Department of Computer Science
Series of Publications A
Report A-2018-2

Crowdsensed Mobile Data Analytics

Ella Peltonen

To be presented, with the permission of the Faculty of Science of the University of Helsinki, for public examination in Auditorium PIII, Porthania, City Center, Helsinki on February 26th, 2018 at 12 o'clock noon.

University of Helsinki
Finland

Supervisor
Prof. Sasu Tarkoma, University of Helsinki, Finland
Dr. Petteri Nurmi, University of Helsinki, Finland

Pre-examiners
Prof. Mika Ylianttila, University of Oulu, Finland
Prof. Cristian Borcea, New Jersey Institute of Technology, USA

Opponent
Prof. Nicholas Lane, University of Oxford, United Kingdom

Custos
Prof. Sasu Tarkoma, University of Helsinki, Finland

Contact information

Department of Computer Science
P.O. Box 68 (Gustaf Hällströmin katu 2b)
FI-00014 University of Helsinki
Finland

Email address: info@cs.helsinki.fi
URL: http://www.cs.helsinki.fi/
Telephone: +358 2941 911, telefax: +358 9 876 4314

ISSN 1238-8645
ISBN 978-951-51-4051-7 (paperback)
ISBN 978-951-51-4052-4 (PDF)
Computing Reviews (1998) Classification: H.2.8, H.1.1, H.1.2
Helsinki 2018
Unigrafia

Crowdsensed Mobile Data Analytics

Ella Peltonen

Department of Computer Science
P.O. Box 68, FI-00014 University of Helsinki, Finland
ella.peltonen@cs.helsinki.fi
https://www.cs.helsinki.fi/u/peltoel/

PhD Thesis, Series of Publications A, Report A-2018-2
Helsinki, February 2018, 100+91 pages
ISSN 1238-8645
ISBN 978-951-51-4051-7 (paperback)
ISBN 978-951-51-4052-4 (PDF)

Abstract

Mobile devices, especially smartphones, are nowadays an essential part of everyday life. They are used worldwide and across all the demographic groups - they can be utilized for multiple functionalities, including but not limited to communications, game playing, social interactions, maps and navigation, leisure, work, and education. With a large on-device sensor base, mobile devices provide a rich source of data. Understanding how these devices are used help us also to increase the knowledge of people's everyday habits, needs, and rituals. Data collection and analysis can thus be utilized in different recommendation and feedback systems that further increase usage experience of the smart devices.

Crowdsensed computing describes a paradigm where multiple autonomous devices are used together to collect large-scale data. In the case of smartphones, this kind of data can include running and installed applications, different system settings, such as network connection and screen brightness, and various subsystem variables, such as CPU and memory usage. In addition to the autonomous data collection, user questionnaires can be used to provide a wider view to the user community. To understand smartphone usage as a whole, different procedures are needed for cleaning missing and misleading values and preprocessing information from various sets of variables. Analyzing large-scale data sets - rising in size to terabytes - requires understanding of different Big Data management tools, distributed computing environments, and efficient algorithms to perform suitable data analysis and machine learning tasks. Together, these procedures and methodologies aim to provide actionable feedback, such as recommendations and visualizations, for the benefit of smartphone users, researchers, and application development.

This thesis provides an approach to a large-scale crowdsensed mobile analytics. First, this thesis describes procedures for cleaning and preprocessing mobile data collected from real-life conditions, such as current system settings and running applications. It shows how interdependencies between different data items are important to consider when analyzing the smartphone system state as a whole. Second, this thesis provides suitable distributed machine learning and statistical analysis methods for analyzing large-scale mobile data. The algorithms, such as the decision tree-based classification and recommendation system, and information analysis methods presented in this thesis, are implemented in the distributed cloud-computing environment Apache Spark. Third, this thesis provides approaches to generate actionable feedback, such as energy consumption and application recommendations, which can be utilized in the mobile devices themselves or when understanding large crowds of smartphone users. The application areas especially covered in this thesis are smartphone energy consumption analysis in the case of system settings and subsystem variables, trend-based application recommendation system, and analysis of demographic, geographic, and cultural factors in smartphone usage.

Computing Reviews (1998) Categories and Subject Descriptors:
H.1.1 Information Systems, Value of information
H.1.2 User/Machine Systems, Human factors
H.2.8 Information Systems, Data mining

General Terms:
Crowdsensing, Mobile Devices, Data Analytics

Additional Key Words and Phrases:
Data Cleaning, Machine Learning, Large-scale Data Analysis

Acknowledgements

The funded PhD position of Doctoral Programme in Computer Science (DoCS) has made it possible for me to focus full time on my PhD research and travel to important conferences of my research area. I also extend my gratitude to Nokia Foundation that awarded me the Scholarships in 2015 and 2016. These external fundings made it possible for me to visit University College London, UK, during the academic year 2015 – 2016.

I would like to thank my supervisors, Professor Sasu Tarkoma and Dr Petteri Nurmi, who have supported me through the ups and downs of my PhD process. Dr Eemil Lagerspetz, with whom I have worked from my undergraduate traineeships, has been an important co-worker during all these years. I am also grateful to my co-authors: Professor Stephan Sigg from Aalto University, Dr Mirco Musolesi and Dr Abhinav Mehrotra from University College London, and Jonatan Hamberg and all the other research assistants of the Carat project. Thank you for your brilliant ideas and precious discussions.

For me, one of the best parts of becoming a researcher has been meeting so many outstanding people around the world. There are too many names to list them all, but I would like to especially thank the following people for their support and encouragement: Professor Cecilia Mascolo and Dr Eiko Yoneki from the University of Cambridge, UK, Dr Aarathi Prasad from Skidmore College, US, and Dr Denzil Ferreira and Dr Susanna Pirttikangas from the University of Oulu, Finland. N^2 Women and ACM-W Europe have provided me with insightful networks for discussion and meeting great researchers.

Many co-workers, colleagues, and friends of mine have supported me beyond reason - I will always remember you warmly. At the University of Helsinki, Dr Pirjo Moen and the Niklander family have offered me invaluable help on every kind of practicalities and everyday problems. I would also like to extend my thanks to everyone who hosted me during my research visits and trips all around the world, especially the people in the Intelligent Social Systems Lab at University College London, the Computer Laboratory at the University of Cambridge, and the Insight Centre at University College Cork, Ireland, together with the Jokela and Gibbs families in the UK.

My own family in Finland has supported me beyond all expectations, with warmth, trust, and purrs. Last, all my love and gratitude belongs to my husband Iivari. *This is hard, but that's how I wanted it.*

In Cork, Ireland, January 30, 2018

Ella Peltonen

Contents

1 Introduction **1**
- 1.1 Motivation 1
- 1.2 Problem Statement 2
- 1.3 Methodology 4
- 1.4 Thesis Contributions 8

2 Background: Crowdsensing for Mobile Devices **13**
- 2.1 Mobile Crowdsensing 14
- 2.2 Data Cleaning and Processing 15
- 2.3 Generating Recommendations 16
  - 2.3.1 Energy Recommendations 17
  - 2.3.2 Application Recommendations 18
- 2.4 Analyzing Mobile Usage 19
- 2.5 Large-Scale Data Analysis 21

3 The Carat Project **23**
- 3.1 Collecting Large-scale Mobile Data 23
- 3.2 The Carat Data Statistics 24
- 3.3 User Background Questionnaire 25
- 3.4 Limitations of the Carat Dataset 28
- 3.5 Ethical Considerations 28

4 Cleaning and Preprocessing Crowdsensed Mobile Data **31**
- 4.1 Nominal and Ordinal Attributes 32
- 4.2 User-changeable System Settings 33
- 4.3 Subsystem Variables 34
- 4.4 Energy Measurements 37
- 4.5 Detecting Country 38
- 4.6 Applications 40
- 4.7 Application Categories 41

5 Methodology for Analyzing Crowdsensed Data 43
5.1 Information Metrics 43
5.1.1 Energy Impact of System Settings and Subsystems . 44
5.2 Trend Mining 47
5.3 Analyzing Similarity of Usage 48
5.3.1 Demographic Usage Differences 49
5.3.2 Geographic Usage Differences 50

6 Decision Making and Actionable Recommendations 55
6.1 Energy Modeling of System Settings 58
6.2 Application Trend Based Recommendations 62
6.3 Insights into Demographic, Geographic, and Cultural Factors in Mobile Usage 65
6.3.1 Demographic Factors 66
6.3.2 Geographic Factors 69
6.3.3 Cultural Factors 73

7 Conclusions 79
7.1 Summary of the Main Findings 79
7.2 Implications of the Research 82
7.3 Limitations 83
7.4 Future Work 84
7.5 Conclusion 85

References 87

Research Theme A: Mobile Energy Consumption 103
Research Paper I: Energy Modeling of System Settings: A Crowdsourced Approach 103
Research Paper II: Constella: Crowdsourced System Setting Recommendations for Mobile Devices 115

Research Theme B: Mobile Application Usage 137
Research Paper III: Exploiting Usage to Predict Instantaneous App Popularity: Trend Filters and Retention Rates 137
Research Paper IV: The Hidden Image of Mobile Usage: Uncovering the Impact of Geographic and Demographic Factors . . 161

Chapter 1

Introduction

1.1 Motivation

Mobile devices, especially smartphones, are nowadays an important part of everyday life. Different mobile applications support work life, well-being, education, and leisure time. Because smartphones are flexible and easy to carry, they have replaced multiple single-purpose devices, such as regular mobile phones, pocket cameras, gaming consoles, maps, and navigators. To enable all these multipurpose functionalities, smartphones have to implement different sensing capabilities on their programming interface. Because of this, smartphones provide a rich source of different types of data available: sensor readings, running applications, system settings, and different subsystem variables, such as CPU and memory usage. This information, especially collected from multiple devices, can provide important insights in how people behave and what kind of needs they have in their everyday life.

Guo et al. [1] define *crowdsensing* as a large-scale sensing paradigm based on user-companioned everyday devices, including, for example, mobile phones, tablets, and many wearable devices. In the future, many new household devices, such as smart TVs, fridges, and cars, will join this Internet-connected crowd. Crowdsensing is based on collaboration of a heterogeneous *crowd* of smart devices. Analysis of that kind of data collected from multiple devices can provide novel insights and help to consider what is normal in the device community. Sometimes the term *crowdsourcing* is used in the same meaning, but often it involves human-provided input, whereas crowdsensing indicates an autonomous process where a crowd of devices is used as self-supporting sensors [2].

Ganti et al. [3] remind us that there are challenges, but also a lot of new opportunities in crowdsensing applications. Smartphones and other mobile devices have become efficient with computational power, storage space, and communication capabilities. Mobile devices are largely carried along everywhere people go and whatever they do. These features also make smartphones different than traditional sensor networks, where sensor functionality and location were often considered for a single purpose only.

Often a cloud or single virtual machines are used for back-end processes, such as managing data collection, data cleaning and processing, and the actual analysis phase. Because smart devices produce easily large amounts of data in a comparably short period of time, also techniques and technologies related to Big Data processing and distributed computing environments have to be considered. The data analysis output, for example, feedback, visualizations, and recommendations, can thus be sent back to the devices from the back-end service.

This thesis focuses on crowdsensing for smart devices, especially smartphones. It will cover three key topics: crowdsensed data collection, data cleaning and processing procedures, and it will present three example cases of how crowdsensed data analytics can be utilized. These example cases are the following: First, we show how system settings and subsystem variables of the smartphones can be adjusted to save energy and provide longer battery life. Second, we analyze application trends and present a methodology to improve application recommendations based on the actual success of different applications. Third, we analyze mobile users worldwide and suggest mobile usage as a novel cultural factor to define cultural boundaries between countries.

1.2 Problem Statement

Holistic understanding of smartphone crowdsensed data is an important open research topic. Complex interdependencies between application usage, system settings, and different subsystem variables, together with a need for real-life data, make holistic analysis challenging. This thesis aims to provide techniques and methods for analyzing mobile usage in the wild and generating actionable recommendations for optimizing smartphone functionalities, such as energy efficiency, recommendation of suitable applications, and understanding smartphone usage as a whole.

Jagadish et al. [4] define challenges for Big Data processing, which are relevant to the crowdsensing applications especially taking into account the amount of data smartphones are capable of producing in a short period of time. Four of these challenges that are especially covered in this thesis, are:

- **Data acquisition.** The programming interfaces of the smartphones usually provide a wide set of sensors and other readings also for third-party developers. These can be utilized for data collection. In Section 2 we discuss in more detail for which purposes mobile data have been collected.

- **Information extraction and cleaning.** Crowdsensed data is only rarely usable directly, but there is a need for preprocessing and cleaning procedures. In Section 4 we present attributes that are easy to collect from smartphone platforms, and what kind of cleaning procedures we have applied to these attributes.

- **Modeling and analysis.** The large scale of crowdsensed mobile data sets is own challenge alone. In Section 5 we discuss distributed systems and algorithms used to scope performance and effectiveness of the analysis procedures. We also give examples of how these methodologies have been utilized in our work.

- **Interpretation.** Understanding the analysis results is crucial when aiming to provide recommendations that are of real utility back to the devices. In Section 6, we present use cases for actionable, human-readable recommendations and decision making based on the crowdsensed data analysis.

Taking into account these challenges, the research questions considered in this thesis can be listed as the following:

RQ1. How do different data attributes have to be cleaned and preprocessed to produce a reliable picture of the system state?

RQ2. How can crowdsensed data be used to present crucial factors of a smartphone's system state?

RQ3. What are the effects of subsystem variables, system settings, and their combinations to smartphone energy consumption?

RQ4. How can smartphone energy consumption be improved by recommending better system state and subsystem variables?

RQ5. How can mobile recommendation systems be improved by analyzing application popularity?

RQ6. What can be learned about mobile application usage and popularity in real-life crowdsensed data?

RQ7. How does mobile application usage reflect differences in user population?

RQ8. What can be learned about cultural, demographical, and geographical differences in crowdsensed smartphone usage?

Figure 1.1 presents how the research questions are covered in the publications listed below in Section 1.4 and also shortly summarizes methodologies involved in each research question. The first four research questions closely relate to smartphone energy analysis, even if findings and methodologies may be useful also in other application areas. RQ1 reflects a need for real-life data to understand actual usage cases and environments when studying smartphone usage and, for example, energy consumption. RQ2 studies how data gathered by a crowdsensed system need to be preprocessed and cleaned to produce reliable results. RQ3 derives analysis of complex interdependencies between system settings and subsystem variables, and RQ4 presents how these interdependencies can be modeled to generate actionable, human-understandable energy recommendations.

RQ5 and RQ6 relate to application usage analysis. First, RQ5 manages application popularity based on real-life crowdsensed data and answers the question, what happens after applications are installed to the device? Second, RQ6 focuses on the question how usage information can be utilized for application recommendation systems. RQ7 and RQ8 aim to deepen the understanding of smartphone usage in the wild. RQ7 delivers information about the effect of culture and demography in smartphone application usage, and RQ8 aims to describe smartphone usage as a modern cultural factor in benefit of the research community.

1.3 Methodology

Machine learning algorithms and statistical tests are crucial to understand interdependencies and relationships in the crowdsensed data. To generate actual value out of the analysis output, we have to consider how these results are presented in a human-readable, understandable and actionable way. The aims of large-scale crowdsensed data analysis include providing useful information out of the data to be used, for example, making decisions, generating recommendations, and showing helpful visualizations based on the data.

In the continuous sensing process, better usage suggestions on the device side would also generate back to the data and its analysis process. This phenomenon can be called the *continuous feedback loop.* Figure 1.2 presents

Figure 1.1: Research questions and their matching publications along with the methodology used.

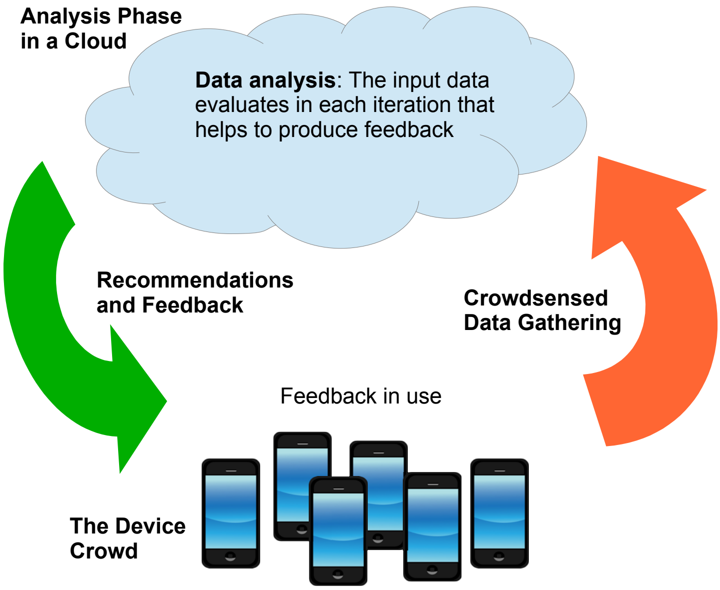


Figure 1.2: An example of a continuous feedback loop for crowdsensing applications.

an example of the continuous feedback loop, where data collected from a crowd of mobile devices is evaluated in the cloud back-end, and learning output is sent back to the devices as recommendations and feedback.

Figure 1.3 visualizes the whole process required for crowdsensed systems applying machine learning procedures and actionable feedback loop, where devices are used not only to collect the data, but also benefit the analysis output. The main phases of the system can be listed as the following, numbers of the list matching the ones in Figure 1.3:

1. A smartphone application developed for data readings and collection to perform the actual crowdsensing phase.

2. A back-end service or a cloud computing environment to manage load balancing, data storage, and the data cleaning and analysis procedures, which are next given in more detail.

3. **Data cleaning and preprocessing** to handle missing data items, unexpected values, and develop further information from attribute combinations and their interdependencies. For example, this thesis

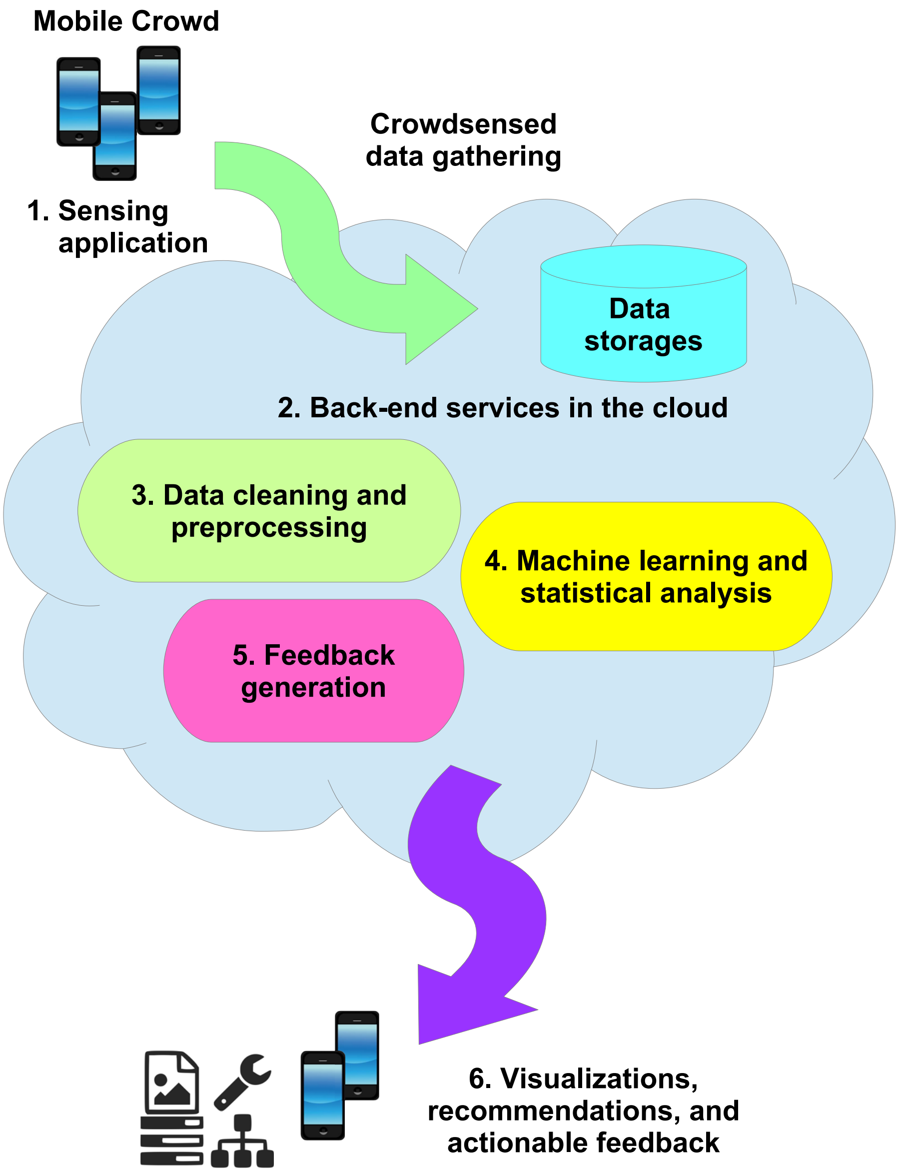


Figure 1.3: Example of a crowdsensing system that utilizes machine learning and actionable feedback.

gives approaches to clean system settings and subsystem variables by defining their reasonable operation ranges, developing general categorized usage of running applications, and present country based on network and timezone information.

4. **Machine learning algorithms** to provide statistical information, data models, and novel knowledge from the data. For example, this thesis uses information analysis - mutual and conditional mutual information - to present statistical associations, decision trees to model transactions between system states, retention rates and trend filters to understand application popularity, and the Kullback-Leibler divergence to analyze differences in application usage.

5. **Post-processing of algorithms' output** to provide actionable recommendations, feedback, visualizations, etc, to the devices and analysis environments. For example, this thesis presents how to provide energy recommendations based on system settings and subsystem variables, how to improve application recommendations based on the trend filtering, and what can be learned about cultural, demographical, and geographical differences in mobile usage.

6. The devices and other end-users, such as developers and researchers, utilizing the output of the data analysis.

The main contributions of this thesis are to give approaches for (i) the crowdsensed data cleaning and preprocessing, which is challenging with the data collected from real-life conditions, (ii) providing suitable machine learning and statistical analysis procedures that can handle large amounts of data in a sufficient period of time, and (iii) generating actionable feedback, such as recommendations and human-readable analysis results, that can be utilized in the mobile devices themselves or when understanding large crowd of smartphone users.

1.4 Thesis Contributions

The author of this work contributes the following published articles and manuscripts under revision. When referring to *the author*, it indicates the author of this thesis. These publications and manuscripts also construct the outline of this thesis, and the main focus has been given to the work the author has contributed herself.

Publication I: Energy Modeling of System Settings: A Crowdsourced Approach. Ella Peltonen, Eemil Lagerspetz, Petteri Nurmi, and Sasu Tarkoma. Published in the Proceedings of the IEEE International Conference on Pervasive Computing and Communications, PerCom '15, St. Louis, MO, USA, March 23-27, 2015.

Contribution: The author was in the lead of the planning of the publication, implementing necessary distributed data mining and statistical analysis algorithms, analyzing the data, and writing the publication. The data collection itself is based on the earlier work done in the Carat project lead by Dr Eemil Lagerspetz. Dr Petteri Nurmi and Prof. Sasu Tarkoma gave important contributions to the planning and writing processes of the publication.

Publication II: Constella: Crowdsourced System Setting Recommendations for Mobile Devices. Ella Peltonen, Eemil Lagerspetz, Petteri Nurmi, and Sasu Tarkoma. Published in Pervasive and Mobile Computing, Volume 26, February 2016, pages 71 - 90.

Contribution: The publication extends Publication I with a novel recommendation system for energy consumption of system settings and subsystem variables. Some parts of the work is based on the author's Master's Thesis published in 2013 at the University of Helsinki[1]. The author was responsible for implementing the decision tree-based recommendation system, perform the data analysis procedures, and write the publication. Dr Eemil Lagerspetz, Dr Petteri Nurmi, and Prof. Sasu Tarkoma contributed to the planning and writing process of the publication.

Manuscript I: Exploiting Usage to Predict Instantaneous App Popularity: Trend Filters and Retention Rates. Stephen Sigg, Eemil Lagerspetz, Ella Peltonen, Petteri Nurmi, and Sasu Tarkoma. A preprint is available in `https://arxiv.org/abs/1611.10161`. Under submission and review to a journal publication.

Contribution: The publication was lead by Prof. Stephan Sigg who delivered the main ideas, methodology, and structure of the publication. The author contributed by participating in the planning of the publication, and implementing and running the application recommendation system for the validation and use case of the trend filter analysis. The author also

[1] `http://hdl.handle.net/10138/40924`

gave comments through the process and participated in the writing of the publication together with other authors.

Manuscript II: The Hidden Image of Mobile Usage: Uncovering the Impact of Geographic and Demographic Factors. Ella Peltonen, Eemil Lagerspetz, Jonatan Hamberg, Abhinav Mehrotra, Mirco Musolesi, Petteri Nurmi, and Sasu Tarkoma. Under submission and revision to a journal publication.

Contribution: The publication started in collaboration between the author and researchers at University College London, Dr. Mirco Musolesi and Dr. Abhinav Mehrotra. Most of the ideas that lead to the publication were delivered through the author's research visit to University College London. The author was in the lead of the data analysis work, planning the additional data gathering, such as the user background questionnaires, and constructing the publication. Jonatan Hamberg and Dr Eemil Lagerspetz contributed significantly to the implementation of the questionnaire and data collection system, and together with Dr Petteri Nurmi and Prof. Sasu Tarkoma, they participated by sharing ideas and in the writing process.

The thesis is organized as follows: Section 2 provides the state of the art for mobile crowdsensing, presents the mobile dataset used as a source of the analysis of the listed articles, and considers ethical issues related to the crowdsensing mobile data. Section 4 discusses data cleaning procedures and techniques, and presents the main attributes available in mobile devices without complicated permission policies. Section 5 discusses distributed machine learning and statistical analysis techniques used to generate the results in the listed articles. Section 6 presents the main use cases of this work, including actionable feedback and recommendation systems for smartphones. Finally, Section 7 concludes the thesis with a summary of the main findings, discussion of limitations, and possibilities for relevant future work.

To summarize, the contributions of this thesis are the following:

- The thesis provides an approach for the **crowdsensed mobile data cleaning and preprocessing**, which is challenging with the data collected from real-life conditions. This thesis shows how interdependencies and relationships between different context factors are important to consider when analyzing mobile usage and aims to understand the smartphone system state as a whole.

- This thesis provides suitable **distributed machine learning and statistical analysis** procedures that can handle large amounts of data in a sufficient period of time. The algorithms, such as the decision tree-based classification and recommendation system, and information analysis methods presented in this thesis, are implemented in the distributed cloud-computing environment Apache Spark.

- This thesis provides approaches to **generating actionable feedback**, such as recommendations and human-readable analysis results, which can be utilized in the mobile devices themselves or when understanding large crowds of smartphone users. Understanding smartphone usage as a whole provides insights in how people use their devices and which kind of needs they have for, for example, better battery life and finding new and more successful applications.

Chapter 2

Background: Crowdsensing for Mobile Devices

Mobile devices, such as smartphones, tablets, and smart watches, are nowadays an important part of everyday life[1]. Mobile devices are nowadays used instead of several previous hand-held devices, such as cameras, navigators, and gaming consoles. In addition to applications, smart devices come with a set of various sensors, settings, and other functionalities sometimes hidden from the user. Always carried along and interacted with around 60 times per day [5], they provide a rich source of information on the everyday habits of their users.

Crowdsensing mobile usage data from large sets of users worldwide provides an access to the real everyday life of people. No laboratory simulations can provide such detailed and well covered information, because the amount of possible usage combinations of different applications and system settings rises to incalculable. On the other hand, application programming interfaces of modern smartphone platforms provide various sets of easy to access attributes. Indeed, smart device usage information can be increasingly collected through non-obtrusive instrumentation of the device. For example, the Carat [6][2] and Device Analyzer projects [7, 8][3] have collected smartphone crowdsensed data worldwide.

Experiments conducted through a combination of laboratory measurements, such as power meter measurements, and a large-scale analysis of crowdsourced measurements demonstrate that the crowdsensing method-

[1]Newzoo ranked top 50 countries by the number of smartphone users, with average smartphone penetration of 39.4% or total 2.4 bn smartphone users: `https://newzoo.com/insights/rankings/top-50-countries-by-smartphone-penetration-and-users/`.

[2]The Carat project: `http://carat.cs.helsinki.fi/`

[3]The Device Analyzer project: `https://deviceanalyzer.cl.cam.ac.uk/`

ology is capable of constructing models that accurately capture complex interdependencies between system settings, sensors, and usage contexts, providing an accurate view of the *system state* of the device. In contrast with previous works, which have predominantly focused on capturing the effects of specific sensors, system settings or applications [9, 10], a methodology presented in this thesis focuses on interdependencies and the device as a whole.

2.1 Mobile Crowdsensing

This thesis and multiple previous projects consider mobile devices and its system state as a sensor. A wide sensor base of mobile devices makes crowdsensing possible to be utilized for multiple purposes, and all the possible application areas are impossible to list. A great part of previous work has focused on analyzing device- or user-specific patterns, for example, identifying potential malware infections on the smartphones [11], analyzing network traffic and what it can reveal about the device and its user [12], or identifying and characterizing the current user of the device [13].

As carry-on devices, smartphones are easy to utilize as sensors in various conditions. One of the popular application areas is transportation mode sensing, which often utilizes sensors like accelerometer, location information, cell tower availability, and other network signals. For example, Koukoumidis et al. [14] present a system called SignalGuru that uses a smartphone's camera to predict and analyze traffic signals on roads. Hemminki et al. [15] use accelerometer and GPS location points to detect current transportation mode, such as bus, train, or walking.

Mobile devices work as sensors also indoors in contrast to, for example, GPS and network signals possible unaccessible or weak indoors. For example, images captured by camera may be used to deliver information about the usage context. Radu et al. [16] monitor indoor Wi-Fi networks, Gao et al. [17] model indoor structures and landmarks, and Chon et al. [18] present a methodology to deliver information of the place from images and audio files collected by mobile crowdsensing.

A great interest has been given to recommendation systems that help users, for example, to gain a longer battery life or choose more useful applications. In general, analyzing large-scale smartphone usage data provides an access to a rich source for knowledge. Next, we consider the state of the art in the mobile crowdsensing application areas that are especially focused on in this thesis.

2.2 Data Cleaning and Processing

The term *data cleaning* describes a process where errors, inconsistencies, and missing items in the data set are removed, replaced, or otherwise handled [19]. Data cleaning aims to improve data quality and remove misleading values, for example, unnecessary default values that may affect the reliability of the statistical distributions significantly. Data cleaning is often mentioned as one of the key challenges when analyzing and processing Big Data [20] and especially the data automatically collected from sensing devices [21, 22, 23].

Based on the study of Strong et al. [24] from the year 1997, the data quality has been an important issue at least the last twenty years. Rahm and Do [19] provide an early review for data cleaning and preprocessing procedures. They list, for example, the following challenges and problems that are especially relevant for cleaning crowdsensed smartphone data:

- *Cryptic values and abbreviations* are common in smartphone environments where any spare data transmission should be reduced due to the network costs (in terms of both energy consumption and money). That may lead to shortened values and presenting nominal values as integers, for example. In the data analysis phase, interpretation of the data values should be considered right, and possible varying presentation forms standardized so that comparison between different device models is possible.

- *Illegal values* are, for example, min and max values should not be outside reasonable or permissible range. For example, the battery temperature cannot be very high or very low due to the sensor capability to read the lithium battery, and CPU usage should be given between 0 and 1, or respectively, 0% to 100%.

- *Misspellings and the like* can appear in user-changeable settings, for example, a wrongly selected timezone setting can be considered such. A reasonable amount of system settings is adjusted automatically or the user can only choose from the limited range of options, such as screen brightness setting is often adjusted by a slider. Thus, the risk of totally inconsiderable user-based inputs is quite small.

- *Missing values* can appear in the data due to a technical error, limited access to the resource, or the presence of a default value that may indicate a missing value. The missing values have to be recognized, removed, and at least, not included in the data processing and analyzing phases.

- *Varying value representations* can appear due to, for example, different manufacturers' own changes in the API. Especially missing values can be indicated as, for example, null, NaN, none, 0, or by a default value. These values have to be recognized and combined, so that their value can be considered as the same.

- *Violated attribute dependencies* mean situations where two or more data factors should be corresponding, but for some reason they are not. For example, that may be the case when the time between two samples does not match the distance traveled between them, for example, it is not possible to travel hundreds of kilometers in several minutes.

Data cleaning and management for different sensor readings have been covered in some previous literature. They focus especially on sensor readings in unreliable or noisy environments [25, 26]. To mention some relevant examples, Williamson et al. [22] study data cleaning for wearable devices, and Tong et al. [27] propose the CrowdCleaner for web-based crowdsensed data.

The sensor-based readings are often proposed to be cleaned by machine learning or other statistic approaches. Park et al. [21] use data cleaning methods for accelerometers and light sensors using thresholds to prevent outliers, episode dictionaries to model expected measurements, and the longest common subsequences to detect errors and noise in the data. Also Jeffery et al. [23, 28] present methodologies to manage missed and unreliable data readings. Several database repairing schemes are also studied and presented in the literature [29, 30].

In some cases human input is required for successful cleaning. Chu et al. [31] use crowdsourcing to validate appropriate patterns in the data. More often human work is involved to set parameters and threshold values [32], if they are not possible to learn by statistical and other autonomous methods. In our approaches, we prefer combining autonomous and human-driven approaches, for example, setting "natural" thresholds whenever available but validating findings by statistical methods.

2.3 Generating Recommendations

Recommendations are a way to introduce users to better usage policies and help them to learn hidden features of their smart devices. Great interest has been given to help users understand their devices' energy consumption in terms of gaining a longer battery life. Another important topic considers choosing the right applications out of millions of them available in the app markets. Next, this thesis covers the current state of the art related to these topics.

2.3.1 Energy Recommendations

Mobile energy profiling refers to the process of characterizing the energy consumption of a mobile device, including running applications, system settings, sensors, and other subsystem variables and hardware components. Energy profiling is typically carried out by constructing one or more statistical models that can be correlated with specific system states with energy consumption patterns. The goal of the energy modeling is to identify energy bottlenecks at runtime and to provide actionable recommendations on how the lifetime can be improved.

The previous research provides some insights in how people consider their device's battery life and how they tend to charge the device. Banerjee et al. [33] conduct an user study showing that, for example, users tend to leave their smartphones charging overnight or whenever it is otherwise possible. They also provide a method to save energy especially focusing on the screen brightness. Rahmati et al. [34, 35] study how people interact with their device's batteries and show that people can be divided into two groups: those who charge regularly once or more a day regardless of the battery level, and those who follow notifications and feedback given by a battery manager. Ferrera et al. [36] study how understandable different battery interfaces are, and note that users tend to have very limited knowledge what to do when they face battery problems.

Improving the user's understanding of the battery lifetime of their devices requires human-readable energy recommendations. These recommendation systems can provide warnings of bug-behavior applications, which for example, Banerjee et al. [37] suggest in their study. Ma et al. [38] present a system called eDoctor that monitors battery drain and gives suggestions about possible energy-hungry applications and suspicious system events, such as heavy network traffic. Pathak et al. [39] focus on monitoring the operation system and especially abnormal CPU usage of the device. Shye et al. [40] also focus on analyzing the effect of CPU and screen brightness on the battery life.

The measurements for constructing energy models can be gathered either using specialized hardware in laboratory conditions, such as the Monsoon power monitor [4] or BattOr [41], or through the battery interface of the device [6, 42, 43]. Benefits of the data-driven approaches include capability

[4]Monsoon Power Monitor: `https://www.msoon.com/LabEquipment/PowerMonitor/`

to catch a large variety of real-life use cases. For example, Falaki et al. [10] conduct an analysis of smartphone usage patterns, revealing that usage patterns contain significant variation across users and that personalized application usage models are essential for accurate prediction of battery drain.

Agarwal et al. [44] build in MobiBug a data-driven approach for energy diagnosis. The DeviceAnalyzer project [8, 45] is gathering rich measurements of mobile device state, but the data has not yet been used for large-scale analysis, and its high sampling cycle (even 100,000 per day from a single device) can itself lead to unexpected and increased energy consumption.

The Carat application [6] is known as the first collaborative energy profiler that performs its analysis with large-scale crowdsensed data. To the best of our knowledge, the Constella model [46, 47] that bases on the data collected in the Carat project, is the first model capable of constructing fine-grained energy effects from crowdsourced measurements.

2.3.2 Application Recommendations

Choosing the most suitable applications out of millions available is becoming a popular topic in the recommendation research field. Most application markets integrate some version of recommendation systems by themselves, for example, Google Play supports both personalized recommendations and country-specific "featured" and most popular application listings. Also, several academic and commercial recommendation systems that focus on suggesting new applications to the end users have been proposed. These systems typically operate exclusively on top of a cloud back-end, requiring large amounts of teaching data, and relying on computationally intensive matrix factorization methods [48].

Most application recommendation systems operate directly on the marketplace and rely on application popularity, such as installation counts or ratings to generate recommendations [49, 50]. However, studies on mobile usage have shown that ratings and installation counts are often a poor indicator of user interest. Users tend to try out several applications without necessarily ever using them again [51, 52]. Some users may not uninstall unnecessary applications but rather keep them, even if they are tried only once. The same holds for ratings which do not necessarily reflect true user interest. For example, many users give a one star rating for apps that do not function properly on their device [52], and some applications, especially games, even repay for higher ratings. It has been shown that usage patterns are highly contextualized, with many applications only being used in specific contexts [53], for example, tourism or transportation apps in a visited city.

Some popular app recommendation systems include, for example, AppJoy [52] that considers a weighted model where recency, frequency, and duration of interactions are taken into consideration. Other recommendation systems, such as GetJar [54] and Djinn [55], operate on binary usage patterns. AppJoy relies on a constantly running background process that monitors app use, while both GetJar and our technique can be used with crowdsourced, infrequently sampled data. Also other works on integrating context information, such as location or timing, as part of app recommendations have been proposed [53, 56, 57, 58, 59, 60]. Recently, commercial app recommendation systems, such as Aptoide[5] and Cydia[6], have emerged.

Our work in [61] uses application usage collected by crowdsensing from real users and real use cases. It focuses on adapting classic content-based and collaborative filtering techniques for mobile usage. Information learned from the trend analysis can be further used to improve the existing application recommendation systems.

2.4 Analyzing Mobile Usage

In addition to recommendations systems, there are also other essential possibilities for benefiting crowdsensed data from mobile devices. Before this, the full picture of how mobile devices have been used worldwide needs to be covered. Various previous projects have focused, for example, presenting the effect of context, timing, and location on smartphone usage. The main challenges and limitations in these works is related to the lack of worldwide, large-scale data, but in general, they give a picture how and why mobile devices are used.

Ferreira et al. [62] present that social and spatial context have a strong influence on application usage in general. They show that mobile applications are more often used at home and alone, and a large part of interactions with the phone can be considered as a "micro-usage", such as checking notifications or just killing time. Hiniker et al. [63] show that app usage reflects both instrumental (for some purpose) and ritualistic (more habitual) behavior. The instrumental use can be, for example, looking up opportunities and utilities, tracking sport or health activity, or getting in touch with other people. The ritualistic usage includes different kinds of "time killing" activities such as browsing blogs or news, playing games, or checking social media.

[5]The Aptoide meta-store: `http://m.aptoide.com/`

[6]The Cydia package management software for jailbroken iPhones: `https://www.cydiaios7.com/`

Multiple studies show that application usage reflects diurnal and daily variation. Falaki et al.[10] perform a statistical analysis and show the existence of the diurnal patterns with significantly risen activity during daytime hours compared to nighttime hours. On the other hand, they note that the exact patterns of individual users vary. Xu et al. [64] show that news apps are the most popular in the early morning and sports apps in the evening. Böhmer et al. [65] also note the risen popularity of news as well as the built-in music app in the morning hours, Google Maps in the early evening hours, and several games and e-readers in the late evenings. Both studies agree on the risen application usage when moving around, with not only traveling applications and maps, but also video and multimedia apps. The same effect might be seen in the risen energy consumption when moving around instead of staying stationary [46]. On the other hand, smartphones are still widely used for communication purposes and the communication apps are used evenly during the day [65]. Also, Jones et al. [66] study how often the apps are revisited and show that the usage patterns depend on the application and its functionality.

Verkasalo [60] shows that the location has significant correlation how smartphones are used. Xu et al. [64] study geographical differences in application usage in the US and show that 20% of applications can be considered local. They also present that the US users tend to have multiple applications for the same purpose, for example, several news applications. Petsas et al. [67] show the similar effect that the most popular apps gain the most downloads, and the users tend to have several apps from the same categories. In general, user preferences for application usage seem to be highly clustered.

Several studies show that there are also demographic and cultural boundaries in application usage. Seneviratne et al. [68] demonstrate that application usage reflects the user's gender and age. Zhao et al. [69] study over 100.000 Chinese smartphone users and find out that they can be clustered to descriptive groups, such as, "evening learners", "young parents", "financial users", and "cat lovers". They show that there is correlation between gender, age, and income level to the application usage. Lim et al. [70] analyze application download decisions across countries, finding the importance of pricing, reviews, and app descriptions to vary across countries. Kang et al. [71] compare the US and Korean smartphone users in terms of culture and basic need, such as belongingness and self-actualization.

Mobile usage can also be used to identify cognitive or personal states. Chittaranjan et al. [72] present that smartphone usage correlates with the users' Big Five personality traits. A system called MoodScope uses applications usage patterns and other smartphone sensors to identify the user's mood [73]. Lathia et al. [74, 75] present the EmotionSense system that uses smartphones to track human behavior and changes in it. Sandstrom et al. [76] use smartphone-based crowdsensing to show that people's feelings vary in different locations and situations.

In addition to everyday mood and emotions, smartphones may help with mental illnesses. Gruenerbl et al. [77] show that smartphone sensors can be used to aid even psychiatric diagnosis. They use an accelerator to measure physical motion and GPS traces to detect travel patterns and aim to predict manic episodes of bipolar disorder patients. The MoodScope system's results are also shown to correlate with the PhQ-9 depression scores [78].

Understanding mobile usage may provide researchers and other parties valuable information of people's daily life patterns and their common needs and preferences [79]. Obviously, that kind of knowledge also benefits marketing and consumer targeting.

2.5 Large-Scale Data Analysis

Because of computational power and especially battery lifetime are limited in smartphones, a current popular approach is to collect and analyze crowdsensed data on the back-end servers, which often means introducing cloud-computing services or a cluster of virtual machines. Large-scale data processing power has become available for many users, developers, and researchers thanks to the new cloud-computing environments that do not require heavy hardware investments, but only a credit card. Amazon Web Services [7] and Microsoft Azure [8] are examples of this kind of popular cloud-computing services. The newest addition to the easy-to-access data analysis family is Gluon [9], a collaboration project between Amazon and Microsoft.

Even if these cloud-based computing resources are well available, there are challenges in implementing effective machine learning support for mobile crowdsensing. Understanding distributed environments and implementation of scalable analysis algorithms becomes crucial, when data size and diversity increase rapidly. Distributed environments require new paradigms compared to the traditional single-machine computing. MapReduce [80, 81, 82] has been seen as a leading new computational paradigm of the field, implemented

[7]`https://aws.amazon.com/`

[8]`https://azure.microsoft.com/`

[9]`https://github.com/gluon-api/gluon-api/`

in Hadoop [10] and often used together with its machine learning libraries, for example, Mahout [11] and SystemML [83].

Apache Spark [12] [84] provides a fast programming interface and supplementary features to the MapReduce paradigm together with its machine learning library MLlib [85] and programming interface MLbase [86]. These machine learning platforms implement many of the key functionalities for data analysis, such as, statistical tools for hypothesis testing and machine learning algorithms for classification, regression, clustering, recommendation making, topic modeling, and association analysis, and so on.

Users' reluctance to participate in crowdsensing projects is seen as a challenge, as well as researchers' lack of skills for mobile development [87]. Systems like AWARE [88] help researchers to launch their crowdsensing projects on a single platform without deep knowledge of smartphone app development for multiple platforms. Also, systems like AWARE already have a user base available, which reduces marketing and user acquisition costs.

The Carat application [6] uses its own data collection procedures and performs the analysis in the AWS Elastic Compute Cloud (EC2) service[13]. We implement our algorithms with the Spark platform whenever there is no library algorithm available or for some reason it does not fit the purpose intended. For example, information metrics used in our work, such as mutual information and conditional mutual information presented in Section 5.1, are not currently part of the MLlib library. From user point of view, the Carat provides actionable feedback from their battery life, which might have been a crucial element for gathering such a large user base.

[10] http://hadoop.apache.org
[11] http://mahout.apache.org
[12] http://www.spark-project.org
[13] https://aws.amazon.com/ec2/

Chapter 3

The Carat Project

Launched in June 2012 and still operating, the Carat application [6, 46] has been used to collect worldwide mobile usage data from Android and iOS devices. The project has been started in collaboration between the University of Helsinki, Finland, and University of California, Berkeley, USA. To the best of our understanding, it is currently one of the most comprehensive crowdsensed mobile data sources available including over 200 million samples from over 780,000 users.

To participate in data collection, users are not required to do anything else except download the application from a stock market: Google Play, App Store, or a separate Android package from the project website [1]. The data is collected to the Amazon EC2 cloud service and stored to the Amazon S3 data storage. Based on the analysis results, the clients show users actionable recommendations that help them to increase their battery life [89].

3.1 Collecting Large-scale Mobile Data

The Carat data collection includes multiple attributes available without extreme permissions. They are, for example, lists of the installed and running applications, user-changeable system settings, such as screen brightness and network type, and subsystem variables, such as CPU usage, memory state, and battery level. Also, user-specific hash identifier (referred to as the user's Carat id), timestamp, device model, and operating system version are recorded among others. Different mobile platforms offer varying list of system attributes, and some Android manufacturers may have included their own limitations to the programming interface. For these reasons, the

[1]The Carat project website: `http://carat.cs.helsinki.fi/`

amount and quality of items in the data may vary by manufacturer and operating system[2].

Because some of the features have been included in the system later than others, information available from specific years can vary. The newest addition is the mobile country code, which has been collected since March 2016. New data items are collected all the time, so that the system can also capture new device models, applications, and other changes in the market.

Originally designed for energy consumption research, the Carat sampling procedure takes a sample every time 1% of battery has been drained. This makes the data collection process very energy-efficient itself, but it also increases the length of time spent between two samples, especially when the smartphone is staying mainly idle. This may set some challenges in the cases where the Carat dataset is used for other than energy-efficiency research, for example, studying usage.

To respect user privacy, the Carat system does not collect any personal or contact information, such as phone numbers, calls or text messages, or exact location information. Ethical considerations are later discussed in Section 3.5. Altogether, country information can be delivered when certain factors of network and timezone are known, as we show in Section 4.5. Preliminary efforts to publish the Carat data for application developers and researchers have also been done [90], and the subset of the data consisting of system settings and subsystem variables has already been published as a part of our work [46]. This dataset is available on our website [3].

3.2 The Carat Data Statistics

Table 3.1 summarizes the statistics of the Carat data in June 2017. The entire Carat data has over 784,000 distinct user records. 48.8% of these were Android devices and 51.2% iPhones. There are more registrations to the system, over 864,000, but it might be that some users never opened the application again, so no samples have been sent to the back-end service. There are almost 215 million samples, and more is coming to the system all the time.

Different mobile platforms provide different context factors for third-party applications depending on their policies. As an open-sourced platform, Android provides the widest range of factors available and utilized by

[2] The full description of the data collection protocol can be find in `https://github.com/carat-project/carat/blob/master/protocol/CaratProtocol.thrift`

[3] The Carat context factor dataset is available in: `http://carat.cs.helsinki.fi/#Research`

Registered users	864,079
Users with samples	784,165
Android users	382,667 (48.8%)
iOS users	401,498 (51.2%)
Samples	214,931,177
Android applications	603,854
iOS applications	167,482
Raw data size	1.2 TB
Compressed data size	315GB

Table 3.1: The Carat data statistics 2nd June 2017.

application developers. Thus, in most cases of this thesis and in our previous work we consider the Android devices and a subset of the Carat data.

For example, we present an energy analysis of system settings and subsystem variables in Section 6.1 based on containing around 11.2 million samples from 150,000 active Android users. Modeling these energy combinations is based on our previous work [46], as well as a recommendation system Constella delivered on the basis of these energy models [47].

In another example that we later discuss in Sections 5.3 and 6.3, we perform a large-scale comparison of application usage in different countries. There we consider a subset of 5.65 million samples from Android devices. For those samples, we can validate the country of origin by a method later described in Section 4.5. To summarize, we compare the mobile country code obtained by the network to the country that is indicated by the timezone attribute. This procedure helps us to detect the country even when the exact GPS or Wi-Fi based location is not available for privacy reasons. The subset contains 25,323 Android users associated with 114 country codes, out of which 44 countries have a significant number of users (100 or more). Figure 3.1 presents the distribution of users whose country of origin can be tested by our methodology. The majority of the users are from the USA, but there is also a strong user base in Finland, India, Germany, and the United Kingdom among others.

3.3 User Background Questionnaire

Understanding who the users are, can provide important new insights to the smartphone usage. To collect more detailed information about the Carat users' demographic background, we sent a voluntary questionnaire within the Carat app to all active Android users. The questionnaire includes basic

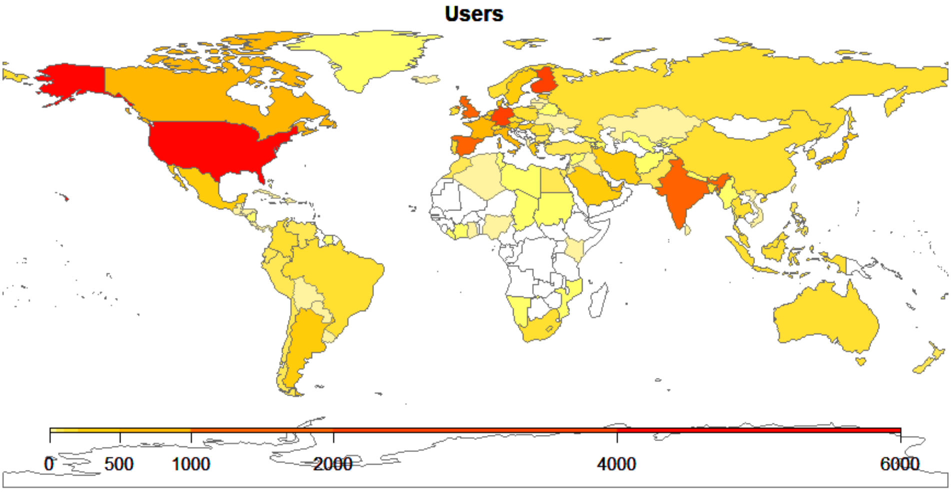


Figure 3.1: Distribution of the Carat users whose country of origin can be validated through their network's mobile country code.

background information, such as gender and age group, and socio-economic status, such as questions related to household situation and annual income. The questionnaire also records the current GPS location of the user if a permission were granted. Only adults (18 years or older) have been able to answer the questionnaire.

The following information has been collected (each question as a single choice):

1. Gender: female, male, or other;

2. Age group: 18-24, 25-34, 35-44, 45-64, or over 65 years old;

3. Current occupation: manager, professional, technician or associate professional, clerical support, sales or services, agricultural or forestry or fishery, craft and trade or plant and machine operations, entrepreneur or freelancer, student, staying at home, retired, or no suitable option;

4. Highest completed education: elementary school or basic education, high school or sixth form or other upper secondary level, vocational school or trade school or other education leading to a profession, undergraduate or lower university degree (Bachelor's or equivalent), professional graduate degree or higher university degree (Master's or equivalent), research graduate degree (PhD or equivalent);

5. Household situation: living alone, living with other adult(s), living alone with under-aged kid(s) (under 18 years old), living with other adult(s) and kid(s);

6. Annual income, compared to the user's country average: much lower, lower, about the same, higher, or much higher;

7. Debt, as percentage of monthly income need to cover it: no debt, or 10%, 25%, 50%, or most of the income;

8. Savings, as a number of months possible to live off it: less than a month, 1-3 months, 4-6 months, 7-12 months, or over a year;

9. Current coarse location, if user agrees to measure it: yes or no, measured automatically if agreed.

The users' answers can be linked to their application usage through their Carat id, a unique hash code generated automatically for each user. The questionnaire received $3,293$ responses from individuals in 44 countries. This corresponds to 14.3% of active Carat users that have the latest Carat version and thus the questionnaires available.

In comparison to the results from a prior questionnaire from 2013 [89], the demographic distributions are quite similar with the exception of user locations, where the majority now coming from Finland instead of the United States. This can be caused by the marketing bias together with the research lead switching from UC Berkeley to University of Helsinki between the studies. Another bias considers gender: 10% of answers come from female and around 87% from men. On the other hand, user questionnaires performed by mobile applications have been reported to have high gender biases before [91].

In terms of occupations, the most represented are professionals (34%), technicians or associate professionals (14%), students (12%), and managers (10%), so our questionnaire respondents are well employed. That may also reflect the general picture of owners of mobile devices. Even if they have become much cheaper in present years, there may still be financial considerations in buying such a device. The distribution of education of the respondents reflects this, too: 35% have an undergraduate degree, 30% have a Master's degree or equivalent, and 5% even have a PhD or research graduate degree. 36% of the answers report their yearly salary is higher than their country's average and 7% that it is much higher. On the other hand, age groups are evenly distributed: 12% of age 18 – 24, 30% of age 25 – 34, 28% of age 35 – 44, 27% of age 46 – 64 and 4% 65 years or older.

Section 6.3.1 later discusses the analysis of how people in different demographic groups use their smartphones. Utilizing also the country attribute, Section 6.3.2 provides comparison between different demographic and geographic influences on the mobile usage.

3.4 Limitations of the Carat Dataset

As discussed before, the Carat user population – or at least those who voluntarily take also the questionnaires – seems to be biased towards well-educated and affluent males. Since the Carat application itself does not collect any background data, it is hard to say how well these distributions represent the Carat user population in general. Because it has been mainly marketed as an energy-saving application, the user base might be biased towards people having energy issues in their smartphones.

The sampling period of the Carat application is set based on the energy consumption: whenever the battery level changes, the system collects a sample. These samples are sent to the cloud only if the actual Carat application has been opened to avoid unwanted and potentially costly and energy-influencing network traffic. This data collection method means that the time distance between two samples is unpredictable and may vary a lot between different users, usage cases, and device models. This makes utilizing certain sensors, such as accelerometer and gyroscope, mostly impossible and the Carat application does not collect this kind of data features requiring more dense and interval-based sampling.

Some limitations, such as missing items and misleading default values, can be managed by the data cleaning technologies we later discuss in Section 4. On the other hand, these methodologies are never absolutely complete, for example, in the case the default value given by the device manufacturer seems to be coherent.

3.5 Ethical Considerations

Privacy and data security have become important issues for the crowdsensed data analysis [3]. User-accompanied devices may reveal users' daily routines and locations of home and workplace, also for malicious purposes, and unwanted marketing may become irritating in some cases. This is why we take especially care of user privacy when working with the automatically collected crowdsensed data.

The Carat system only considers aggregate-level data which contains no personally identifiable information, such as exact location, calls, text messages, or phone numbers. Instead of the GPS location, only a distance between two successive samples is stored to the database. Even if application data and other possibly revealing information is collected, they are not trusted to any third parties without the full consent of how the data will be used. Our previous work [90] discusses our possible data sharing policies and plans in more detail. For example, application names can be hashed or displaced with descriptive categorical names, such as ”game” or ”flashlight”, when the data is studied by third-party researchers or developers. It is also possible, that developers can only gain access to the data collected from their own application.

The privacy protection mechanisms of Carat are detailed in our previous work [6]. The data collection of the Carat application is also a subject to the IRB process of University of California, Berkeley. Users of Carat are informed about the collected data and give their consent from their devices when installing the application from the app market.

User questionnaires performed as a part of understanding the background of the Carat users have been approved on 14 June 2016 by the IRB process of the University of Helsinki, Finland. Participation in the study has been voluntary and the users have been informed about the data collection and management procedures. During the questionnaires, the exact location of the user or some other privacy-sensitive information, such as mental state and personality tests, have been collected but only with the consent of the user.

Chapter 4

Cleaning and Preprocessing Crowdsensed Mobile Data

Crowdsensing for smart devices is based on automatized data gathering processes. Hence, there is a possibility for unsuccessful readings and errors, for example, in case the device itself is in an incapable state, or the manufacturer or network operator limits accesses to certain factors. A good example of this kind of behavior can be seen when Apple closed access to the list of running processes from third-party developers on the iOS version 9 [1]. The operation system Sailfish in the Jolla phones is claimed to support also Android applications with an emulator, but in reality, most of the sensor readings though the emulator were unsuccessful [2]. Instead of coherent values, manufacturers and network operators may provide different default values, replacements, or empty fields. In addition, there is always a risk of programming bugs especially in autonomous processes.

Smartphones are considered highly privacy sensitive devices, as discussed before in Section 3.5. For that reason, there is a lack of some information, for example, in the Carat data no exact location information has been collected. Some useful information may be missing due to technical features, such as application information provides only a slight view of the actual functionality of the application. Thus, we need methods for developing new information from existing data attributes.

Mobile devices provide a rich source for different settings, applications, and other features that describe the usage context of the device. Some of them are only possible to collect when special permissions are received,

[1] Preventing *sysctl()* call in iOS 9: `https://developer.apple.com/videos/play/wwdc2015/703/`

[2] Jolla cannot provide compatibility: `https://jolla.zendesk.com/hc/en-us/articles/201440787`

Context Factor	**Mean**	**Std**	**Median**
CPU use	75%	33%	91%
Distance traveled	680.5 m	53.23 km	0 m
Distance (> 0)	867.06 m	2.66 km	5.85 m
Battery voltage (V)	3.78	0.61	3.84
Screen brightness	61.82	87.96	-1
Screen brightness (0-255)	128.03	85.71	109
Temperature (°C)	29.27	5.75	30
Wi-Fi signal strength (dBm)	-61.29	13.02	-61

Table 4.1: Summary statistics of selected context factors. Previously published [46, 47].

some of them are more easily available. In this work, we are interested in features, later called also *context factors*, which does not require heavy permission policies or come with standard permission routines. Together, these factors define the *system state* of the device.

4.1 Nominal and Ordinal Attributes

Context factors consist of both nominal and ordinal attributes. For nominal variables we use the different possible values as the categories, such as network type that indicates information of Wi-Fi or mobile, and applications come with their process names along other information, such as the human-readable name and information whether they are running background or foreground. Most of the context factors, such as screen brightness, battery temperature, and CPU use, are ordinal-valued. Managing different data types at the same time requires preprocessing, for example, discretization of the ordinal-valued factors.

Another challenge is set by default and missing values, that may seem obscure, for example, large negative values when considered missing battery temperature or screen brightness provided out of normal setting range. Some context factors come with possible calculation mistakes, for example, distance traveled between two samples may seem to be thousand of kilometers because of missing or default value in the location information of another sample.

For nominal variables we use all the different values as categories. To simplify the comparison of the context factors, we discretize ordinal-valued into categories using an equal frequencies procedure, in other words, each factor is divided into categories containing approximately the same number of values. The number of categories is determined empirically and based on observations reported in previous studies related to the field. Summary statistics of selected context factors are given in Table 4.1 and the different categories are next discussed together with descriptions of each factor.

To be specific, we only discuss attributes given by the Android devices in this section. That is because in most of the cases our research only covers the Android devices due to the crucial differences compared to the iOS platform.

4.2 User-changeable System Settings

System settings are collected via the Android programming interface. Characteristically, they are visible to the user via system setting menus and the user has control over them. This also sets the main challenge for managing system settings: there are no proofs that users have adjusted them wisely. At the same time, system settings out of reasonable range can easily be considered as defaults, misreadings, or errors, because users can only control them inside the allowed ranges.

The systems settings considered in this work are the following:

Network type is a categorical attribute describing the current method used by the phone for Internet connectivity, for example, none, Wi-Fi, mobile, or WiMAX, depending on the technology used. When the network type equals mobile, detailed information about the connectivity type is given by the attribute mobile network type. The user can modify the setting by choosing the preferred networking strategy, such as allowing mobile data connection or connecting to the preferred Wi-Fi available. Some general settings, such as flight mode, affect the network type by suppressing all the network connectivity.

Mobile data status describes the current status of the mobile data interface. It is given as a categorical attribute and has one of the following values: connected, disconnected, connecting, or disconnecting. The user can modify the status allowing or disallowing the mobile data connection.

Mobile network type is a categorical attribute that specifies the mobile data transfer standard currently being used on the phone. Examples of values it can take include LTE, HSPA, GPRS, EDGE, and UMTS. The list of possible mobile networks is broad and depends on the technologies available in each country and by each operator. The user has the best control over the mobile network type setting when choosing a data plan, which are widely marketed under the generalized names of 3G, 4G, and so on. Some devices also allow the user to choose the preferred technology later on, for example, in the case of lacking network coverages in rural areas.

Roaming describes whether mobile network traffic outside of the own operator network is allowed. The value of the attribute is either disabled or enabled, given as a binary attribute (0 or 1). The setting is possible to modify in the networking settings, but also the network operator may disallow it or it can be disabled in the customer data plan.

Screen brightness refers either to a manually adjusted brightness value, given as a numeric value between 0 and 255 where a larger value implies brighter screen, or automatic setting, given by value -1. Some devices provide 256 as the largest value (full screen brightness), but all the other values out of range [-1, 256] can safely be disregarded. The automatic setting can vary in some devices, for example, based on the sensing of the outside light. Therefore, knowing the setting parameter may not give the actual brightness value that is currently used.

Based on the Carat data, when screen brightness is manually controlled, the mean is around 128, or the exact midpoint. The distribution of the values, shown in 4.1, indicates that almost the entire range of screen brightness values is used, making it difficult to categorize screen brightness values in a meaningful way. While small brightness values generally have lower energy impact than higher values, or even automatic settings, they usually occur only in specific situations, such as at night or while reading a book in a dimly lit room. As these values are encountered very infrequently, their overall energy benefits are small compared to using automatic setting. Based on these observations, we opt for a binary split into manual and automatic brightness especially when studying energy consumption. For some other application areas, a different kind of split might be more useful.

4.3 Subsystem Variables

Subsystem variables are not directly available as a user-modifiable system setting, but can give information about the state of the smartphone. For example, if we notice a decreased Wi-Fi link speed or signal strength, we can recommend that the user try to use the mobile network instead of Wi-Fi in this context. In the energy analysis, these factors provide important insights into what happens inside the smartphone.

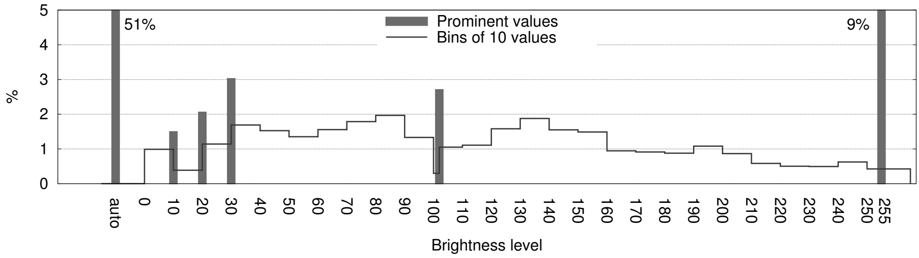


Figure 4.1: The frequency of all screen brightness settings. Previously published [47].

Misreadings, defaults, and missing data points set the most important challenge when managing and analyzing subsystem variables. At least, they should come from a reasonable range and match the given Android API description. Because manufacturers may set their own defaults, missing values or unsuccessful readings, for example, it is not straightforward to define "good" values.

The subsystem variables considered in this work are the following:

Battery health is a categorical attribute determined by the smart battery interface of the Android device. Values of the attribute are vendor-specific, with examples of common values being Good, Bad, Overheat, and Unknown failure. The value Good is the most common value in the Carat data. It is even so common that any other value might be considered an abnormal behavior.

Battery temperature is the temperature of the battery given in Celsius degrees. We only considered positive values, because in normal conditions the battery should not freeze under zero degrees. Also, some device manufacturers seem to provide high negative values when readings of the sensor are not available. With median 30 °C and mean 29.27 °C, presented in Table 4.1, we consider discretization over this value: over or under. Depending on the research objectives, a more sophisticated split might provide more information.

Battery voltage describes the current battery capacity in Volts. The safe operating voltage of a smartphone Lithium-Ion battery is in the range 3 - 4.2V. The nominal voltage of such batteries is typically 3.7V. The mean, the median, and the standard deviation presented in Table 4.1 reflect this very closely. We consider three categories for voltage: Low (0 - 3V), Medium

(3 - 4.2V), High (4.2V+). Values lower than 0 and greater than 5 were considered defaults or unsuccessful measurements.

CPU usage is a percentage (0-100%) that describes the fraction of the CPU currently used. We consider measurements that reflect the percentage of time the CPU is active. The CPU usage should be given by a value between 0 and 1, and all the other values were considered defaults or missing values. The mean and median in Table 4.1 indicate that CPUs are mostly active. We split the CPU use around the mean, resulting in three categories: Low (0 - 42%), Medium (43 - 85%), and High (86 - 100%).

Distance traveled is a location-based measurement between two samples, given in meters. For privacy reasons, the Carat application does not gather the exact location of the user, but uses distance measurements to determine whether the device has been moving or not, for example, in a car. Most of the values are during stationary periods or with little movement, especially when taking into account around 100 meters standard error of coarse localization services. Based on this observation and large statistical standard deviations, we consider a split between stationary (less than 100 meters, later referred to just as 0 meters) and non-stationary behavior (100 meters or more, later referred to as greater than 0).

Mobile data activity describes how the mobile data interface is being used. The value of this categorical attribute is one of the following: none, out, in, or inout. Mobile data activity has cross-effect on multiple settings that allow data connectivity in general, but in contrast to them, this factor describes the actual occurrence. For example, the user might have allowed the mobile data connection, but for some reason it may not be available.

Wi-Fi link speed is given in Mbps and is determined by the Android API. The attribute does not provide the actual speed used, but the capability of the closest cell tower. Thus, it better describes the maximum capacity available than real usage, and we do not consider it much in our work. The Wi-Fi link speed attribute might be useful if it is known that two or more devices are connected to the same cell tower and share the connection and thus the bandwidth, too.

Wi-Fi signal strength is given in dBm and is determined by the Android API. We only consider RSS values in the range [-100, 0] due to the technical limitations. Good Wi-Fi signal strength values are normally between -30 and

-10dBm, and the worst, while still being connected, is -95dBm. We consider four categories: Bad (-100 to -75dBm), Average (-74 to -61dBm), Good (-60 to -49dBm) and Excellent (-49 to 0dBm). The mean RSS is between the Average and the Good levels, and the Excellent and the Bad levels are within one standard deviation. These values are in line with typical values used in Wi-Fi positioning literature, and they were also restrictively tested to match four "bars" in the user interface.

4.4 Energy Measurements

Energy impact of system settings and subsystem variables is an important new research field. To measure the energy consumption of the device, we consider timestamps and battery levels reported by Carat and develop *energy rates*. These reflect normalized energy consumption per time unit, more formally defined as:

$$\text{Energy rate} = \Delta\text{battery level}/\Delta\text{time} \tag{4.1}$$

The methodology used to derive rates and the validity of using energy rates as a measurement for battery consumption has been validated and presented in previous work by Oliner et al. [6].

The energy rate distribution coarsely follows power distance: fewer rates of high energy consumption, in other words, only hours of total battery life, and most of them indicating discharge level considerable normal. We compare discharge rates routinely as consumption per second, but they can also be interpreted to more human-readable format, as hours of battery life in the given system state, as follows:

$$h = \frac{\frac{100}{rate}}{3600} \tag{4.2}$$

The difference between two different system states can thus be denoted as *battery life gain*. It measures how changes in context factors influence the lifetime of a device on average. We usually give the battery life gain as percentages compared to the average, but also actual hours of battery lifetime left in the given combination of context factors might be considered.

As an example, Table 4.2 presents battery life gains of selected subsystem variables and system settings we have studied in our work [46]. High CPU use obtains the highest energy loss. The benefit of maintaining a balanced CPU load is significant, as medium CPU use produces +5.72% energy benefit compared to the average use. For screen brightness, the automatic setting seems to improve battery life significantly, providing even +6.29%

Context Factor	Value	BL Gain
CPU use	Low (0–42%)	+3.24%
CPU use	Medium (43–85%)	+5.72%
CPU use	High (86–100%)	-2.48%
Distance traveled	None	-0.76%
Distance traveled	>0	+8.20%
Battery voltage	Low (0–3V)	-16.60%
Battery voltage	Medium (3–4.2V)	-0.76%
Battery voltage	High (4.2V+)	+69.08%
Screen brightness	Manual	-4.96%
Screen brightness	Automatic	+6.29%
Wi-Fi signal strength	Bad (-100 – -75 dBm)	-2.29%
Wi-Fi signal strength	Average (-74 – -61 dBm)	+4.00%
Wi-Fi signal strength	Good (-60 – -49 dBm)	+6.29%
Wi-Fi signal strength	Excellent(-48 – 0 dBm)	+7.63%

Table 4.2: Battery life gains of selected context factors. Previously published [46, 47].

better battery life compared to the average. Manual brightness, in contrast, shows a major loss of battery life (-4.97%). Also, Wi-Fi signal strength has a dominant effect on the energy consumption. When the Wi-Fi signal strength is considered Bad, there is -2.29% power loss compared to the average. Moving to the area of at least the average signal strength helps to gain more battery life.

4.5 Detecting Country

All the useful information is not possible to read directly from the Android API, but is derived from other collected factors. To protect user privacy, the Carat system does not gather any location information. Instead, Carat collects different attributes about the network usage, especially Mobile Country Code (MCC) as well as the current timezone. In our work [92], we propose a method to detect the country of the user without exact location information, but only using the MCC and timezone attributes.

A mobile country code (MCC) is a three-digit value tied to a mobile network. Each MCC corresponds to a single two-letter IANA country code[3]. Unfortunately, the MCC is not available on Wi-Fi-only devices, such as

[3] `http://www.iana.org/time-zones`

tablets, and some CDMA networks. From the beginning of March 2016 until May 2017, the Carat dataset has 5.65 million samples with valid MCCs.

There are 69.7 million samples with the timezone information available in the Carat dataset. The Android devices follow the IANA timezone database format and give the timezones presented as the continent and the closest big city, for example, America/New_York or Europe/London. These values can be further translated to the two-letter country codes (later referred to as CC) similar to the MCC codes.

Both mobile country codes and timezone-based country codes can sometimes have errors or they can be misconfigured. We compare the MCC and CC codes, and find that out of 5.83 million samples with valid MCC and CC values, these two indicate the same country in 97% of the samples. In those 3% of samples where MCC and CC indicate a different country, there are distinct neighbor countries such as small European states, and nearby countries in the same timezone. Difference in MCC and CC may be caused by cross-border usage of the network infrastructures in neighboring countries. Because some devices allow the user to adjust the timezone, there is a possibility of misleading selections. For example, both Europe/Athens and Europe/Helsinki represent the GMT+2 timezone but Athens is shown first in the alphabetical list, so it might be chosen also by users outside Greece for convenience.

Together with the automatic data collection in Carat, there are several volunteering questionnaires run, as described in Section 3.3. 1153 users have shared us their GPS location coordinates (latitude and longitude). We compare these locations to the user's sample history, take the MCC codes from all the user's samples, and find that, for 97% of the users, the coordinates match the most common MCC among all the samples. This means these people have been inside a single country for most of the time, and help to trust the MCC and CC analysis as a country information source.

For large-scale comparison of application usage in different countries, we consider a subset of 5.65 million samples in which the timezone-based CC and MCC fields match. The MCC is obtained from the cellular network infrastructure, and automatically converted to a two-character country code. We compare MCC with the country that the city of the timezone field corresponds to. This procedure increases the reliability of detecting the country of the user, when the exact GPS or Wi-Fi-based location is not available for privacy reasons. The subset contains 25,323 users associated with 114 country codes, out of which 44 countries have a significant number of users (100 or more). The majority are based in the USA, with strong user bases also in Finland, India, Germany, and the United Kingdom.

4.6 Applications

Running and installed applications can be seen as a fundamental factor of smartphone usage. Different applications provide functionalities for well-being, education, and leisure. There are plenty of applications available on the app markets, for example, 2.2 million applications in the Google Play store and 2 million in AppStore[4]. Different features can impact the choice of the application, for example, search results, user recommendations, and application description, and in the longer run, energy usage, performance, and user experience. Without applications and their wide functionalities, even smartphones would remain as regular phones. Interesting research questions include application usage comparison, for example, between individual users or user groups, countries, cultural areas, and so on.

The Carat application collects the following information from the running and installed applications in the device: A package name is the real name of the application, for example, `com.facebook.katana` is the package name for the main Facebook application. Also this human-readable application name will be collected, together with information if the app is a system app or update to a system app, it's human-readable version code, signatures of the app from `PackageInfo.signatures`, and package that installed the application, for example, in case of service and library applications. Importance is an attribute to describe whether the application is running foreground or background, or if it's status is something else provided by Android API, such as visible, service, or empty.

Application information collected by Carat has previously been used to study their energy consumption [6, 89] and malware prediction [11]. In this work, we present two example cases of application usage analysis: first, we analyze application usage trends in Section 6.2 that is based on Manuscript I attached to this thesis [61], and second, we present demographic, geographic, and cultural boundaries in mobile application usage in Section 6.3 that is based on Manuscript II attached to this thesis [92].

There are several challenges of analyzing application usage. First, any application seems similar to the system, and separating system applications from applications actually installed and used by the user is not always straightforward. In some cases there is the system application status provided by an importance attribute, but it is not mandatory and sometimes the difference between system app and another functionality app may be difficult to define. For example, many manufacturers and network operators

[4]Numbers of existing applications are estimates: `http://www.statista.com/statistics/276623/number-of-apps-available-in-leading-app-stores/`

provide their own, preinstalled applications for messaging, file management, picture processing, calendars, and emails. Second, there is no knowledge of the application's functionality provided in the data. The name of the application, and categories and descriptions given by the app market, may give good guesses for which purpose the app is meant, but this information first has to be gathered and processed. Thus, we next to consider this functionality or labeling problem in more detail.

4.7 Application Categories

Often it is more useful to study what applications do or are used for than the usage of single applications alone. For example, there are several messaging applications with basically the same functionalities but different popularity and language bases around the world. To avoid language and marketing biases, we can consider applications through the categories they belong to, such as communication, social apps, and different game genres.

There are two ways to label applications: by hand, which is clearly a very ineffective method, or by using categories already provided by the application markets. To obtain this categorization, we fetch the application descriptions of all the applications existing in the Carat dataset as HTML files from the Google Play store. Then we map the application names to the corresponding categories. This way each user's category usage can be detected. In October 2016, there were 55 categories on Google Play. The Carat dataset contains 97,000 different applications including system processes, out of which 54,776 applications are available in Google Play with at least one category assigned. Some apps can have multiple categories, such as family oriented action games may belong to categories *Family pretend* and *Action games*. To avoid inconvenience, these apps were considered once in every category they belong in.

There are some challenges regarding using Google Play as a source for category detection. Google Play is not available in China, so Chinese users cannot be studied. In addition, not all the applications are available in Google Play at all. For the same reason, the iPhone devices have to be excluded, even if there is plenty of iOS data in the Carat dataset. It is not possible to fetch application descriptions or categories from App Store in the same way as we can do with Google Play.

The number of categories of Google Play has increased over the years, but some application categories may still be too broad. For example, the *Tools* category contains a lot of general applications, for example, keyboards for different alphabets, flashlight apps, and other utilities such

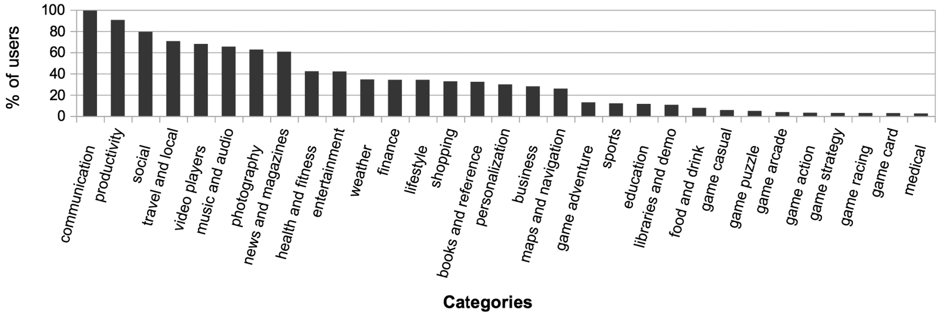


Figure 4.2: Distribution of users (%) between the top 31 (out of 55) Google Play categories. Previously published [92].

as Carat, with very different use cases. Similarly, almost every user uses the *Communication* category, since it contains common messaging apps, some of which are preinstalled on most smartphones, for example, Google's Hangouts, Facebook Messenger, and WhatsApp. When looking at the most used categories, leading categories are these two large ones: *Tools* (100% of users used) and *Communications* (99% used). Figure 4.2 shows the next popular categories, including, for example, other wide topics such as *Productivity*, *Social*, and *Travel and Local*.

Chapter 5

Methodology for Analyzing Crowdsensed Data

Data analysis methodology targets finding novel insights to the crowdsensed data. When successful, these methods are capable of constructing models that can capture complex interdependencies between the context factors. Challenges on analyzing crowdsensed data are highly related to the way data have been collected: there is often a need for cleaning and preprocessing, as discussed in Section 4.

On the other hand, mobile crowdsensing systems are capable to produce large amounts of multidimensional data in comparably short time, depending on how many attributes have been collected and the sampling period. This causes a need to perform effective machine learning procedures in a distributed environment. This Section focuses on methodological approaches to analyze large-scale mobile crowdsourced data in a suitable environment.

Several key statistical methods and algorithms essential to our work will be introduced. We introduce information metrics from statistical tools to measure the association between attributes, and methodologies to understand popularity, trend, and application usage. The output of this analysis procedure can be later utilized when making decisions or building recommendations systems, as will be discussed later in Section 6.

5.1 Information Metrics

Information metrics are used to measure the strength of statistical association between different context factors, such as system settings, subsystem variables, application usage, or any other nominal or discretized ordinal context factor available. Results of the information metrics can be used to rank features and provide insights to understand the effect of a given context factor or a set of factors to measurable value, such as, energy consumption, popularity, or usage.

Mutual information. To measure statistical association, we consider the *mutual information* (MI) between two context factors. For assessing the influence of a single context factor X to the target factor Z, the MI is formally defined as:

$$MI(X, Z) = \sum_{z \in Z} \sum_{x \in X} p(x, z) \cdot log\left(\frac{p(x, z)}{p(x) \cdot p(z)}\right). \tag{5.1}$$

Conditional Mutual Information. For higher order combinations containing two or more context factors (denoted as X and Y in case of two factors) and the target factor (denoted as Z), the *conditional mutual information* (CMI) is formally defined as follows:

$$CMI(X, Y|Z) = \sum_{z \in Z} \sum_{y \in Y} \sum_{x \in X} p(x, y, z) \cdot \log\left(\frac{p(z) \cdot p(x, y, z)}{p(x, z) \cdot p(y, z)}\right) \tag{5.2}$$

By using CMI to analyze the impact of context factors combinations, we can identify combinations that are as informative as possible while at the same time minimizing redundancy between the different factors. Accordingly, this usage style of information metrics can be understood analogously to the use of (conditional) mutual information for the feature selection techniques in machine learning research [93].

5.1.1 Energy Impact of System Settings and Subsystems

Next, we demonstrate these information metrics by examining how context factors (system settings and subsystem variables in this case) affect the energy consumption of the mobile devices. The work has been previously published in two Publications attached to this thesis [46, 47]. We derive a ranking for different factors based on their mutual information values, presented in Table 5.1. The results of the MI analysis are well in line with previous research [10, 40]. In particular, the major individual impact of CPU use and traveled distance on battery consumption is clearly observable. The results also contain some exceptions to the findings in the previous studies. The most prominent example is screen brightness, which is commonly considered as the most battery-heavy feature. In our analysis, screen

Context Factor	**MI Estimate**
CPU use	1.330
Distance traveled	1.069
Battery temperature	0.143
Battery voltage	0.099
Screen brightness	0.030
Mobile network type	0.019
Network type	0.018
Wi-Fi signal strength	0.014
Wi-Fi link speed	0.014
Mobile data status	0.013
Mobile data activity	0.005
Battery health	0.004
Roaming	0.0002

Table 5.1: Context factors' impact on energy consumption, ordered by the mutual information estimate. Previously published [46].

brightness results in a lower score than many other attributes. This may be due to the fact that screen brightness often happens to be high for some reason, for example, the device has been used to play a game with heavy graphics assistance.

Similarly, we derive an energy effect for the context factor pairs by considering the conditional mutual information. The results are listed in Table 5.2. Compared to the results of individual context factors, the combination of two factors gives more accurate explanations of the battery consumption. CPU use gains significantly higher impact when combined with another factor than when considered alone. Also factors related to network connection, such as Wi-Fi signal strength and network type, are more prominent when considered in conjunction with another context factor. Capturing this kind of nuances in consumption is particularly beneficial when giving suggestions to the end user on how to improve battery life. As an example, we can observe that changing another system setting can help to improve battery life in cases where high CPU use is mandated, such as, when playing a game.

The top context factors according to energy consumption are battery voltage, CPU use, battery temperature, and movement (distance traveled) of the device, or combinations of these context factors. The effects of these factors are mediated by other factors, which in turn can cause significant increases or decreases in the energy consumption.

Context Factors		**CMI**
Battery voltage	CPU use	4.29
CPU use	Screen brightness	2.17
Battery temperature	CPU use	2.07
CPU use	Distance traveled	1.81
CPU use	Wi-Fi signal strength	1.69
Battery voltage	Distance traveled	1.53
Battery temperature	Distance traveled	1.28
Distance traveled	Screen brightness	1.26
CPU use	Wi-Fi link speed	1.12
Battery voltage	Screen brightness	1.08
Wi-Fi link speed	Wi-Fi signal strength	0.99
Mobile data status	Network type	0.95
Network type	Wi-Fi signal strength	0.85
CPU use	Mobile network type	0.80
Battery temperature	Screen brightness	0.79
Distance traveled	Wi-Fi signal strength	0.75
Network type	Wi-Fi link speed	0.64
Mobile data status	Wi-Fi signal strength	0.60
Battery temperature	Battery voltage	0.56
Distance traveled	Wi-Fi link speed	0.54
Battery voltage	Wi-Fi signal strength	0.53
Mobile data status	Wi-Fi link speed	0.46
CPU usa	Network type	0.42
Distance traveled	Mobile network type	0.37
CPU use	Mobile data status	0.32
Battery voltage	Wi-Fi link speed	0.27
CPU use	Mobile data activity	0.27
Screen brightness	Wi-Fi signal strength	0.26
Distance traveled	Network type	0.20

Table 5.2: Top of the conditional mutual information estimates for pairs of context factors for energy consumption rates. Previously partially published [46].

5.2 Trend Mining

Application popularity and trend analysis is an important part of understanding how smartphones are used. Some studies on mobile application usage have characterized factors that drive download decisions [67, 94, 95] without being able to determine what happens once the app has been installed. Some previous works have focused on overall usage and how that is influenced by contextual factors [53, 96, 97]. Some analytics companies use retention rates[1] to describe successfulness of the apps. To the best of our understanding, our study in attached Manuscript I [61] is the first to independently analyze, what happens once the application has been installed.

Retention Rates Retention rate on day d is defined as the percentage of users that continue using the application d days after the first usage. To estimate retention rates, we identify for each user and application the first and last time the user launched the application.

Retention rates of the first week are presented in Figure 5.1. First day retention rates for applications with at least 10 users are close to 50%, compared to 80% reported by many analytics companies[2]. For applications with at least 1000 users the retention rate rises to 62%. For the most popular 100 applications, the first day retention rate is even as high as 68% and after 7 days the retention rate remains higher than 50%. This analysis shows that the retention rates largely depend on the initial number of users, and popular apps stay healthy in terms of user base for longer periods of time.

Trend Analysis. While retention rate reflects the long-term attractiveness of an application to individual users, it does not cover instantaneous popularity, usage trends, or seasonal patterns. Figure 5.2 presents usage patterns of some automatically selected (based on the peak detection algorithm) applications from first day of usage up until 100 days of usage. The example indicates that application usage patterns do not always follow a simple falloff pattern suggested by retention rates. Several rising trends and again falling trends can be seen in Figure 5.2. For some and possibly various reasons, these apps have became substantive again.

[1] `http://info.localytics.com/blog/the-8-mobile-app-metrics-that-matter`

[2] `http://andrewchen.co/new-data-shows-why-losing-80-of-your-mobile-users-\ is-normal-and-that-the-best-apps-do-much-better/`

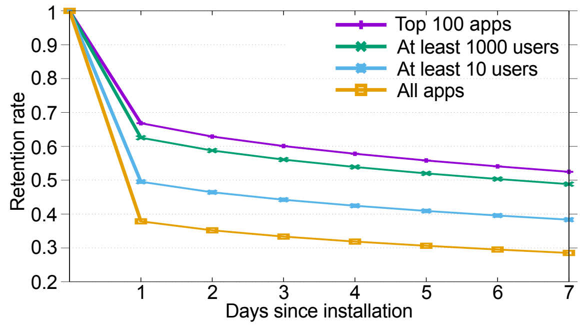


Figure 5.1: Retention rates of the first week. Previously published [61].

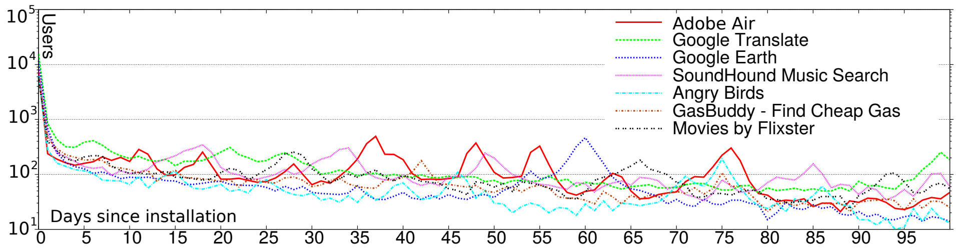
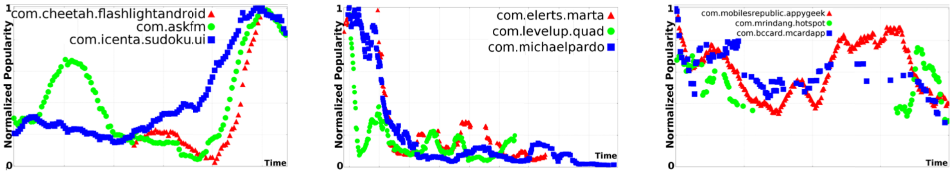


Figure 5.2: App usage patterns up to 100 days. Previously published [61].

Based on these observations, we develop in [61] a methodology to determine the application life cycle. We characterize application trend patterns to the following classes: *Flop*, *Hot*, *Dominant*, or *Marginal* apps. Figure 5.3 presents example cases of the Flop, Hot, and Dominant applications. When applying this trend analysis to the Carat dataset, we find that 40% are *Marginal* apps with a very limited user base in general. In the remaining 60%, the following patterns can be found: 0.4% are *Dominant* or gaining constantly high popularity, 1% are *Flops* or falling in continuously popularity, and 7% are *Hot* or continuously rising in popularity. These findings can be utilized in application recommendation systems, as later presented in Section 6.2.

5.3 Analyzing Similarity of Usage

Analyzing application usage data from a large population of people can provide important insights in how applications and smartphones are used in general in the wild. Because smartphones are largely considered to be important daily life devices, this kind of data analysis opens the doors to the routines, habits, and rituals people practice in their daily life.



(a) Hot apps. (b) Flop apps (c) Dominant apps

Figure 5.3: Applications trend patterns with example apps. Previously published [61].

The Carat dataset contains around 55, 000 applications. To get a generally representative picture of their functionality, we map them to the Google Play categories (currently 55 possible) as presented in Section 4.7, but the same metric may also be used with an application-based approach.

Usage Vectors. We generate binary category vectors for each user considering whether that user has used a category or not (1 for usage, 0 for none). For each country (or any similar group of users), we construct the probability distribution of category usage within the country, represented as the fraction of users in the country having used that category.

Formally, for each category $c_i \in C, C = c_1, c_2, ..., c_k$ where k is a number of categories, we define the probability of its use within a country n as

$$c_{i,n} = \frac{\sum_j u_{i,j} \in U_n}{|U_n|} \tag{5.3}$$

where U_n is the set of users in country n and $u_{i,j}$ is 1 if user j used category i and 0 otherwise.

Now $C_n = c_{1,n}, c_{2,n}, ..., c_{k,n}$, is the category use probability vector for country n.

5.3.1 Demographic Usage Differences

To understand demographic differences in the application usage, we need to benefit from the definition of the category usage vectors above and the mutual information metric described in Section 5.1. We consider different demographic attributes collected from the Carat user by the questionnaire described in Section 3.3: age group, gender, current occupation, highest completed education, household situation, and the following economic factors: amount of debt, amount of savings, and level of monthly salary. In addition to these, we consider country as a similar factor affecting smartphone usage.

Attribute	Mutual Information Gain
Country	4.60
Occupation	2.78
Education	2.14
Savings	2.12
Debt	1.99
Salary	1.96
Age	1.94
Household	1.57
Gender	0.59

Table 5.3: Demographic attributes sorted by information gain against application usage. Previously published [92].

To use categorical application usage as a target factor for the mutual information metric, we consider the usage vector as a single data element. Thus, the mutual information can be considered between each demographic attribute and the application usage vectors. Table 5.3 describes the results of the analysis, sorted by the information gain. We can see that the country attribute is characterized by the highest information gain compared to the other attributes, such as gender and age, whose information is significantly lower in comparison. The high information gain for country strongly motivates to detect in more detail how countries differ in terms of mobile use.

Similar analysis can be performed conversely between individual application categories and the demographic attribute as a target factor. Table 5.4 shows the results of this analysis. For country, the category of the *Weather* applications gives the best information gain, probably because weather is more predictable in some countries than in others. Occupation is related, for example, to the *Business* and *Finance* applications, that is probably caused by academic and professional workers benefiting from the mobile techniques in their work. The household attribute that indicates whether the person is living alone, with other adults, or with kids, is best described by categories of family-related applications, such as *Family music videos*, *Parenting*, and *Dating*, the last probably for those living alone.

5.3.2 Geographic Usage Differences

Understanding how different attributes affect application usage provides us with interesting insights, but is needed also to compare and cluster users and user groups together. Comparison of application usage between users,

Attribute	**App categories with highest information gain**
Country	Weather, Game action, Finance, Family pretend
Occupation	Business, Game adventure, Finance, Family pretend
Education	Finance, Game adventure, Shopping, Music and audio
Savings	Game adventure, Game simulation, Entertainment, Personalization
Debt	Finance, Books and references, Game simulation, Family music video
Salary	Game adventure, Business, Game casual, Game simulation
Age	Game adventure, Weather, Business, Family music video
Household	Family music video, Dating, Family action, Parenting
Gender	Business, Game casual, Personalization, Books and reference

Table 5.4: Application categories that gain the highest information against each demographic attribute. Previously published [92].

or a set of users, requires a suitable similarity metric. We use there the Kullback-Leibler divergence (KL), which is a relative entropy metric used to detect how a probability distribution diverges from another.

Kullback-Leibler Divergence. To compare the usage vectors with each other, we use the Kullback-Leibler divergence (KL). For two probability vectors it is formally defined as

$$KL(C_n||C_m) = \sum_{i=1}^{k} C_n(i) \log \left(\frac{C_n(i)}{C_m(i)} \right) \tag{5.4}$$

However, since the KL divergence is not symmetric and it does not satisfy the triangle inequality, in our analysis we use the logarithmic sum of two-way KL divergences as a distance metric, so that the distance between two user countries is given by

$$dist(C_n, C_m) = \log \left(KL(C_n||C_m) + KL(C_m||C_n) \right) \tag{5.5}$$

As an example case, we present how the KL divergence is used to compare usage in different countries. The work is presented in the attached Manuscript II [92]. Figure 5.4 gives a dendrogram presentation based on the KL divergence between 44 countries well represented in the Carat

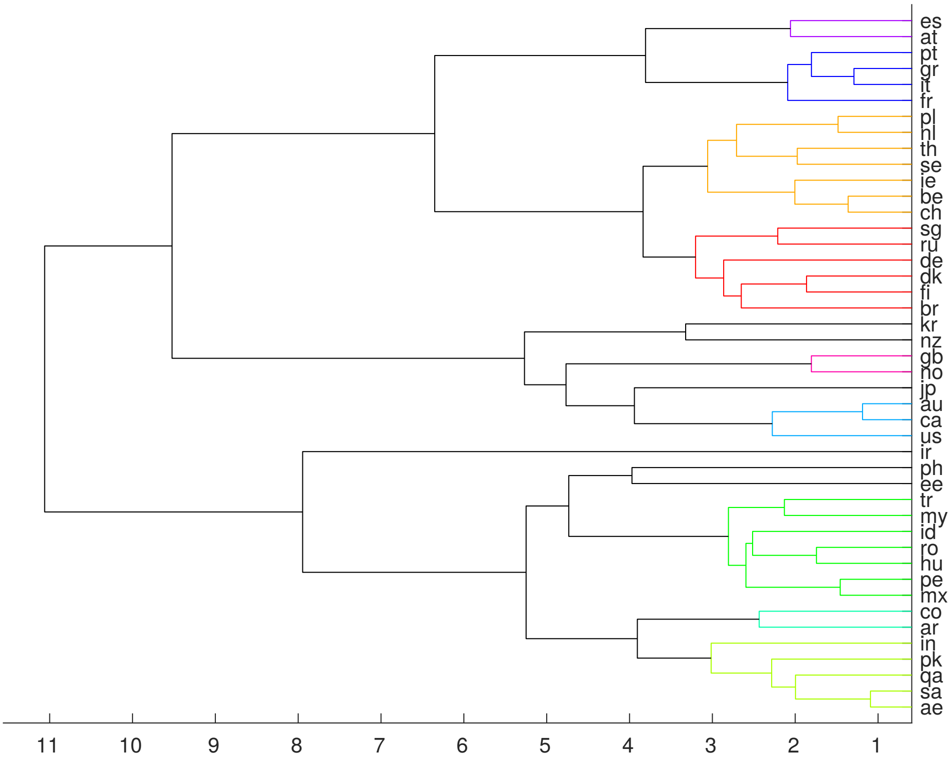


Figure 5.4: A dendrogram visualization of the Kullback-Leibler divergence between countries. Previously published [92].

data. There, we can see three main branches in the dendrogram. The topmost group contains mostly European countries. Its subgroups roughly correspond to southern (the top five countries), central (the next seven), and eastern Europe (the last six in the group). Brazil (br) may be included due to the effect of language or history.

The middle branch in the dendrogram or the next group contains English-speaking countries such as the USA, Australia, Canada, New Zealand, the United Kingdom, and other countries with early adopters of the Carat app, such as South Korea (kr) and Japan (jp). Norway (no) may be included because of its location near the United Kingdom. The latter three countries may also be included because the Carat user questionnaires have been only presented in English, so those familiar with English applications may have answered the questionnaire more readily than others.

The third main branch or group consists of the rest of the countries, with some meaningful demographical or geographical groups, such as Columbia (co) and Argentina (ar) in South America, and the Arab Emirates (ae),

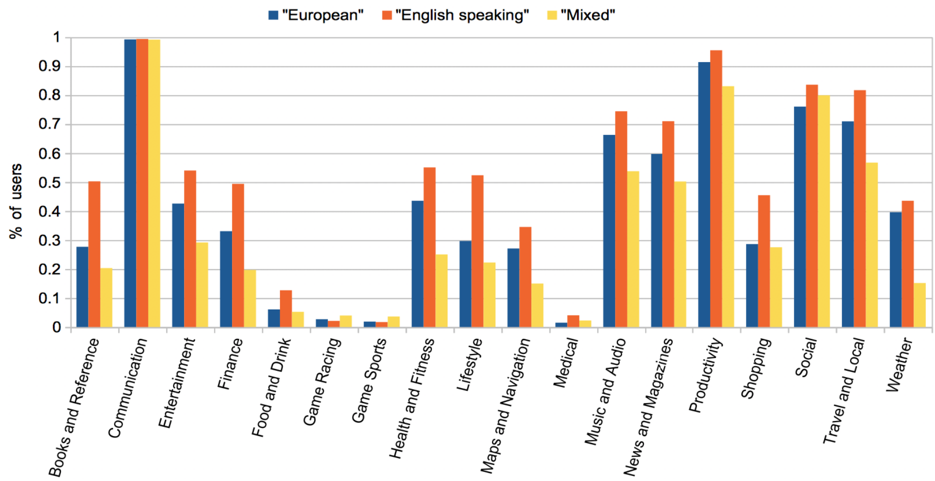


Figure 5.5: Usage of three main country clusters in several statistically significant categories. Previously published [92].

Saudi Arabia (sa), Qatar (qa), Pakistan (pk) and India (in) in Asia. Iran (ir), the Philippines (ph) and Estonia (ee) were not grouped close to other countries.

These three main groups visible in Figure 5.4 follow certain geographical, cultural, and language boundaries. These differences between application usage are also visible when looking at the application category data in greater detail. In Figure 5.5, we compare application category usage in certain categories strongly correlated with different cultural value factors. In general, the "English speaking" group uses a wider set of applications, and it seems to be statistically significantly high in almost every application category.

The "Mixed" group is characterized by lower application usage across the board, but higher than the other groups in two categories: *Sports* and *Racing games*. Some categories, such as *Food and Drink*, *Medical*, and *Shopping* are almost equally popular in both of the groups "non-English European" and "Mixed", but surpassed by the "English-speaking" group. *Weather* apps are, on the other hand, popular in the groups "non-English European" and "English-speaking", but less used in the "Mixed" group. That may be because the weather clearly is more predictable in some areas than others.

Communication apps are very popular in all the groups. Although the "Mixed" group has low usage in most categories, it also has very high usage of the *Productivity* and *Social* applications. On the other hand, the "English-speaking" group has the highest usage in almost all categories. This may be due to the fact that almost all apps have an English version, and many services, retailers, restaurants, and public places in Europe and the USA have dedicated apps[3].

Demographic, cultural, geographical, and other differences in application usage may be utilized by marketing, social research, and many other areas. Later in Section 6.3 we analyze deeper these differences with different use cases, and compare application usage to the existing cultural factor model.

[3] McDonalds France is available in English: `https://play.google.com/store/apps/details?id=com.md.mcdonalds.gomcdo&hl=en`.
The city of Wien has a dedicated mobility app: `https://play.google.com/store/apps/details?id=at.wienerlinien.wienmobillab`.
Hyde Park club dedicated app: `https://play.google.com/store/apps/details?id=com.hydepark`.
The Finnish weather map in English: `https://play.google.com/store/apps/details?id=com.nordicweather_sadetutka`.
Sydney's Central Park has an app to aid sightseeing: `https://play.google.com/store/apps/details?id=com.beaconmaker.android.centralpark`.

Chapter 6

Decision Making and Actionable Recommendations

Understanding smartphone usage as a whole can provide novel insights to the user's needs for smartphone functionality. The most important target here is to provide the users with the best smartphone usage experience possible. Figure 6.1 provides an example of questions that need to be answered to improve smartphone utility. In terms of performance and energy-efficiency, there are choices to do for the network connectivity: mobile data or Wi-Fi connection? When can the user charge the device next time, and what is the overall condition of the battery? How many applications has the user installed, and how many of them are actually in use? What needs does the user have, and which applications meet these needs the best?

The user's location and occupation may affect the set of application functionalities in need: local transportation apps to support commuting, the best utility apps for a more effective working life, and maybe some task management apps to balance work and leisure? There might be a favorite game even if it uses a lot of battery life, but the user favors continuing to play it. The users cannot always choose applications by themselves, but are supposed to use those that are popular in the community they are living in, for example, for networking and social media. And when there are new applications available in the app market, how can the user know whether they are worth installing?

To understand these needs and answer these kinds of questions, we need to focus on analyzing the context of the user - including their daily routines, functionalities they require from their smartphones, and so on - and the context of the device including, for example, the device model and operation system version as well as memory and CPU loads. Supporting the smartphone user experience requires understanding the battery life

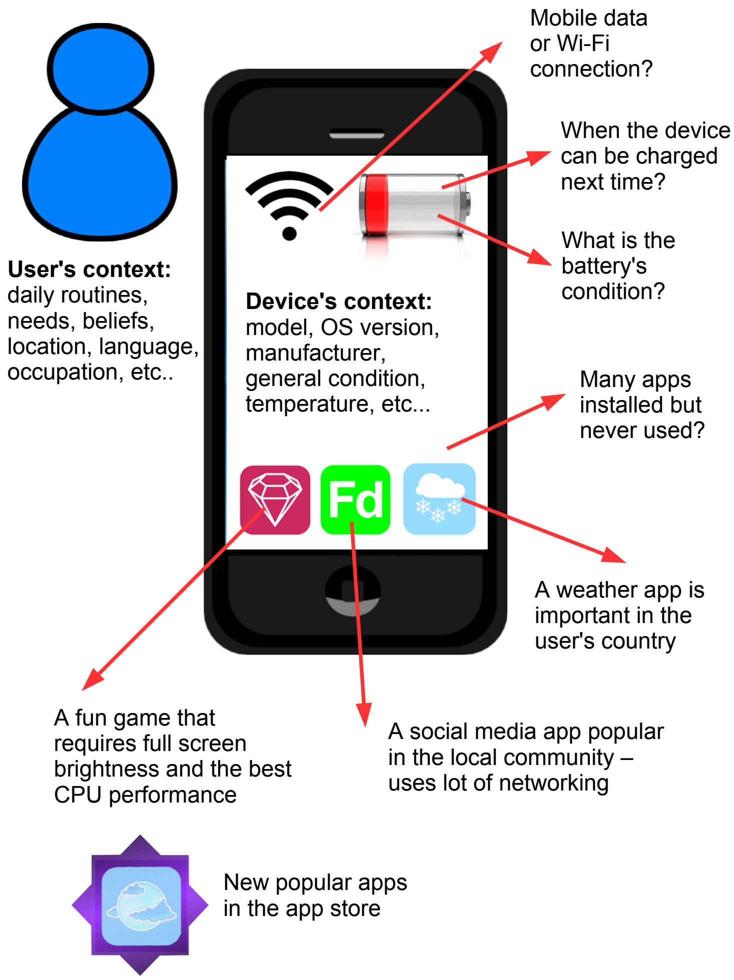


Figure 6.1: An example: Overall picture of the user's needs and smartphone usage as a whole.

limitations, user's needs, and in general the factors that have an influence on smartphones.

In this section, we present three example cases for making decisions and actionable recommendations based on the crowdsensed mobile data that has been collected in the Carat project. In every case, the methodology presented in this thesis for data cleaning and preprocessing as well as machine learning and statistical analysis of the results, are required. These examples aim to provide actionable feedback and new insights about the smartphone usage based on the crowdsensed data analytics. These example cases, with corresponding publications, can be listed as the following:

- **Energy recommendations for system settings and subsystem variables.** Many context factors of smartphones affect the device's energy consumption, but there are often complex interdependencies between them, which make it difficult to determine the optimal energy saving policy and right settings for a given situation. We present a system called Constella to provide actionable and human-readable energy recommendations for system settings and subsystem variables. The analysis work has previously been published in the attached Publication I [46] and the decision tree-based recommendation system in the attached Publication II [47].

- **Application trend analysis.** Understanding what happens after an application has been installed to the device, helps us to value the potentiality of applications. We present a novel app-usage behavior trend measure that provides instantaneous information about popularity of applications. Based on the application trends, traditional app recommendation systems can be evaluated and improved. The work is based on the attached Manuscript I [61].

- **Demographic, geographic, and cultural effects on mobile application usage.** Smart devices and functionalities they provide are nowadays an integral part of everyday life and part of modern life. Based on the user questionnaire performed for the Carat users, we study demographic, geographic, and cultural effects on smartphone usage. In addition to this, we also propose a method to use application usage data as a modern cultural factor. Understanding demographic factors, geographic differences, and cultural boundaries in application usage supports application developers, social researchers, and other people involved in the application ecosystem. The work is based on the attached Manuscript II [92].

6.1 Energy Modeling of System Settings

The processing and transmission power of smartphones continues to grow [98], while their battery technology remains largely unchanged [99]. Consequently, energy efficiency remains a high priority for current smartphone operating systems, and increasingly, for applications. The importance of energy efficiency has also been highlighted in several user studies, which have shown that users actively take measures to optimize the power consumption of their device [89, 100, 101]. Understandably, longer battery life provides better user experience and less struggle to find out the next charging possibility.

Mobile users are often forced to actively seek countermeasures to prolong the lifetime between successive recharges [33, 36]. Examples of these countermeasures include killing battery hungry applications or tasks, and manipulating context factors either through switching off specific sensors or adjusting individual system settings. Previous research has predominantly focused on the former task [6, 37, 38, 39]. In our previous work [46, 47], we demonstrate how the crowdsensed data-analysis approach can be used to obtain new insights into battery consumption. Especially, we provide a novel method to measure the energy effect of the combinations of different system settings and subsystem variables.

To demonstrate our methodology, we next provide an analysis of the energy effect of certain selected context factors. We have selected CPU use (Low, Medium, or High) and temperature (over or under 30°Celsius) from subsystem variables, and distance (motion or stationary) and screen brightness (automatic or manual) from system settings. Preprocessing of these context factors have been discussed in Sections 4.2 and 4.3. In all cases of the example, the network connection type has been a cellular data connection. Table 6.1 presents the estimated time in hours to drain the battery from 100% to 0%, while actively using a smartphone with the given context factor and value combination. With different values of CPU use, battery temperature, movement, and screen brightness, the battery life time ranges from 3.45 to 9.12 hours.

Table 6.1 demonstrates that the main deciding factor for battery life is the temperature: the lower the temperature, the longer the battery life. Traveling instead of staying still seems to increase battery life. This may be due to users driving and not using their mobile phones. After these factors, the CPU is the most dominant, and changing screen brightness brings the smallest, but still significant, battery life differences. These results show that while the CPU use alone is a good indicator of energy consumption, significant battery life gains can be obtained by considering more complex

Battery temp.	**Distance traveled**	**CPU use**	**Screen brightness**	**Battery life (h)**
Under 30°C	>0	Low	Automatic	8.83 – 9.12
Under 30°C	>0	Low	Manual	8.49 – 8.82
Under 30°C	>0	High	Automatic	8.09 – 8.24
Under 30°C	>0	Medium	Automatic	7.65 – 7.89
Under 30°C	>0	Medium	Manual	7.34 – 7.60
Under 30°C	>0	High	Manual	7.27 – 7.41
Under 30°C	None	Medium	Automatic	6.57 – 6.64
Under 30°C	None	Low	Automatic	6.28 – 6.35
Under 30°C	None	Medium	Manual	6.13 – 6.20
Under 30°C	None	Low	Manual	5.88 – 5.96
Under 30°C	None	High	Automatic	5.78 – 5.82
Over 30°C	>0	Low	Automatic	5.08 – 5.22
Under 30°C	None	High	Manual	5.00 – 5.04
Over 30°C	>0	Low	Manual	4.73 – 4.88
Over 30°C	>0	High	Automatic	4.62 – 4.69
Over 30°C	>0	Medium	Automatic	4.59 – 4.70
Over 30°C	>0	Medium	Manual	4.28 – 4.39
Over 30°C	None	Medium	Automatic	4.25 – 4.29
Over 30°C	>0	High	Manual	4.08 – 4.14
Over 30°C	None	Medium	Manual	4.06 – 4.09
Over 30°C	None	Low	Automatic	4.02 – 4.06
Over 30°C	None	High	Automatic	3.91 – 3.94
Over 30°C	None	Low	Manual	3.74 – 3.78
Over 30°C	None	High	Manual	3.45 – 3.46

Table 6.1: Estimated battery life in hours for selected combinations of four context factors. Previously published [46, 47].

context factor combinations. In addition to this, battery temperature and distance traveled can be used together to predict battery life very well.

The complex combinations of the context factors, such as those listed in Table 6.1, can be used to decide which factors to change to improve battery life, while keeping others constant. For example, while moving and playing a game, the CPU use is often high. If the phone can be kept relatively cool, 78% more battery life can be expected compared to warmer battery (increase from 4.08h to 7.27h). Further savings can be obtained by switching screen brightness from manual to the automatic setting.

Based on these observations, we deliver a recommendation system that can effectively summarize relationships between the context factor combinations and present transmission from a system state to another in a comparable easy manner. Constella [47] is a novel recommendation system for system settings and subsystem variables.

Constella relies on a decision tree-based recommendation model for capturing the energy impact of different context factors at once. Decision trees have been shown to provide a user-friendly and understandable representation for complex relationships [102], which is essential for improving users' trust in the recommendations. The decision tree model also provides a compact and compressible representation of relevant information which can be efficiently stored and used on a smartphone without a considerable impact on battery life.

The decision tree organizes context factor combinations into a logical structure and turns them into human-readable and actionable recommendations. The tree model can be learned efficiently on the cloud-computing back-end, and sent back to each client device. The clients can then generate recommendations independently by following paths of potential system state changes within the decision tree model. This makes it possible to generate energy recommendations also offline whenever the usage context changes: indoors and outdoors, with or without network connectivity, and so on. In the future, separate decision trees can be computed, for example, in the case of different applications or application combinations used.

Figure 6.2 presents an example of the Constella decision tree. Implementation details and more detailed examples are provided in Publication II attached to this thesis [47]. The example tree splits for three context factors: first the network type that splits the data into three parts. For each sub-branch, we can calculate an expected value of battery lifetime (EV). On the second level, there is a split by screen brightness after the network type "mobile" and by distance traveled after network type "Wi-Fi". If the device currently remains connected to the Internet via the mobile network with manual screen brightness, we can find at least two one-step changes to consider: if EV4 is better than EV5 (the current expected value), we can suggest switching to automatic screen brightness instead of the manual setting. If EV2 is also better than EV5, we can also show the recommendation for changing the network type. With more than one step, we can also go deeper in the tree, depending on the size and depth of the tree.

To remain clear these examples present only a limited number of the context factors. In reality, the trees will have more splits and options

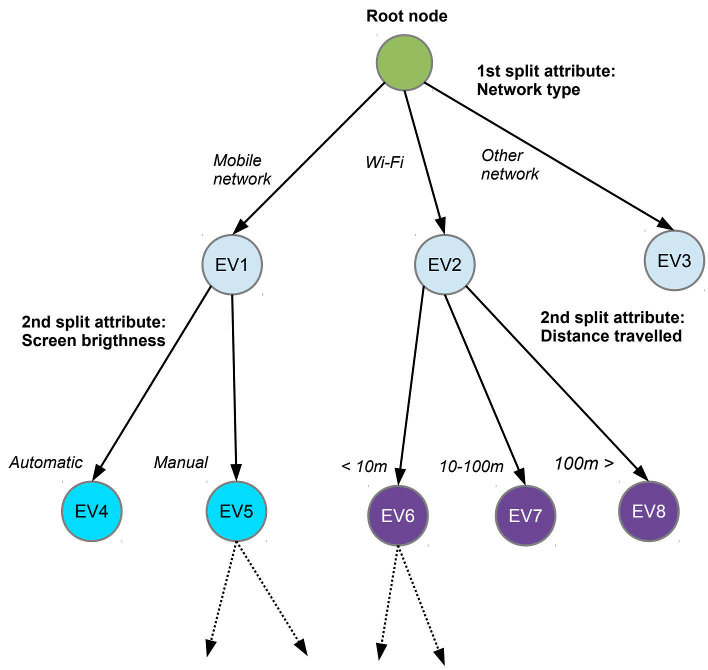


Figure 6.2: Example of the decision tree used for energy recommendations. EV = Expected value of battery lifetime in a given node. Previously published [47].

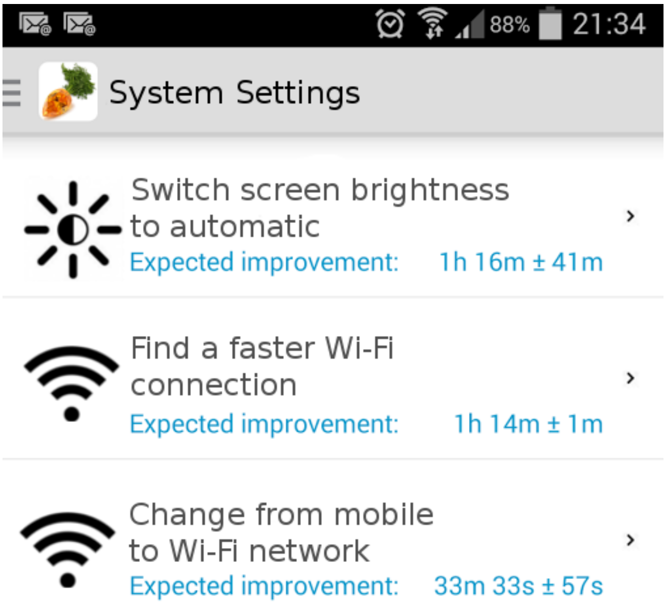


Figure 6.3: An example of the context factor energy recommendations. Previously published [47].

available. Figure 6.3 shows how the system setting and subsystem variable recommendations might be seen in the user interface. By adjusting the settings as suggested, users also participate in the continuous feedback loop with new and more informative data items to gather.

6.2 Application Trend Based Recommendations

Next, we focus on one of the fundamentals in the smartphone functionality: the applications. Choosing the right application for the right purpose is an important open research question. More generally, choosing an application that has any future in terms of upcoming new versions, security updates, and development support, is crucial. Our analysis of application trends focuses on app life cycle that characterizes usage behavior as discussed in Section 5.2. Next, we consider how trend information can affect application recommendations. The work is attached to this thesis as Manuscript I [61].

We implement the Slope One Prediction model that compares a user's profile to other users with similar usage history [103]. Slope One is considered to be a representative example of current state-of-the-art app recommenders, and also other popular systems based on a closely similar approach [55, 104].

The Slope One Prediction and especially its well-known implementation in the AppJoy system [103] operates on so-called usage scores, which are constructed by aggregating the following information: (i) time elapsed since the last interaction with an app, (ii) frequency of the user interactions with an app, and (iii) total duration of time the user has interacted with an app. Due to the infrequent sampling period of the Carat application, we focus on the amount of interactions, in other words, how often the application has been seen in the user's sample history.

Recommendation model. The Slope One Prediction model compares a user's profile to other users with similar application usage history. Formally, we define $S(u)$ as the set of applications used by user u. Given an application i and user u, we define $R_{u,j}$ as the set of relevant applications j used by other users together with i, in other words, $R_{u,j} = \{i | i \in S(u), j \notin S(u), \#S_{i,j} > 0\}$ where $S_{i,j}$ is the set of users who have used both i and j. The relevance of application j for user u is then given by:

$$P(u_j) = \frac{1}{size(R_{u,j})} \sum_{i \in R_{u,j}} (dev_{i,j} + u_i). \tag{6.1}$$

Here dev is the average of the usage scores between users who have used both i and j:

$$dev_{i,j} = \sum_{w \in S_{i,j}} \frac{v_{w \vdash j} - v_{w \vdash j}}{size(S_{j,i})}. \tag{6.2}$$

Given the relevance scores $P(u_j)$, the algorithm returns the top$-N$ items with the highest score as recommendations.

We run the Slope One-based recommendation system together with our trend filter analysis for a subset of the Carat data containing $4,500$ users and their $1,000$ most frequently used applications. We select October 2014 to be the test period due to little seasonal fluctuations, and as training data we selected all the Carat data accumulated between January 2014 and September 2014. Given the test data, we used the Slope One to generate recommendations in an incremental fashion for each of the four weeks in October 2014.

We counted (i) how many recommended applications can be classified as *Flop* or *Hot* apps by the trend filter analysis, and (ii) how they compare with the total number of the *Flop* and *Hot* apps. We also calculated

Week	Rec. Hots	Rec. Flops	Total Hots	Total Flops
1	8	5	219	163
2	7	6	229	158
3	8	7	232	154
4	10	9	225	150

Table 6.2: The *Hot* and *Flop* apps in the 20 best recommendations out of top 1000 applications during a month. Previously published [61].

the following evaluation metrics proposed in the previous literature [105]: temporal diversity, novelty, and accuracy. Diversity presents how the recommendations change over time, whereas novelty describes how many new recommendations there are seen compared to the later ones. The novelty of the recommendations relates closely to the trends, because changes in the application trends should affect new recommendations. Formally these metrics are defined as follows:

$$diversity(L_1, L_2, N) = \frac{|L_2 \setminus L_1|}{N} \tag{6.3}$$

$$novelty(L_1, N) = \frac{|L_1 \setminus A_t|}{N} \tag{6.4}$$

$$accuracy(L_1, A) = \frac{size(L1 \cap A)}{size(A)} \tag{6.5}$$

Table 6.2 presents an analysis of the top-20 recommendations given to all the users during the period of four weeks. In each row there is first the number of the week (from the beginning of October), and then in order the number of recommended *Hot* apps, the number of recommended *Flop* apps, the total number of *Hot* apps in the week, and the total number of *Flop* apps in the week. Table 6.3 gives statistics of diversity, novelty, and accuracy, first considering all the given recommendations and then without the *Flop* applications.

In Table 6.2 we can see that the number of *Hot* apps recommended for each week is small and comparable to the number of recommended *Flops*. Given that we have generated in total 90,000 recommendations for 4,500 users each week, the amount of *Hot* recommended corresponds to a very small percentage of the entire set of recommendations. More than 200 applications each week can be classified as *Hot*, and about 160 as *Flop*. On average, only 3.6% *Hot* apps are recommended, compared to 4.3% *Flops*.

Week	Diversity	Novelty	Accuracy	Div. w/o Flops	Nov. w/o Flops	Acc. w/o Flops
1	-	-	0.02	-	-	0.02
2	0.80	0.98	0.03	0.90	0.90	0.12
3	0.62	0.81	0	0.54	0.73	0.10
4	0.56	0.75	0.11	0.50	0.68	0.11

Table 6.3: Diversity, novelty, and accuracy statistics of the 20 best recommendations out of top 1000 applications during a month. Previously published [61].

Table 6.3 shows that when the *Flops* are removed from the recommendations, both novelty and diversity decrease, but accuracy increases slightly. The main reason for this behavior is that the metrics used to generate recommendations require a sufficient amount of usage before an app is recommended. However, once sufficient usage has been observed, the app can already be past its "best before" date as the recommendation model does not differentiate between the *Hot* and *Flop* apps.

Integrating usage trend information as part of the recommendation process can help to overcome this issue and improve the overall quality of the application recommendations. We suggest that, among others, the applications with the *Flop* pattern might be reasonable to remove completely, and the weight given to the *Hot* applications might be increased. Thus, we can warn users for using applications that might be losing their popularity and soon becoming less supported when their developers' focus changes to the new projects. Focusing on the *Hot* applications users gain the benefits of the applications with a strong user base: security updates and developing towards new features.

6.3 Insights into Demographic, Geographic, and Cultural Factors in Mobile Usage

Sometimes mobile applications are used not because they are popular in general, but because they are popular in the users' context: in their country or among their family and friends. On the other hand, not only the application's popularity affects the usage of the applications, but also the application functionality, and the user's background, needs, and desires. It is easily understandable that people of different background and demography consider different use cases more important than others.

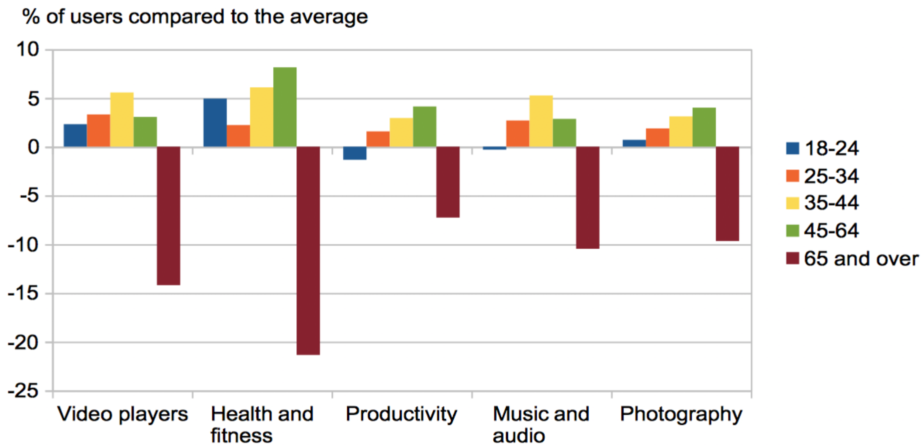


Figure 6.4: Comparison between different age groups.

Next, we show that indeed, demography, geography, and culture play an important role when considering application usage. Understanding these factors in the mobile application usage will help many stakeholders in their work, including application developers, social researchers, and other parties involved in cultural studies or mobile application ecosystem.

6.3.1 Demographic Factors

The user questionnaire run for the Carat user base (described in Section 3.3) provides background information to analyze in more detail how people of different age, education level, and so on use their smartphones. We call these features demographic factors on smartphone usage.

From the questionnaire data we choose four factors for more detailed analysis: age group, education, occupation, and household situation. Education and occupation gain a high mutual information in the analysis performed in Section 5.3.1, and the age and household situation are included because of the general interest. The country information gained the highest mutual information and it will be discussed next in Section 6.3.2. Out of all the answers, we consider those groups with ten or more responses. For example, there are hundreds of students and professionals, but only a few respondents staying at home with kids, or working in agriculture. There are 55 different application categories in Google Play. For convenience, the categories were sorted by the highest standard deviation among the usage of each answer group to highlight the differences between the groups.

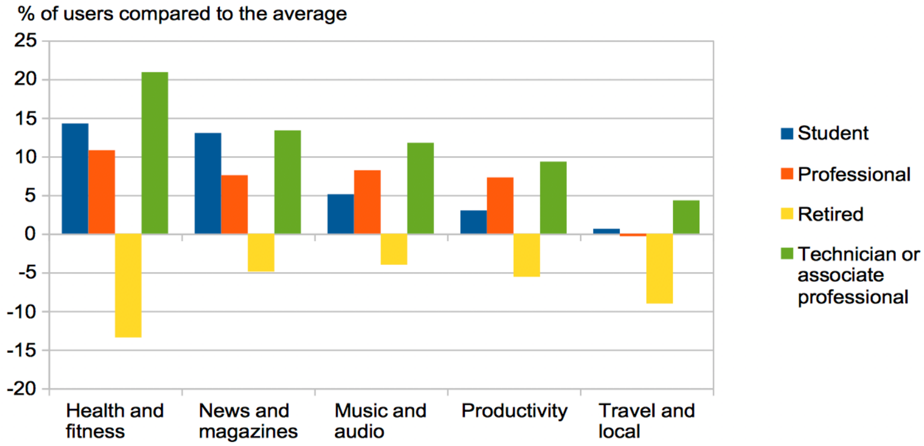


Figure 6.5: Comparison between different occupation groups.

Figure 6.4 presents the comparison between the age groups, all of them included: 18 – 24 (12% of respondents), 25 – 34 (30%), 35 – 44 (28%), 45 – 64 (27%), and over 65 years old (4%). The underage children were excluded from the study. The application categories considered interesting due to the standard deviation-based analysis are *Video players*, *Health and Fitness*, *Productivity*, *Music and Audio*, and *Photography*. The graph presents comparison the average usage, in percentages. Compared to the other user groups, elder people tend to use less of all the categories. Especially "trendy" categories, such as *Health and Fitness* do not gain popularity in this group. Elder people might be excluded from the marketing and target audience of these apps, even if caring for your health does not become less important with age. On the other hand, the health apps are the most popular in people of 45 – 64 years old. People of working age (35 – 44 years old) seem to be the most active users of *Video players* and *Music and Audio* - both categories that might be considered to gain their greatest audience from young people. It is possible that these kinds of applications provide relaxation during work days and thus gain their popularity in this age group.

Figure 6.5 compares different occupational backgrounds. Out of 13 possible choices in the questionnaire, we consider the following four groups: students (12% of answerers), professional (34%), retired (5%), and technician or associate professional (14%). The application categories considered in this case are: *Health and Fitness*, *News and Magazines*, *Music and Audio*, *Productivity*, and *Travel and Local*. Retired people follow the same pattern considered in the case of elder people in general: they seem to use less than

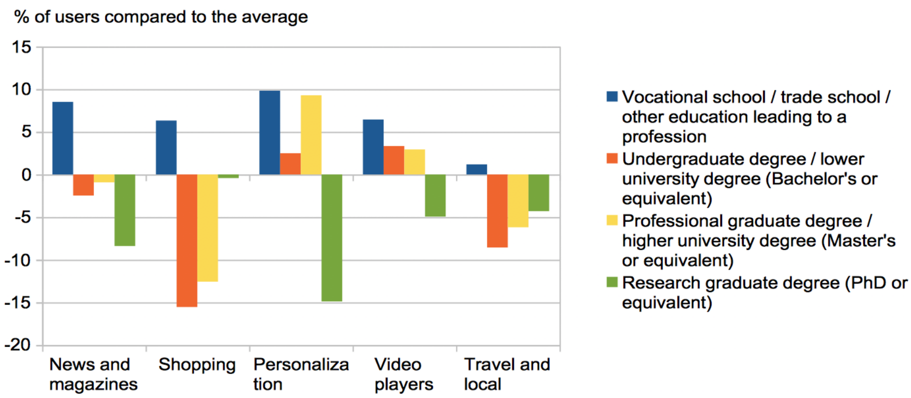


Figure 6.6: Comparison between different education backgrounds.

average in all the categories. Professional and associate professionals seems to use considerable amounts of the productivity applications, maybe to help in their work. Technicians and associate professionals have the highest usage of the traveling apps in comparison: maybe they gain more free time than students and professionals, and use it for more smartphone-oriented tasks compared to retired people.

The comparison between education levels is presented in Figure 6.6. The following groups are considered out of seven different options: vocational school or trade school or other education leading to a profession (11% of respondents), undergraduate or lower university degree (Bachelor's or equivalent) (35%), professional graduate degree or higher university degree (Master's or equivalent) (30%), and research graduate degree (PhD or equivalent) (5%). The application categories considered are the following: *News and Magazines*, *Shopping*, *Personalization*, *Video players*, and *Travel and Local*. Interestingly, people with vocational school or corresponding seem to use all the categories more than people with other educational backgrounds. Especially the *News and Magazines* and *Shopping* categories are more popular among them than the other groups. It seems that the highest educational group including PhD and corresponding use the *Shopping* applications as much as the average, but the lower university degree holders use them significantly less. On the other hand, having a PhD seems to reduce a need for *Personalization* apps, as well as *News and Magazines* and *Video players*.

Figure 6.7 presents the differences between household situations. There the following groups are considered: living alone (19% of respondents), living with other adult(s) (48%), living alone with under-aged kid(s) (30%),

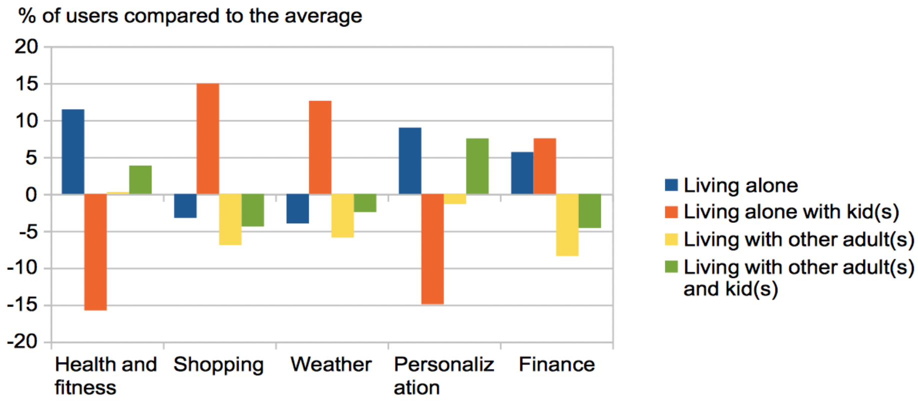


Figure 6.7: Comparison between different household situations.

and living with other adult(s) and kid(s) (3%). The following application categories are considered: *Health and Fitness*, *Shopping*, *Weather*, *Personalization*, and *Finance*. People taking care of their kids alone seems to prefer *Shopping* and *Whether* applications highly compared to the people in other household situations. At the same time, their time seems to be limited for the *Health and Fitness* and *Personalization* apps. Living alone without other adults seems to reduce a need for the *Finance* applications, maybe because there is no other person to help with financial issues. *Health and Fitness* are the most popular among people living alone, too.

6.3.2 Geographic Factors

Comparing demographic information across countries gives us an insight to application usage worldwide. Understanding how demographically speaking similar people - for example, people of the same age, education, and occupation - use their smartphones in different countries can provide important insights on geographic and cultural boundaries. As already discussed in Section 6.3.1, we study in more detail several demographic factors including occupation and education. We also include the household status to highlight some common, interesting clusters. Countries with less than 10 respondents to the questionnaire are excluded, leaving 21 in total out of 44 countries included in the comparative study in Section 5.3.2.

In Figure 6.8, we compare the four most widely represented occupations (student, professional, retired, and technician or assistant professional) within 21 countries. In Figure 6.9, we present a similar comparison between the best represented educational levels (education leading to a profession,

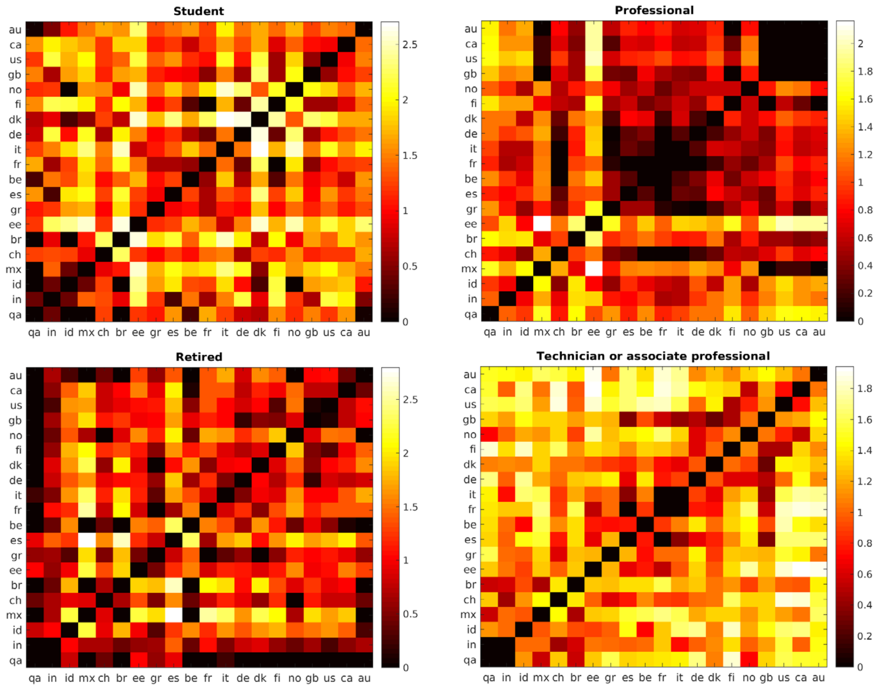


Figure 6.8: Comparison between different occupation groups in the selected countries. The colormaps are based on the KL differences.

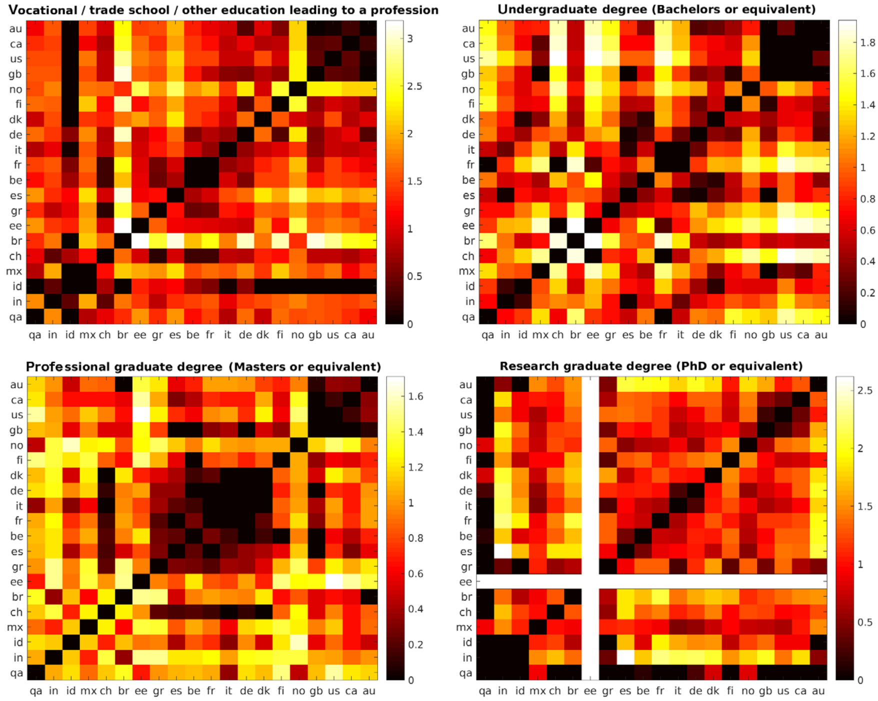


Figure 6.9: Comparison between different education backgrounds in the selected countries. The colormaps are based on the KL differences. As far as the PhD degree is concerned, the values for Estonia (ee) are missing.

Bachelor's degree, Master's degree, and PhD equivalent degree). In both figures, the darker color indicates closeness (the KL divergence between countries close to 0) and lighter color a longer distance (the higher KL divergence, see Section 5.3.2).

As seen in Figure 6.8, professionals in Australia, Canada, the USA, and the United Kingdom use application categories similarly, indicated as a dark cluster in the North-Eastern corner of the colormap. The same cluster is visible in all the educational groups in Figure 6.9, and we may conclude that highly educated people or those working as professionals seem to use their mobile devices similarly in these Western, English-speaking countries.

Another cluster is visible in the South-Western corner of the colormaps, including Qatar (qa), India (in), and Indonesia (id). Especially students and people with PhD or equivalent degree are presented in this cluster, indicating similarities in application usage of academic people in these countries. It is possible that these groups also have a higher smartphone penetration, and

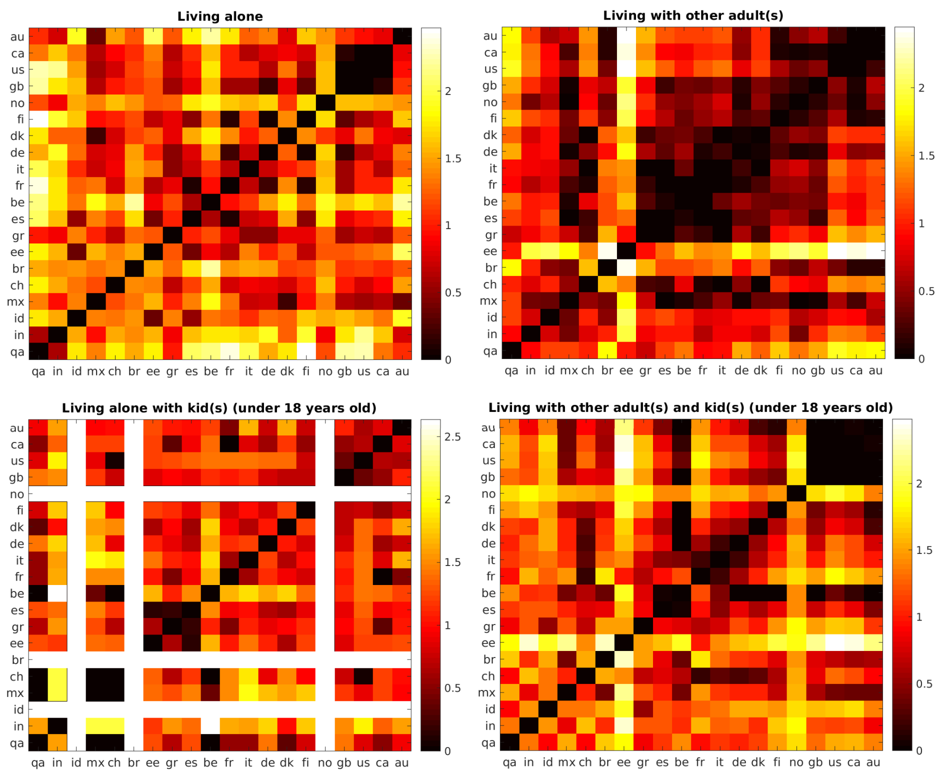


Figure 6.10: Comparison between different household situations in the selected countries. The colormaps are based on the KL differences.

the use of English may be required in studies. Highly educated people are often considered to be builders of rising societies, and they may adapt faster to new technologies such as mobile device functionalities in everyday use.

For professionals and Master's degree holders there is also a third cluster in the middle of the colormaps. This cluster includes European countries: Denmark (dk), Germany (de), Italy (it), France (fr), Belgium (be), Spain (es), and Greece (gr). The application category usage of this group is different from the previously mentioned English-speaking cluster, as also seen in Figure 5.4 presented in Section 5.3.2.

Interestingly, students seem to use applications differently in each country, as there are no clear clusters in the students' colormap. This might be because university students may travel to faraway countries to study, while, for example, vocational studies are commonly done in the same or nearby countries. The colormap of retired people is darker than the others overall, which means their application use is more similar through all the considered countries, but are few strongly similar clusters. This may be a result of people having used different sets of applications, not adopting new ones as a group. Technicians and associate professionals have the lightest colormap in comparison, indicating that the countries have the highest distances (and thus less overall similarity) in this occupational category. This may be caused by the wide range of actual professions and people with different smartphone needs in the category.

6.3.3 Cultural Factors

In addition to the demographic and geographic factors, we also study how culture affects mobile usage. Culture is a wide concept to define with some difficulties. The culture of an individual or group can encompass all aspects of life. For example, the Cambridge English Dictionary defines culture as "the way of life, especially the general customs and beliefs, of a particular group of people at a particular time." Elements of culture may include habits, rituals, and beliefs, as well as ways to perform everyday actions.

Cultural Value Model. In empirical research, Hofstede's Cultural Values Model (VSM) [106, 107] is used with wide variety to represent cultural values between countries [108]. The VSM model consists of six factors, given by a country, that are made by questionnaire studies in different countries around the world.

The VSM model has been previously used, for example, to study culture in IT corporations [109], evaluate tourist services [110], study international ethics [111], evaluate consumer decision making [112], analyze Doodle scheduling responses [113], and model emoji usage in different countries [114]. The VSM model is not free from criticism. Especially, McSweeney [115] questions the validity of defining culture boundaries based on politically agreed national areas. Also, the model does not include minorities or subcultures inside countries, or take into account immigration and emigration in the global world, previously referred to as transnational mobility [116]. These need to be taken into account when analyzing the results: assumptions can be made only to present the measured cultural values, leaving the larger picture of culture harder to capture.

In our work attached to this thesis as Manuscript II [92], we evaluate differences between countries by comparing mobile usage to the six VSM factors[1], described in the following:

- **Power distribution (PDI)** describes whether unequal power distributions are expected and accepted in the population. Cultures with higher power distribution tend to be more hierarchical and persist more inequalities compared to the cultures with lower power dimension.
- **Individualism versus collectivism (IDV)** describes how much members of the population are supposed to take care of themselves or stay integrated to the group, such as family. In cultures with high individualism people define themselves as "I", compared to the stronger "we" feeling in countries with lower individualism.
- **Masculinity versus femininity (MAS)** describes strength of masculine and feminine roles in the population, for example, in working life. Cultures of high masculinity are more competitive, compared to the lower masculinity (higher femininity) that stands on collaboration and modesty.
- **Uncertainty avoidance (UAI)** describes whether members of the population feel either comfortable or uncomfortable in new, unstructured, or unpredictable situations. A high level of uncertainty avoidance implicates stricter codes of planning and caring for the future, compared to more relaxed cultures of the lower score of this factor.
- **Long versus short-term orientation (LTO)** describes how members of the population accept delays in either social, material, or emotional gratification. Cultures with a high score of this factor are more future-planning compared to those that score lower.

[1] The open-sourced VSM data matrix is available in: `http://www.geerthofstede.nl/dimension-data-matrix`

- **Indulgence versus restraint (IVR)** describes whether any gratifications are allowed to be relatively free (having fun by themselves) or regulated by strict norms of the population. A high indulgent score reflects higher importance of free leisure time compared to restraint cultures of a lower indulgence score.

Several methodologies to clean and process the application usage data have already been given in this thesis: Section 4.5 describes the process to collect the country information, Section 4.6 how application data is collected and cleaned, and Section 4.7 how application categories are delivered. In Section 5.3, we present how application usage data is converted to the usage vectors that are more flexible to process with different statistical methods. Countries' usage vectors are considered as the average of users, belonging to the given country, that have used a certain category.

Next, we correlate the usage of all the application category and VSM factor pairs separately. Table 6.4 summarizes the results and lists the categories that have the highest positive or negative correlation for the VSM factors.

Table 6.4 shows us several findings. For example, a low power distance that indicates low hierarchy in the culture, correlates significantly to the use of *Entertainment* applications and other leisure-related categories, such as *Travel and Local*, *Sports*, and *Music and Audio*. These same categories together with, for example, *Health and Fitness* are mostly related to individualist cultures.

Collectivist cultures, those with higher power distance, and cultures considered feminine seem to value family related categories, such as *Family create*, *Education games*, and *Family pretend*. Masculine cultures correlate with high use of *Personalization* apps. Long-term-oriented cultures seem to prefer *Sport, Casual* and *Word games*, as well as *Social* apps. In short-term-oriented cultures, there is a preference for *Role playing games* and a need for *Weather* apps as well as *Comics*.

It is noticeable that categories with high correlations differ from those with the highest usage in general (as presented in Figure 4.2), indicating that the differences are more sophisticated and complex by nature. A similar correlation analysis can also be performed reversely. There are nine categories that correlate less than 0.2 (or -0.2, similarly) to at least five VSM factors. The category *Dating* correlates slightly more (0.26) only to the Individualism versus collectivism, and the category *Events* to the Masculinity versus femininity (0.21). *Game role playing* has very low impact in five categories, but gains more than 0.3 correlation to Long versus short-term orientation. In addition to these, the category *Beaty* and a list of

	Power distance (PDI)
ρ	*Categories*
$<$ -0.5	Music & audio, Entertainment, Weather
$<$ -0.4	News & magazines, Productivity, Travel & local, Sports, Libraries & demo
$<$ -0.3	Game trivia, Photography, Finance, Communication, Auto & vehicles, Game card
$>$ 0.3	Family create
$>$ 0.4	Game action
	Individualism versus collectivism (IDV)
ρ	*Categories*
$<$ -0.4	Family create, Game action
$<$ -0.3	Game education
$>$ 0.4	Books & references, Photography, Libraries & demo, Education, Finance, Game words, Medical, Family music video
$>$ 0.5	Auto & vehicles, Productivity, Sports
$>$ 0.6	Weather, News & magazines, Travel & local, Health & fitness, Music & audio, Entertainment
	Masculinity versus femininity (MAS)
ρ	*Categories*
$<$ -0.4	Family pretend
$<$ -0.3	Game board
$>$ 0.3	Personalization
	Uncertainty avoidance (UAI)
ρ	*Categories*
$<$ -0.4	Parenting, News & magazines, Family music video, Game words
$<$ -0.3	Education, Family education, House & home, Entertainment, Books & references, Family brain games
$>$ 0.3	Family create
$>$ 0.4	Game action
	Long versus short-term orientation (LTO)
ρ	*Categories*
$<$ -0.4	Game sports
$<$ -0.3	Family music video, Game word, Social, Game casual
$>$ 0.3	Maps & navigation, Game role playing
$>$ 0.4	Comics, Weather
	Indulgence versus restraint (IVR)
ρ	*Categories*
$>$ 0.3	Sports, Photography, Communication, Game words
$>$ 0.4	Music & audio, Family music video, News & magazines, Entertainment, Books & references

Table 6.4: The best category correlations to VSM factors with 44 countries.

different game categories gain low correlation to every VSM factor. Indeed, there are certain categories that are in general more independent from the cultural model than the others. These categories can provide us with insights to the applications that are similarly important through all the studied countries.

To summarize, mobile usage reflects geographic, demographic, and cultural boundaries and at the same time, it cannot be truly explained only by those societal and cultural factors. We propose mobile usage as a novel societal factor to consider in future studies that apply smart devices worldwide.

Chapter 7

Conclusions

7.1 Summary of the Main Findings

This thesis has proposed methods and approaches to clean, analyze, and utilize crowdsensed mobile data for actionable feedback and human-readable recommendations. To summarize the main findings of the work, this section revisits the research questions provided in the beginning of the thesis in Section 1.2. Those research questions and their proposed, summarized answers are the following:

RQ1. **How do different data attributes have to be cleaned and pre-processed to produce a reliable picture of the system state?**

There is a need for cleaning mobile crowdsensed data before it is used as an input of the analysis systems: the data contains misreadings, missing values, manufacturer-specific default values, and other items that have to be considered in more detail. We show that using natural thresholds and statistical analysis of the items' value ranges we can include the valid data items in the analysis set and remove invalid ones. The cleaning procedure has to be separately defined for each crowdsensed data attribute, and that is where understanding the collected data becomes so crucial.

RQ2. **How can crowdsensed data be used to present crucial factors of a smartphone's system state?**

Our work with system setting and subsystem variable analysis has shown that these context factors, indeed, have a crucial effect on smartphone energy consumption. Our findings are in line with the previous literature in the case of single context factors, and our results are possible to validate with the laboratory measurements, too. Indeed, the crowdsensed data provides new insights to the real-life use cases that are not even possible to model in limited laboratory conditions.

RQ3. **What are the effects of subsystem variables, system settings, and their combinations to smartphone energy consumption?**

We have shown that not only single factors affect smartphone energy consumption, but the crowdsensed data reveals that the combinations give even more detailed insights to the energy-hungry system settings and subsystem variables. We show that the most accurate energy impact is revealed when the system state of the device is analyzed as a whole, taking into account all the possible combinations that system settings, subsystem variables, and running applications can combine. In this kind of analysis, the crowdsensed data provides valuable new insights and gives possibilities to look for an almost unlimited number of real-life system states compared to the more dependent laboratory environments.

RQ4. **How can smartphone energy consumption be improved by recommending better system state and subsystem variables?**

We propose a novel energy recommendation system Constella that can take into account the whole system state of the device, including subsystem variables, system settings, and applications. Most importantly, the combinations of these context factors can be covered in the effective, decision tree-based approach. The Constella recommendation system relies on the concept of the continuous feedback loop where the data items are collected from the crowd, processed and analyzed in the back-end cloud-computing environment, and the value of the results is then sent back to the devices as human-readable, actionable recommendations. Thus, the value of the analysis will be returned to the sources of the data, also, to benefit of the future learning loops.

RQ5. **How can mobile recommendation systems be improved by analyzing application popularity?**

We suggest that the trend filtering methodology can be used to improve current application recommendation systems. We show that the traditional recommendation systems tend to favor also applications that are already falling in popularity. We show how the trend filters can be used to improve recommendation results by filtering out the *Flop* applications that are already losing their popularity or increasing the weight given to the *Hot* applications that show their potentiality by gaining a significant user base fast.

RQ6. **What can be learned about mobile application usage and popularity in real-life crowdsensed data?**

Understanding application usage in the wild provides several new insights in how applications become popular or fall in popularity. Our trend-filtering approach suggests a novel methodology to represent popularity in addition to the traditional retention rates used by the marketing analysis companies. We can characterize applications as *Hot*, *Flop*, *Marginal*, or *Dominant* based on their lifetime success in the crowd of mobile devices.

RQ7. **How does mobile application usage reflect differences in user population?**

We compare crowdsensed mobile application usage to the existing Cultural Value Model and find out that, indeed, there are correlations between mobile application usage and known cultural factors. Our study suggests that in the future, mobile application usage could be seen as an additional demographical factor or at least as a reflection of local societies. Our approach can potentially help different social research areas and application developers targeting international markets.

RQ8. **What can be learned about cultural, demographical, and geographical differences in crowdsensed smartphone usage?**

In addition to the knowledge of how applications are used as a whole, different demographic, geographic, and cultural factors have shown to have a significant effect to the mobile usage patterns. We study worldwide crowdsensed application usage and show that, indeed, there are differences and similarities between certain areas. For example, we can categorize 44 countries into three groups, including mainly the English-speaking countries, the continental European countries, and the mixed group of various Asian and Middle-Eastern countries. Our research suggests that in addition to the demographic factors, such as age group, gender, occupation, or education, the country is an essential source of information when studying mobile usage in the wild.

7.2 Implications of the Research

The research in this thesis has shown that the crowdsourced data can and should be utilized in the cases where previously only laboratory measurement would be considered. Especially when studying real-life effects and use cases in the wild, the large-scale crowdsensed mobile data provides essential insights impossible or too expensive to model alone in laboratory conditions. This is especially important in an energy consumption perspective, where various different usage situations need to be included in reliable models.

The crowdsensed approach allows modeling real-life system state combinations and reveals important interdependencies between different context factors. This is especially seen in our energy modeling work, where we show that system setting and subsystem variable combinations provide more complex information about the battery life and overrun any single sensor-based approaches to understanding mobile energy consumption.

The crowdsensed mobile data provides a possibility for independent, large-scale studies that are not related to the marketing companies or mobile manufacturers. For example, we have presented an independent study of retention rates that provides information about application popularity. We suggest novel methods to better understand application popularity and trend patterns, and mobile usage all around the world, which provides important insights for multiple parties involved in the mobile ecosystem.

We have shown that mobile application usage, indeed, reflects demographic, geographic, and cultural factors, which is crucial for application developers to take into account when targeting their products worldwide. There are several clear design implications: Understanding the target audience, their needs, and habits regarding the mobile usage helps to design suitable system features. Knowing the popularity of certain categories in different areas eases the definition of general terms for the applications and make them easier to find. Understanding cultural differences in application use helps to both target or generalize applications in the worldwide market.

7.3 Limitations

Even if the crowdsensed approaches can provide beneficial information to utilize in many application areas, there are still existing challenges. Even if many easy to use systems have been introduced, in most of the cases, the best suitable crowdsensing application requires mobile development skills and understanding of large-scale computing paradigms and distributed machine learning algorithms. Not every researcher or developer has time or opportunity to study all the necessary skills.

When it comes to mobile crowdsensing, data cleaning still has challenges not fully solved. For example, every new addition to the context factor set has to be studied independently to understand its natural value thresholds and statistical distributions. When new data items have been collected and, for example, new and more effective device models introduced, also the statistical distributions and features used in the old models may become dated. This means not only the learning phase but also the data cleaning procedures have to be updated regularly.

The data collection itself sets its own challenges. Even if it is known that the crowdsensing systems can provide large amounts of data in a comparably short period of time, there still exists the cold start problem to first gain the user base and then get enough data to start the analysis phases. Many existing machine learning applications are based on previously collected data sets and only rarely fully online-based learning systems are proposed. Without sufficient training data, starting a new crowdsensing project and implementing necessary machine learning procedures may be difficult.

User acquisition has often been seen as an important challenge where there really is no silver bullet to solve it. The Carat system relies on energy recommendations it gives back to the users as an additional value. The truth still seems to be that not many people want to participate in research projects without any other benefit. Giving out money or gift cards may be out of the budget for many research teams, and even then, gaining a representative user population might be challenging. First, when considering mobile crowdsensing, only users with an appropriate smartphone can be studied. There is always a group of people left outside, for example, in the case of the Carat project, only Android and iPhone users can be considered and even there, iOS provides significantly less information out of the device compared to the Android system.

Second, people owning a suitable smart device and volunteering to participate in the research task do not necessarily correspond to the full population using these devices. Based on the user questionnaires run for the Carat population, there is a high bias towards well-educated men working in professional occupations. Women and less educated people consist of a clear minority. Interestingly, the age group seems to be the least biased attribute, because even elder people are well represented in the Carat questionnaires. Without fully working recruiting strategies, it seems hard to gain a well representative user population.

Crowdsensing smartphone data gives a great responsibility to researchers managing such privacy-sensitive information. Users should trust the policies and storage strategies involved in the analysis process, and in the case of industrial applications, also trust that no information learned from the data is used harmfully or unpleasantly. Privacy and security questions still need to be covered in more detail in the future.

7.4 Future Work

This thesis has presented several use cases for crowdsensed mobile data analytics. On the other hand, there are still open questions and novel application areas where the mobile devices and the Carat data can be utilized. For example, the energy analysis can be enlarged to cover combinations of different applications together with system settings and subsystem variables. The Constella recommendation system should be analyzed in the wild with its recommendations sent back to the user community. Thus, the real-life effects of this kind of recommendation systems could be tested and evaluated.

This thesis focuses on a single energy decision tree constructed from the entire dataset, but it could be more beneficial to consider each device model separately, or based on the user profiles. Together with understanding the usage context and application usage history, also the energy recommendations could be improved even further. For example, the network infrastructures differ between countries and different user populations have different needs for their smartphones, so it is reasonable to consider that also their energy profiles vary. Personal energy plans might be one of the next topics to investigate. Characterizing users and their needs for their smartphones could be used to generate more accurate application recommendations.

The demographic, cultural, and geographical analyses of this thesis have focused on a limited number of data attributes available in the questionnaire run for the Carat user base. More information about the user context could be collected, including but not limited to, for example, the user's personality traits, mental state, and daily habits and routines. From a cultural point of view, also different personal beliefs, religion, and political opinions might be considered, as well as identification with minorities and subcultures. There is previous work about these topics, but the Carat user base provides a special opportunity to study these topics worldwide with real users and usage cases in the wild.

The user context analysis together with understanding of global and local application trends provide a rich input also for new application recommendation systems. The current trend analysis presented in this thesis focuses on the global trends, but in the future also local trends and popularity inside certain demographical subgroups or geographical areas would provide fruitful results.

7.5 Conclusion

This thesis has presented how crowdsensed mobile data can be used in benefit of energy diagnosis, application popularity analysis, and demographic insights. Smartphones have become a crucial part of modern everyday life and it is clear that they and corresponding new smart devices in the future will continue this trend. People have become used to being connected and relying on a single device of multiple integrated functionalities for fun and leisure as well as for work and education.

The crowdsensed data can be utilized for several application areas, but it still also has challenges to solve. For example, the autonomous data collection requires intensive cleaning strategies for functional information retrieval. Large-scale data collection sets challenges for running effective machine learning procedures and statistical analysis also in the cloud-computing environment. Analysis results of the learning algorithms have to be converted to human-readable form and visualizations to fully utilize their value in the future. For example, recommendation systems and decision-making tools can utilize crowdsensed data effectively.

This thesis has proposed methodologies to collect, clean, analyze, and form the value out of the crowdsensed data. As an opinion of the author, there are more pros than cons in the crowdsensed mobile data analytics. The open challenges of the field only help to make applications and methodologies stronger and easier to utilize in the future.

References

[1] Bin Guo, Zhiwen Yu, Xingshe Zhou, and Daqing Zhang. From participatory sensing to mobile crowd sensing. In *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2014 IEEE International Conference on*, pages 593–598. IEEE, 2014.

[2] Dejun Yang, Guoliang Xue, Xi Fang, and Jian Tang. Incentive mechanisms for crowdsensing: Crowdsourcing with smartphones. *Biological Cybernetics*, 24(3):1732–1744, 2016.

[3] Raghu K Ganti, Fan Ye, and Hui Lei. Mobile crowdsensing: current state and future challenges. *IEEE Communications Magazine*, 49(11), 2011.

[4] HV Jagadish, Johannes Gehrke, Alexandros Labrinidis, Yannis Papakonstantinou, Jignesh M Patel, Raghu Ramakrishnan, and Cyrus Shahabi. Big data and its technical challenges. *Communications of the ACM*, 57(7):86–94, 2014.

[5] Daniel Hintze, Philipp Hintze, Rainhard D. Findling, and René Mayrhofer. A large-scale, long-term analysis of mobile device usage characteristics. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies (IMWUT)*, 1(2):13:1–13:21, June 2017.

[6] Adam J. Oliner, Anand P. Iyer, Ion Stoica, Eemil Lagerspetz, and Sasu Tarkoma. Carat: Collaborative energy diagnosis for mobile devices. In *Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems*, SenSys '13, pages 10:1–10:14, New York, NY, USA, 2013. ACM.

[7] Daniel T Wagner, Andrew Rice, and Alastair R Beresford. Device Analyzer: Large-scale mobile data collection. In *Big Data Analytics workshop (in conjunction with ACM Sigmetrics 2013)*. ACM, 2014.

[8] Daniel T Wagner, Andrew Rice, and Alastair R Beresford. Device analyzer: Understanding smartphone usage. In *International Conference on Mobile and Ubiquitous Systems: Computing, Networking, and Services*, pages 195–208. Springer, 2013.

[9] Andrew Rice and Simon Hay. Decomposing power measurements for mobile devices. In *Proceedings of the 2010 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 70–78. IEEE, 2010.

[10] Hossein Falaki, Ratul Mahajan, Srikanth Kandula, Dimitrios Lymberopoulos, Ramesh Govindan, and Deborah Estrin. Diversity in smartphone usage. In *Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services (MobiSys 2010)*, pages 179–194. ACM, 2010.

[11] Hien Truong, Eemil Lagerspetz, Petteri Nurmi, Adam Oliner, Sasu Tarkoma, and N. Asokan. The company you keep: Mobile malware infection rates and inexpensive risk indicators. In *Proceedings of the 23rd International Conference on World Wide Web*, WWW '14, pages 39 – 50. ACM, 2014.

[12] Qiang Xu, Jeffrey Erman, Alexandre Gerber, Zhuoqing Morley Mao, Jeffrey Pang, and Shobha Venkataraman. Identifying diverse usage behaviors of smartphone apps. In *Proceedings of the 11th ACM SIGCOMM Internet Measurement Conference*, IMC '11, 2011.

[13] Pascal Welke, Ionut Andone, Konrad Blaszkiewicz, and Alexander Markowetz. Differentiating Smartphone Users by App Usage. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '16, 2016.

[14] Emmanouil Koukoumidis, Li-Shiuan Peh, and Margaret Rose Martonosi. Signalguru: leveraging mobile phones for collaborative traffic signal schedule advisory. In *Proceedings of the 9th international conference on Mobile systems, applications, and services*, pages 127–140. ACM, 2011.

[15] Samuli Hemminki, Petteri Nurmi, and Sasu Tarkoma. Accelerometer-based transportation mode detection on smartphones. In *Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems*, page 13. ACM, 2013.

[16] Valentin Radu, Lito Kriara, and Mahesh K Marina. Pazl: A mobile crowdsensing based indoor wifi monitoring system. In *Network and Service Management (CNSM), 2013 9th International Conference on*, pages 75–83. IEEE, 2013.

[17] Ruipeng Gao, Mingmin Zhao, Tao Ye, Fan Ye, Yizhou Wang, Kaigui Bian, Tao Wang, and Xiaoming Li. Jigsaw: Indoor floor plan reconstruction via mobile crowdsensing. In *Proceedings of the 20th annual international conference on Mobile computing and networking*, pages 249–260. ACM, 2014.

[18] 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*, pages 481–490. ACM, 2012.

[19] Erhard Rahm and Hong Hai Do. Data cleaning: Problems and current approaches. *IEEE Bulletin of the Technical Committee on Data Engineering*, 23(4):3–13, 2000.

[20] Ibrahim Abaker Targio Hashem, Ibrar Yaqoob, Nor Badrul Anuar, Salimah Mokhtar, Abdullah Gani, and Samee Ullah Khan. The rise of "big data" on cloud computing: Review and open research issues. *Information Systems*, 47:98–115, 2015.

[21] Kyungseo Park, Eric Becker, Jyothi K Vinjumur, Zhengyi Le, and Fillia Makedon. Human behavioral detection and data cleaning in assisted living environment using wireless sensor networks. In *Proceedings of the 2nd International Conference on PErvasive Technologies Related to Assistive Environments*, page 7. ACM, 2009.

[22] James Williamson, Qi Liu, Fenglong Lu, Wyatt Mohrman, Kun Li, Robert Dick, and Li Shang. Data sensing and analysis: Challenges for wearables. In *Design Automation Conference (ASP-DAC), 2015 20th Asia and South Pacific*, pages 136–141. IEEE, 2015.

[23] Shawn R Jeffery, Gustavo Alonso, Michael J Franklin, Wei Hong, and Jennifer Widom. Declarative support for sensor data cleaning. *Lecture Notes in Computer Science*, 3968:83–100, 2006.

[24] Diane M Strong, Yang W Lee, and Richard Y Wang. Data quality in context. *Communications of the ACM*, 40(5):103–110, 1997.

[25] Eiman Elnahrawy and Badri Nath. Cleaning and querying noisy sensors. In *Proceedings of the 2nd ACM international conference on Wireless sensor networks and applications*, pages 78–87. ACM, 2003.

[26] Dimitrios Lymberopoulos, Athanasios Bamis, and Andreas Savvides. Extracting spatiotemporal human activity patterns in assisted living using a home sensor network. *Universal Access in the Information Society*, 10(2):125–138, 2011.

[27] Yongxin Tong, Caleb Chen Cao, Chen Jason Zhang, Yatao Li, and Lei Chen. Crowdcleaner: Data cleaning for multi-version data on the web via crowdsourcing. In *Data Engineering (ICDE), 2014 IEEE 30th International Conference on*, pages 1182–1185. IEEE, 2014.

[28] Shawn R Jeffery, Gustavo Alonso, Michael J Franklin, Wei Hong, and Jennifer Widom. A pipelined framework for online cleaning of sensor data streams. In *Data Engineering, 2006. ICDE'06. Proceedings of the 22nd International Conference on*, pages 140–140. IEEE, 2006.

[29] Xu Chu, Ihab F Ilyas, and Paolo Papotti. Holistic data cleaning: Putting violations into context. In *Data Engineering (ICDE), 2013 IEEE 29th International Conference on*, pages 458–469. IEEE, 2013.

[30] Maksims Volkovs, Fei Chiang, Jaroslaw Szlichta, and Renée J Miller. Continuous data cleaning. In *Data Engineering (ICDE), 2014 IEEE 30th International Conference on*, pages 244–255. IEEE, 2014.

[31] Xu Chu, John Morcos, Ihab F Ilyas, Mourad Ouzzani, Paolo Papotti, Nan Tang, and Yin Ye. Katara: A data cleaning system powered by knowledge bases and crowdsourcing. In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, pages 1247–1261. ACM, 2015.

[32] Michele Dallachiesa, Amr Ebaid, Ahmed Eldawy, Ahmed Elmagarmid, Ihab F Ilyas, Mourad Ouzzani, and Nan Tang. Nadeef: a commodity data cleaning system. In *Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data*, pages 541–552. ACM, 2013.

[33] Nilanjan Banerjee, Ahmad Rahmati, Mark D. Corner, Sami Rollins, and Lin Zhong. Users and batteries: Interactions and adaptive energy management in mobile systems. In John Krumm, Gregory D. Abowd, Aruna Seneviratne, and Thomas Strang, editors, *UbiComp 2007:*

Ubiquitous Computing, volume 4717 of *Lecture Notes in Computer Science*, pages 217–234. Springer Berlin Heidelberg, 2007.

[34] Ahmad Rahmati, Angela Qian, and Lin Zhong. Understanding human-battery interaction on mobile phones. In *Proceedings of the 9th international conference on Human computer interaction with mobile devices and services*, pages 265–272. ACM, 2007.

[35] Ahmad Rahmati and Lin Zhong. Human–battery interaction on mobile phones. *Pervasive and Mobile Computing*, 5(5):465–477, 2009.

[36] Denzil Ferreira, Eija Ferreira, Jorge Goncalves, Vassilis Kostakos, and Anind K. Dey. Revisiting human-battery interaction with an interactive battery interface. In *Proceedings of Ubicomp 2013*. ACM, 2013.

[37] Abhijeet Banerjee, Lee Kee Chong, Sudipta Chattopadhyay, and Abhik Roychoudhury. Detecting energy bugs and hotspots in mobile apps. In *Proceedings of the 22Nd ACM SIGSOFT International Symposium on Foundations of Software Engineering*, FSE 2014, pages 588–598, New York, NY, USA, 2014. ACM.

[38] Xiao Ma, Peng Huang, Xinxin Jin, Pei Wang, Soyeon Park, Dongcai Shen, Yuanyuan Zhou, Lawrence K. Saul, and Geoffrey M. Voelker. eDoctor: automatically diagnosing abnormal battery drain issues on smartphones. In *Proceedings of the 10th USENIX conference on Networked Systems Design and Implementation*, pages 57–70, Berkeley, CA, USA, 2013. USENIX Association.

[39] Abhinav Pathak, Abhilash Jindal, Y. Charlie Hu, and Samuel P. Midkiff. What is keeping my phone awake?: Characterizing and detecting no-sleep energy bugs in smartphone apps. In *Proceedings of the 10th International Conference on Mobile Systems, Applications, and Services*, MobiSys '12, pages 267–280, New York, NY, USA, 2012. ACM.

[40] Alex Shye, Benjamin Scholbrock, and Gokhan Memik. Into the wild: Studying real user activity patterns to guide power optimizations for mobile architectures. In *Proceedings of the 42nd Annual IEEE/ACM International Symposium on Microarchitecture*, pages 168–178, New York, NY, USA, 2009. ACM.

[41] Aaron Schulman, Thomas Schmid, Prabal Dutta, and Neil Spring. *Demo: Phone Power Monitoring with BattOr*, 2011. ACM Mobicom 2011. Available at http://www.stanford.edu/ aschulm/battor.html.

[42] Lide Zhang, Birjodh Tiwana, Zhiyun Qian, Zhaoguang Wang, Robert P. Dick, Zhuoqing Morley Mao, and Lei Yang. Accurate online power estimation and automatic battery behavior based power model generation for smartphones. In *Proceedings of the 8th IEEE/ACM/IFIP international conference on Hardware/software codesign and system synthesis*, pages 105–114, New York, NY, USA, 2010. ACM.

[43] Fengyuan Xu, Yunxin Liu, Qun Li, and Yongguang Zhang. V-edge: Fast self-constructive power modeling of smartphones based on battery voltage dynamics. In *Proceedings of the 10th USENIX Conference on Networked Systems Design and Implementation*, pages 43–56, Berkeley, CA, USA, 2013. USENIX Association.

[44] Sharad Agarwal, Ratul Mahajan, Alice Zheng, and Victor Bahl. Diagnosing mobile applications in the wild. In *Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks*, pages 22:1–22:6, New York, NY, USA, 2010. ACM.

[45] Daniel T Wagner, Andrew Rice, and Alastair R Beresford. Device Analyzer: Large-scale mobile data collection. *ACM SIGMETRICS Performance Evaluation Review*, 41(4):53–56, 2014.

[46] Ella Peltonen, Eemil Lagerspetz, Petteri Nurmi, and Sasu Tarkoma. Energy modeling of system settings: A crowdsourced approach. In *the 2015 IEEE International Conference on Pervasive Computing and Communications*, PerCom '15, pages 37–45, March 2015.

[47] Ella Peltonen, Eemil Lagerspetz, Petteri Nurmi, and Sasu Tarkoma. Constella: Crowdsourced system setting recommendations for mobile devices. *Pervasive and Mobile Computing*, 26:71 – 90, 2016. Thirteenth International Conference on Pervasive Computing and Communications (PerCom '15).

[48] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8), 2009.

[49] Fredrik Boström, Petteri Nurmi, Akos Vetek, Patrik Floréen, Péter Boda, Tianyan Liu, and Tiina-Kaisa Oikarinen. Capricorn - An intelligent user interface for mobile widgets. In *Proceedings of the 10th*

International Conference on Human-Computer Interaction (MobileHCI), pages 328–330. ACM, 2008.

[50] Andrea Girardello and Florian Michahelles. Appaware: Which mobile applications are hot? In *Proceedings of the 12th International Conference on Human Computer Interaction with Mobile Devices and Services*, MobileHCI '10, pages 431–434, New York, NY, USA, 2010. ACM.

[51] Hossein Falaki, Ratul Mahajan, Srikanth Kandula, Dimitrios Lymberopoulos, Ramesh Govindan, and Deborah Estrin. Diversity in smartphone usage. In *Proceedings of the 8th International Conference on Mobile Systems, Applications, and Services*, MobiSys '10, pages 179–194, New York, NY, USA, 2010. ACM.

[52] Bo Yan and Guanling Chen. Appjoy: Personalized mobile application discovery. In *Proceedings of the 9th International Conference on Mobile Systems, Applications, and Services*, MobiSys '11, pages 113–126, New York, NY, USA, 2011. ACM.

[53] Matthias Böhmer, Brent Hecht, Johannes Schöning, Antonio Krüger, and Gernot Bauer. Falling asleep with angry birds, facebook and kindle: A large scale study on mobile application usage. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services*, MobileHCI '11, pages 47–56, New York, NY, USA, 2011. ACM.

[54] Kent Shi and Kamal Ali. Getjar mobile application recommendations with very sparse datasets. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '12, pages 204–212, New York, NY, USA, 2012. ACM.

[55] Alexandros Karatzoglou, Linas Baltrunas, Karen Church, and Matthias Böhmer. Climbing the app wall: Enabling mobile app discovery through context-aware recommendations. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, CIKM '12, pages 2527–2530, New York, NY, USA, 2012. ACM.

[56] Stefano Mizzaro, Marco Pavan, Ivan Scagnetto, and Ivano Zanello. A context-aware retrieval system for mobile applications. In *Proceedings of the 4th Workshop on Context-Awareness in Retrieval and Recommendation*, CARR '14, pages 18–25, New York, NY, USA, 2014. ACM.

[57] Christoffer Davidsson and Simon Moritz. Utilizing implicit feedback and context to recommend mobile applications from first use. In *Proceedings of the 2011 Workshop on Context-awareness in Retrieval and Recommendation*, CaRR '11, pages 19–22, New York, NY, USA, 2011. ACM.

[58] Wolfgang Woerndl, C. Schueller, and R. Wojtech. A hybrid recommender system for context-aware recommendations of mobile applications. In *Data Engineering Workshop, 2007 IEEE 23rd International Conference on*, pages 871–878, April 2007.

[59] Konglin Zhu, Lin Zhang, and Achille Pattavina. Learning geographical and mobility factors for mobile application recommendation. *IEEE Intelligent Systems*, 32(3):36–44, 2017.

[60] Hannu Verkasalo. An international study of smartphone usage. *International Journal of Electronic Business*, 1/2:158–181, 2011.

[61] Stephen Sigg, Eemil Lagerspetz, Ella Peltonen, Petteri Nurmi, and Sasu Tarkoma. Exploiting usage to predict instantaneous app popularity: Trend filters and retention rates. *Under review, preprint available at `https://arxiv.org/abs/1611.10161`.*

[62] Denzil Ferreira, Jorge Gonçalves, Vassilis Kostakos, Louise Barkhuus, and Anind K. Dey. Contextual experience sampling of mobile application micro-usage. In *Proceedings of the 16th International Conference on Human-computer Interaction with Mobile Devices & Services*, MobileHCI '14, 2014.

[63] Alexis Hiniker, Shwetak N. Patel, Tadayoshi Kohno, and Julie A. Kientz. Why would you do that? predicting the uses and gratifications behind smartphone-usage behaviors. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '16, pages 634–645, New York, NY, USA, 2016. ACM.

[64] Qiang Xu, Jeffrey Erman, Alexandre Gerber, Zhuoqing Mao, Jeffrey Pang, and Shobha Venkataraman. Identifying diverse usage behaviors of smartphone apps. In *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference*, pages 329–344. ACM, 2011.

[65] Matthias Böhmer, Brent Hecht, Johannes Schöning, Antonio Krüger, and Gernot Bauer. Falling asleep with Angry Birds, Facebook and

Kindle: A large scale study on mobile application usage. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services*, 2011.

[66] Simon L Jones, Denzil Ferreira, Simo Hosio, Jorge Goncalves, and Vassilis Kostakos. Revisitation analysis of smartphone app use. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 1197–1208. ACM, 2015.

[67] Thanasis Petsas, Antonis Papadogiannakis, Michalis Polychronakis, Evangelos P. Markatos, and Thomas Karagiannis. Rise of the planet of the apps: A systematic study of the mobile app ecosystem. In *Proceedings of the 2013 Conference on Internet Measurement Conference*, IMC '13, pages 277–290, New York, NY, USA, 2013. ACM.

[68] Suranga Seneviratne, Aruna Seneviratne, Prasant Mohapatra, and Anirban Mahanti. Your installed apps reveal your gender and more! *Mobile Computing and Communications Review*, 18:55–61, 2014.

[69] Sha Zhao, Julian Ramos, Jianrong Tao, Ziwen Jiang, Shijian Li, Zhaohui Wu, Gang Pan, and Anind K. Dey. Discovering different kinds of smartphone users through their application usage behaviors. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '16.

[70] Soo Ling Lim, Peter J. Bentley, Natalie Kanakam, Fuyuki Ishikawa, and Shinichi Honiden. Investigating country differences in mobile app user behavior and challenges for software engineering. *IEEE Transactions on Software Engineering*, 41:40–64, 2014.

[71] Seok Kang and Jaemin Jung. Mobile communication for human needs: A comparison of smartphone use between the us and korea. *Computers in Human Behavior*, 35:376 – 387, 2014.

[72] Gokul Chittaranjan, Jan Blom, and Daniel Gatica-Perez. Who's who with big-five: Analyzing and classifying personality traits with smartphones. In *Proceedings of the 2011 15th Annual International Symposium on Wearable Computers*, ISWC '11, pages 29–36, Washington, DC, USA, 2011. IEEE Computer Society.

[73] Robert LiKamWa, Yunxin Liu, Nicholas D. Lane, and Lin Zhong. Moodscope: Building a mood sensor from smartphone usage patterns. In *Proceeding of the 11th Annual International Conference on Mobile*

Systems, Applications, and Services, MobiSys '13, pages 389–402, New York, NY, USA, 2013. ACM.

[74] Neal Lathia, Veljko Pejovic, Kiran K. Rachuri, Cecilia Mascolo, Mirco Musolesi, and Peter J. Rentfrow. Smartphones for large-scale behavior change interventions. *IEEE Pervasive Computing*, 12(3):66–73, 2013.

[75] Neal Lathia, Kiran K. Rachuri, Cecilia Mascolo, and Peter J. Rentfrow. Contextual dissonance: Design bias in sensor-based experience sampling methods. In *Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '13, pages 183–192, New York, NY, USA, 2013. ACM.

[76] Gillian M Sandstrom, Neal Lathia, Cecilia Mascolo, and Peter J Rentfrow. Putting mood in context: Using smartphones to examine how people feel in different locations. *Journal of Research in Personality*, 69:96–101, 2017.

[77] Agnes Gruenerbl, Venet Osmani, Gernot Bahle, Jose C. Carrasco, Stefan Oehler, Oscar Mayora, Christian Haring, and Paul Lukowicz. Using smart phone mobility traces for the diagnosis of depressive and manic episodes in bipolar patients. In *Proceedings of the 5th Augmented Human International Conference*, AH '14, pages 38:1–38:8, New York, NY, USA, 2014. ACM.

[78] G Drake, E Csipke, and T Wykes. Assessing your mood online: acceptability and use of moodscope. *Psychological medicine*, 43(7):1455–1464, 2013.

[79] Sha Zhao, Julian Ramos, Jianrong Tao, Ziwen Jiang, Shijian Li, Zhaohui Wu, Gang Pan, and Anind K Dey. Who are the smartphone users?: Identifying user groups with apps usage behaviors. *GetMobile: Mobile Computing and Communications*, 21(2):31–34, 2017.

[80] Jeffrey Dean and Sanjay Ghemawat. MapReduce: Simplified data processing on large clusters. In *Proceedings of the Sixth Symposium on Operating Systems Design and Implementation*, OSDI '04, 2004.

[81] Jeffrey Dean and Sanjay Ghemawat. MapReduce: Simplified data processing on large clusters. *Communications of the ACM*, 51(1):107–113, 2008.

[82] Jeffrey Dean and Sanjay Ghemawat. MapReduce: A flexible data processing tool. *Communications of the ACM*, 53(1):72–77, 2010.

[83] Amol Ghotingm, Rajasekar Krishnamurthy, Edwin Pednault, Berthold Reinwald, Vikas Sindhwani, Shirish Tatikonda, Yuanyuan Tian, and Shivakumar Vaithyanathan. SystemML: Declarative machine learning on MapReduce. In *Proceedings of IEEE 27th International Conference on Data Engineering*, ICDE '11. USENIX Association, 2011.

[84] Matei Zaharia, Mosharaf Chowdhury, Tathagata Das, Ankur Dave, Justin Ma, Murphy McCauley, Michael J. Franklin, Scott Shenker, and Ion Stoica. Resilient distributed datasets: A fault-tolerant abstraction for in-memory cluster computing. In *Proceedings of NSDI '12: 9th USENIX Symposium on Networked Systems Design and Implementation*, NSDI '12, pages 15–28. USENIX Association, 2012.

[85] Xiangrui Meng, Joseph Bradley, Burak Yavuz, Evan Sparks, Shivaram Venkataraman, Davies Liu, Jeremy Freeman, DB Tsai, Manish Amde, Sean Owen, et al. Mllib: Machine learning in apache spark. *The Journal of Machine Learning Research*, 17(1):1235–1241, 2016.

[86] Tim Kraska, Ameet Talwalkar, John Duchi, Rean Griffith, Michael F. Franklin, and Michael Jordan. MLbase: A distributed machine-learning system. In *Proceedings of 6th Biennial Conference on Innovative Data Systems Research*, CIDR '13, 2013.

[87] Yu Xiao, Pieter Simoens, Padmanabhan Pillai, Kiryong Ha, and Mahadev Satyanarayanan. Lowering the barriers to large-scale mobile crowdsensing. In *Proceedings of the 14th Workshop on Mobile Computing Systems and Applications*, page 9. ACM, 2013.

[88] Denzil Ferreira, Vassilis Kostakos, and Anind K Dey. Aware: mobile context instrumentation framework. *Frontiers in ICT*, 2:6, 2015.

[89] Kumaribaba Athukorala, Eemil Lagerspetz, Maria von Kügelgen, Antti Jylhä, Adam J. Oliner, Giulio Jacucci, and Sasu Tarkoma. How Carat Affects User Behavior: Implications for Mobile Battery Awareness Applications. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, New York, NY, USA, 2014. ACM.

[90] Ella Peltonen, Eemil Lagerspetz, Petteri Nurmi, and Sasu Tarkoma. Too big to mail: On the way to publish large-scale mobile analytics data. In *Big Data (Big Data), 2016 IEEE International Conference on*, pages 2374–2377. IEEE, 2016.

[91] José S Marcano Belisario, Jan Jamsek, Kit Huckvale, John O'Donoghue, Cecily P Morrison, and Josip Car. Comparison of self-administered survey questionnaire responses collected using mobile apps versus other methods. *Cochrane Database of Systematic Reviews*, (7), 2015.

[92] Ella Peltonen, Eemil Lagerspetz, Jonatan Hamberg, Abhinav Mehrotra, Mirco Musolesi, Petteri Nurmi, and Sasu Tarkoma. The hidden image of mobile usage: Uncovering the impact of geographic and demographic factors. *Under review.*

[93] François Fleuret. Fast binary feature selection with conditional mutual information. *Journal of Machine Learning Research*, 5:1531–1555, December 2004.

[94] Mark Harman, Yue Jia, and Yuanyuan Zhang. App store mining and analysis: Msr for app stores. In *Proceedings of the 9th IEEE Working Conference on Mining Software Repositories*, MSR '12, pages 108–111, Piscataway, NJ, USA, 2012. IEEE Press.

[95] Hee-Woong Kim, Hyun Lyung Lee, and Jung Eun Son. An exploratory study on the determinants of smartphone app purchase. In *The 11th International DSI and the 16th APDSI Joint Meeting, Taipei, Taiwan*, 2011.

[96] Mark de Reuver, Harry Bouwman, Nico Heerschap, and Hannu Verkasalo. Smartphone Measurement: do People Use Mobile Applications as they Say they do? In *Proc. International Conference on Mobile Business (ICMB)*, 2012.

[97] Lenin Ravindranath, Jitendra Padhye, Sharad Agarwal, Ratul Mahajan, Ian Obermiller, and Shahin Shayandeh. Appinsight: Mobile app performance monitoring in the wild. In *OSDI*, volume 12, pages 107–120, 2012.

[98] C. H. (Kees) van Berkel. Multi-core for mobile phones. In *Proceedings of the Conference on Design, Automation and Test in Europe (DATE '09)*, pages 1260–1265, 2009.

[99] D. Linden and T. B. Reddy. *Handbook of Batteries.* McGrew-Hill Professional, 2001. 3rd edition.

[100] Khai N. Truong, Julie A. Kientz, Timothy Sohn, Alyssa Rosenzweig, Amanda Fonville, and Tim Smith. The design and evaluation of a

task-centered battery interface. In *Proceedings of the12th International Conference on Ubiquitous Computing*, 2010.

[101] Denzil Ferreira, Anind K. Dey, and Vassilis Kostakos. Understanding human-smartphone concerns: A study of battery life. In *Proceedings of the 9th International Conference on Pervasive Computing*, 2011.

[102] Brian Y. Lim, Anind K. Dey, and Daniel Avrahami. Why and why not explanations improve the intelligibility of context-aware intelligent systems. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '09, pages 2119–2128, New York, NY, USA, 2009. ACM.

[103] Bo Yan and Guanling Chen. Appjoy: Personalized mobile application discovery. In *Proceedings of the 9th International Conference on Mobile Systems, Applications, and Services*, MobiSys '11, pages 113–126, New York, NY, USA, 2011. ACM.

[104] Kent Shi and Kamal Ali. Getjar mobile application recommendations with very sparse datasets. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '12, pages 204–212, New York, NY, USA, 2012. ACM.

[105] Neal Lathia, Stephen Hailes, Licia Capra, and Xavier Amatriain. Temporal diversity in recommender systems. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '10, pages 210–217, New York, NY, USA, 2010. ACM.

[106] S. Ronen and O. Shenkar. Clustering countries on attitudinal dimensions: A review and synthesis. *Academy of Management Review*, 10:435–454, 1985.

[107] Viv J. Shackleton and Abbas H. Ali. Work-related values of managers: A test of the hofstede model. *Journal of Cross-Cultural Psychology*, 21:109–118, 1990.

[108] Bradley L Kirkman, Kevin B Lowe, and Cristina B Gibson. A quarter century of culture's consequences: A review of empirical research incorporating hofstede's cultural values framework. *Journal of International Business Studies*, 37(3):285–320, 2006.

[109] I. P. L. Png, B. C. Y. Tan, and Khai-Ling Wee. Dimensions of national culture and corporate adoption of it infrastructure. *IEEE Transactions on Engineering Management*, 48(1):36–45, Feb 2001.

[110] John C. Crotts and Ron Erdmann. Does national culture influence consumers' evaluation of travel services? a test of hofstede's model of cross-cultural differences. *Managing Service Quality: An International Journal*, 10(6):410–419, 2000.

[111] Richard A. Bernardi and Steven T. Guptill. Social desirability response bias, gender, and factors influencing organizational commitment: An international study. *Journal of Business Ethics*, 81(4):797–809, Sep 2008.

[112] Kendall Goodrich and Marieke de Mooij. How 'social' are social media? a cross-cultural comparison of online and offline purchase decision influences. *Journal of Marketing Communications*, 20(1-2):103–116, 2014.

[113] Katharina Reinecke, Minh Khoa Nguyen, Abraham Bernstein, Michael Näf, and Krzysztof Z Gajos. Doodle around the world: Online scheduling behavior reflects cultural differences in time perception and group decision-making. In *Proceedings of the 2013 Conference on Computer Supported Cooperative Work*, pages 45–54. ACM, 2013.

[114] Xuan Lu, Wei Ai, Xuanzhe Liu, Qian Li, Ning Wang, Gang Huang, and Qiaozhu Mei. Learning from the ubiquitous language: An empirical analysis of emoji usage of smartphone users. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '16, pages 770–780, New York, NY, USA, 2016. ACM.

[115] Brendan McSweeney. Hofstede's model of national cultural differences and their consequences: A triumph of faith - a failure of analysis. *Human relations*, 55(1):89–118, 2002.

[116] Janet Vertesi, Silvia Lindtner, and Irina Shklovski. Transnational HCI: Humans, computers, and interactions in transnational contexts. In *CHI'11 Extended Abstracts on Human Factors in Computing Systems*, pages 61–64. ACM, 2011.

Included Publications and Manuscripts