Capturing Daily Student Life by Recognizing Complex Activities Using Smartphones

Christian Meurisch, Artur Gogel, Benedikt Schmidt, Timo Nolle,
Frederik Janssen, Immanuel Schweizer, Max Mühlhäuser
Technische Universität Darmstadt
Hochschulstraße 10, D-64289 Darmstadt, Germany
{meurisch,gogel,schmidt,nolle,janssen,schweizer,max}@tk.tu-darmstadt.de

ABSTRACT

In-depth understanding of student life is essential to proactively support students in their academic educations. However, there is no work that identifies and recognizes a sufficient set of activities to capture a daily student life since complex activity recognition is still challenging. In this paper, we address this issue by recognizing 10 complex student activities such as learning, attending a lecture, or sleeping. We first identify these relevant student activities by conducting a pre-study with 21 students aiming to get an insight into their daily lives. Based on this outcome, we design our sensing applications to collect an appropriate dataset including user-annotated ground-truth data from 163 students over 4 weeks. We investigate different multi-class hierarchical approaches, as well as compare general models against individual models. The results show that our approaches consistently outperform the baseline classifiers and achieve F1-scores of 82.3% for the 1st level, 83.4% and 72.7% for both 2nd levels (study-related and non-study-related activities) on a merged activity set using individual models. The findings offer a novel way to capture a daily student life and can be used to support or guide students in their study.

CCS CONCEPTS

•Human-centered computing → Ubiquitous and mobile computing; •Applied computing → *Life and medical sciences;*

KEYWORDS

student life, user behavior analysis, mobile sensing, activity recognition, heterogeneous data sources, smartphone

ACM Reference format:
Christian Meurisch, Artur Gogel, Benedikt Schmidt, Timo Nolle, Frederik Janssen, Immanuel Schweizer, Max Mühlhäuser. 2017. Capturing Daily Student Life by Recognizing Complex Activities Using Smartphones. In *Proceedings of the 14th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services, Melbourne, VIC, Australia, November 7–10, 2017 (MobiQuitous 2017),* 10 pages.
DOI: 10.1145/3144457.3144472

1 INTRODUCTION

A student life is characterized by many factors like behavior, psychological and mental state or social contacts having a bearing on the student's academic performance. Research works that assess and predict the academic performance (e.g., [40, 41]), mental states or issues (e.g., mood, emotion, or depression [4]) and social contacts [8] already exist. To proactively support students or intervene in daily life an in-depth understanding of student life and human behavior is essential. Activities are the most basic form of observable human behavior. Recognizing activities is, thus, the foundation of many context-aware applications, e.g., personal assistants [26, 29] or anticipatory mobile computing systems [25, 32].

Recognizing simple, physical activities such as walking is well understood [21]. It is, to a certain degree, baked into most mobile operating systems. Detecting more complex activities, e.g., student activities like attending a lecture, which are composed of several simple activities, however, is still a major challenge [24]. Currently, the most extensive assessment of student life using smartphones was conducted by Wang et al. [40]. However, this work is limited to only three behavioral classifiers, namely physical activities, conversation, and sleep detection. Hence, at the moment, there is no work that identifies and sufficiently recognizes a set of student activities to capture an entire day in the life of a student.

In this paper, we address this issue by recognizing 10 complex student activities using multi-class and two-level hierarchical recognition approaches. Since it is unclear which activities are relevant for this task, we first conduct a pre-study with 21 students. We group the resulting activities in *study* and *non-study* related activities which represent the first level of our hierarchical approach. For the second level, we select the 10 most relevant activities, namely attending *lecture*, attending *exercise*, *learning alone*, *collective learning*, *briefing* (denoted as study related activities) as well as *sleeping*, *transition* between places, *working*, *eating* and *leisure* activities (denoted as non-study related activities).

Based on this insight, we design our main study to collect an appropriate user-annotated dataset over four weeks from 163 participants. We automatically collect heterogeneous data from physical and virtual sensors of smartphones as well as online social network information. In addition, participants manually annotate their performed activity to get ground-truth data using a context-triggered interface [28]. To further gain our understanding of student activities and tune the feature extraction process, we give an extended insight into daily student life by analyzing the collected user-annotated dataset.

We investigate different multi-class and hierarchical approaches to achieve high recognition results. The key advantage of using

MobiQuitous 2017, Melbourne, VIC, Australia

ACM 978-1-4503-5368-7/17/11…$15.00
DOI: 10.1145/3144457.3144472

a hierarchical approach is the inherently complexity reduction of our 10-class classification problem, first, into a binary problem for deciding whether an activity is study related or not. Given this context, it simplifies the next recognition of the five detailed activities like learning or sleeping for the specific second level 5-class classifier. Moreover, we compare general models against individual models, where we show a slightly higher performance for individual models. To further optimize our results, we experimented with merged activity classes, which are hard to distinguish with our extracted features (e.g., attending a lecture or exercise group).

In summary, the contributions of this paper are threefold:

- We conduct a pre-study with 21 participants to determine most relevant student activities to capture a student life: attending *lecture*, attending *exercise*, *learning alone*, *collective learning*, *briefing* (study related activities) as well as *sleeping*, *transition* between places, *working*, *eating* and *leisure* activities (non-study related activities).
- Using smartphones, we collect a large user-annotated data set over 4 weeks from 163 students with an extensive set of sensor data from online social networks, physical and virtual sensors. The extensive analysis of the derived dataset yields in-depth insights into daily student life.
- To the best of our knowledge, this is the first work that captures a comprehensive set of up to 10 relevant student activities using a two-level hierarchical recognition approach and individual models as best setup

The remainder of this paper is organized as follows. First, we define the term complex activity. Second, we provide an overview of the related work. Third, we report the conducted pre-study to specify the study design. After, the resulting dataset is described and an insight into student life is given. We describe the extracted features from this collected data set. The paper closes with a presentation of the results, a discussion of them, and conclusion.

2 ACTIVITY TERMINOLOGY

In the following we give a clear definition of the term *complex activity* as we believe that a common understanding of what we mean with this term is a necessary prerequisite. Activity generally refers to something done by an agent, based on the capacity to act. This immediately refers to behavior. Still, activity is not necessarily restricted to mere behavior but has the dimension of "what the doing does" to paraphrase Foucault, including aspects like actor intention, objective effect, perception and cultural practices.

Activity detection in computer science currently focuses on the behavior dimension, using sensors to detect an activity like walking or sitting [21]. Work beyond this exists [24], introducing concepts like composite activities, e.g., activities composed of basic activities. This follows the intuition that recurring and easy to observe behavior forms basic activities which in combination form composite activities. However, recurrence and simplicity of observation highly depend on the used sensor. Walking is simple to detect with a gyroscope and accelerometer but complex with a video – however, the complexity of an activity should not depend on the way it is observed. To avoid a definition of complexity which depends on the observer, we define the term complex activity as a behavior that instantiates a cultural practice. The cultural practice has an intention – for the individual and for the environment – which goes beyond behavior. For example reading is the process of decoding signs (which holds for braille reading as well) which is a basic activity. The process of learning the complex activity is composed of basic activities like reading with the intention of collecting information and creating embodied knowledge as rational capacity.

3 RELATED WORK

Activity recognition is a fundamental technique required for context-aware personalized applications for many emerging computing areas, e.g., pervasive computing or smart environments. Early approaches leveraging wearables were concerned with inferring activities from accelerometer data [1]. Recently, mobile phones feature an increasing amount of sensors [22] providing the perfect platform for activity recognition. Simple activity recognition support is now build into most modern operating systems. State of the art approaches detect simple activities (e.g., standing, walking) with an accuracy of more than 90% relying on accelerometer data only (e.g., [6, 7, 20, 21, 30]). Other approaches are concerned with energy-efficiency to provide support for continuous recognition [44].

Recognizing complex activities, which can be comprised of several simple activities, however, is still in its infancy. Current approaches utilize either *external* or *wearable* sensors [24]. External sensors are most common in controlled environments, e.g., smart homes. Here, they can be embedded in target objects [14], e.g., a mug, or installed into the environment, e.g., cameras. Obviously, the detection of a limited set of activities in these controlled environments is not challenging. However, these methods are bound to certain locations and do not scale.

Most work with a focus on wearable sensors focuses either on dedicated on-body sensors or mobile phones. Activity recognition using dedicated on-body sensors has been shown to work very well for a diverse set of activities [1, 38], e.g., brushing teeth, watching TV, or group activities like volleyball or football. Without dedicated sensors, mobile phones are the most common platform for activity recognition. Thus, a lot of research is concerned with activity recognition using data collected solely via the mobile phone. However, the built-in physical sensors are insufficient for the classification of complex activities. State of the art approaches leverage different types of auxiliary information, e.g., app usage statistics [16] or community context [23]. However, none of them adds more than only a single source of auxiliary information.

In this paper, we utilize multiple heterogenous data sources to detect complex activities using smartphones. In the domain of student activities, complex activities necessary for modeling an entire student life are not well-researched. Wang et al. currently present the most extensive assessment of student life considering three behavioral classifiers relying on smartphones [40]. However, there is no existing work that recognizes a comprehensive set of student activities needed to capture an entire student day (in contrast to, e.g., all-day nurse activity recognition [17]). This work addresses the issue to further enhance existing works (e.g., SmartGPA [41], or [42]) to support students. Assessing human life and behaviors is also one relevant contribution for anticipatory mobile computing systems which guide users and intervene in daily life [32, 35].

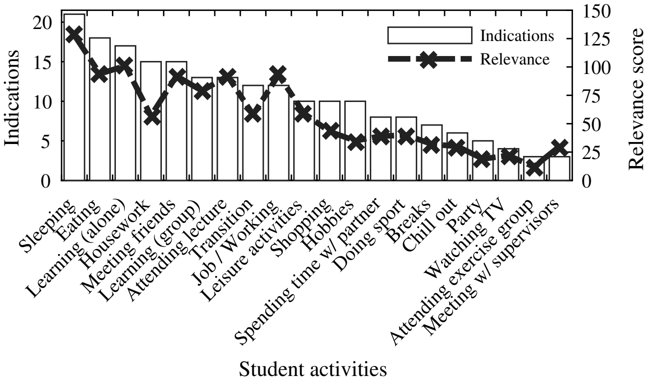


Figure 1: Resulting student activities from the pre-study ordered by their indications

4 PRE-STUDY

Since related work does not give insights which complex activities are relevant to capture an entire student life, we first conducted a pre-study. For that, we interviewed 21 students of our university in a 2-phase survey, where students should (1) *list all considerable daily activities taking longer than 15 minutes that describe their entire day as a student*. After this free text question was filled out, we grouped the reported activities and student should (2) *rank ten out of these 20 clustered activities by their relevance for their day as a student*. We then calculate a relevance score as the sum of reverse ranking positions ranging from ten points for first place indications to one point for position ten indications.

On average, the interviewed students (6 female, 15 male) are 26.3 ± 2.5 years old and pursue their Master in computer science. Figure 1 shows the result of the survey, where student activities are ordered by their indications. Not surprisingly, all students state *sleeping* with the highest relevance for their day, followed by *eating* and *learning alone*. Notably, *housework* is mentioned by over 70% of students, albeit it is marked as not so relevant. On the contrary, *attending lecture* and *working* are relatively relevant for the students. Based on these findings (cf. Fig. 1), we group similar activities together, e.g., *hobbies* and *watching tv* to *leisure activities*. Then we select the most relevant (grouped) activities. Table 1 shows the ten resulting activities consisting of five *study* related and five *non-study* related activities.

5 STUDY DESIGN

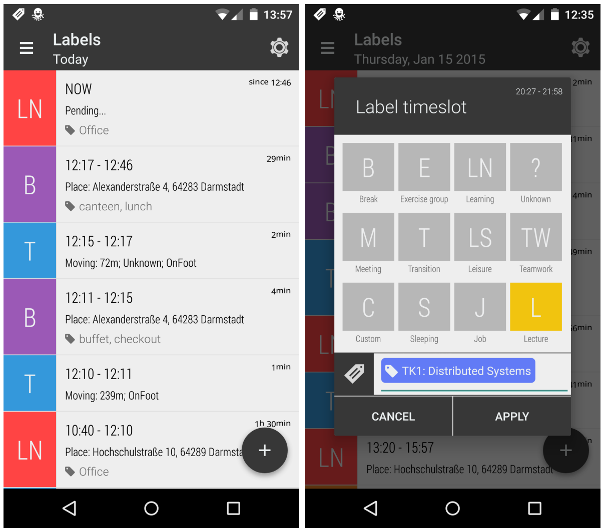
On the basis of our pre-study results, we design the main study to collect a large appropriate dataset for evaluating our hierarchical recognition approach of student activities. In this section, we describe the subject group, the study procedure, and our data collection tools. We also discuss compliance and data quality issues as well as the counter-measures taken.

5.1 Participants

For the study, we recruited 195 students of our university, out of which 163 students completed the four-week study and filled out the corresponding survey about demographical questions (i.e., dropout rate of 16.4%). In the following, we only present results from these 163 students. These students are predominately male students (76% male vs. 24% female) pursuing their Masters of computer science (90% Master's degree vs. 10% Bachelor's degree). The average age of participants was 25.4 years. The participants self-assessed their average technical skills as 3.7 on a 5-point rating scale.

Abbr.	Activity	Description
L	Lecture	Attending a lecture
EX	Exercise	Attending a exercise
LN	Learning$_{ind}$	Individual university-related task
LC	Learning$_{col}$	Collective university-related task
B	Briefing	Meeting w/ a university supervisor
S	Sleeping	State of hibernation w/ closing eyes
T	Transition	Movement between places
W	Working	Pursuing a job
E	Eating	Breaks for food intake
LS	Leisure	Non-study activity, e.g., doing sport

Table 1: Selected activities consisting of 5 study (upper) and 5 non-study activities (lower) to capture a student's life



NOW since 12:46
Pending...
Office

12:17 - 12:46 29min
Place: Alexanderstraße 4, 64283 Darmstadt
canteen, lunch

12:15 - 12:17 2min
Moving: 72m; Unknown; OnFoot

12:11 - 12:15 4min
Place: Alexanderstraße 4, 64283 Darmstadt
buffet, checkout

12:10 - 12:11 1min
Moving: 239m; OnFoot

10:40 - 12:10 1h 30min
Place: Hochschulstraße 10, 64289 Darmsta

20:27 - 21:58
TK1: Distributed Systems
13:20 - 15:57
Place: Hochschulstraße 10, 64289 Darmstadt

(a) Main screen (b) Annotation dialog

Figure 2: User-annotation app

5.2 Study Procedure

The study is embedded in a practical class and consists of a kickoff or training event, and data collection. We briefed all students at a training event on how to use the collection and annotation tools. Here, students did also sign a data release form in person to ensure mutual understanding of the extensive data collection necessary for the study. For the data collection, all students use their own Android smartphones which results in supporting challenges due to several operation system versions and various models from different manufacturers. To ensure a smooth collection, we provide weekly supporting hours to fix individual issues or give advice to users on how to use their smartphone in an appropriate way for obtaining both high data quality and high quantity of data.

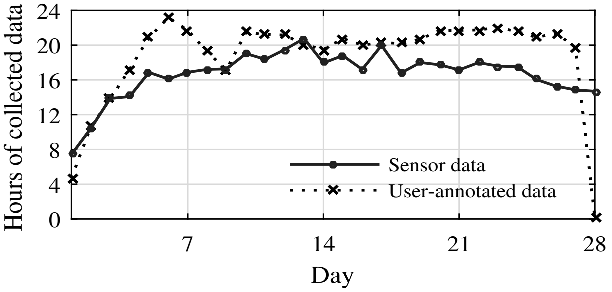


Figure 3: Medial temporal daily coverage of student data collected within the four-week user study

Sensor type	Count	Count$_{filt}$	Indiv. Count$_{filt}$
Physical sensors			
Acceleration	16,491,889	13,320,753	116,685 ± 614,500
Location	25,480,791	19,670,036	173,123 ± 421,949
Light	162,388,462	112,549,018	1,071,761 ± 3,697,698
Interaction sensors			
Ringtone mode	9,665,981	6,662,505	58,469 ± 125,074
Browsing history	291,614	202,101	19,899 ± 84,176
Application usage	6,020,233	3,041,929	27,175 ± 20,711
Virtual sensors			
Google activities	70,277,072	52,816,051	459,839 ± 1,244,175
Total	290,616,042	208,262,393	1,926,954 ± 6,208,286

Table 2: Self-tracking data

5.3 Data Collection

Given the outline of our study in the last sections, the data collection is the most crucial step in studying activity recognition. It is also instrumental to understand the tools given to the students to annotate the data. Therefore, we discuss the two published tools for the data collection in this section: (1) an automatic sensing application, namely *Kraken.me*, which automatically collects sensor data in the background and transmits these data efficiently to a central server [36]; (2) a context-triggered user-annotation tool, namely *Labels*, providing a user interface for the students to specify activity time slots and attach annotations (cf. Fig. 2) [28].

5.3.1 Automatic and Continuous Sensing. The sensing platform provides energy-efficient user tracking across several platforms (i.e., Android, iOS, Windows, Linux) [27, 37]. In this study, we only use the Android-based application which runs as a background service collecting data from the mobile phone and connected services, e.g., social networks [36]. The application also provides user authentication and secure upload of the data to a central server instance, as well as offers data visualization through a web dashboard.

5.3.2 Context-triggered Annotation Interface. The annotation application builds upon the data collected by the sensing platform [28]. The app presents users with an intuitive user-supportive interface to annotate their activities (cf. Tab. 1) for collecting ground truth data. The user is presented with a time view splitting the day into time slots (cf. Fig. 2a). These slots are automatically generated depending on the user's context, but can be merged, deleted, and modified manually. The user can select a single predefined annotation label and add multiple tags per time slot (cf. Fig. 2b). The time slot detection is based on place changes detected by periodically executing a clustering algorithm [11]. In the end, each time slot is associated with exactly one activity label and all 24 hours of a given day should be annotated.

5.4 Compliance and Data Quality

Recognition accuracy not only depends on the number of available instances, but also on the annotation quality. However, manually checking the quality of all annotations is impossible. To ensure high data quality, we implement five quality assurance measures. Two indicators are linked to the quantity of annotated data: (1) *tracked time* which indicates the percentage of time the data collection has remained active, i.e., was not killed by the user or a dead battery; and (2) *average annotation labels per time slot* which reveals the dimension of time slots without any annotations. The remaining three quality indicators are: (3) *tagging* where we assume that users provide better data, as they do spend more time contemplating about their activities; (4) *time to click* is the minimal time that users need to perceive, reflect, choose, and select the correct label (cf. Model Human Processor [5]) while available labels are always presented in a random order to the user; and (5) *time to annotate* addresses the issues that humans are not able to accurately recall activities after one day is elapsed [34]. These quality measures are already displayed to the users during the study [28]. Moreover, we instruct and motivate participants to provide high-quality annotations by additional non-monetary incentives.

6 COLLECTED DATASET

In this section, we describe and characterize the dataset collected within the conducted 4-week study. Moreover, we illustrate and discuss life patterns of our participating students.

6.1 Sensing Data Characterization

Given the aforementioned data collection applications, we collect a large dataset consisting of over 290 million raw sensor values in total from various sensors, social network data, and user-annotated data from 163 participants over four weeks. We gather ground truth data of over 31,000 labels, i.e., user-annotated activity data provided by the participating students. Figure 3 illustrates the medial temporal coverage of both automatic sensing data and user-annotated data. On average, we have data accounting for approximately 70.1% (i.e., 17.1 hours of user-annotated sensor data per day) of the time since the phones have been deployed. The missing data can be due to lazy annotation behavior, especially, for some users in the first days, data corruption, or, mainly, powered-off devices. Especially, the error of powered-off devices is hard to avoid since users often turn off their phone at night or the battery dies. These errors and annotation issues are caught by our quality indicators, and are thus filtered from the dataset.

In the following, we report our resulting dataset contains data from three different providers: automatic sensing data, crawled online social network data, and user-annotated data. In comparison to other community datasets (e.g., MIT Reality Mining [10], Dartmouth CenceMe, Device Analyzer [39], Samsung CS, StudentLife

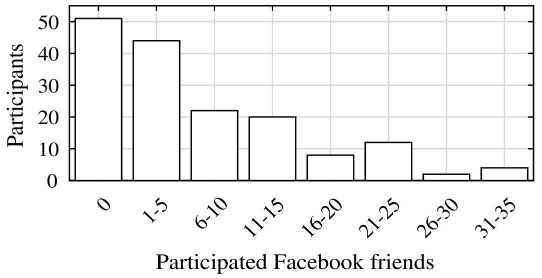


Figure 4: Crawled Facebook friends of participating users

Abbr.	Activity	Count	Count$_{filt}$	Indiv. Count$_{filt}$
L	Lecture	1,867	1,281	123 ± 23
EX	Exercise	379	292	3.9 ± 3.5
LN	Learning$_{ind}$	4,491	3,615	31.7 ± 19.0
LC	Learning$_{col}$	892	661	7.1 ± 5.0
B	Briefing	438	311	3.7 ± 3.5
S	Sleeping	3,393	2,618	22.6 ± 10.8
T	Transition	9,751	6,044	52.1 ± 27.2
W	Working	751	671	10.5 ± 9.0
E	Eating	4,552	3,278	28.5 ± 20.4
LS	Leisure	4,520	3,450	30.3 ± 24.5
Total		31,034	22,221	20.2 ± 13.1

Table 3: Manually user-annotated data

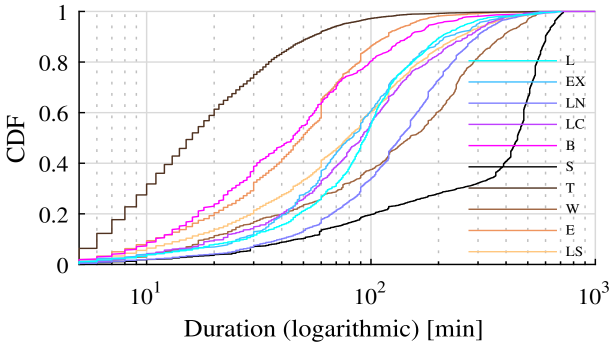


Figure 5: Cumulative distribution function (CDF) for the durations of student activities

dataset [40], or Nokia Lausanna Data Collection Campaign) [43], our dataset is unique in its diversity.

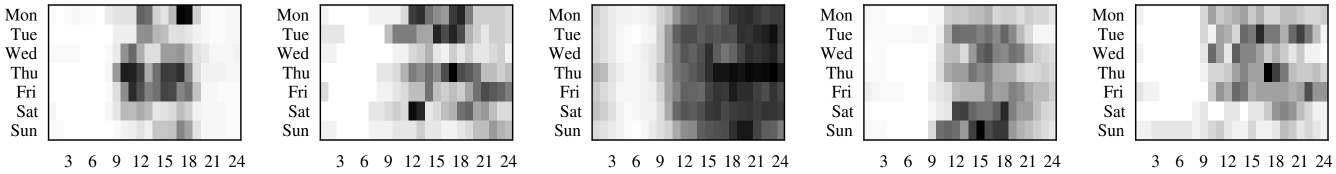
6.1.1 Automatic Sensing Data. Using the previously described sensing application, we automatically collect a large dataset from different sensors in the background. Table 2 lists the total amount of raw sensor data (*count*), the filtered amount ($count_{filt}$), and the filtered amount per participant ($indiv.\ count_{filt}$) applying our quality measures described in the previous section. The data are grouped by their sensor type: *physical* (e.g., location), *interaction* (e.g., app interaction) and *virtual sensors* (e.g., Google activities). The sensing app strives for a balance between high sampling rate and low energy consumption for each sensors [36]. For example, the sampling rate of location sensor depends on the strength of the user's movement, i.e., we reduce the sampling rate if the smartphone is still, while we increase the sampling rate if the smartphone is moving, especially in vehicles. This is reflected by the difference in the amount of values collected by individual sensors.

6.1.2 Social Data. To consider group behaviors and dynamics [18], we need to capture who knows each other. For that, students have given us the permission to crawl their Facebook profile and friends lists. While we could query the social networks on the phone itself, the computational burden would result in additional energy drain. Hence, we extracted the account information on the phone and collected the data in the back-end using the public APIs of the respective social network. We create a social graph based on these friends lists to identify social ties. Figure 4 shows a histogram of the relationship between our participants. More than 50 students do not know any other participating students in Facebook (31.3%); 44 students know at least 1 to 5 other instrumented students (27.0%). Four students even know more than 30 of their participating classmates (2.5%). On average, each participant knows about 7.2 other participants within the study in the social network Facebook. Network density, an indicator for general level of connectedness, is equal to 0.09, i.e., the graph is not densely connected. For friendship networks, however, in which links are more difficult to create, it is entirely sufficient. This data was later leveraged to build the co-located friends network [8].

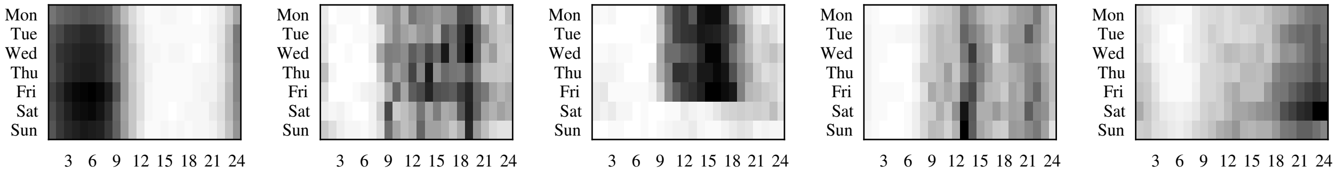
6.1.3 User-annotated Data. A total of $52,609$ time slots were automatically or manually created during the study [28], out of which $31,034$ (59.0%) have been annotated with our ten activity labels by the users. Inspired by [31], we will also consider unlabeled time slots in future research. However, in this paper, we consider $22,221$ (71.6%) labeled time slots remaining after applying our quality measures as discussed in the previous section to filter random annotating users. Table 3 presents the original (*count*), the filtered ($count_{filt}$), and the individual filtered ($indiv.\ count_{filt}$) annotation label distribution for our 10 requested activities. Obviously, some activities are more frequent than others. Frequency depends on the activity type and duration (cf. Fig. 5). For instance, students usually sleep one long time slot per day (M = 375.2min, SD = 209.3min, MD = 450min). Transitions between places occur far more frequently and are much shorter (M = 26.6min, SD = 40.2min, MD = 16min). Thus, we have considerably more labels for transition than for sleeping. It is also notable that the individual count reveals a high standard deviation or a high scatter, which is caused by various behaviors and different daily activities.

6.2 Student Life Characterization

Students engage in different activities during the day and the week. We figure out that a typical student's week consists of 31.4% of study related activities (i.a., learning or attending lectures) and 68.6% of non-study related activities (i.a., sleeping or leisure activities) during the last four weeks of lecture time. Figure 6 shows the collective temporal distribution of these daily activities for all participated students in the study. The darker the slot, the higher the number of students performing the specific activity in the slot. We can observe that students *attend lectures* between 9am to 6pm, with high peaks on Thursday and Friday (cf. Fig. 6a). Confirming [40],



(a) Lecture (*L*) **(b) Exercise (*EX*)** **(c) Learning alone (*LN*)** **(d) Collective learning (*LC*)** **(e) Briefing (*B*)**



(f) Sleeping (*S*) **(g) Transition (*T*)** **(h) Working (*W*)** **(i) Eating (*E*)** **(j) Leisure (*LS*)**

Figure 6: Temporal distribution of student's daily activities revealed from our collected dataset (the intensity reflects the number of participants over the time of day, where *black* is a high population and *white* a low population)

this period can be labeled as *day epoch.* For *attending exercises*, Monday, Tuesday and Thursday are dominant across the week (cf. Fig. 6b). Note that some courses took also place on Saturday during the observed period. However, we see highly entropic behaviors in attending courses through our participants since master students are able to enroll in different courses towards their classmates. In Figure 6c, we see that students *learn* most of the time alone between 9am to midnight independent from the day of week. Even some students learn in the night. When students *learn together*, they meet between 9am to 6pm with a high peak on the weekend (cf. Fig. 6d). Most *briefing* events with the supervisor take place in day epoch during the weekdays (cf. Fig. 6e).

Unsurprisingly, the most dominant non-study activity is *sleeping* (cf. Fig. 6f). Students go to bed at different times, but it stands out that most students sleep between midnight and 9am. The wake up times range from 9am to noon, and the sleep times start between 10pm and midnight. Equal to [40], the switch from evening to night starts roughly at midnight and 12am to 9am is termed as *night epoch.* The remaining period between 6pm and 12am is considered as *evening epoch. Transitions* between places are irregular distributed through the day period, but with clear peak lines at 6pm over all days when students go home (cf. Fig. 6g). In Figure 6h, we see that *working* times for earning money are well-defined in the day epoch during weekdays. Since students have an entropic lifestyle the *eating* times are not well-defined and distributed over the day and evening epochs (cf. Fig. 6i). We notice only trends around 1pm and 9pm. Figure 6j shows the *leisure* activities like socializing, sport, chill out, or shopping activities. We observe that leisure activities are concentrated to the evening epoch, especially on Friday and Saturday evening. Based on this extended insight into student behaviors, we align the following feature extraction process.

7 FEATURE EXTRACTION

Using machine learning methods, we first start with the feature extraction from our collected dataset and build derived values, so-called *features* [2]. Table 4 gives a simplified overview of our 42 extracted features and their respective value ranges. For better clarity, we assign them into so-called *feature groups* consisting of *spatial*, *temporal*, *environmental*, *connectivity*, *movement*, *community*, *interaction*, and *previous activity* features. While spatial, temporal, environmental, and basic movement features are used in a variety of related approaches, community, interaction, previous activity features as well as the combination of all of them are rather unconventional.

Based on the insight that many activities highly correlate with locations, we extract relevant spatial features from the mobile device's location sensor. Since the unprocessed use of coordinates (i.e., latitude and longitude values) provides little benefit for using them as feature, we extract significant *places* of a user from his traces of coordinates using [19] with empirical determined temporal and spatial parameters of $t = 15min$ and $d = 25m$. We then apply heuristics, user-specific metadata and geospatial operations to semantically label the created clusters as *home*, *university*, *work*, *other place*, and *no place.* Since this work targets detecting student activities, we characterize the *university place* more precisely by determining stays in the main building of computer science, where all lectures and exercises take place, the library, the canteen, the remaining campus, and out of the campus. We use the most dominant place as feature within a time slot. We further calculate the average *accuracy* of location values provided by the device's location sensor. Depending on the available location provider, the collected location values show different accuracies, i.e., if the user is indoor, his mobile device may have access to WiFi or cellular network which provides a lower localization accuracy than GPS having outdoor available. As last spatial features, we determine the staying *coverage* for each place within a given time slot.

Temporal features reflect the temporal dimension of an activity. This follows the insight that many activities correlate with time. Examples are sleeping at night, eating at lunchtime or attending a lecture at a specified time. To capture this correlation the following features are extracted: a distinction of *weekend* and *weekday*, the *day of week*, the *start* and *end hour* as well as the *duration* of each labeled activity. For a more detailed view, we split a day into

Feature group	Feature	Value range
Spatial	Place	{home, university, work, other or no place}
	Place$_{university}$	{main building, library, canteen, campus, none}
	Accuracy$_{location}$	$\{0 - \infty\}$ [m]
	Coverage$_{for\ each\ place}$	$\{0.0 - 100.0\}$ [%]
Temporal	Weekend	{false, true}
	Day of week	{Mo,Tu,We,Th,Fr,Sa,Su}
	Hour of day$_{start/end}$	$\{0 - 23\}$ [h]
	Time slot$_{start/end}$	$\{0 - 95\}$ (15-min split)
	Duration	$\{0 - \infty\}$ [min]
	Midnight overlapping	{false, true}
Environment	Luminance *min/max/avg/mdn/std*	$\{0 - \infty\}$ [lux]
Connectivity	Cell./Wifi available	$\{0.0 - 100.0\}$ [%]
	Cell./Wifi connected	$\{0.0 - 100.0\}$ [%]
Movement	Physical activity *still/foot/bicycle/vehicle*	$\{0.0 - 100.0\}$ [%]
	Acceleration intensity *min/max/avg/mdn/std*	$\{-\infty - +\infty\}$
	Distance covered	$\{0 - \infty\}$ [m]
Community	People$_{nearby}$	$\{0 - \infty\}$
	Friends$_{nearby}$	$\{0 - \infty\}$
	Roommates$_{nearby}$	$\{0 - \infty\}$
Interaction	Smartphone usage	$\{0.0 - 100.0\}$ [%]
	App interactions	$\{0.0 - \infty\}$ [1/min]
	URL requests	$\{0 - \infty\}$
	Ringtone mode (*silent*)	$\{0.0 - 100.0\}$ [%]
Activity	Previous and second-last activity	{L,EX,LN,LC,B, S,T,W,E,LS}

Table 4: Extracted features and their range of values grouped by meaningful *feature groups* for a better overview

15-minute time slots and extract features regarding *start* and *end* time slots of an activity. Moreover, we detect whether an activity overlaps midnight or not, i.e., if an activity such as sleeping is performed on two consecutive days (e.g., from Monday 9pm to Tuesday 7am).

Assuming a relation between light conditions and activities (e.g., sleeping) the *minimum*, *maximum*, *average*, *median* and *standard deviation* values of the physical luminance sensor for the given time slot are extracted as feature.

For some activities such as transition, stays at home or at the university, connectivity features could be interesting. For that, we extract the availability of networks and the coverage of the mobile device connected to these networks (i.e., *cellular* and *WiFi networks*) within an activity time slot. For instance, if the student is at home, we assume his mobile device is mainly connected to his wireless home network. While the student is on the way, we assume his mobile device is mainly connected to the mobile cellular network of his provider.

Since complex activities are a complex compound of simple activities, *physical activities* may be a relevant indicator for them. For instance, if a user sleeps, his mobile device is motionless or still. In contrast, if a user is on the way, the accelerometer sensor of his accompanying device detects these changes. Using the Google Play Activity Recognition API we extract *still*, on the *foot*, *bicycle*, and in *vehicle* features. We further calculate the *minimum*, *maximum*, *average*, *median* and *standard deviation* intensity values of the mobile device's acceleration. The last movement feature represents the *covered distance* by the user within a given time slot.

Community features represent information about other instrumented people co-located with the user. In [23], the community feature shows an improvement in activity recognition, once the activity of related people are considered. We extend this idea as we do not only consider the *community of friends*, but also the *community of roommates* and the *community of people nearby* [8]. We get the information if students are friends from Facebook, and if students are roommates (e.g., dorm or living community) by using heuristics of temporal (e.g., nights) and spatial (e.g., at home) proximities. These community features could be interesting, e.g., for distinguishing between attending a lecture or a exercise group, which differ in the number of participating students.

Interactions with the mobile device could also indicate complex activities, e.g., people play with their smartphone more frequently by having a break than working in a job. To capture these interactions, we extract the *smartphone usage* coverage within a given time slot, the *app interaction* frequency, and the number of *URL requests* in the browser app. We also extract information about the *ringtone mode* of a mobile device with two states of silent and loud, i.e., we capture if the user sets his device in silent mode within a given time slot, e.g., when attending a meeting or a lecture.

Since the daily life of humans is shaped by activity patterns (e.g., sleeping > transition from home to university > attending a lecture) [33], we consider the user's activity history by extracting the *previous* and *second-last activities* immediately performed before the given activity.

All considered sources can typically be collected with a mobile phone. Since the data is composed of activity labels as ground-truth with respective sensor data, we generate features for each of these activity labels or annotated time slots (aka *instances*). In the following, we will use these resulting 22,221 instances consisting of 42 features and one ground-truth activity class label.

8 RESULTS

In this section, we report and discuss the results of applying conventional multi-class and 2-level hierarchical approaches on the collected dataset. Moreover, we train *general* and *individuals models* to compare the performances of them.

As model validation technique, we apply *10-fold cross-validation* to assess how the results generalize to an independent dataset [2], i.e., we train the models on nine folds and test on the remaining fold. We repeat this method for all ten folds, such that each fold once served as test set. The final result is averaged across all iterations. We evaluate using F_1*-score* (harmonic mean of precision and recall):

$$F_1 = \frac{2 \times precision \times recall}{precision + recall}$$

$$precision = \frac{TP}{TP + FP} \qquad recall = \frac{TP}{TP + FN}$$

where TP, FP and FN are counts of true positives, false positives and false negatives [31]. To assess the quality of our classifiers

Activities		**10-class classifier**				**Binary classifier** (1^{st})				**5-class classifier** (2^{nd})				**5-class classifier** (2^{nd})			
		General		Individual		General		Individual		General		Individual		General		Individual	
1^{st} *level*	2^{nd} *level*	F_1	*ROC*	F_1	*ROC*	F_1	*ROC*	F_1	*ROC*	F_1	*ROC*	F_1	*ROC*	F_1	*ROC*	F_1	*ROC*
Study	**L**	56.8	92.9	35.3	77.7	61.0	88.1	55.9	87.0	63.6	90.1	44.1	74.2	–			
	EX	5.7	79.3	3.7	30.4					7.1	70.1	4.5	25.0				
	LN	58.1	87.8	50.1	84.2					86.1	89.2	79.4	83.1				
	LC	15.0	85.7	13.2	59.3					22.0	77.3	20.8	53.4				
	B	4.1	73.9	0.6	31.2					11.5	85.6	6.4	32.5				
Non-study	**S**	81.9	97.5	83.9	96.3	88.8	88.1	88.8	87.0	–				85.4	98.0	87.0	97.3
	T	79.6	96.0	78.8	95.1									84.1	95.4	82.4	94.5
	W	52.3	91.1	26.6	45.7									62.7	93.9	29.2	46.8
	E	50.2	84.2	42.4	80.5									59.7	86.0	51.5	83.9
	LS	39.6	80.7	38.3	77.5									57.1	85.8	51.7	82.5
Baseline (*ZeroR*)		10.5	50.0	15.6	44.6	61.0	50.0	61.5	46.5	47.7	49.9	47.4	33.9	18.6	50.0	25.9	45.1
Weighted average		**59.5**	**89.4**	**63.3**	**88.5**	**81.2**	**88.1**	**81.8**	**87.0**	**68.5**	**87.1**	**68.3**	**80.5**	**72.1**	**91.6**	**75.7**	**91.3**

Table 5: Classification results for *10 activities* covering a student daily life using a conventional approach of 10-class classification and a hierarchical approach with a binary classifier on the 1^{st} level and one 5-class classifier each for study and non-study related activities on the 2^{nd} level by training general and individual models

or measure the aggregated classification performance, we use the *area under the ROC curve* metric [12]. The ROC curve shows the true positive rate (sensitivity) against the false positive rate (specificity) [9].

For automatic classification, we use *WEKA* as data mining software [15]. As baseline classifier for our activity recognition, we experimented with a range of supervised classifiers including SVM, Random Forest, LMT, and Kstar. Random Forest - an ensemble learning algorithm of decision trees for classification [3] - consistently achieved the best results. Therefore, we will report following results using *Random forest with 100 trees as base classifier.* As baseline classifier, we use *ZeroR*, which always predicts the majority class.

8.1 Initial Activity Set (10-class)

Table 5 shows the classification results for both the conventional approach facing with a 10-class classification problem and the hierarchical approach dividing the problem into two sub-problems: (1) binary classification which decides whether an activity is study related or not; and (2) two 5-class classifications for further distinguishing five activities (e.g., learning, sleeping) in each of the two branches. We further compare the general model against individual models. However, the individual models are only trained on a filtered set of 116 users provided us enough data to have a sufficient number of training and test instances, while the general model considers all instances from 163 students.

As seen in Table 5, the 10-class classifier achieves an overall F_1-score of 59.5% for the general model and 63.3 for the individual models, while the baseline is merely 10.5% or 15.6%. Especially three study-related activities, namely *briefing* (4.1% or 0.6%), attending *exercise* (5.7% or 3.7%), and *collective learning* (15.0% or 13.2%), cannot be classified accurately due to their relatively small count of instances. While activities with a higher number of instances perform much better, e.g., up to 81.9% or 83.9% for *sleeping*. However, remind that the classifier is facing a complex 10-class classification problem with imbalanced instances, especially the distribution of study and non-study related instances is highly skewed (6,160 *vs* 16,061). Testing over- and undersampling as well as one-vs-all and one-vs-one schemes does not result in a performance gain [13].

To overcome these issues and to simplify the classification problem, we first distinguish the study (*SA*) and non-study context (*NA*) using a binary classifier. This way, we achieve an overall F_1-score of 81.2% (general model) or 81.8% (individual models) while the baseline is 61.0% or 61.5% (cf. Tab. 5). We can observe that the classifiers on the second level achieve a high performance of 88.8% for *non-study* activities but only 61.0% or 55.9% for *study* related ones. However, as expected, the binary classifier is able to accurately separate both contexts. Moreover, the ROC curve of our classifier shows good aggregated classification performances (87.0% – 88.1%), which reflects the quality of the classifier.

Given that the study or non-study context depends on the use case or the application, we then apply a 5-class classifier to further distinguish five detailed activities. Considering study activities, the classifier is able to improve the activity recognition performance of all activities, e.g., up to 86.1% F_1-score for *learning alone* (cf. Tab. 5). Attending *lecture* can also be recognized accurately with 63.6%. However, *collective learning*, *briefing*, and attending a *exercise* remains at a poor level (having F_1-scores between 4.5% and 22.0%). Nevertheless, the overall F_1 performance of 68.5% (general model) or 68.3% (individual models) is at a higher level compared to the baseline of 47.7% or 47.4% (5-class classification problem).

The same holds for the other branch (non-study activities). The overall performance is 72.1% (general model) or 75.7% (individual models) while the baseline is only 18.6% or 25.9% (cf. Tab. 5). Thus, the 5-class classifier clearly outperforms the baseline classifier and very accurately detects non-study activities such as *transition* between places with over 82% and *sleeping* with over 85%. The three remaining activities (namely *working*, *eating*, and *leisure*) can be detected with performances in the region of 50 – 60%. It is notable that the ROC curve is over 93%, which means that our classifier has excellent aggregated classification performance.

Activities		**7-class classifier**				**Binary classifier (1^{st})**				**3-class classifier (2^{nd})**				**4-class classifier (2^{nd})**			
		General		Individual		General		Individual		General		Individual		General		Individual	
1^{st} level	*2^{nd} level*	F_1	*ROC*	F_1	*ROC*	F_1	*ROC*	F_1	*ROC*	F_1	*ROC*	F_1	*ROC*	F_1	*ROC*	F_1	*ROC*
Study	**L+EX**	57.0	91.0	36.3	79.7	62.2	88.1	57.2	86.8	66.1	87.6	48.3	74.9	–			
	LN	55.4	87.9	48.4	84.5					87.1	89.3	80.1	84.4				
	LC+B	9.1	81.6	11.8	64.9					23.8	79.7	26.8	62.1				
Non-study	**S**	82.5	97.6	84.3	96.4	89.0	88.1	89.0	86.8	–				85.7	98.0	86.9	97.2
	T	82.0	96.1	79.5	95.3									85.1	95.4	81.9	94.5
	W	52.8	91.4	24.4	46.1									58.2	93.8	27.5	46.9
	LS+E	65.1	84.0	63.3	83.1									80.8	91.3	79.1	90.7
Baseline (*ZeroR*)		15.3	50.0	20.3	45.6	61.0	50.0	61.5	46.5	47.7	49.9	49.3	35.7	26.6	50.0	34.4	45.7
Weighted average		**66.6**	**89.9**	**68.6**	**89.3**	**81.7**	**88.1**	**82.3**	**86.8**	**72.6**	**87.4**	**72.7**	**82.4**	**82.2**	**94.0**	**83.4**	**93.5**

Table 6: Classification results for *7 merged activities* covering a student daily life using a conventional approach of 7-class classification and a hierarchical approach with a binary classifier on the 1^{st} level and one 3-class classifier for study and one 4-class classifier for non-study related activities on the 2^{nd} level by training general and individual models

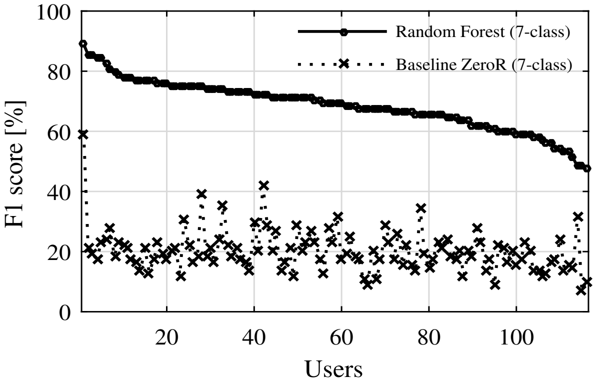


Figure 7: F_1-score over a filtered set of 116 students provided enough data for training individual models (7 activities)

8.2 Merged Activity Set (7-class)

Since some activities like attending a *lecture* or attending a *exercise group* are hard to distinguish with our extracted features, we experiment with merging these activities together to get (1) more instances per class, and (2) better recognition results for a subset of activities. More precisely, we merge lecture (L) and exercise group (EX), collective learning (LC) and briefing or meeting (B) as well as leisure (LS) and eating (E) together resulting in 7 activities in total.

Table 6 shows the classification results for our 7 merged activities: the 7-class classifier achieves a weighted average F_1-score of 66.6% for the general model and 68.6% for individual models and has an excellent ROC value of over 89%. Considering the hierarchical approach, we see that the average performance values are over 81% for the 1st level. Especially, the non-study related activities can be recognized by outstanding 89% F_1-score, while study activities are only recognized by 62.2% or 57.2%. The classifiers on the 2nd level also show very good F_1-scores of 72 – 83%. Especially, learning, sleeping, transition, and newly the merged activity class of leisure and eating can be detected with F_1-scores over 80%. All classifiers still outperform the baseline classifiers distinctly. Reducing the classification problem from 10 to 7 classes, we achieve good recognition results for each class except for the merged class *LC+B* (collective learning and briefing), which remains quite challenging ($F1 \sim 10\%$) with our features.

Figure 7 gives an detailed view of the performances of our individual trained models per student. We see that all individual models easily outperform the baseline classifiers trained on the same instances. Considering a 7-class classification problem, it is outstanding that the best users achieve a F_1-score over 90%, while the worst users are lower than 50%. On average, the individual models achieve a F_1-score of 68.6% with a very good ROC of 89.3%.

In summary, our classifiers consistently outperform the baselines and the training of individual models results in a slightly higher performance than the training of one general model. As seen, the hierarchical approach strongly reduces the complexity of the classification problem, i.e., this approach makes decisions on different recognition levels. The distinction between both branches, study and non-study activities, can already be sufficient for some use cases while the recognition of fine-grained activities is necessary for some other use cases.

9 CONCLUSION

In this paper, we have identified and recognized relevant complex activities such as learning, attending a lecture, or sleeping to capture a daily student life. For that, we first conducted a pre-study with 21 students aiming to get an insight into their daily lives. Based on these findings, we designed our main study and collected a large user-annotated dataset from 163 students over 4 weeks. Our data analysis yielded a number of insight into student life to better understand student behaviors. We investigated different multi-class and hierarchical recognition approaches as well as compare general models against individual models. The results show that our approaches consistently outperform the baseline classifiers and achieve F_1-scores of 82.3% for the 1st level and 72.7% or 83.4% for the 2nd levels on a merged activity set using individual models. Researchers from areas such as anticipatory mobile computing or existing works can benefit from our results by utilizing previously inaccessible detailed knowledge about student lives. Further research can also use our findings to support or guide students in their study.

ACKNOWLEDGMENTS

This work has been co-funded by the German Federal Ministry of Education and Research (BMBF) Software Campus project "PersonalAssistant" [01IS12054] and by the LOEWE initiative (Hessen, Germany) within the NICER project.

REFERENCES

[1] Ling Bao and Stephen S Intille. 2004. Activity Recognition from User-annotated Acceleration Data. In *Pervasive Computing*. Springer, 1–17.

[2] Christopher M Bishop et al. 2006. *Pattern Recognition and Machine Learning*. Vol. 4. Springer New York.

[3] Leo Breiman. 2001. Random Forests. *Machine learning* 45, 1 (2001), 5–32.

[4] Luca Canzian and Mirco Musolesi. 2015. Trajectories of Depression: Unobtrusive Monitoring of Depressive States by Means of Smartphone Mobility Traces Analysis. In *17th International Conference on Ubiquitous Computing (UbiComp'15)*. ACM, 1293–1304.

[5] Stuart K Card, Thomas P Moran, and Allen Newell. 1986. The Model Human Processor: An Engineering Model of Human Performance. *Handbook of Human Perception* 2 (1986).

[6] Vivek Chandel, Anirban Dutta Choudhury, Avik Ghose, and Chirabrata Bhaumik. 2013. AcTrak-unobtrusive activity detection and step counting using smartphones. In *10th International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services (MobiQuitous'13)*. Springer, 447–459.

[7] Tanzeem Choudhury, Sunny Consolvo, Beverly Harrison, Jeffrey Hightower, Anthony LaMarca, Louis LeGrand, Ali Rahimi, Adam Rea, G Bordello, Bruce Hemingway, et al. 2008. The Mobile Sensing Platform: An Embedded Activity Recognition System. *IEEE Pervasive Computing* 7, 2 (2008).

[8] Justin Cranshaw, Eran Toch, Jason Hong, Aniket Kittur, and Norman Sadeh. 2010. Bridging the Gap Between Physical Location and Online Social Networks. In *12th International Conference on Ubiquitous Computing (UbiComp'10)*. ACM, 119–128.

[9] Jesse Davis and Mark Goadrich. 2006. The Relationship Between Precision-Recall and ROC Curves. In *23rd International Conference on Machine Learning (ICML'06)*. ACM, 233–240.

[10] Nathan Eagle and Alex Pentland. 2006. Reality Mining: Sensing Complex Social Systems. *Personal and Ubiquitous Computing* 10, 4 (2006), 255–268.

[11] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. 1996. A Density-based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In *2nd International Conference on Knowledge Discovery and Data Mining (KDD'96)*, Vol. 96. 226–231.

[12] Cesar Ferri, José Hernández-Orallo, and Peter A Flach. 2011. A Coherent Interpretation of AUC as a Measure of Aggregated Classification Performance. In *28th International Conference on Machine Learning (ICML'11)*. 657–664.

[13] Mikel Galar, Alberto Fernández, Edurne Barrenechea, Humberto Bustince, and Francisco Herrera. 2011. An Overview of Ensemble Methods for Binary Classifiers in Multi-class Problems: Experimental Study on One-vs-one and One-vs-all Schemes. *Pattern Recognition* 44, 8 (2011), 1761–1776.

[14] Dawud Gordon, Markus Scholz, and Michael Beigl. 2014. Group Activity Recognition Using Belief Propagation for Wearable Devices. In *18th International Symposium on Wearable Computers (ISWC'14)*. ACM, 3–10.

[15] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H Witten. 2009. The WEKA Data Mining Software: an Update. *SIGKDD explorations newsletter* 11, 1 (2009), 10–18.

[16] Jon C Hammer and Tingxin Yan. 2014. Exploiting Usage Statistics for Energy-efficient Logical Status Inference on Mobile Phones. In *18th International Symposium on Wearable Computers (ISWC'14)*. ACM, 35–42.

[17] Sozo Inoue, Naonori Ueda, Yasunobu Nohara, and Naoki Nakashima. 2015. Mobile Activity Recognition for a Whole Day: Recognizing Real Nursing Activities With Big Dataset. In *17th International Conference on Ubiquitous Computing (UbiComp'15)*. ACM, 1269–1280.

[18] Kasthuri Jayarajah, Youngki Lee, Archan Misra, and Rajesh Krishna Balan. 2015. Need Accurate User Behaviour?: Pay Attention to Groups!. In *17th International Conference on Ubiquitous Computing (UbiComp'15)*. ACM, 855–866.

[19] Jong Hee Kang, William Welbourne, Benjamin Stewart, and Gaetano Borriello. 2005. Extracting Places from Traces of Locations. *ACM SIGMOBILE Mobile Computing and Communications Review* 9, 3 (2005), 58–68.

[20] S. Khalifa, M. Hassan, and A. Seneviratne. 2015. Pervasive Self-powered Human Activity Recognition Without the Accelerometer. In *13th International Conference on Pervasive Computing and Communications (PerCom'15)*. IEEE, 79–86.

[21] Jennifer R Kwapisz, Gary M Weiss, and Samuel A Moore. 2011. Activity Recognition using Cell Phone Accelerometers. *SIGKDD Explorations Newsletter* 12, 2 (2011), 74–82.

[22] Nicholas D Lane, Emiliano Miluzzo, Hong Lu, Daniel Peebles, Tanzeem Choudhury, and Andrew T Campbell. 2010. A Survey of Mobile Phone Sensing. *Communications Magazine* 48, 9 (2010), 140–150.

[23] Nicholas D Lane, Li Pengyu, Lin Zhou, and Feng Zhao. 2014. Connecting Personal-scale Sensing and Networked Community Behavior to Infer Human Activities. In *16th International Conference on Ubiquitous Computing (UbiComp'14)*. ACM, 595–606.

[24] Oscar D Lara and Miguel A Labrador. 2013. A Survey on Human Activity Recognition using Wearable Sensors. *Communications Surveys & Tutorials* 15, 3 (2013), 1192–1209.

[25] Christian Meurisch. 2016. Intelligent Personal Guidance of Human Behavior Utilizing Anticipatory Models. In *UbiComp'16: Adjuncts*. ACM, 441–445.

[26] Christian Meurisch, Maria-Dorina Ionescu, Benedikt Schmidt, and Max Mühlhäuser. 2017. Reference Model of Next-generation Digital Personal Assistant: Integrating Proactive Behavior. In *UbiComp'17: Adjuncts*. ACM, 149–152.

[27] Christian Meurisch, Bennet Jeutter, Wladimir Schmidt, Nickolas Gündling, Benedikt Schmidt, Fabian Herrlich, and Max Mühlhäuser. 2017. An Extensible Pervasive Platform for Large-scale Anticipatory Mobile Computing. In *41st Annual Computer Software and Applications Conference (COMPSAC'17)*, Vol. 1. IEEE, 459–464.

[28] Christian Meurisch, Benedikt Schmidt, Michael Scholz, Immanuel Schweizer, and Max Mühlhäuser. 2015. Labels - Quantified Self App for Human Activity Sensing. In *UbiComp'15: Adjunct Publication*. ACM.

[29] Christian Meurisch, Stefan Wullkotte, Stefan Niemczyk, Florian Kohnhäuser, Max Mühlhäuser, et al. 2017. NICER911: Ad-hoc Communication and Emergency Services Using Networking Smartphones and Wireless Home Routers. In *18th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc'17*. ACM, 33.

[30] Emiliano Miluzzo, Nicholas D Lane, Kristóf Fodor, Ronald Peterson, Hong Lu, Mirco Musolesi, Shane B Eisenman, Xiao Zheng, and Andrew T Campbell. 2008. Sensing Meets Mobile Social Networks: The Design, Implementation and Evaluation of the CenceMe Application. In *6th International Conference on Embedded Network Sensor Systems (SenSys'08)*. ACM, 337–350.

[31] Le T Nguyen, Ming Zeng, Patrick Tague, and Joy Zhang. 2015. Recognizing New Activities with Limited Training Data. In *19th International Symposium on Wearable Computers (ISWC'15)*. ACM, 67–74.

[32] Veljko Pejovic and Mirco Musolesi. 2015. Anticipatory Mobile Computing: A Survey of the State of the Art and Research Challenges. *Comput. Surveys* 47, 3 (2015), 47.

[33] Fabio Pianese, Xueli An, Fahim Kawsar, and Hiroki Ishizuka. 2013. Discovering and Predicting User Routines by Differential Analysis of Social Network Traces. In *14th International Symposium and Workshops on a World of Wireless, Mobile and Multimedia Networks (WoWMoM'13)*. IEEE, 1–9.

[34] Gertrude Raffel. 1934. The Effect of Recall on Forgetting. *Journal of experimental psychology* 17, 6 (1934), 828.

[35] Benedikt Schmidt, Sebastian Benchea, Rüdiger Eichin, and Christian Meurisch. 2015. Fitness Tracker or Digital Personal Coach: How to Personalize Training. In *UbiComp'15: Adjunct Publication*. ACM, 1063–1067.

[36] Immanuel Schweizer, Roman Bärtl, Benedikt Schmidt, Fabian Kaup, and Max Mühlhäuser. 2014. Kraken.me Mobile: The Energy Footprint of Mobile Tracking. In *MobiCASE'14*. IEEE, 82–89.

[37] Immanuel Schweizer and Benedikt Schmidt. 2014. Kraken.me: Multi-device User Tracking Suite. In *UbiComp'14: Adjunct Publication*. ACM, 853–862.

[38] Yujin Tang and Chihiro Ono. 2016. Detecting Activities of Daily Living from Low Frequency Power Consumption Data. In *13th EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services (MobiQuitous'16)*. 38–46.

[39] Daniel T Wagner, Andrew Colin Rice, and Alastair R Beresford. 2013. Device Analyzer: Understanding Smartphone Usage. In *10th International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services (MobiQuitous'13)*. Springer, 195–208.

[40] Rui Wang, Fanglin Chen, Zhenyu Chen, Tianxing Li, Gabriella Harari, Stefanie Tignor, Xia Zhou, Dror Ben-Zeev, and Andrew T Campbell. 2014. Studentlife: Assessing Mental Health, Academic Performance and Behavioral Trends of College Students using Smartphones. In *16th International Conference on Ubiquitous Computing (UbiComp'14)*. ACM, 3–14.

[41] Rui Wang, Gabriella Harari, Peilin Hao, Xia Zhou, and Andrew T Campbell. 2015. SmartGPA: How Smartphones Can Assess and Predict Academic Performance of College Students. In *17th International Conference on Ubiquitous Computing (UbiComp'15)*. ACM, 295–306.

[42] Jun-Ichiro Watanabe, Kazuo Yano, and Saki Matsuda. 2013. Relationship Between Physical Behaviors of Students and Their Scholastic Performance. In *10th International Conference on Ubiquitous Intelligence and Computing and 10th International Conference on Autonomic and Trusted Computing (UIC/ATC'13)*. IEEE, 170–177.

[43] Evan Welbourne and Emmanuel Munguia Tapia. 2014. CrowdSignals: A Call to Crowdfund the Community's Largest Mobile Dataset. In *16th International Conference on Ubiquitous Computing (UbiComp'14): Adjuncts*. ACM, 873–877.

[44] Zhixian Yan, Vigneshwaran Subbaraju, Dipanjan Chakraborty, Archan Misra, and Karl Aberer. 2012. Energy-efficient Continuous Activity Recognition on Mobile Phones: An Activity-adaptive Approach. In *16th International Symposium on Wearable Computers (ISWC'12)*. ACM, 17–24.