2.3. COMBINING CONTEXT CATEGORIES

One benefit of detecting multiple context aspects is that together they can potentially add information to the context or improve the accuracy of the detected context. Often activities are performed in specific locations, for example sleeping in the bedroom and having dinner in the kitchen. Sub research question 2.4 (SRQ2.4) is used to investigate whether this expected benefit will be recognized.

SRQ2.4 *To what extent can Location context information improve the determination of the context category Activity?*

To be able to make conclusions about the added value of knowing the location for activity recognition, first the accuracy of activity recognition without knowing the location has to be determined. A significant improvement in accuracy when the location is added as input to the activity classification is expected. The maximum improvement shall depend on the association between the location labels and the activity labels for the dataset used, as a strong association gives predictive power to the location label. Sub research question 2.5 is used to measure the association between these two types of labels:

SRQ2.5 *To what extend are Location and Activity associated with each other for the dataset used to recognize the context category Activity?*

3.3. RESEARCH CONTINUATION

The subjects of research for phase two described within this chapter are chosen because they supposed to provide valuable information about how context recognition can be refined. Due to the explorative nature of the defined research, the sub research questions for phase two will be answered using experiments, which will be discussed in the next chapter.

II

PHASE 2

4
METHODOLOGY

In Chapter 3 a new set of sub research questions for the second phase of this research is presented. This chapter describes how these sub research questions will be answered. Any form of research to answer the sub research questions requires data to analyze. The collection of data and analysis on it are considered two separate parts and are therefor discussed separately within this chapter.

4.1. SCOPE
This research is an initial exploration. Due to a limited timeframe of the research, only a exploratory answer can be given to the (sub) research questions. Also due to this limited timeframe and for privacy reasons only data collected by the student is used. Because everybody behaves differently, the results of this study are expected to differ when using data collected by someone else. The target of this research is explorative, and not to have a (near) perfect solution. That is also the reason why the accuracy of different classifiers are not compared for their performance or score, and the used classifiers are not extensively optimized.

4.2. COLLECTING DATA
Different methods can be applied to give an answer to the research questions, but each method requires to have smartphone data to analyze. Existing data sets that could be used are searched for and are presented in Chapter 2. Since none of them are labeled with multiple levels of detail and the available data sources are not sufficient, these public available datasets cannot be used. Consequently data has to be gathered for this research. Building a model to generate this data allows to have a large amount of data, but a real life like model is very hard to make. Does a result say something about the data and algorithms or about the model?

Because having real life data is important for this research, real data is captured on a smartphone. Preferably a large group of people will be used to capture and label data for long period of time (say weeks to months). Unfortunately it is a big challenge to find people to label context at the required detail level, since it will require a lot of effort to consequently entering the labels when context changes. The correctness of the labels could be questionable because people may make mistakes without making annotations. Finally, privacy could get a problem because a lot of private data such as location information and visible WiFi access points will be captured. Although a large amount of data could be hashed, a very large test group is required to not being able to trace back data to a test group member.

For these reasons it is decided to only capture data on the smartphone of the student. Consequently, the results by using data from other people are expected to differ. Due to the explorative nature of this research, it is not considered a problem as long as it is taken into account during this study.

Although for training a context recognition classifier a large amount of data is preferably available, collecting data for about ten weeks is considered long enough to capture the most important different contexts and have a reasonable set of data to answer the sub research questions. The amount of data is, however, too low for training a competitive classifier with a fairly good accuracy. For such a target the data also has to be collected using a large number of volunteers to be able to generalize the classifier. Because the data to collect is privacy sensitive information, since it contains, amongst others, location information and and human behavior patterns.

4.3. ANALYSIS

4.3.1. DATA SOURCE IMPORTANCE (SRQ2.1)

Different *feature selection* or *feature subset selection* methods could be used to filter out features that do not significantly impact the accuracy of the detection methods. For *SRQ2.1 (Which data sources have the largest impact on the detecting results for context categories Location and Activity?)* however, the impact of different data sources need to be investigated, each with one or more features. The purpose of this research question is not to filter out those features that have little impact on the accuracy, but to compare the impact of different data sources. One way to do this is using the statistics produced by feature selection algorithms to select the most important features.

According to Kohavi and John, and Guyon and Elisseeff, feature selection methods can be categorized into *filter methods*, *wrapper methods*, and *embedded methods* [KJ97; GE03]. Filter methods use a statistic analysis as a preprocessing step before training the classifier and this algorithm is independent of any learning algorithm. Wrapper methods are algorithms that compare the predictive power of a classifier when using different subsets of features. In contrast with filter and wrapper methods, which are preprocessing steps in the training process, embedded methods are built into the classifier and are therefor also specific for the algorithm used.

Although wrapper methods and especially embedded methods can perform well for feature selection, filter methods are relatively simple and they suffice for calculating an indication of the importance of smartphone data sources. Filter methods are often based on associations between features and the classes used in a dataset. Since that part is of most interest for determining which are the most important features for the given dataset, only statistics is applied to the given dataset to calculate for each feature the association with a) the building-level location class, b) the room-level location class, and c) the activity class.

4.3.2. MULTIPLE LEVELS OF DETAIL (SRQ2.2 & SRQ2.3)

To answer sub research questions *SRQ2.2 (To what extent can basic level user context be determined for context category Location, using raw smartphone data sources?)* and *SRQ2.3 (To what extent can detailed level user context be determined for context category Location, using raw smartphone data sources?)*, a context recognition classifier will be build. Context recognition has been implemented using ontology- rule-based algorithms, but by far machine learning techniques are used. Because this proves machine learning techniques work well for context recognition, and the student has more experience with those than ontology- and rule-based algorithms, the context recognition classifier will be build using machine learning techniques.

For this research standard machine learning techniques will be used as the goal is not to produce a (nearly) perfect system, but to explore the possibilities of using machine learning techniques for the purpose mentioned earlier. Since the goal is to predict classes, techniques such as (multivariate) linear or polynomial regression are not applicable.

Kotsiantis reviewed different supervised classification techniques, namely Decision trees, Naive Bayes, Neural Networks, K Nearest Neighbors (knn) and Support Vector Machine (SVM) [KZP07]. He ranked different aspects of these algorithms such as general accuracy, dataset size in terms of number of features and number of examples, and tolerance for irrelevant features or missing data. The dataset to use is small with a relatively large number of features. The purpose of the classification requires a reasonable accuracy, but highly optimized algorithms are not necessary. When the techniques discussed by Kotsiantis are compared for their support for these aspects, decision trees seem to be the most appropriate. Therefor, decisions tree will be used as a context recognition classifier.

To answer the sub research questions *SRQ2.2* and *SRQ2.3*, two machine learning sessions are used in which a decision tree with a) the data labeled with the building level location (basic context) and b) the data labeled with the room level location (detailed context) will be trained. The training and test data sets are randomly selected from the original data set where 80% is used for the training set and 20% is used for the test set. The results from these two sessions qualify how well the building and room level locations can be predicted using raw smartphone data.

A large number of smartphone data sources are captured, from which most are continuous (floating point) numbers, but some of them are categories (see Chapter 5). Unfortunately, machine learning algorithms typically do not handle categorical values well. A technique called One-hot encoding can be used to transform a categorical feature into multiple numerical features. A feature is added for each category, where the value is set to one if it is the category from the original feature, or 0 otherwise.

4.3.3. COMBINING CONTEXT CATEGORIES (SRQ2.4 & SRQ2.5)

For sub research question *SRQ2.4 (To what extent can Location context information improve the determination of the context category Activity?)*, the data set is labeled using the activities and the accuracy is determined using a Decision tree. Then the room- and building-level location is subsequently added to the original dataset, where the location ID representing the location label is transformed into multiple features using One-hot encoding. Then the accuracy is again determined for the dataset with the room-level location and for the dataset with the building-level location, using the same Decision tree algorithm. An increase in accuracy when the room level context label is incorporated in the dataset is a measure for the extent to which activity recognition can be improved using location information.

Calculating the association between two categories, which is required to answer sub research question *SRQ2.5 (To what extend are Location and Activity associated with each other for the dataset used to recognize the context category Activity?)*, can be done using Cramer's V, a method for calculating the association between two categorical variables. Since for answering sub research question *SRQ2.5* the association between pairs of labels (activity and location) has to be calculated, Cramer's V is applicable for this purpose. Cramer's V is based on the Pearson's chi-squared test (χ^2) and has as result a value between 0 and 1, where 0 means no association and 1 means full association. Its formula is given in Equation 4.1

$$V = \sqrt{\frac{\chi^2}{n \cdot (min(k,r) - 1)}} \tag{4.1}$$

where n is the total number of observations, k is the number of columns, r is the number of rows, and $min(k,r)$ selects the smallest of k and r.

When the association between two data sources need to be calculated, first a contingency table is calculated, which is a table which holds the number of occurrences for all combinations of categories from vectors A and B. For this table then the *Pearson's chi-squared statistic* (χ^2) value is calculated, which represents the extent to which two categorical variables are dependent to each other.

Cramer's V works well for calculating the association between two categorical values, but due to its symmetric nature it cannot calculate the predictive power of one variable predicting the other. The *uncertainty coefficient*, also called Theil's U, does allow to calculate this predictive power. Specific to our purpose it allows to calculate the predictive power of the location label to predict the activity label and vice versa.

To answer sub research question 2.5, both Cramer's V and the uncertainty coefficient are calculated between the building-level location and the activity as well as the room-level location and the activity. Comparing the results of the two outcomes gives insight in the predictability of activities for different levels of detail for location context.

4.4. VALIDATION

Data is collected for ten weeks in spring. The period of the year when data is collected can affect the measure results. For example the time of sun rise and dawn can have effect on the data captured from a light sensor. Also people likely behave differently in winter than in summer with regard to the activities being performed and the locations visited.

Apart from the different time of year where data can be captured, the used dataset only contains data gathered by one person, and therefor will likely not generalize well for other people because their performed activities are probably different and the visited locations certainly are. Also their behavior patterns and how the smartphone is used by and carried with the owner differs between persons. Not only differences between smartphone users will affect the generalizability, but also differences between the smartphones used can influence the accuracy. Saha et al. noted that microphone and WiFi fingerprints differ very per smartphone (model).

This all makes the results of this research probably not reproducible for data collected by other users on other smartphones. When using the same dataset, the results will be reproducible though. As mentioned at the beginning of this chapter, the goal of this research is not to build a (near) perfect context recognition solution, but to explore the possibilities for refining context recognition on smartphones. That models used within this research are not generalizeable is known upfront and was a conscious choice.

Data from different data sources captured at a certain time do not represent the state at exact the same time due to differences in sampling, sequential sampling of data sources, buffering of data sources by the operating system for energy saving purposes, or low-pass filters applied to the data sources. These differences can make data sources not represent context at exact the same time. Data sources which are thought not to change very fast are sampled once per capture, but other data sources are sampled during halve a minute, to better accommodate for variations in time. The variations in time are expected to have effect mainly when changing context because then measurements can quickly change, for example when the user stops walking. Also a time inaccuracy is introduced with changing the context labels, which can affect the combination of context data with labels if this inaccuracy is not accommodated for. For this, a margin before and after changing the context labels is used in which a capture is not annotated with a context label. This margin is five seconds after the start of the label session, and thirty seconds before ending the session. The start margin is chosen so that there is enough time to actually start with the labeled context after labeling. The end margin is larger because it often takes a little bit longer to change the context labels when context changes.

Before the data gathering app is used collect data, its data generated by sensors is validated using debug sessions and temporary storage within the app. The validity of the data is examined and combined with this also the hardware is checked for correctly delivering data. Validation of collected data is done manually using the daily send data e-mail.

5
Collecting data

Context data is collected on a smartphone, using an app which automatically collects data from different data sources in certain intervals. Each set of data collected at the same time is called a *capture*.

Another smartphone is used to label the context. Another app has the ability to start and stop sessions with one or more context labels. Both apps send their data every day by e-mail and these data files are later combined with the label files. This context labeling app is not used on the smartphone which which is gathering data to prevent influencing data captures with entering labels. It runs on a second smartphone, which is only used for this purpose. smartphone

5.1. Capturing sensor data

5.1.1. Which data is captured

A variety of data sources are captured during a capture. Only data sources available on a Samsung Galaxy S7 are used, since this it the type of smartphone used by the student which captures the smartphone data. Only low-level data is used, which is data directly received from a data source. No extensive algorithms are used, and no data sources which are inherited by or result from context reasoning are used.

Varying data sources

Some data sources provide data which can change significantly within a few seconds. For example data generated by an accelerometer sensor. Consequently a single capture of such a data source might only represent the state for about a few milliseconds, but will provide rather a low amount of information about the context. Instead of one snapshot, data will be gathered for about 30 seconds and for this recording the following metrics, which are called Sensor metrics within this document, will be gathered:

- first value of the capture
- last value of the capture
- average value
- median value

- maximum value
- minimum value
- standard deviation of the value

These metrics are expected to provide valuable information about the recording, and enables saying something about for example change over time, while limiting the amount of storage needed and the amount of features produced. Thereby for this research only data shall be used which is not heavily processed, and the Sensor metrics fit this requirement because of their simplicity.

DATA SOURCES

The following table shows which data sources are captured and for each data source which data is determined and stored.

Data source	**Elements**
Capture date and time	Capture start and end date and time
Accelerometer	Sensor metrics for X, Y and Z axis
Game rotation vector	Sensor metrics for X, Y and Z axis
Gravity	Sensor metrics for X, Y and Z axis
Gyroscope	Sensor metrics for X, Y and Z axis
Linear acceleration	Sensor metrics for X, Y and Z axis
Magnetic field	Sensor metrics for X, Y and Z axis
Rotation vector	Sensor metrics for X, Y and Z axis
Light	Sensor metrics for the amount of light
Pressure	Sensor metrics for the amount of pressure
Proximity	Sensor metrics for the amount of proximity
	Number of events captured during the capture
Step detector	Number of steps during the capture
Significant motion	Number of significant motion events during the capture
WiFi	BSSID's and signal level of visible WiFi access points
	BSSID of the connected WiFi access point (if any)
Battery	Whether the phone charging or not
	Battery percentage
GPS	Latitude and longitude
	Speed
	Altitude

Table 5.1: Data sources captured on the smartphone

The data provided by a part of these data sources speak for itself, but others, especially for the motion sensors, need a little explanation.

Accelerometer The accelerometer measures acceleration of the smartphone on each of the X-, Y-, and Z- axis. The real direction and value of the smartphones' acceleration is represented by the resulting vector of these three values. Usually, an accelerometer Integrated Circuit (IC) provides acceleration data including the value of gravity. When a smartphone is at rest, the accelerometer should show the same results as the Gravity sensor.

Gravity The value of gravity measured on each of the X-, Y-, and Z- axis. These values can be acquired using low-pass filtered accelerometer data [Ande].

Gyroscope A gyroscope measures the angular speed on each of the X-, Y-, and Z- axis and is often used to detect orientation changes.

Linear acceleration The acceleration measured on each of the X-, Y-, and Z- axis, not including gravity. The results can be acquired using high-pass filtered accelerometer data [Ande].

Magnetic field The magnetic field measured on each of the X-, Y-, and Z- axis, and can for example be used as a compass. A magnetic field sensor, called a *magnetometer*, can be used to represent the orientation of the smartphone.

Rotation vector The rotation vector is a virtual sensor which uses data from other sensors to provide the angular rotation around the X-, Y-, and Z-axis. Its purpose is comparable with the gyroscope, but the representation is more accurate and computational efficient.

Game rotation vector The game rotation vector is the same as the rotation vector, but the rotation vector uses the geomagnetic field, whereas the game rotation vector uses a relative reference point. The latter allows for more accurate relative rotations, which is preferable when the orientation is used to control the smartphone view, for example when playing games or watching virtual reality.

Light The ambient light measured. The sensor is often located at the front of the smartphone and its output is for example used to adjust the screen brightness under different environment light conditions in order to optimize the visibility of content shown.

Proximity A proximity sensor is used to detect the presence or absence of objects nearby the front of the smartphone. It is for example used to turn of the touch functionality and/or screen during a phone call or to turn off the screen when a cover is closed. Some sensors provide the measured distance to a detected object and others provide only a far/near status [Ande].

Significant motion The significant motion sensor generates a so called *Trigger event* for Android smartphones, when it detects a relatively large sudden motion, for example a shake [Andf]. It can be a hardware solution in the form of an IC, or a software solution using typically the accelerometer [Andb].

Step detector Using other sensors, the step detector generates a trigger event when a step is recognized.

FEATURES

Most features used are directly derived or copied from the data captures as described in Table 5.1, but a few are determined while processing the data files and will be described in more detail here.

Time The following features are added to represent the time of the capture:

- MinuteOfDay
- HourOfDay
- DayOfWeek

As can be noted these features are (single) numbers which represent a time aspect of the capture, with a number which can be compared with other features and can be used to train a context recognition model. The *MinuteOfDay* and *HourOfDay* features are added because they are expected to be important for finding daily human behavior patterns. For example sleeping at night and having breakfast in the morning. The *HourOfDay* feature is the same as the quotient of dividing *MinuteOfDay* by 60. The *HourOfDay* feature is thus redundant, but is added to more easily tracing back captures.

The *DayOfWeek* feature is added to allow the model to detect human behavior patterns which depend on the day of the week, for example to differentiate between work days and weekend days.

WiFi The WiFi data source the available access points for a capture will be used as features. To be able to include such features in the dataset for context recognition, a One-hot encoding-like mechanism is used to transform the available access points into separate features. Instead of just using 1 or 0 to denote the availability or absence of an access point, the signal strength (RSSI value) is used for available access points and for unavailable access points a RSSI of -100dBm is used, which represents no signal at all.

A disadvantage of a One-hot encoding-like mechanism is that each access point ever visible during a capture will be included as a feature, which results in a very large amount of features. To limit the number of features generated by this technique, only the top 150 most seen access points during the data collection period are used as features.

Next to the visible access points, also a feature is added which denotes whether connected or not to a WiFi network.

Data is gathered using a smartphone app which automatically acquires data from all data sources specified at certain intervals. Since the smartphones which are available for testing purposes all run Android, the data gathering app will be targeted for that platform. Preferably data is captured very often, so that the amount of data within the period of data collection is as large as possible. Unfortunately, a continuous capture of sensor data requires the smartphone to keep running and sensors to be active all the time, which will drown the battery very fast; the battery of the device under test will not even last for half a day, without using the smartphone intensively. For functionality such as capturing data in intervals, Android has background tasks. A disadvantage of using background tasks is that captures cannot be performed very frequently or at fixed intervals due to Android's energy saving features introduced since Android 6.0 (API level 23), when *Doze mode* and *App standby mode* was introduced [Anda]. Doze mode lets background tasks be deferred to maintenance windows which are scheduled with intervals of several minutes, thereby minimizing the time the processor needs to run and optimizing the time where the smartphone can be put into sleep mode. App Standby Mode lets apps use less resources the longer the app is not used. Android 7 introduced Light-Doze mode, a lightweight (less restricted) form of Doze, which is entered directly when the screen is turned off [Andc]. Even more restrictions to background tasks execution is introduced in android 8.0 and 9.0, and all these features and limitations result in varying intervals between captures of a few seconds up to tens of minutes.

Data collected is automatically send to the student by e-mail on a daily basis. This allows for easy collection, processing and storage of the data files without the need to connect the smartphone to a PC, with the risk of affecting the data captures because this can introduce user behavior specific for the data collection and processing.

5.2. LABELING CONTEXT

To be able to label the context, an Android app is used in which live sessions with sets of labels can be started and stopped. During the day, each time a labeled activity or location changes the current session is manually stopped. A new session can then be started with a new set of labels. Only activities and locations are labeled which are expected to be often used and have a large enough duration to have a likely data capture during the session. Labels for different levels of detail are used together to be able to later use any of the detail levels.

Table 5.2 shows the activity labels used during the data capture period and for which data is captured within the label session. The building-level location labels annotated on the capture data are *family-home*, *in-transit*, *home*, and *office*. For building-level locations *home* and *office*, also a detailed location label is annotated and these room-level location labels are shown in Table 5.3. For other building-level locations no room-level locations are added since the data capture period was considered too short to have enough data captures per room-level location to analyze.

Activity labels				
bike	childcare	lunch	sleeping	visiting
breakfast	dinner	personal-care	study	watching-tv
car	diy	reading	toilet	working

Table 5.2: Activity context labels captured on the smartphone

Sometimes a label session is forgotten to end on time. This is annotated as a special label session so that the faulted label session can be fixed (e.g. the end time is lowered or the session is removed). This is done manually in the label files before further processing them.

Home room-level location labels			Office room-level location labels		
babyroom	childroom	office	desk	logistics	toilet
bathroom	kitchen	toilet	kantina	meetingroom	
bedroom	livingroom				

Table 5.3: Location context labels captured on the smartphone

5.3. PROCESSING AND LABELING CAPTURES

The data and label files are manually downloaded to folders on the local system. The label files are then manually inspected and corrected. Sessions can be annotated with faults, such as when a session is forgotten to stop on time or when a label is incorrect. Such sessions are edited in the file (for example by changing the end date or removing the session) and subsequently the corresponding annotation is removed.

All files in the labels and data folders are then loaded into a self written .NET application which purpose is to transform these files into a labeled data set. The inherited features are calculated based on the provided data files. For each data capture, a corresponding label session is searched for and the data capture is labeled when it is performed within the label session. To minimize the influence of timing inaccuracy between the two smartphones and the inaccuracies in the physical transition between contexts (session is started a little bit too early or ended too late), a margin is used to restrict the time in which a capture has to be performed within the label session. This margin is five seconds after the start of the label session, and thirty seconds before ending the session. The start margin is chosen so that there is enough time to actually start with the labeled context after labeling. The end margin is larger because it often takes a little bit longer to change the context labels when context changes.

6
ANALYSIS

Within the analysis, two different datasets are used which are a subset of the global dataset of features extracted from captures which are annotated with at least one label:

1. **Building-level dataset** - The dataset with examples which contain both an activity and a building-level location label.
2. **Room-level dataset** - The dataset with examples which contain both an activity and a room-level location label.

This chapter is subdivided into three sections, where each of the sections cover the analysis of one of the research subjects presented in Chapter 3. The methodology for this analysis is presented in Chapter 4.

6.1. DATA SOURCE IMPORTANCE (SRQ2.1)

For the first subject of the second part of this research the importance of different data sources with regard to activity and location recognition is investigated. The sub research question for this section is discussed in Chapter 3, and is defined as follows:

SRQ2.1 *Which data sources have the largest impact on the detecting results for context categories Location and Activity?*

To determine the importance of the different smartphone data sources for activity and location recognition, associations between each feature and the location and activity classes are calculated for both the building-level and the room-level datasets. The location and activity classes are categorical values and Cramer's V is used to compare the activity and location classes with categorical features, while *Correlation ratio*, denoted with η, is used for comparing the activity and location classes with continuous features.

Before any association between the features and classes can be calculated, any 'constant' features have to be removed, because association cannot be calculated for such features. For this, the variance for each feature of the building-level and room-level datasets is calculated. Out of 340 features, 32 of them for the building-level dataset and 45 for the room-level dataset are constant. For both datasets the average X, Y, and Z values for the

Game rotation vector, Gyroscope and Rotation vector sensors are constant, as well as the number of significant motion events, number of steps and 21 different WiFi access points. For the room-level dataset the median X, Y, and Z value of the Gyroscope and ten more WiFi access points are also constant.

For both the Building-level and Room-level datasets, the associations between each feature and the location and activity classes are calculated. Appendix A shows the results for association values larger than or equal to 0.3; features with a association value below 0.3 are considered not associated for the provided dataset and are therefor not included.

6.1.1. FEATURE RANKING FOR BUILDING-LEVEL LOCATION RECOGNITION

Table 6.1 is a summary of Table A.1 and shows an ordered list of associations between each feature and the location class from the building-level dataset. Consecutive WiFi access point RSSI value features (unique access points) within this list are taken together, because identification of specific access points is not relevant for determining which data sources are important for detecting activities and locations.

What can be noticed is that most features with high association are related to the WiFi data source, where the top of the list is dominated by the signal level of specific WiFi access points. This can be expected because WiFi access points are likely bounded to a fixed location and can hence denote a location. The smartphone is connected to WiFi for building-level locations *family-home* and *home*, but not for *in-transit* and *office*. because the majority of the examples from the dataset have either the *home* or *office* class, the high association between the location class and the WiFi-IsConnected feature can be explained.

Another notable result is the association between the battery charging feature and the building-level location class. Because the phone is mainly charged at home and not at the other locations from the dataset, this result can be expected.

The associations for the WiFi-IsConnected and Battery-IsCharging features are notably high because of user behavior patterns since this behavior is revealed at specific locations. Although it cannot be checked with the datasets used, other users are also expected to have behavior patterns for these and other features, but the variation on these patterns will vary between users.

As can be expected for a building-level location, the GPS latitude and longitude are also included in this list. GPS speed is, however, not so obvious. When calculating the mean GPS speed for each building-level location class of the dataset, the results, shown in Table 6.2, clarify why such a high association is found. Because the *In transit* location usually implies a movement, this consequently results in a significant (GPS) speed, while for the other building-level location classes the smartphone is generally not moving, at least not at the speeds for the *In transit* class.

Another notable data source present at the top of this list is the accelerometer, especially those features related to the Z-axis. Though the accelerometer is primarily used for detecting movement, the absolute values for each of the three axis can provide the orientation of the smartphone when it is not moving. The Z-axis for example can differentiate between holding the smartphone vertically or horizontally. The first is mostly true when the smartphone is being used or when it is put into a pocket, while the latter is true when the smartphone is laid down on a surface and is (mostly) not being used. When the orientation of the smartphone has such a large impact on the building-level location detection, this means that user behavior could be more important for location detection than expected

Feature	η / V *	**Feature**	η / V *
WiFi-IsConnected	0.98 *	*3 WiFi access points*	0.39
16 WiFi access points	0.61 to 0.93	Accelerometer-StdY	0.39
Battery-IsCharging	0.61 *	*7 WiFi access points*	0.37 to 0.39
5 WiFi access points	0.60 to 0.61	Accelerometer-MaxZ	0.36
GPS-Speed	0.52	*1 WiFi access point*	0.36
Accelerometer-MinZ	0.51	Gravity-MedianZ	0.36
2 WiFi access points	0.50	*1 WiFi access points*	0.35
Accelerometer-AverageZ	0.49	Gravity-StdMagnitude	0.35
Accelerometer-MedianZ	0.49	LinearAcceleration-StdMagnitude	0.35
1 WiFi access point	0.48	*2 WiFi access points*	0.34 to 0.35
Accelerometer-FirstZ	0.48	Accelerometer-StdMagnitude	0.34
1 WiFi access point	0.48	MagneticField-MinZ	0.33
Accelerometer-LastZ	0.48	Gravity-AverageZ	0.33
Gravity-MaxZ	0.47	Gravity-FirstZ	0.33
2 WiFi access points	0.46 to 0.47	Gyroscope-StdX	0.33
LinearAcceleration-LastZ	0.45	*1 WiFi access point*	0.33
LinearAcceleration-MaxZ	0.44	Accelerometer-AverageX	0.33
3 WiFi access points	0.44	*1 WiFi access point*	0.32
Accelerometer-MaxX	0.44	Accelerometer-MinY	0.32
GPS-Longitude	0.43	Accelerometer-MedianX	0.32
2 WiFi access points	0.42	MagneticField-MedianZ	0.31
GPS-Latitude	0.41	*1 WiFi access point*	0.31
Accelerometer-StdX	0.41	MagneticField-LastZ	0.31
3 WiFi access points	0.40	Accelerometer-LastX	0.31
Gravity-StdZ	0.40	Gravity-AverageX	0.30
LinearAcceleration-StdZ	0.39	*1 WiFi access point*	0.30

Table 6.1: An overview of the features with the highest association with the Building-level locations for the Building-level dataset. Features with an association value (η for correlation ratio and V for Cramer's V, for which the features are marked with a star) below 0.3 are not included because they are considered uncorrelated. Consecutive features in this ordered list for the RSSI value of WiFi access points are taken together as the specific access points are irrelevant here.

upfront.

Besides the accelerometer, also other motion sensors appear at the rank list. The gravity sensor is essentially a low-pass filtered accelerometer and due to this it will provide the same information as the accelerometer when the smartphone is not being moved. The magnetometer (magnetic field sensor) also provides information about the orientation of

Building-level Location	Mean GPS speed
home	0.0164
in-transit	8.0773
family-home	0.1498
office	0.1026

Table 6.2: The mean GPS speed calculated for each building-level location of the building-level dataset

the smartphone. It's Z-axis can also be used to differentiate between holding the smartphone horizontally or vertically. The gyroscope and linear acceleration provide information about movements of the smartphone and not about its orientation. Motion sensors are not expected to be associated with a (room-level) location because movement of a smartphone does not provide any direct information about the location, but the measurements show that a association is found, although not very strong for the gyroscope. It is expected that these motion sensors are associated with the location classes because of *human behavior patterns*: rooms have different functions which make humans behave differently. For example the standard deviation of the Gyroscope X axis, which is a measure of the amount of movement around the X axis, is significant higher for the building-level class *In transit* than the other building-level classes.

6.1.2. FEATURE RANKING FOR ROOM-LEVEL LOCATION RECOGNITION

Table A.2 shows the calculated associations between each feature and the location classes within the Room-level dataset. Again a large number of WiFi access point signal strength features appear at the top of this list, which could be explained because the signal level of an access point differs room to room. Since the battery is mainly charged in the car and in the bedroom at home, the high association for this feature could also be explained. As for the building-level location, the feature WiFi-IsConnected is also highly associated with the room-level location. This is because the smartphone is connected to WiFi at home but not at the office, and hence for the rooms at home it is connected, but for those at the office it is not.

The high associations between the GPS coordinates and the room-level location are not expected due to the typical accuracy of a few meters when not in vicinity of buildings, and the fact that the majority of the room-level classes are within a building. The median latitude and longitude do however approach the real location, though the locations often tend to represent a location just outside of the real location. Because most locations are at a side of the building, this can help for the high associations.

Just as for the building-level locations, also for the room-level locations a large amount of motion sensor features are included in the list. These features are also expected to be associated with the location class because of human behavior patterns: the smartphone is expected to be handled differently, for example the orientation of the smartphone, at certain locations. These human behavior patterns are also indicated by the association between the *MinuteOfDay* (and linear coherent *HourOfDay*) feature and room-level location classes. These associations are expected because locations are typically visited during the same time frames.

These human behavior patterns are also made clear because of the association between the battery percentage and the location class. Because the smartphone is mainly charged at night during sleep, the battery percentage decreases throughout the day and a pattern for visiting locations will result in a high association between the location classes and the battery percentage.

6.1.3. FEATURE RANKING FOR ACTIVITY RECOGNITION

Tables A.3 and A.4 show the calculated associations between the features the activity classes for respectively the building- and room-level datasets. Again those features with an association value below 0.3 are not included in this list. Globally these tables are pretty much the same, although the association values and the order of features differ of course. One important difference is that the features from the Time data source have a significant stronger association with the activity classes than with the location classes. This indicates that human behavior patterns are even more important for detecting activities.

Notable is that features which are not thought to have a direct relation with activities are highly associated, for example the WiFi and GPS features. Activities are performed at a limited set of locations, and thus location information seem to be important for (improving) detecting activities. In Section 6.3 the association between activities and locations are discussed in detail, as well as the added value of using location information to predict activities.

For detecting activities, motion sensors are thought to provide the most important information. The measurements show however that other features can be of even more interest. The relative low association values for motion sensors can be explained by looking at the activity classes that will be trained for detection: the majority of activities are *non or low physical*. Because of this the orientation of the smartphone can be of interest, but information related to movements will be less useful.

6.1.4. MOST IMPORTANT DATA SOURCES

In order to give insight into and compare the importance of the data sources investigated, each feature is categorized into five ranges of 0.2 by their association values, for Activity and Location, both Building- and Room-level datasets. The constant features, discussed at the beginning of this chapter, are not included because association cannot be calculated for those features. Then the frequency of association value categories for each dataset is calculated and shown in Tables B.1, B.3, B.2, and B.4, where the frequency distributions are shown for activity and location, for the room- and building level datasets. In Table 6.3 the sum of these four tables is shown.

Notice that that the WiFi, GPS and Battery data sources have highly associated features, followed by the accelerometer, time, linear acceleration, gravity and gyroscope. These datasources are therefor considered to have the largest impact on the detection results for context categories Location and Activity. For the game rotation vector features are found which have moderate association with the activity or location classes. The rest of the data sources have no features with an association value above 0.4 and are therefor considered less important.

Data source	**Number of features in association value range**				
	1.0 to 0.8	**0.8 to 0.6**	**0.6 to 0.4**	**0.4 to 0.2**	**0.2 to 0.0**
WiFi	39	55	90	163	153
GPS	4	1	5	1	5
Battery	3	4		1	
Accelerometer		8	45	32	3
Time		6		2	4
Linear acceleration		2	11	20	55
Gravity		1	17	50	20
Gyroscope		1	4	29	36
Game rotation vector			3	35	38
Magnetic field				40	48
Light				28	
Rotation vector				23	53
Proximity					32
Pressure					28

Table 6.3: The frequency distribution of the feature ranks per data source. For both the building- and room level data sets, the association between each feature and the activity and location classes is classified in the association value ranges shown in this table. Per data source the frequencies of feature classes are summed for the building- and room-level datasets and for activity and location recognition.

6.2. MULTIPLE LEVELS OF DETAIL (SRQ2.2 & SRQ2.3)

The goal of this section is to investigate to what extent different levels of location context can be recognized using raw smartphone data sources. The sub research questions *SRQ2.2* and *SRQ2.3* are defined in Chapter 3 but also provided here for quick reference. The *basic level* and *detailed level* are translated into respectively the *Building-level* and *Room-level* datasets.

SRQ2.2 *To what extent can basic level user context be determined for context category Location, using raw smartphone data sources?*

SRQ2.3 *To what extent can detailed level user context be determined for context category Location, using raw smartphone data sources?*

When these sub research questions are answered, based on the results a conclusion and recommendation can be made about whether it is feasible to develop more detailed location recognition algorithms. To answer the sub research questions, a Scikit learn decision tree is trained for location recognition on both the building- and room-level datasets [Bui+13]. Scikit learn uses an optimized version of the Classification and Regression Trees (CART) algorithm [lea], which look like the C4.5 decision tree algorithm.

The constant features (features with zero variance), discussed in the previous chapter, are ignored because they will not be usefull for context recognition. The accuracy, precision, recall and F1 score are calculated for each trained decision tree to be able to compare the results of these trees. In order to calculate those metrics, the dataset is randomly divided into a training set (80% of the samples from the dataset) and a test set (the remaining 20% of the dataset) using the function `train_test_split`. Because the dataset is relatively small, the `stratify` option is used to make sure the percentage of occurrences of a class in the training and test sets equals the configured split percentage (80% for the training set and 20% for the test set in this particular case). This way it is assured that each class has occurrences in both the training and test sets.

Not only these metrics are calculated, but also a confusion matrix is calculated per trained tree to give insight in how well the decision tree predicts its classes.

The training set is used to construct the decision tree and the test set is used to calculate the metrics of the constructed decision tree, as well as a confusion matrix. The term *run* is used for the process of constructing the decision tree, the training phase, and calculating the metrics for the constructed decision tree, the test phase.

It turned out that, due to the relatively small datasets and the random selection of samples from the dataset, the calculated metrics vary between separate runs. The actual metrics can therefor not be represented by a single measurement. To overcome this, 100 runs are executed and then the mean and standard deviation of the metrics are calculated over these runs. The variation does not only affect the calculated metrics, but also the confusion matrices. Therefor for each of the 100 runs the confusion matrices are summed.

First consider the calculated confusion table for location recognition on the building-level dataset, which is presented in Table 6.4.This dataset contains four different classes, for which the decision tree is trained for using the training set. The decision tree is then used to predict the classes of the examples in the test set. The confusion matrix shows for each true class (shown in the rows) how often each possible class (shown in the columns) is predicted totally within the 100 test runs. It turned out that, for example, the true class *home* was totally predicted 47564 times correctly, but 27 times the wrong class *in-transit* was predicted. Overall the confusion matrix shows that the building-level location can be predicted reasonably good. One thing that can be noted is that largest error is for the true class *in-transit* where classes *family-home* and *office* are often predicted, and vice versa.

True classes \ Predicted classes	*home*	*in-transit*	*family-home*	*office*
home	**47564**	27	7	2
in-transit	36	**2463**	178	223
family-home	13	183	**3102**	2
office	0	233	7	**16560**

Table 6.4: Confusion matrix for location prediction using the building-level dataset. Summed over 100 training and test sequences.

True classes \ Predicted classes	*home-childroom*	*home-kitchen*	*home-bedroom*	*home-toilet*	*home-bathroom*	*home-babyroom*	*home-livingroom*	*home-office*	*office-toilet*	*office-meetingroom*	*office-desk*	*office-logistics*	*office-kantina*
home-childroom	**1155**	84	40	61	213	122	88	28	0	1	5	3	0
home-kitchen	102	**2567**	32	79	139	23	433	23	0	0	0	0	2
home-bedroom	40	27	**32100**	19	46	11	31	20	0	0	6	0	0
home-toilet	82	89	8	**95**	43	16	163	1	0	3	0	0	0
home-bathroom	210	140	57	30	**387**	87	72	10	0	2	5	0	0
home-babyroom	160	33	20	19	79	**335**	4	39	2	2	2	4	1
home-livingroom	59	438	11	147	129	6	**4910**	0	0	0	0	0	0
home-office	25	5	49	0	6	26	0	**1587**	0	0	2	0	0
office-toilet	0	0	0	0	0	0	0	1	**899**	41	103	45	11
office-meetingroom	3	3	1	0	2	4	1	0	64	**1645**	30	39	8
office-desk	2	0	1	1	0	1	2	0	85	30	**6706**	39	33
office-logistics	3	4	2	2	2	1	0	1	42	57	36	**1294**	56
office-kantina	3	1	0	0	0	17	2	0	11	3	21	55	**487**

Table 6.5: Confusion matrix for location prediction using the room-level dataset. Summed over 100 training and test sequences.

The calculated confusion table for location recognition on the room-level dataset is provided in Table 6.5. This matrix shows that room-level location recognition also performs well, although the error is larger than for building-level location. Because of the larger amount of classes and the fact that the classes are physically closer to each other than for the building-level locations, the larger amount of error could be expected. That those classes which are physically closer to each other give more error can be seen in the matrix: the largest amount of error is generated for those classes within the same building (*home* or *office*). The class unbalance discussed earlier can be seen easily within this matrix, but it is also clear that the unbalance does not severely affect the training of the decision tree. An unbalanced could lead to a misleading accuracy when a dominant class is predicted often, but the confusion matrix shows that this is not the case. It can also be noted that for each true class except *home-toilet*, the majority of predictions are correct.

For each of the 100 test runs the metrics *accuracy*, *precision*, *recall*, and *F1 score* are also calculated. The function to calculate the last three metrics is `precision_recall_fscore_support`, provided by Scikit learn. To calculate the precision, recall and F1 score for a multi-class prediction, an averaging strategy has to be chosen. For the target prediction, the *weighted* strategy is chosen because it can deal with an unbalanced dataset. For each of the metrics, the mean and standard deviation are calculated over the 100 runs performed, and are shown in Table 6.6.

	Accuracy		**Precision**		**Recall**		**F1 score**	
	avg	**std**	**avg**	**std**	**avg**	**std**	**avg**	**std**
Building-level	0.9879	0.0045	0.9874	0.0042	0.9871	0.0045	0.9871	0.0045
Room-level	0.9181	0.0100	0.9200	0.0097	0.9181	0.0100	0.9175	0.0097

Table 6.6: Decision tree metrics for location prediction using the building- and room-level datasets

The building-level location can be predicted with high accuracy, precision, recall, and F1 score, which all are between 98.7% and 98.8%. The standard deviation of these metrics is reasonably low with a value below a half percent. Correct prediction of the room-level location is a little bit harder, as can be seen by the metric values which are between 91.7% and 92.0%. The standard deviation for the results of the room-level predictions are also larger with a value around one percent.

Conclusion The location can be recognized with high accuracy, precision, recall and F1 score, especially the building-level location. The number of classes for the building-level location is smaller than for the room-level dataset, which is a benefit for the building-level location recognition decision tree and can (partially) be the reason for the better results. In reality however, if more detailed levels are used, also more classes can be identified due to hierarchical character of location classes. This means that a larger number of classes for a more detailed level is a realistic situation.

It is important to note that the results are based on an over idealized situation because the dataset only contains data from one person, which results almost sure in an overfitted decision tree. Including data from other persons will definitely affect the results negatively. This problem is identified upfront and it was a conscious choice to only use data from one person to explore the possibilities for recognizing a more detailed location. The latter seems to be possible according to the results of this section.

6.3. COMBINING CONTEXT CATEGORIES (SRQ2.4 & SRQ2.5)

The purpose of this section is to investigate whether combining information from different context categories can help improve the context recognition result. The sub research questions that will be answered in this section are discussed in Chapter 3, and are provided here again:

SRQ2.4 *To what extent can Location context information improve the determination of the context category Activity?*

SRQ2.5 *To what extend are Location and Activity associated with each other for the dataset used to recognize the context category Activity?*

6.3.1. IMPROVING ACTIVITY RECOGNITION USING LOCATION INFORMATION

On same way as in Section 6.2, a decision tree is trained with the activity classes from the building- and room-level datasets. The difference between these two datasets for activity recognition is that the building-level dataset is a superset of the room-level dataset. The room-level dataset only contains activities performed at any of the detailed level locations at *home* or at the *office*, where the building-level dataset also contains activities performed with building-level location classes *in-transit* and *family-home*.

The confusion matrices for activity recognition on the room- and building-level datasets are respectively shown in Tables 6.8 and 6.7. Notice that overall the decision tree seems to work well, but some classes seems to be harder to predict correctly. Especially combinations with the *childcare* class, but also with *sleeping*, *reading*, and *watching-tv*, have a significant amount of error which decrease the overall result.

The results from the decision trees trained for activity recognition so far will be used as a baseline for comparing it with results when the location information is included as input to the classifier. On the same way as without location information included, a decision tree is trained with the activity classes from the building- and room-level datasets. But now the location class (building-level for the building-level dataset; room-level for the room-level dataset) is one-hot encoded into new features and are added to the building- and room-level datasets. The decision tree algorithm is trained again for activity recognition for 100 times and the results are used to calculate the mean and standard deviation of the accuracy, precision, recall, and F1 scores. These metrics, together with the metrics from activity recognition without location information, are shown in Tables 6.9 and 6.10 for respectively the building- and room-level datasets.

For activity recognition on the building-level dataset, no improvement is found when the building-level location is added to the dataset. For the room-level dataset, however, an improvement between five and six percent is found for the metrics calculated.

Conclusion With these measurements *SRQ2.4* can be answered. For the room-level dataset, the accuracy, precision, recall, and F1 score of activity recognition increases with five to six percent when location information, represented by the room-level location class, is added to the dataset. For the building-level dataset no improvement can be found when adding the building-level location to the dataset for activity recognition.

True classes \ Predicted classes	*diy*	*breakfast*	*sleeping*	*reading*	*toilet*	*personal-care*	*childcare*	*watching-tv*	*lunch*	*dinner*	*working*	*study*
diy	**525**	1	0	2	24	40	289	50	6	13	9	41
breakfast	5	**718**	2	2	37	57	101	1	4	1	14	58
sleeping	2	5	**21351**	1325	0	4	1903	106	0	2	0	2
reading	6	0	1306	**1487**	2	19	227	144	0	0	1	8
toilet	28	26	0	2	**938**	51	145	73	40	7	237	53
personal-care	56	53	0	14	54	**357**	268	34	9	14	16	25
childcare	250	110	1904	243	115	299	**3381**	668	41	224	43	222
watching-tv	33	3	115	120	62	37	718	**1919**	28	142	47	176
lunch	0	4	0	0	8	6	50	35	**580**	9	271	37
dinner	26	0	3	1	10	16	212	155	9	**1143**	3	22
working	8	15	0	4	224	29	37	49	285	2	**9903**	44
study	23	31	6	15	36	18	204	167	22	26	57	**2295**

Table 6.7: Confusion matrix for activity prediction using the room-level dataset. Summed over 100 training and test sequences.

True classes \ Predicted classes	*diy*	*car*	*breakfast*	*sleeping*	*reading*	*toilet*	*personal-care*	*childcare*	*watching-tv*	*visiting*	*lunch*	*dinner*	*working*	*bike*	*study*
diy	**566**	1	2	1	2	17	42	282	27	0	6	14	9	0	31
car	3	**1710**	1	4	1	14	2	17	2	74	7	0	127	34	4
breakfast	2	6	**723**	3	2	20	59	109	7	0	1	1	7	0	60
sleeping	2	7	2	**21408**	1248	3	16	1914	89	0	1	0	0	0	10
reading	9	0	0	1321	**1464**	1	10	245	138	0	0	1	0	0	11
toilet	24	21	28	8	2	**767**	59	115	67	2	28	10	413	5	51
personal-care	51	2	43	21	15	48	**328**	285	39	0	10	12	15	0	31
childcare	251	20	104	1948	241	86	278	**3406**	662	4	42	222	18	0	218
watching-tv	27	0	2	123	113	41	39	718	**1952**	0	32	158	33	0	162
visiting	0	61	0	0	0	2	0	4	0	**3186**	0	0	13	28	6
lunch	2	4	2	0	0	5	3	61	43	0	**508**	10	337	0	25
dinner	14	0	0	0	1	17	17	231	163	0	10	**1137**	1	0	9
working	11	118	13	0	0	422	22	24	45	19	345	0	**13912**	108	61
bike	0	15	0	0	0	5	0	0	0	14	0	0	90	**467**	9
study	22	1	43	18	20	30	18	245	174	18	17	16	82	4	**2192**

Table 6.8: Confusion matrix for activity prediction using the building-level dataset. Summed over 100 training and test sequences.

6.3.2. ASSOCIATION BETWEEN LOCATION AND ACTIVITY

In order to answer *SRQ2.5* the association between the activity and location classes is calculated for both the building-level and the room-level datasets, using Cramer's V and the

Location	Accuracy		Precision		Recall		F1 score	
	avg	std	avg	std	avg	std	avg	std
No	0.7697	0.0140	0.7725	0.0133	0.7697	0.0140	0.7691	0.0133
Yes	0.7696	0.0149	0.7733	0.0140	0.7696	0.0149	0.7687	0.01466
	-0.0001		+0.0008		-0.0001		-0.0004	

Table 6.9: Decision tree metrics for activity prediction comparison with and without inclusion of location information, using the building-level dataset

Location	Accuracy		Precision		Recall		F1 score	
	avg	std	avg	std	avg	std	avg	std
No	0.7508	0.0159	0.7538	0.0155	0.7508	0.0159	0.7501	0.0153
Yes	0.8094	0.0160	0.8115	0.0157	0.8094	0.0160	0.8092	0.01658
	+0.0506		+0.0577		+0.0586		+0.0591	

Table 6.10: Decision tree metrics for activity prediction comparison with and without inclusion of location information, using the room-level dataset

uncertainty coefficient.

For the room-level dataset the association calculated using Cramer's V is 0.67. Using the uncertainty coefficient, the calculated value of association for predicting the activity using the location class is 0.71 and for predicting the location using the activity class this value is 0.65. These values indicate that the room-level location and activity are associated, although not very strong, and that activities can slightly better predict locations than locations can predict activities. An explanation for the latter is that activities are more often performed on a single location and thus a single location is linked to multiple activities. This makes the predictive power of an activity predicting the location larger than the predictive power of a location predicting the activity.

For the building-level dataset an even larger difference between the directions of the uncertainty coefficient for the activity and location classes is found. For activities predicting the location the result is 0.95 whilst for locations predicting the activity it is 0.40. Since only four building-level location classes are identified and a multiple of activities classes, and because specific activities are often linked to a singe building-level location, this could be explained.

Conclusion The results show that for the datasets considered, the location and activity classes are associated, although the value varies between the building- and room-level datasets and between the location and activity classes.

Apart from the added value of including two categories into the context, the results from this section show that combining context categories can improve the accuracy of predicting these classes separately. It has to be noted though that the results will likely differ for other subjects and that the improvement for activity recognition when using the location class as a feature might be a little overrated because the location class has to be predicted itself, thereby introducing inaccuracy.

7

Conclusions and future work

7.1. Conclusions

The first phase of this research consists of an exploration of the concept of (user) context, what scientific work is done related to context recognition, and how the second part of the research can be arranged to contribute to the subject of refining context. The definition of context and taxonomies are described in Chapter 2, which answers sub research question one of the first phase (SRQ1.1). It is found that most of the related work with regard to context recognition focuses on the *Activity* category. Less often related work focused on the *Location* category and occasionally *Personal*, *Social* and *Environment* categories are studied. All literature found for context recognition used a limited number of features and thus only a few data sources, but also the amount of classes the classifier is trained for is limited. The classes used in the classifier are rather basic and high level. For activity recognition often a few (physical) activities or Activities of Daily Living (ADL) are considered; for location recognition the locations are often limited to the building-level locations.

Based on the conclusions of Chapter 2, research for the second part of this research is defined in Chapter 3, which is the answer for sub research question two of the first part (SRQ1.2). Refining context can be done by improving the accuracy of existing or earlier studied context recognition applications or algorithms (deepening), or can be done by adding more context information (broadening). Because improving existing context recognition applications or algorithms is considered not a realistic target for the limited time frame of the graduation project, the focus is put on context broadening. One subject of the second part of this research is to investigate which data sources are important for location and activity recognition, since related work only takes a limited number of data sources into account.

For the datasets and classes used, the most important data sources for location and/or activity recognition turned out to be WiFi, GPS, battery information, accelerometer, current time, linear acceleration, gravity, gyroscope and game rotation vector.

Another aspect of refining context that in Chapter 3 turned out to be an important subject of research is to explore to which extent more detailed location context can be recognized. This part of the research is covered by sub research questions two (SRQ2.2) and five (SRQ2.3) and consists of investigating how well a detailed level of location context (called the room-level) can be classified, compared to the basic level (called the building-level).

For the building-level dataset, the building-level location can be predicted with a high accuracy, precision, recall and F1 score of 98 percent. The room-level location can be predicted with a slightly lower accuracy of around 92 percent. These scores are very high, but since the datasets contain data from only one person, the trained decision tree does not generalize and likely overfits the data from the dataset. The latter is identified upfront and because of the explorative nature of the research, a conscious choice is made that exploration is more important compared to a general working context recognition application. What can be concluded from this part of the research is that it is possible to have context recognition predict a more detailed location compared to the basic level used in related studies.

The third subject of research as defined in Chapter 3 is to investigate whether combining different context categories can help improving context recognition. The idea behind this is that for example when the location is known, using this location to improve the recognition of an activity, because activities are expected often to be performed at specific locations. The latter will be investigated using sub research question five of the second part (SRQ2.5), which aims to give insight to association between activities and locations. Sub research question 2.4 (SRQ2.4) on the other hand is defined to investigate to what extent activity recognition can be improved using location information.

A comparison between activity recognition without knowing the location and activity recognition with location information used as input, shows that no improvement is found when building-level location information is included as input to the activity recognition algorithm, but including room-level location information increases the accuracy, precision, recall and F1 score with five to six percent for the used room-level dataset. This is considered a significant improvement. It should be mentioned though that knowing the location is not a realistic situation: the location itself will probably have to be recognized also, which introduces an inaccuracy that will likely decrease the improvement. Though, it is seen that location recognition, at least for the datasets considered, has higher accuracy than activity recognition. When the location can be predicted with high accuracy, it might still improve activity recognition.

For sub research question 2.5 (SRQ2.5) the association between the location and activity classes is investigated. It turns out that the direction of association is important, e.g. whether an activity has predictive power for the location or whether the location has predictive power for the activity. This directional association can be calculated using the uncertainty coefficient. For the room-level dataset the result of the uncertainty coefficient is 0.71 for location to activity and 0.65 for activity to location. In both directions a reasonably high association is found. For the building-level dataset the uncertainty coefficient returns 0.95 for activity to location and 0.40 for location to activity. For this dataset considered, the activities are highly associated with the location, but the location is very less associated with the activity.

Supported by the conclusions of the sub research questions, a conclusion can be provided for the main research question, which is the following:

RQ *In what manner can the concept of 'user context' be refined using smartphone sensors and data sources?*

The concept of context can be refined by broadening, e.g. adding more context information, or by deepening, by adding more detailed information. It is found that existing studies on context recognition typically focus on one category of context. Context can thus be refined by including multiple context categories. An advantage of including multiple categories is that they can be used to improve recognition of other categories, as was found for sub research question four of the second part. A reason why this is possible is because the location and activity classes are associated with each other.

More detail can be added to the concept of context by training a classifier with a more detailed level of context. For this research a detail level of location context is investigated, which is called the room-level location, and for the dataset used this room-level location can be predicted with high accuracy. Although the classifier is not generally applicable because it is trained for only one person, and thus will likely be overfitted, the results look promising for further research. The datasets used to train the classifiers contain a large number of features from a diverse set of smartphone datasources. Although some of them are not thought to be very relevant for context recognition, and are not used in literature before, they do seem to provide valuable information for context recognition. For example the WiFi data source is found to be important for both location and activity recognition.

7.2. DISCUSSION

This research has an explorative nature and as such it has some limitations which are important to note. The data is collected only by one person and thus the analysis based on this data does probably not generalize well. Letting a group of volunteers to collect and actively label context data for a ten week period requires a lot of discipline and time from this volunteers. This in combination with the privacy sensitive information collected makes it hard to find such a group of volunteers which is large enough for analyzing general working principles. It was a conscious choice to collect data from only one person to be able to explore the possibilities of refining context and not to create a general working principle.

When data from another person, or when another another period of time or other activities and locations are used to reproduce the analysis of this research, the results will likely differ. Though the same datasets could be used to repeat the analysis of this research, which will give comparable results. A limitation of the datasets used is that the classes are not balanced, which means the occurrences of different classes in the dataset are not equal. Where possible this is taken into account, but a balanced dataset, which requires to collect more data for specific classes, would improve the results.

7.3. FUTURE WORK

Due to the explorative nature and limited time of this research, many aspects related to this research are still open for further research. One important aspect is to investigate to which extent it is possible to perform context recognition, and specifically location and activity recognition, using data collected by multiple persons, preferably for group as large as possible.

Also other data sources can be considered. For this research the majority of motion sensors are investigated, but the added value of other data sources such as Bluetooth and cellular, but also more high level data sources such as the agenda and smartphone usage characteristics, can be investigated. These smartphone usage characteristics are part of human behavior pattern analysis, which can potentially add important information to the concept of context. For example using human behavior patterns, an application can predict that a person will ride a bike from home to work at a certain time. The latter could be very valuable information for applications to become smarter and more helpful to the user, for example to warn for rain or to advice the user to leave earlier because of (expected) delays on the road.

Within this research it is investigated to what extent activity recognition can be improved when location information is included as input to the activity recognition algorithm. For this research the location information included is the location class from the dataset, but this is not realistic for an actual application because the location need to be recognized itself, thereby introducing an uncertainty. Future work can focus on how well such a two staged context recognition algorithm works. Also improving the accuracy of the separate context categories can be investigated. Location recognition can potentially be improved or made more flexible when unsupervised learning is used to find location clusters, which is something that could be investigated in the future.

Finally, future research can be dedicated to multi-label context recognition. This subject is already introduced in Chapter 3, but is not further investigated within this research. It is a promising development for a practical application of recognizing multiple aspects of context, for example classes from multiple context categories, or for classes of the same context category but with different levels of detail.

A

Feature association data

A.1. Feature vs Location association (Building-level)

Feature	η / V *
WiFi-IsConnected	0.978 *
WiFi-849ca6686d66-RSSI	0.931
WiFi-f09fc2f11b48-RSSI	0.837
WiFi-704ca588e6d0-RSSI	0.832
WiFi-704ca588e6c8-RSSI	0.804
WiFi-c4a36653f8f0-RSSI	0.798
WiFi-704ca588cd79-RSSI	0.794
WiFi-704ca588e6c9-RSSI	0.794
WiFi-62a36653f8f1-RSSI	0.789
WiFi-704ca588cd78-RSSI	0.761
WiFi-10bf48e7237c-RSSI	0.760
WiFi-344dea9a6185-RSSI	0.759
WiFi-a45d36414641-RSSI	0.676
WiFi-704ca588e6e1-RSSI	0.662
WiFi-704ca588e6e0-RSSI	0.646
WiFi-d8b6b7cc3248-RSSI	0.625
WiFi-ac22057c31c2-RSSI	0.610
Battery-IsCharging	0.607 *
WiFi-ae22157c31c2-RSSI	0.606
WiFi-38d82f12423b-RSSI	0.597
WiFi-48d343f4f3f1-RSSI	0.586
WiFi-fa8fca569800-RSSI	0.578

WiFi-880355bbc910-RSSI	0.559
GPS-Speed	0.517
Accelerometer-MinZ	0.507
WiFi-788a202dd670-RSSI	0.500
WiFi-704ca588e6e8-RSSI	0.497
Accelerometer-AverageZ	0.491
Accelerometer-MedianZ	0.486
WiFi-ccce1ed72795-RSSI	0.484
Accelerometer-FirstZ	0.484
WiFi-4c1b865b85a9-RSSI	0.482
Accelerometer-LastZ	0.481
Gravity-MaxZ	0.473
WiFi-344dea887c2e-RSSI	0.472
WiFi-6a4dea887c2f-RSSI	0.461
LinearAcceleration-LastZ	0.447
LinearAcceleration-MaxZ	0.444
WiFi-fa8fca8220a0-RSSI	0.442
WiFi-d0b2c426ea8c-RSSI	0.438
WiFi-fa8fca8ad97f-RSSI	0.438
Accelerometer-MaxX	0.436
GPS-Longitude	0.434
WiFi-d2b2c426ea8d-RSSI	0.422
WiFi-dea26600a7c7-RSSI	0.419
GPS-Latitude	0.411
Accelerometer-StdX	0.410
WiFi-704ca588cee0-RSSI	0.398
WiFi-704ca588eb48-RSSI	0.398
WiFi-461ca825e90c-RSSI	0.397
Gravity-StdZ	0.397
LinearAcceleration-StdZ	0.392
WiFi-704ca588eb49-RSSI	0.392
WiFi-704ca588eb51-RSSI	0.392
WiFi-704ca588eb50-RSSI	0.391
Accelerometer-StdY	0.387
WiFi-965330691d7c-RSSI	0.386
WiFi-001daaeeb3c0-RSSI	0.385

WiFi-624dea9a6186-RSSI	0.383
WiFi-844765ce0f9c-RSSI	0.382
WiFi-704ca588cee1-RSSI	0.381
WiFi-704ca588cd80-RSSI	0.365
WiFi-000cf65c9830-RSSI	0.365
Accelerometer-MaxZ	0.359
WiFi-6abe537c0f72-RSSI	0.358
Gravity-MedianZ	0.357
WiFi-54be537c0f71-RSSI	0.352
Gravity-StdMagnitude	0.351
LinearAcceleration-StdMagnitude	0.350
WiFi-704ca588cee8-RSSI	0.349
WiFi-98e7f4abcec8-RSSI	0.343
Accelerometer-StdMagnitude	0.338
MagneticField-MinZ	0.332
Gravity-AverageZ	0.331
Gravity-FirstZ	0.330
Gyroscope-StdX	0.329
WiFi-b0b98a60ccff-RSSI	0.327
Accelerometer-AverageX	0.325
WiFi-e81cba09db19-RSSI	0.322
Accelerometer-MinY	0.318
Accelerometer-MedianX	0.315
MagneticField-MedianZ	0.310
WiFi-724a775a596b-RSSI	0.309
MagneticField-LastZ	0.309
Accelerometer-LastX	0.307
Gravity-AverageX	0.302
WiFi-004a775a596a-RSSI	0.302

Table A.1: The association between features and the Building-level locations for the Building-level dataset. Only features with an association value (η for correlation ratio and V for Cramer's V, for which the features are marked with a star) >= 0.3 are shown.

A.2. FEATURE VS LOCATION ASSOCIATION (ROOM-LEVEL)

Feature	η / V *
GPS-Longitude	0.992
WiFi-IsConnected	0.987 *
WiFi-849ca6686d66-RSSI	0.975
WiFi-704ca588e6d0-RSSI	0.974
WiFi-704ca588e6c8-RSSI	0.974
GPS-Latitude	0.973
WiFi-704ca588e6c9-RSSI	0.972
Battery-IsCharging	0.956 *
WiFi-704ca588e6e8-RSSI	0.919
WiFi-704ca588cd79-RSSI	0.913
WiFi-965330691d7c-RSSI	0.905
WiFi-38d82f12423b-RSSI	0.895
WiFi-704ca588cd78-RSSI	0.892
WiFi-704ca588e6e0-RSSI	0.891
WiFi-704ca588e6e1-RSSI	0.877
WiFi-c4a36653f8f0-RSSI	0.865
WiFi-f09fc2f11b48-RSSI	0.847
WiFi-62a36653f8f1-RSSI	0.847
WiFi-461ca825e90c-RSSI	0.843
WiFi-e81cba09db18-RSSI	0.838
WiFi-e81cba09db19-RSSI	0.836
WiFi-704ca588eb51-RSSI	0.827
WiFi-dea26600a7c7-RSSI	0.825
WiFi-da0f99878481-RSSI	0.822
WiFi-704ca588eb48-RSSI	0.811
WiFi-704ca588cee0-RSSI	0.801
WiFi-704ca588cd80-RSSI	0.796
WiFi-880355bbc910-RSSI	0.793
WiFi-704ca588eb49-RSSI	0.790
WiFi-704ca588eb50-RSSI	0.783
WiFi-344dea9a6185-RSSI	0.766
WiFi-704ca588cee1-RSSI	0.764
WiFi-e81cba09db20-RSSI	0.760

WiFi-d8b6b7cc3248-RSSI	0.749
WiFi-704ca588cee8-RSSI	0.749
Accelerometer-AverageZ	0.736
Battery-Percentage	0.735
Accelerometer-MedianZ	0.734
Accelerometer-FirstZ	0.729
Accelerometer-LastZ	0.727
Accelerometer-MinZ	0.726
HourOfDay	0.681
MinuteOfDay	0.680
WiFi-c04a002d3846-RSSI	0.675
WiFi-788a202dd670-RSSI	0.672
Accelerometer-MaxZ	0.647
WiFi-38d82f11a986-RSSI	0.635
Gravity-MaxZ	0.633
Accelerometer-StdX	0.619
LinearAcceleration-MaxZ	0.614
Accelerometer-MinY	0.613
LinearAcceleration-LastZ	0.612
WiFi-8416f9792b7c-RSSI	0.601
WiFi-52d82f11a987-RSSI	0.601
WiFi-d0b2c426ea8c-RSSI	0.600
WiFi-d2b2c426ea8d-RSSI	0.586
WiFi-3a431d6ed2d4-RSSI	0.576
WiFi-c89346347fe0-RSSI	0.569
Accelerometer-MedianY	0.549
Accelerometer-AverageY	0.548
WiFi-38437d6ed2d4-RSSI	0.546
WiFi-001daaeeb3c0-RSSI	0.546
Accelerometer-StdY	0.542
WiFi-844765ce0f9c-RSSI	0.536
Accelerometer-LastY	0.530
WiFi-880355c2383a-RSSI	0.528
Gravity-MedianY	0.525
Accelerometer-FirstY	0.525
Gravity-FirstY	0.518

Gravity-AverageY	0.511
Gravity-MedianZ	0.507
WiFi-e81cba09db21-RSSI	0.501
Gravity-AverageZ	0.500
WiFi-98e7f4abcec8-RSSI	0.498
WiFi-925c142aceab-RSSI	0.497
Accelerometer-MaxX	0.491
LinearAcceleration-StdZ	0.490
Accelerometer-StdMagnitude	0.479
WiFi-905c442aceab-RSSI	0.477
WiFi-a42bb0200046-RSSI	0.476
Accelerometer-AverageX	0.474
Gravity-StdZ	0.467
WiFi-000cf65c9830-RSSI	0.467
Accelerometer-MedianX	0.460
WiFi-b0b98a60ccff-RSSI	0.460
WiFi-00e04c91d48b-RSSI	0.451
GameRotationVector-MedianX	0.451
Accelerometer-LastX	0.451
Gravity-LastY	0.448
WiFi-c80e1404d738-RSSI	0.446
GameRotationVector-FirstX	0.443
Gravity-FirstZ	0.441
WiFi-724a775a596b-RSSI	0.440
LinearAcceleration-StdMagnitude	0.438
Accelerometer-MinX	0.438
WiFi-a063913374c0-RSSI	0.435
Accelerometer-FirstX	0.434
WiFi-004a775a596a-RSSI	0.433
WiFi-fa8fca8ad97f-RSSI	0.432
WiFi-ca0e1404d738-RSSI	0.422
Gyroscope-MinX	0.421
WiFi-fa8fca8220a0-RSSI	0.419
WiFi-4c1b8627a8df-RSSI	0.413
WiFi-6abe537c0f72-RSSI	0.412
LinearAcceleration-MinZ	0.408

WiFi-54be537c0f71-RSSI	0.406
WiFi-b827ebf3d980-RSSI	0.404
WiFi-9c5c8eb6f300-RSSI	0.399
Gravity-LastZ	0.399
Gravity-MedianX	0.396
Gravity-FirstX	0.395
Gyroscope-MaxX	0.393
Light-Max	0.392
Light-Median	0.391
Light-Average	0.391
MagneticField-MinZ	0.389
Gravity-MinY	0.387
Light-Last	0.387
MagneticField-LastZ	0.381
Gravity-StdMagnitude	0.381
GameRotationVector-MedianY	0.378
Accelerometer-MaxY	0.376
Light-First	0.374
GameRotationVector-StdMagnitude	0.374
WiFi-624dea9a6186-RSSI	0.373
MagneticField-MedianZ	0.373
Gravity-AverageX	0.372
WiFi-5a957f42e8c0-RSSI	0.361
Gravity-MaxY	0.361
GameRotationVector-MaxZ	0.360
Light-Min	0.360
MagneticField-AverageZ	0.356
WiFi-a42bb0cdc3f7-RSSI	0.355
WiFi-00e04c878b08-RSSI	0.347
WiFi-d8b6b7cc324c-RSSI	0.346
Gyroscope-MinY	0.344
Gyroscope-MaxY	0.344
WiFi-d0667b03506d-RSSI	0.344
WiFi-f29fc2f21b48-RSSI	0.342
GameRotationVector-StdY	0.341
Gravity-MinZ	0.340

Gyroscope-MinZ	0.337
GameRotationVector-FirstY	0.333
LinearAcceleration-AverageZ	0.333
GPS-Altitude	0.332
WiFi-14918253509e-RSSI	0.330
LinearAcceleration-LastX	0.328
Gyroscope-MaxZ	0.327
WiFi-2c957f42e8c3-RSSI	0.327
WiFi-3a9d9205b804-RSSI	0.320
Gyroscope-StdMagnitude	0.318
GameRotationVector-StdX	0.313
GameRotationVector-MaxX	0.313
WiFi-b2c287e56ac0-RSSI	0.310
RotationVector-StdX	0.310
WiFi-9c5c8eb6f301-RSSI	0.310
WiFi-72d82f269a33-RSSI	0.308
MagneticField-StdZ	0.302
WiFi-f4068d61326d-RSSI	0.301

Table A.2: The association between features and the Room-level locations for the Room-level dataset. Only features with an association value (η for correlation ratio and V for Cramer's V, for which the features are marked with a star) >= 0.3 are shown.

A.3. FEATURE VS ACTIVITY ASSOCIATION (BUILDING-LEVEL)

Feature	η / V *
WiFi-IsConnected	0.947 *
WiFi-849ca6686d66-RSSI	0.920
Battery-IsCharging	0.905 *
WiFi-f09fc2f11b48-RSSI	0.847
WiFi-38d82f12423b-RSSI	0.840
WiFi-704ca588e6d0-RSSI	0.816
WiFi-c4a36653f8f0-RSSI	0.813
WiFi-62a36653f8f1-RSSI	0.803
WiFi-704ca588e6c8-RSSI	0.791
WiFi-704ca588e6c9-RSSI	0.780
WiFi-704ca588cd79-RSSI	0.768
WiFi-10bf48e7237c-RSSI	0.760
WiFi-880355bbc910-RSSI	0.760
WiFi-344dea9a6185-RSSI	0.746
WiFi-704ca588cd78-RSSI	0.737
HourOfDay	0.719
MinuteOfDay	0.718
WiFi-d8b6b7cc3248-RSSI	0.706
Battery-Percentage	0.696
WiFi-a45d36414641-RSSI	0.676
WiFi-788a202dd670-RSSI	0.676
WiFi-704ca588e6e1-RSSI	0.644
Gyroscope-StdX	0.636
WiFi-704ca588e6e0-RSSI	0.631
WiFi-ac22057c31c2-RSSI	0.610
GPS-Speed	0.608
WiFi-ae22157c31c2-RSSI	0.606
Accelerometer-MinZ	0.587
WiFi-48d343f4f3f1-RSSI	0.587
WiFi-fa8fca569800-RSSI	0.578
Accelerometer-AverageZ	0.559
WiFi-844765ce0f9c-RSSI	0.557
WiFi-d0b2c426ea8c-RSSI	0.554

Accelerometer-MedianZ	0.553
Accelerometer-LastZ	0.551
Accelerometer-FirstZ	0.549
Accelerometer-StdY	0.543
WiFi-d2b2c426ea8d-RSSI	0.542
WiFi-c04a002d3846-RSSI	0.536
LinearAcceleration-MaxZ	0.536
Accelerometer-StdX	0.535
Gravity-MaxZ	0.535
WiFi-001daaeeb3c0-RSSI	0.533
Accelerometer-MinY	0.532
WiFi-3a431d6ed2d4-RSSI	0.524
LinearAcceleration-LastZ	0.523
WiFi-c89346347fe0-RSSI	0.522
Gyroscope-StdMagnitude	0.521
GPS-Longitude	0.519
WiFi-98e7f4abcec8-RSSI	0.504
Accelerometer-MaxX	0.497
WiFi-38d82f11a986-RSSI	0.495
WiFi-704ca588e6e8-RSSI	0.494
GPS-Latitude	0.491
WiFi-ccce1ed72795-RSSI	0.488
WiFi-4c1b865b85a9-RSSI	0.483
WiFi-38437d6ed2d4-RSSI	0.479
WiFi-344dea887c2e-RSSI	0.475
Gyroscope-MinX	0.472
LinearAcceleration-StdZ	0.471
Accelerometer-StdMagnitude	0.465
WiFi-6a4dea887c2f-RSSI	0.464
Gravity-StdZ	0.464
Accelerometer-MedianY	0.464
Gyroscope-MaxX	0.462
WiFi-000cf65c9830-RSSI	0.459
WiFi-fa8fca8220a0-RSSI	0.457
Accelerometer-AverageY	0.454
WiFi-52d82f11a987-RSSI	0.453

Accelerometer-FirstY	0.453
Accelerometer-LastY	0.452
WiFi-8416f9792b7c-RSSI	0.451
WiFi-880355c2383a-RSSI	0.449
WiFi-fa8fca8ad97f-RSSI	0.444
WiFi-b0b98a60ccff-RSSI	0.441
Gravity-MedianY	0.431
Gravity-FirstY	0.429
LinearAcceleration-StdMagnitude	0.429
Accelerometer-MaxZ	0.423
WiFi-461ca825e90c-RSSI	0.421
WiFi-724a775a596b-RSSI	0.420
Gravity-AverageY	0.419
WiFi-6abe537c0f72-RSSI	0.417
WiFi-004a775a596a-RSSI	0.416
Gravity-MedianZ	0.414
WiFi-dea26600a7c7-RSSI	0.411
WiFi-624dea9a6186-RSSI	0.411
GameRotationVector-MedianX	0.404
Gravity-StdMagnitude	0.403
Accelerometer-AverageX	0.402
WiFi-9c5c8eb6f300-RSSI	0.401
WiFi-54be537c0f71-RSSI	0.399
WiFi-704ca588cee0-RSSI	0.398
MagneticField-MinZ	0.396
Gravity-AverageZ	0.395
WiFi-704ca588eb50-RSSI	0.394
WiFi-704ca588eb48-RSSI	0.389
GameRotationVector-FirstX	0.386
WiFi-704ca588eb51-RSSI	0.385
WiFi-704ca588eb49-RSSI	0.382
Accelerometer-MedianX	0.382
MagneticField-MedianZ	0.381
MagneticField-LastZ	0.380
WiFi-c80e1404d738-RSSI	0.380
Gravity-AverageX	0.377

WiFi-704ca588cd80-RSSI	0.376
WiFi-704ca588cee1-RSSI	0.373
Gravity-MinY	0.371
Accelerometer-LastX	0.371
Gravity-LastY	0.371
Gravity-MedianX	0.370
WiFi-ca0e1404d738-RSSI	0.370
WiFi-a42bb0200046-RSSI	0.369
Gravity-LastZ	0.367
Gravity-FirstZ	0.366
MagneticField-AverageZ	0.365
WiFi-965330691d7c-RSSI	0.364
Accelerometer-FirstX	0.359
WiFi-925c142aceab-RSSI	0.356
Gyroscope-StdZ	0.354
Gyroscope-MinZ	0.354
WiFi-704ca588cee8-RSSI	0.352
Gyroscope-MaxY	0.347
Gravity-FirstX	0.347
Gyroscope-MinY	0.344
Accelerometer-MaxY	0.343
Gyroscope-MaxZ	0.343
WiFi-e81cba09db19-RSSI	0.342
WiFi-905c442aceab-RSSI	0.338
Gravity-MinZ	0.338
WiFi-a42bb0cdc3f7-RSSI	0.337
Light-Average	0.336
Gyroscope-StdY	0.336
GameRotationVector-StdMagnitude	0.334
WiFi-5a957f42e8c0-RSSI	0.334
Light-First	0.333
WiFi-4c1b8627a8df-RSSI	0.333
MagneticField-StdZ	0.332
RotationVector-MedianX	0.329
WiFi-00e04c91d48b-RSSI	0.328
WiFi-9c5c8eb6f301-RSSI	0.324

Light-Median	0.322
Gravity-LastX	0.320
Light-Last	0.318
LinearAcceleration-AverageZ	0.312
WiFi-72d82f269a33-RSSI	0.311
Accelerometer-MinX	0.310
GameRotationVector-MedianY	0.310
Gravity-MaxY	0.309
LinearAcceleration-MinZ	0.309
WiFi-2c957f42e8c3-RSSI	0.308
WiFi-3a9d9205b804-RSSI	0.304
WiFi-ccaf7885ff3f-RSSI	0.301

Table A.3: The association between features and the activities for the Building-level dataset. Only features with an association value (η for correlation ratio and V for Cramer's V, for which the features are marked with a star) >= 0.3 are shown.

A.4. FEATURE VS ACTIVITY ASSOCIATION (ROOM-LEVEL)

Feature	η / V *
GPS-Longitude	0.948
WiFi-IsConnected	0.944 *
GPS-Latitude	0.930
Battery-IsCharging	0.903 *
WiFi-849ca6686d66-RSSI	0.882
WiFi-704ca588e6d0-RSSI	0.818
WiFi-38d82f12423b-RSSI	0.813
WiFi-704ca588e6c8-RSSI	0.798
WiFi-f09fc2f11b48-RSSI	0.789
WiFi-704ca588e6c9-RSSI	0.787
WiFi-704ca588cd79-RSSI	0.774
WiFi-704ca588cd78-RSSI	0.751
WiFi-c4a36653f8f0-RSSI	0.750
WiFi-62a36653f8f1-RSSI	0.739
WiFi-880355bbc910-RSSI	0.723
Battery-Percentage	0.720
HourOfDay	0.717
MinuteOfDay	0.715
WiFi-344dea9a6185-RSSI	0.665
WiFi-704ca588e6e1-RSSI	0.657
WiFi-d8b6b7cc3248-RSSI	0.646
WiFi-704ca588e6e0-RSSI	0.645
WiFi-788a202dd670-RSSI	0.634
WiFi-c04a002d3846-RSSI	0.525
WiFi-844765ce0f9c-RSSI	0.520
WiFi-3a431d6ed2d4-RSSI	0.513
WiFi-d0b2c426ea8c-RSSI	0.507
WiFi-c89346347fe0-RSSI	0.504
Accelerometer-MinZ	0.503
WiFi-d2b2c426ea8d-RSSI	0.496
WiFi-001daaeeb3c0-RSSI	0.493
WiFi-38d82f11a986-RSSI	0.489
WiFi-704ca588e6e8-RSSI	0.485

Accelerometer-StdX	0.482
Accelerometer-AverageZ	0.475
WiFi-38437d6ed2d4-RSSI	0.473
WiFi-98e7f4abcec8-RSSI	0.470
LinearAcceleration-MaxZ	0.470
Accelerometer-MedianZ	0.469
Accelerometer-LastZ	0.467
Accelerometer-FirstZ	0.464
LinearAcceleration-LastZ	0.455
WiFi-461ca825e90c-RSSI	0.447
WiFi-52d82f11a987-RSSI	0.446
WiFi-8416f9792b7c-RSSI	0.446
Accelerometer-StdY	0.440
WiFi-880355c2383a-RSSI	0.435
Gravity-MaxZ	0.432
Accelerometer-MinY	0.430
WiFi-000cf65c9830-RSSI	0.413
Accelerometer-AverageY	0.405
Accelerometer-MedianY	0.405
WiFi-b0b98a60ccff-RSSI	0.402
WiFi-704ca588cee0-RSSI	0.401
Accelerometer-LastY	0.400
WiFi-dea26600a7c7-RSSI	0.4
LinearAcceleration-StdZ	0.397
WiFi-704ca588eb51-RSSI	0.394
WiFi-704ca588eb50-RSSI	0.391
Accelerometer-FirstY	0.391
Accelerometer-MaxX	0.389
WiFi-704ca588eb48-RSSI	0.387
Gravity-AverageY	0.386
WiFi-fa8fca8220a0-RSSI	0.386
Gravity-MedianY	0.386
WiFi-724a775a596b-RSSI	0.383
WiFi-704ca588cd80-RSSI	0.382
WiFi-004a775a596a-RSSI	0.380
Accelerometer-StdMagnitude	0.379

Accelerometer-MaxZ	0.377
Gravity-StdZ	0.376
WiFi-704ca588eb49-RSSI	0.376
WiFi-fa8fca8ad97f-RSSI	0.372
WiFi-704ca588cee8-RSSI	0.372
WiFi-9c5c8eb6f300-RSSI	0.370
WiFi-704ca588cee1-RSSI	0.369
WiFi-c80e1404d738-RSSI	0.367
WiFi-6abe537c0f72-RSSI	0.367
LinearAcceleration-StdMagnitude	0.363
Gravity-FirstY	0.363
WiFi-e81cba09db19-RSSI	0.363
Light-Max	0.359
WiFi-a42bb0200046-RSSI	0.359
WiFi-ca0e1404d738-RSSI	0.355
WiFi-965330691d7c-RSSI	0.352
WiFi-624dea9a6186-RSSI	0.349
WiFi-54be537c0f71-RSSI	0.349
WiFi-925c142aceab-RSSI	0.349
Gravity-LastZ	0.345
Light-Average	0.343
Light-Median	0.342
Gravity-LastY	0.339
Gyroscope-MinX	0.339
Light-Last	0.337
Accelerometer-MaxY	0.337
WiFi-5a957f42e8c0-RSSI	0.336
WiFi-905c442aceab-RSSI	0.333
MagneticField-MinZ	0.333
WiFi-4c1b8627a8df-RSSI	0.331
WiFi-00e04c91d48b-RSSI	0.326
Light-First	0.326
GameRotationVector-MedianX	0.325
Gyroscope-MaxX	0.319
Gravity-StdMagnitude	0.318
Gyroscope-MinY	0.317

MagneticField-MedianZ	0.316
WiFi-e81cba09db18-RSSI	0.316
MagneticField-LastZ	0.315
Gravity-MinZ	0.315
Gravity-MedianZ	0.315
Accelerometer-AverageX	0.311
GameRotationVector-FirstX	0.309
Gyroscope-MaxY	0.306
WiFi-2c957f42e8c3-RSSI	0.303
WiFi-b827ebf3d980-RSSI	0.302
MagneticField-AverageZ	0.302
WiFi-a063913374c0-RSSI	0.301
Light-Min	0.301
WiFi-a42bb0cdc3f7-RSSI	0.300
Gravity-AverageZ	0.300

Table A.4: The association between features and the activities for the Room-level dataset. Only features with an association value (η for correlation ratio and V for Cramer's V, for which the features are marked with a star) >= 0.3 are shown.

B

Data source importance data

Data source	Number of features in association value range				
	1.0 to 0.8	**0.8 to 0.6**	**0.6 to 0.4**	**0.4 to 0.2**	**0.2 to 0.0**
WiFi	7	14	31	48	30
Battery	1	1			
Time		2			1
Gyroscope		1	3	6	9
GPS		1	2		1
Accelerometer			16	6	
Gravity			7	13	2
Linear acceleration			4	7	11
Game rotation vector			1	11	7
Magnetic field				13	9
Rotation vector				8	11
Light				7	
Proximity					8
Pressure					7

Table B.1: The frequency distribution of the feature ranks per data source. For the building-level data set, the association between each feature and the activity classes is classified in the association value ranges shown in this table. Per data source the frequencies of feature classes are summed for the activity classes from the building-level dataset

Data source	**Number of features in association value range**				
	1.0 to 0.8	**0.8 to 0.6**	**0.6 to 0.4**	**0.4 to 0.2**	**0.2 to 0.0**
WiFi	4	13	18	52	33
GPS	2				2
Battery	1	1			
Time		2			1
Accelerometer			10	12	
Linear acceleration			2	5	15
Gravity			1	16	5
Magnetic field				12	10
Game rotation vector				8	11
Gyroscope				8	8
Light				7	
Rotation vector				2	17
Proximity					8
Pressure					7

Table B.2: The frequency distribution of the feature ranks per data source. For the room-level data set, the association between each feature and the activity classes is classified in the association value ranges shown in this table. Per data source the frequencies of feature classes are summed for the activity classes from the room-level dataset

Data source	Number of features in association value range				
	1.0 to 0.8	0.8 to 0.6	0.6 to 0.4	0.4 to 0.2	0.2 to 0.0
WiFi	5	13	15	32	65
Battery		1		1	
Accelerometer			7	12	3
GPS			3		1
Linear acceleration			2	4	16
Gravity			1	12	9
Gyroscope				7	12
Light				7	
Game rotation vector				6	13
Magnetic field				5	17
Time				2	1
Rotation vector				1	18
Proximity					8
Pressure					7

Table B.3: The frequency distribution of the feature ranks per data source. For the building-level data set, the association between each feature and the location classes is classified in the association value ranges shown in this table. Per data source the frequencies of feature classes are summed for location recognition on the building-level dataset.

Data source	Number of features in association value range				
	1.0 to 0.8	0.8 to 0.6	0.6 to 0.4	0.4 to 0.2	0.2 to 0.0
WiFi	23	15	26	31	25
GPS	2			1	1
Battery	1	1			
Accelerometer		8	12	2	
Linear acceleration		2	3	4	13
Time		2			1
Gravity		1	8	9	4
Game rotation vector			2	10	7
Gyroscope			1	8	7
Rotation vector				12	10
Magnetic field				10	12
Light				7	
Proximity					8
Pressure					7

Table B.4: The frequency distribution of the feature ranks per data source. For the room-level data set, the association between each feature and the location classes is classified in the association value ranges shown in this table. Per data source the frequencies of feature classes are summed for the location classes from the room-level dataset

Bibliography

References

[Abo+99] G. Abowd, A. Dey, P. Brown, N. Davies, M. Smith, and P. Steggles. “Towards a better understanding of context and context-awareness”. In: *Handheld and ubiquitous computing (pp. 304-307), Springer Berlin/Heidelberg* (1999).

[ALC12] Saeed Abdullah, Nicholas D Lane, and Tanzeem Choudhury. “Towards Population Scale Activity Recognition: A Framework for Handling Data Diversity.” In: *AAAI*. 2012.

[Anda] Android. *Android Doze and App standby modes*. `https://developer.android.com/training/monitoring-device-state/doze-standby.html`.

[Andb] Android. *Android motion sensors*. `https://developer.android.com/guide/topics/sensors/sensors_motion`.

[Andc] Android. *Android Platform Power Management*. `https://source.android.com/devices/tech/power/platform_mgmt`.

[Andd] Android. *Detect when users start or end an activity*. `https://developer.android.com/guide/topics/location/transitions`.

[Ande] Android. *Android sensor event documentation*. `https://developer.android.com/reference/android/hardware/SensorEvent.html#values`.

[Andf] Android. *Android trigger event*. `https://developer.android.com/reference/android/hardware/TriggerEvent`.

[Ang+12a] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J.L. Reyes-Ortiz. “Human activity recognition on smartphones using a multiclass hardware-friendly support vector machine”. In: *International workshop on ambient assisted living*. Springer. 2012, pp. 216–223.

[Ang+12b] Davide Anguita, Alessandro Ghio, Luca Oneto, Francesc Xavier Llanas Parra, and Jorge Luis Reyes Ortiz. “Human activity recognition on smartphones for mobile context awareness”. In: *Advances in Neural Information Processing Systems 26: proceedings of the 2012 conference*. 2012, pp. 1–9.

[Ang+13] Davide Anguita, Alessandro Ghio, Luca Oneto, Xavier Parra, and Jorge Luis Reyes-Ortiz. “A public domain dataset for human activity recognition using smartphones.” In: *ESANN*. 2013.

[Ara+16] Oya Aran, Dairazalia Sanchez-Cortes, Minh-Tri Do, and Daniel Gatica-Perez. “Anomaly detection in elderly daily behavior in ambient sensing environments”. In: *International Workshop on Human Behavior Understanding*. Springer. 2016, pp. 51–67.

[Bae+18] Sangwon Bae, Tammy Chung, Denzil Ferreira, Anind K Dey, and Brian Suffoletto. “Mobile phone sensors and supervised machine learning to identify alcohol use events in young adults: Implications for just-in-time adaptive interventions”. In: *Addictive behaviors* 83 (2018), pp. 42–47.

[Bie+18] Joan-Isaac Biel, Nathalie Martin, David Labbe, and Daniel Gatica-Perez. “Bites ‘n’Bits: Inferring Eating Behavior from Contextual Mobile Data”. In: *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1.4 (2018), p. 125.

[BL16] Sourav Bhattacharya and Nicholas D Lane. “From smart to deep: Robust activity recognition on smartwatches using deep learning”. In: *Pervasive Computing and Communication Workshops (PerCom Workshops), 2016 IEEE International Conference on*. IEEE. 2016, pp. 1–6.

[BLP13] Andrey Bogomolov, Bruno Lepri, and Fabio Pianesi. “Happiness recognition from mobile phone data”. In: *Social Computing (SocialCom), 2013 International Conference on*. IEEE. 2013, pp. 790–795.

[Bog+14] Andrey Bogomolov, Bruno Lepri, Michela Ferron, Fabio Pianesi, and Alex Sandy Pentland. “Daily stress recognition from mobile phone data, weather conditions and individual traits”. In: *Proceedings of the 22nd ACM international conference on Multimedia*. ACM. 2014, pp. 477–486.

[Bui+13] Lars Buitinck, Gilles Louppe, Mathieu Blondel, Fabian Pedregosa, Andreas Mueller, Olivier Grisel, Vlad Niculae, Peter Prettenhofer, Alexandre Gramfort, Jaques Grobler, Robert Layton, Jake VanderPlas, Arnaud Joly, Brian Holt, and Gaël Varoquaux. “API design for machine learning software: experiences from the scikit-learn project”. In: *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*. 2013, pp. 108–122.

[Cas+17] Diego Castro, William Coral, Camilo Rodriguez, Jose Cabra, and Julian Colorado. “Wearable-Based Human Activity Recognition Using and IoT Approach”. In: *Journal of Sensor and Actuator Networks* 6.4 (2017), p. 28.

[CBG11] G. Chittaranjan, J. Blom, and D. Gatica-Perez. “Who’s who with big-five: Analyzing and classifying personality traits with smartphones”. In: *15th Annual International Symposium on Wearable Computers (ISWC)*. IEEE. 2011, pp. 29–36.

[CC11] J. Chon and H. Cha. “Lifemap: A smartphone-based context provider for location-based services”. In: *IEEE Pervasive Computing* 10.2 (2011), pp. 58–67.

[Che+13] Zhenyu Chen, Mu Lin, Fanglin Chen, Nicholas D Lane, Giuseppe Cardone, Rui Wang, Tianxing Li, Yiqiang Chen, Tanzeem Choudhury, and Andrew T Campbell. “Unobtrusive sleep monitoring using smartphones”. In: *Pervasive Computing Technologies for Healthcare (PervasiveHealth), 2013 7th International Conference on*. IEEE. 2013, pp. 145–152.

[Che+14] Zhenyu Chen, Yiqiang Chen, Lisha Hu, Shuangquan Wang, Xinlong Jiang, Xiaojuan Ma, Nicholas D Lane, and Andrew T Campbell. “ContextSense: unobtrusive discovery of incremental social context using dynamic bluetooth data”. In: *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*. ACM. 2014, pp. 23–26.

[Che+15] Zhenyu Chen, Yiqiang Chen, Xingyu Gao, Shuangquan Wang, Lisha Hu, Chenggang Clarence Yan, Nicholas D Lane, and Chunyan Miao. “Unobtrusive sensing incremental social contexts using fuzzy class incremental learning”. In: *Data Mining (ICDM), 2015 IEEE International Conference on*. IEEE. 2015, pp. 71–80.

[Cho+12] Yohan Chon, Nicholas D Lane, Fan Li, Hojung Cha, and Feng Zhao. “Automatically characterizing places with opportunistic crowdsensing using smartphones”. In: *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*. ACM. 2012, pp. 481–490.

[CK01] Amanda Clare and Ross D King. “Knowledge discovery in multi-label phenotype data”. In: *European Conference on Principles of Data Mining and Knowledge Discovery*. Springer. 2001, pp. 42–53.

[Con+15] M. Conti, L.V. Mancini, R. Spolaor, and N.V. Verde. “Can’t you hear me knocking: Identification of user actions on android apps via traffic analysis”. In: *Proceedings of the 5th ACM Conference on Data and Application Security and Privacy*. ACM. 2015, pp. 297–304.

[DA00] A.K. Dey and G.D. Abowd. “Towards a better understanding of context and context-awareness”. In: *CHI 2000 Workshop on the What, Who, Where, When, and How of Context-Awareness* (2000).

[DAS01] A.K. Dey, G.D. Abowd, and D Salber. “A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications”. In: *Human-Computer Interaction, 16-2-4, 97-166* (2001).

[DG14] Trinh Minh Tri Do and Daniel Gatica-Perez. “The places of our lives: Visiting patterns and automatic labeling from longitudinal smartphone data”. In: *IEEE Transactions on Mobile Computing* 13.3 (2014), pp. 638–648.

[EP06] Nathan Eagle and Alex Sandy Pentland. “Reality mining: sensing complex social systems”. In: *Personal and ubiquitous computing* 10.4 (2006), pp. 255–268.

[EP09] Nathan Eagle and Alex Sandy Pentland. “Eigenbehaviors: Identifying structure in routine”. In: *Behavioral Ecology and Sociobiology* 63.7 (2009), pp. 1057–1066.

[FHW16] Eibe Frank, Mark A. Hall, and Ian H. Witten. *The WEKA Workbench. Online Appendix for "Data Mining: Practical Machine Learning Tools and Techniques"*. 4th ed. Morgan Kaufmann, 2016.

[GE03] Isabelle Guyon and André Elisseeff. “An introduction to variable and feature selection”. In: *Journal of machine learning research* 3.Mar (2003), pp. 1157–1182.

[GM02] A. Göker and H.I. Myrhaug. “User context and personalisation”. In: *Workshop proceedings for the 6th European Conference on Case Based Reasoning. 2002*. 2002.

[Gooa] Google. *Google Activity Recognition API*. `https://developers.google.com/location-context/activity-recognition/`.

[Goob] Google. *Google Activity Recognition API - Detected Activity*. https://developers.google.com/android/reference/com/google/android/gms/location/DetectedActivity.

[Gui15] Robert E Guinness. "Beyond Where to How: A Machine Learning Approach for Sensing Mobility Contexts Using Smartphone Sensors". In: *Sensors* 15.5 (2015), pp. 9962–9985.

[HGB13] J.R. Hoyos, Jesus Garcia-Molina, and J.A. Botia. "A domain-specific language for context modeling in context-aware systems". In: *The Journal of Systems and Software 86 (2013) 2890-2905* (2013).

[Hor+13] Teerayut Horanont, Santi Phithakkitnukoon, Tuck W Leong, Yoshihide Sekimoto, and Ryosuke Shibasaki. "Weather effects on the patterns of people's everyday activities: a study using GPS traces of mobile phone users". In: *PloS one* 8.12 (2013), e81153.

[IKE13] Ozlem Durmaz Incel, Mustafa Kose, and Cem Ersoy. "A review and taxonomy of activity recognition on mobile phones". In: *BioNanoScience* 3.2 (2013), pp. 145–171.

[Ker] Keras. *Home - Keras documentation*. https://keras.io/.

[Kha+14] Adil Mehmood Khan, Ali Tufail, Asad Masood Khattak, and Teemu H Laine. "Activity recognition on smartphones via sensor-fusion and kda-based svms". In: *International Journal of Distributed Sensor Networks* 10.5 (2014), p. 503291.

[Kim+16] Hyunchoong Kim, Jonghoon Shin, Soohwan Kim, Yohan Ko, Kyoungwoo Lee, Hojung Cha, Seong-il Hahm, and TaeJun Kwon. "Collaborative classification for daily activity recognition with a smartwatch". In: *Systems, Man, and Cybernetics (SMC), 2016 IEEE International Conference on*. IEEE. 2016, pp. 003707–003712.

[KJ97] Ron Kohavi and George H John. "Wrappers for feature subset selection". In: *Artificial intelligence* 97.1-2 (1997), pp. 273–324.

[KS16] Heli Koskimäki and Pekka Siirtola. "Adaptive model fusion for wearable sensors based human activity recognition". In: *Information Fusion (FUSION), 2016 19th International Conference on*. IEEE. 2016, pp. 1709–1713.

[KZP07] Sotiris B Kotsiantis, I Zaharakis, and P Pintelas. "Supervised machine learning: A review of classification techniques". In: *Emerging artificial intelligence applications in computer engineering* 160 (2007), pp. 3–24.

[Lan+11] Nicholas D Lane, Ye Xu, Hong Lu, Shaohan Hu, Tanzeem Choudhury, Andrew T Campbell, and Feng Zhao. "Enabling large-scale human activity inference on smartphones using community similarity networks (csn)". In: *Proceedings of the 13th international conference on Ubiquitous computing*. ACM. 2011, pp. 355–364.

[Lan+14] Nicholas D Lane, Mu Lin, Mashfiqui Mohammod, Xiaochao Yang, Hong Lu, Giuseppe Cardone, Shahid Ali, Afsaneh Doryab, Ethan Berke, Andrew T Campbell, et al. "Bewell: Sensing sleep, physical activities and social interactions to promote wellbeing". In: *Mobile Networks and Applications* 19.3 (2014), pp. 345–359.

[Lan+15] Nicholas D Lane, Sourav Bhattacharya, Petko Georgiev, Claudio Forlivesi, and Fahim Kawsar. "An early resource characterization of deep learning on wearables, smartphones and internet-of-things devices". In: *Proceedings of the 2015 International Workshop on Internet of Things towards Applications*. ACM. 2015, pp. 7–12.

[Lan+16] Nicholas D Lane, Sourav Bhattacharya, Petko Georgiev, Claudio Forlivesi, Lei Jiao, Lorena Qendro, and Fahim Kawsar. "Deepx: A software accelerator for low-power deep learning inference on mobile devices". In: *Information Processing in Sensor Networks (IPSN), 2016 15th ACM/IEEE International Conference on*. IEEE. 2016, pp. 1–12.

[Lan12] Nicholas D Lane. "Community-aware smartphone sensing systems". In: *IEEE Internet Computing* 16.3 (2012), pp. 60–64.

[lea] Scikit learn. *Scikit learn Decision Trees*. `https://scikit-learn.org/stable/modules/tree.html`.

[Lee+12] Hosub Lee, Young Sang Choi, Sunjae Lee, and IP Park. "Towards unobtrusive emotion recognition for affective social communication". In: *Consumer Communications and Networking Conference (CCNC), 2012 IEEE*. IEEE. 2012, pp. 260–264.

[Les+05] J. Lester, T. Choudhury, N. Kern, G. Borriello, and B. Hannaford. "A hybrid discriminative/generative approach for modeling human activities". In: (2005).

[LFK05] L. Liao, D. Fox, and H. Kautz. "Location-based activity recognition using relational Markov networks". In: *Proceedings of the 19th international joint conference on Artificial intelligence*. 2005, pp. 773–778.

[LFK07] Lin Liao, Dieter Fox, and Henry Kautz. "Hierarchical conditional random fields for GPS-based activity recognition". In: *Robotics Research*. Springer, 2007, pp. 487–506.

[LG15] Nicholas D Lane and Petko Georgiev. "Can deep learning revolutionize mobile sensing?" In: *Proceedings of the 16th International Workshop on Mobile Computing Systems and Applications*. ACM. 2015, pp. 117–122.

[LiK+11] Robert LiKamWa, Yunxin Liu, Nicholas D Lane, and Lin Zhong. "Can your smartphone infer your mood". In: *PhoneSense workshop*. 2011, pp. 1–5.

[LiK+13] R. LiKamWa, Y. Liu, N.D. Lane, and L. Zhong. "Moodscope: Building a mood sensor from smartphone usage patterns". In: *Proceeding of the 11th annual international conference on Mobile systems, applications, and services*. ACM. 2013, pp. 389–402.

[Mar+13] Henar Martın, Ana M Bernardos, Josué Iglesias, and José R Casar. "Activity logging using lightweight classification techniques in mobile devices". In: *Personal and ubiquitous computing* 17.4 (2013), pp. 675–695.

[MBG13] Raul Montoliu, Jan Blom, and Daniel Gatica-Perez. "Discovering places of interest in everyday life from smartphone data". In: *Multimedia tools and applications* 62.1 (2013), pp. 179–207.

[Mil+10] Emiliano Miluzzo, Michela Papandrea, Nicholas D Lane, Hong Lu, and Andrew T Campbell. “Pocket, bag, hand, etc.-automatically detecting phone context through discovery”. In: *Proc. PhoneSense 2010* (2010), pp. 21–25.

[Par+13] Pavlos Paraskevopoulos, Thanh-Cong Dinh, Zolzaya Dashdorj, Themis Palpanas, and Luciano Serafini. “Identification and characterization of human behavior patterns from mobile phone data”. In: *Proc. of NetMob* (2013).

[Phi+10] Santi Phithakkitnukoon, Teerayut Horanont, Giusy Di Lorenzo, Ryosuke Shibasaki, and Carlo Ratti. “Activity-aware map: Identifying human daily activity pattern using mobile phone data”. In: *International Workshop on Human Behavior Understanding*. Springer. 2010, pp. 14–25.

[Pir+16] Ivan Pires, Nuno Garcia, Nuno Pombo, and Francisco Flórez-Revuelta. “From data acquisition to data fusion: a comprehensive review and a roadmap for the identification of activities of daily living using mobile devices”. In: *Sensors* 16.2 (2016), p. 184.

[Pir+17a] Ivan Miguel Pires, Nuno M Garcia, Nuno Pombo, and Francisco Flórez-Revuelta. “A Multiple Source Framework for the Identification of Activities of Daily Living Based on Mobile Device Data”. In: *arXiv preprint arXiv:1711.00104* (2017).

[Pir+17b] Ivan Miguel Pires, Nuno M Garcia, Nuno Pombo, and Francisco Flórez-Revuelta. “User Environment Detection with Acoustic Sensors Embedded on Mobile Devices for the Recognition of Activities of Daily Living”. In: *arXiv preprint arXiv:1711.00124* (2017).

[Pir+17c] Ivan Miguel Pires, Nuno M Garcia, Nuno Pombo, Francisco Flórez-Revuelta, and Susanna Spinsante. “Data Fusion on Motion and Magnetic Sensors embedded on Mobile Devices for the Identification of Activities of Daily Living”. In: *arXiv preprint arXiv:1711.07328* (2017).

[Pir+17d] Ivan Miguel Pires, Nuno M Garcia, Nuno Pombo, Francisco Flórez-Revuelta, and Susanna Spinsante. “Pattern recognition techniques for the identification of Activities of Daily Living using mobile device accelerometer”. In: *arXiv preprint arXiv:1711.00096* (2017).

[Pir+18a] Ivan Miguel Pires, Nuno M Garcia, Nuno Pombo, and Francisco Flórez-Revuelta. “Limitations of the Use of Mobile Devices and Smart Environments for the Monitoring of Ageing People.” In: *ICT4AWE*. 2018, pp. 269–275.

[Pir+18b] Ivan Miguel Pires, Nuno M Garcia, Nuno Pombo, Francisco Flórez-Revuelta, Susanna Spinsante, and Maria Canavarro Teixeira. “Identification of activities of daily living through data fusion on motion and magnetic sensors embedded on mobile devices”. In: *Pervasive and Mobile Computing* 47 (2018), pp. 78–93.

[Pir+18c] Ivan Pires, Nuno Garcia, Nuno Pombo, Francisco Flórez-Revuelta, and Susanna Spinsante. “Approach for the development of a framework for the identification of activities of daily living using sensors in mobile devices”. In: *Sensors* 18.2 (2018), p. 640.

[Pir+18d] Ivan Pires, Rui Santos, Nuno Pombo, Nuno Garcia, Francisco Flórez-Revuelta, Susanna Spinsante, Rossitza Goleva, and Eftim Zdravevski. "Recognition of activities of daily living based on environmental analyses using audio fingerprinting techniques: a systematic review". In: *Sensors* 18.1 (2018), p. 160.

[PyT] PyTorch. *PyTorch*. `https://pytorch.org/`.

[Rad+16] Valentin Radu, Nicholas D Lane, Sourav Bhattacharya, Cecilia Mascolo, Mahesh K Marina, and Fahim Kawsar. "Towards multimodal deep learning for activity recognition on mobile devices". In: *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct*. ACM. 2016, pp. 185–188.

[Rad+18] Valentin Radu, Catherine Tong, Sourav Bhattacharya, Nicholas D Lane, Cecilia Mascolo, Mahesh K Marina, and Fahim Kawsar. "Multimodal deep learning for activity and context recognition". In: *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1.4 (2018), p. 157.

[Rey+16] Jorge-L Reyes-Ortiz, Luca Oneto, Albert Samà, Xavier Parra, and Davide Anguita. "Transition-aware human activity recognition using smartphones". In: *Neurocomputing* 171 (2016), pp. 754–767.

[Rey14] Jorge Luis Reyes Ortiz. "Smartphone-based human activity recognition". In: (2014).

[Rey15] Jorge Luis Reyes Ortiz. *Smartphone-based human activity recognition*. Springer, 2015.

[RM15] Ines Riahi and Faouzi Moussa. "A formal approach for modeling context-aware human–computer system". In: *Computers & Electrical Engineering* 44 (2015), pp. 241–261.

[Sah+14] Manaswi Saha, Shailja Thakur, Amarjeet Singh, and Yuvraj Agarwal. "EnergyLens: combining smartphones with electricity meter for accurate activity detection and user annotation". In: *Proceedings of the 5th international conference on Future energy systems*. ACM. 2014, pp. 289–300.

[Sah+18] Jayita Saha, Chandreyee Chowdhury, Ishan Roy Chowdhury, Suparna Biswas, and Nauman Aslam. "An Ensemble of Condition Based Classifiers for Device Independent Detailed Human Activity Recognition Using Smartphones". In: *Information* 9.4 (2018), p. 94.

[San+18] Darshan Santani, Trinh-Minh-Tri Do, Florian Labhart, Sara Landolt, Emmanuel Kuntsche, and Daniel Gatica-Perez. *DrinkSense: Characterizing Youth Drinking Behavior using Smartphones*. Tech. rep. 2018.

[Sci] Scikit-learn. *scikit-learn: machine learning in Python*. `https://scikit-learn.org/stable/`.

[SDA99] Daniel Salber, Anind K Dey, and Gregory D Abowd. "The context toolkit: aiding the development of context-enabled applications". In: *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*. ACM. 1999, pp. 434–441.

[Sho+15] Muhammad Shoaib, Stephan Bosch, Ozlem Durmaz Incel, Hans Scholten, and Paul JM Havinga. "A survey of online activity recognition using mobile phones". In: *Sensors* 15.1 (2015), pp. 2059–2085.

[SKR] Pekka Siirtola, Heli Koskimäki, and Juha Röning. "Personalizing human activity recognition models using incremental learning". In: ().

[SKR11] Pekka Siirtola, Heli Koskimäki, and Juha Röning. "Periodic quick test for classifying long-term activities". In: *Computational Intelligence and Data Mining (CIDM), 2011 IEEE Symposium on*. IEEE. 2011, pp. 135–140.

[SKR16] Pekka Siirtola, Heli Koskimäki, and Juha Röning. "From user-independent to personal human activity recognition models using smartphone sensors". In: *Proc ESANN'16* (2016), pp. 471–476.

[SKR18] Pekka Siirtola, Heli Koskimäki, and Juha Röning. "Experiences with Publicly Open Human Activity Data Sets". In: (2018).

[SR12] P. Siirtola and J Röning. "Recognizing human activities user-independently on smartphones based on accelerometer data". In: *International Journal of Interactive Multimedia and Artificial Intelligence* 1.5 (2012).

[SR13] Pekka Siirtola and Juha Röning. "Ready-to-use activity recognition for smartphones". In: *Computational Intelligence and Data Mining (CIDM), 2013 IEEE Symposium on*. IEEE. 2013, pp. 59–64.

[SR16] Pekka Siirtola and Juha Röning. "Reducing Uncertainty in User-independent Activity Recognition-A Sensor Fusion-based Approach." In: *ICPRAM*. 2016, pp. 611–619.

[Tap+07] E.M. Tapia, S.S. Intille, W. Haskell, K. Larson, J. Wright, A. King, and R. Friedman. "Real-time recognition of physical activities and their intensities using wireless accelerometers and a heart rate monitor". In: *11th IEEE International Symposium on Wearable Computers*. IEEE. 2007, pp. 37–40.

[Ten] TensorFlow. *TensorFlow*. `https://www.tensorflow.org/`.

[TK07] Grigorios Tsoumakas and Ioannis Katakis. "Multi-label classification: An overview". In: *International Journal of Data Warehousing and Mining (IJDWM)* 3.3 (2007), pp. 1–13.

[Uni] Waikato University. *Weka 3: Machine Learning Software in Java*. `https://www.cs.waikato.ac.nz/ml/weka/index.html`.

[Ven+08] Celine Vens, Jan Struyf, Leander Schietgat, Sašo Džeroski, and Hendrik Blockeel. "Decision trees for hierarchical multi-label classification". In: *Machine learning* 73.2 (2008), p. 185.

[Wan+16] Jiang Wang, Yi Yang, Junhua Mao, Zhiheng Huang, Chang Huang, and Wei Xu. "Cnn-rnn: A unified framework for multi-label image classification". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 2285–2294.

[Xu+13] Ye Xu, Mu Lin, Hong Lu, Giuseppe Cardone, Nicholas Lane, Zhenyu Chen, Andrew Campbell, and Tanzeem Choudhury. "Preference, context and communities: a multi-faceted approach to predicting smartphone app usage patterns". In: *Proceedings of the 2013 International Symposium on Wearable Computers*. ACM. 2013, pp. 69–76.

[Yür+16] Özgür Yürür, Chi Harold Liu, Zhengguo Sheng, Victor CM Leung, Wilfrido Moreno, and Kin K Leung. "Context-awareness for mobile sensing: A survey and future directions". In: *IEEE Communications Surveys & Tutorials* 18.1 (2016), pp. 68–93.

[Zha09] Min-Ling Zhang. "M l-rbf: Rbf neural networks for multi-label learning". In: *Neural Processing Letters* 29.2 (2009), pp. 61–74.

[ZZ05] Min-Ling Zhang and Zhi-Hua Zhou. "A k-nearest neighbor based algorithm for multi-label classification." In: *GrC* 5 (2005), pp. 718–721.

GLOSSARY

ADL Activities of Daily Living.

C4.5 Algorithm used to create a decision tree.

CART Classification and Regression Trees.

Cramer's V Method to calculate the association between two sets of categorical classes.

CRF Conditional Random Fields.

CSN Community Similarity Network.

DNN Deep Neural Network.

HAL Hardware Abstraction Layer.

HAR Human Activity Recognition.

HMM Hidden Markov Model.

IC Integrated Circuit.

knn K Nearest Neighbors.

Naive Bayes Probabilistic classifier based on the Bayes theorem.

One-hot encoding Technique to transform a categorial feature into multiple numerical features, one for each category, where the value is set to 1 if it is the category from the original feature, or 0 otherwise.

PQT Periodic Quick Test.

QDA Quadratic Discriminant Analysis.

RBM Restricted Boltzmann Machines.

Sensor metrics Set of metrics captured for a sensor.

SVM Support Vector Machine.

Theil's U Allows to calculate the association between two sets of categorical classes, but the direction of association is also taken into account.

User context Specific context information related to the user of a device.