A MATTER OF ANNOTATION: AN EMPIRICAL STUDY ON IN SITU AND SELF-RECALL ACTIVITY ANNOTATIONS FROM WEARABLE SENSORS

Alexander Hoelzemann
Ubiquitous Computing
University of Siegen
Siegen
alexander.hoelzemann@uni-siegen.de

Kristof Van Laerhoven
Ubiquitous Computing
University of Siegen
Siegen
kvl@eti.uni-siegen.de

ABSTRACT

Research into the detection of human activities from wearable sensors is a highly active field, benefiting numerous applications, from ambulatory monitoring of healthcare patients via fitness coaching to streamlining manual work processes. We present an empirical study that compares 4 different commonly used annotation methods utilized in user studies that focus on in-the-wild data. These methods can be grouped in user-driven, in situ annotations - which are performed before or during the activity is recorded - and recall methods - where participants annotate their data in hindsight at the end of the day. Our study illustrates that different labeling methodologies directly impact the annotations' quality, as well as the capabilities of a deep learning classifier trained with the data respectively. We noticed that in situ methods produce less but more precise labels than recall methods. Furthermore, we combined an activity diary with a visualization tool that enables the participant to inspect and label their activity data. Due to the introduction of such a tool were able to decrease missing annotations and increase the annotation consistency, and therefore the F1-score of the deep learning model by up to 8% (ranging between 82.1 and 90.4 % F1-score). Furthermore, we discuss the advantages and disadvantages of the methods compared in our study, the biases they may could introduce and the consequences of their usage on human activity recognition studies and as well as possible solutions.

***K*eywords** Human-centered computing · ubiquitous computing · user-driven study · annotations

1 Introduction

Sensor-based activity recognition is one of the research fields of Pervasive Computing developed with enormous speed and success by industry and science and influencing medicine, sports, industry, and therefore the daily lives of many people. However, current available smart devices are mostly capable of detecting periodic activities like simple locomotions. In order to recognize more complex activities a multimodal sensor input, such as [1], and more complex recognition models are needed. Many of the published datasets are made in controlled laboratory environments. Such data does not have the same characteristics and patterns as data recorded in-the-wild. Data that belongs to similar classes but is recorded in an uncontrolled versus controlled environment can differ significantly since it contains more contextual information [2]. Furthermore, study participants tend to control their movements more while being monitored [3]. The recording of long-term and real-world data is a tedious, time-consuming, and therefore a non-trivial task. Researchers have various motivations to record such datasets but the technical hurdles are still high and problems during the annotation process occur regularly. With regards to Human Activity Recognition (HAR), recording a long-term dataset always presents the researcher with the problem of developing a methodology that, on one hand, allows precise labels and, on the other hand, does not unnecessarily burden or disturb the study participants. Relying only on self-recall methods, like writing an activity diary, e.g. [4], can result in imprecise time indications that do not necessarily correspond to the actual time periods of an activity. Such incorrectly or noisy labeled data [5] leads a trained

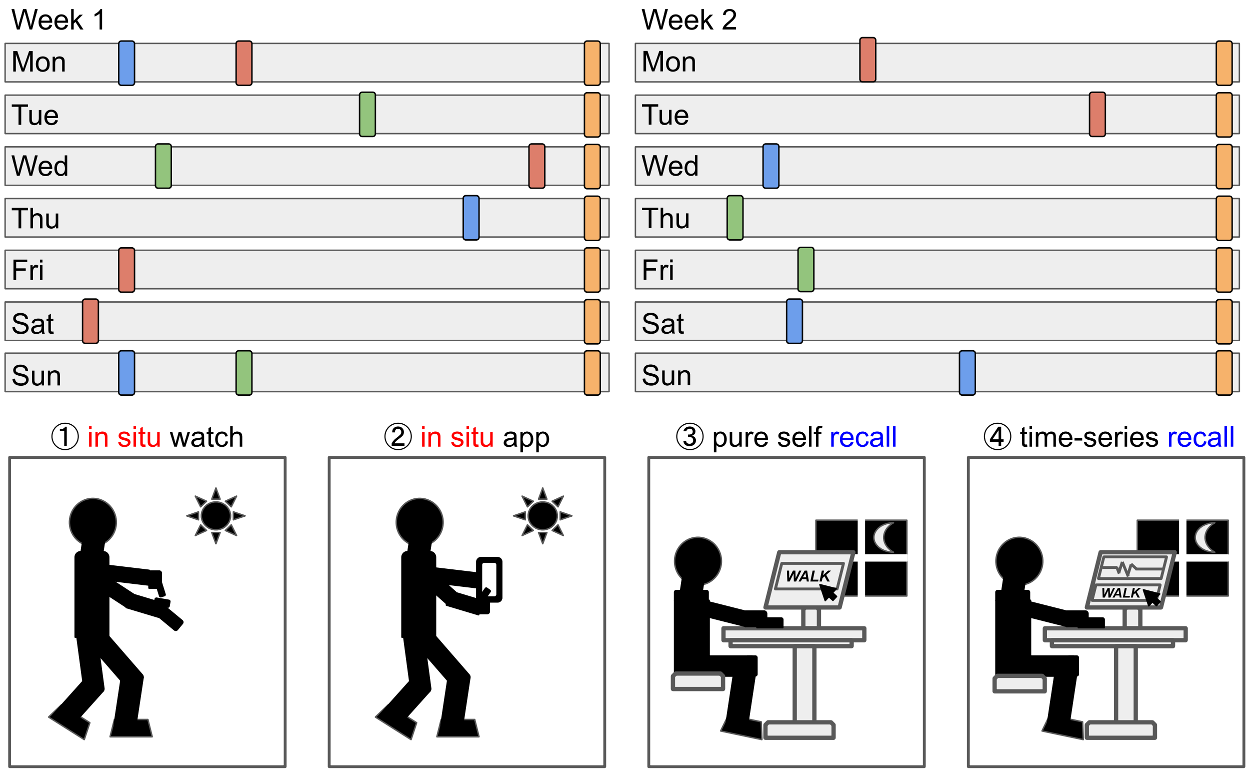

Figure 1: The study participants collected data for 14 days in total and annotated the data with 4 different methods: Labeling ① in situ with a mechanical button, ② in situ with an app, ③ by writing a pure self-recall diary and ④ writing a self-recall diary assisted by a visualization of their time-series data.

model that is less capable of detecting activities reliably, due to unwanted temporal dependencies learned by wrongly annotated patterns [6].

We can see an emerging spotlight on real-world and long-term activity recognition and think that it will be one of the main research challenges that need to be put more in focus in order to overcome current limitations and be capable to recognize complex day-to-day activities. Such datasets, rely heavily on self-recall methods or using additional apps to track movements and set labels either automatically [7] or with the manual selection of a label [8]. Due to these hurdles, many researchers prefer to work with datasets from controlled over uncontrolled environments. As a consequence, only a very limited number of in-the-wild datasets has been published until now.
Contribution: Our study focuses on the evaluation of 4 different annotation methods for labeling data in-the-wild: ① In situ *(lat. on site or in position)* with a button on a smartwatch, ② in situ with the app Strava [1] (an app that is available for iOS and Android smartphones), ③ pure self-recall (writing an activity diary at the end of the day), and ④ *time-series* assisted self-recall with the MAD-GUI [9], which displays the sensor data visually and allows to annotate it interactively. Our study was conducted with 11 participants, 10 males, and 1 female, over 2 weeks. Participants wore a Bangle.js Version 1[2] smartwatch on their preferred hand, used Strava, and completed self-recall annotations every evening. In the first week, the participants were asked to write an activity diary at the end of the day without any helping material and additionally using two user-initiated methods (*in situ button* and *in situ app*) to manually set labels at the start and beginning of each activity. In the second week, the participants were given an additional visualization of the sensor data with an adapted version of the MAD-GUI annotation tool. With the help of this, participants then were instructed to label their data in hindsight with the activity diary as a mnemonic aid. Given labels from both weeks were compared to each other regarding the quality through visual inspection and statistical analysis with regard to the consistency and quantity of missing annotations across labeling methods. Furthermore, we used a Shallow-DeepConv(LSTM) architecture, see Bock *et al.* [10] and Ordoñez *et al.* [11], and trained models with a leave-one-day-out cross-validation method of 6 previously selected subjects and each annotation method.

[1] `https://www.strava.com/`
[2] `https://www.espruino.com/banglejs`

Impact: Annotating data, especially in real-world environments, is still very difficult and tedious. Labeling such data is always a trade-off between accuracy and workload for the study participants or annotators. We raise awareness among researchers to put more effort into exploring new annotation methods to overcome this issue. Our study shows that different labeling methodologies have a direct impact on the quality of annotations. With the deep learning analysis, we proof that this impacts the model capabilities directly. Therefore, we consider the evaluation of frequently used annotation methods for real-world an long-term studies to be crucial in order to give decision makers of future studies a better base on which they can choose the annotation methodology for their study in a targeted way.

2 Related work

Since 2015, only 4 newer datasets that contain wearable sensor data and locomotion activities were released [12]. To the best of our knowledge, a very limited number of datasets are currently publicly available which were recorded in-the-wild, e.g. [13, 14, 15, 16, 17]. Thus, even though [16] and [17] were recorded in an in-the-wild scenario, the participants were equipped with many sensors on different body locations and filmed by a third person during the exercises. Such visible equipment as well as the accompaniment of a third person could have led to a behavior bias [18] in the data since it can alter the participants' movement patterns as participants are constantly reminded that they are currently in the context of a study [3]. Furthermore, multimodal datasets that were recorded with multiple body-worn sensors instead of only one IMU (Inertial Measurement Unit) faced the challenge of proper inter-sensor synchronization [19]. A dataset with a big scope, diverse classes, and accurate annotations, which is recorded by one device that is almost not noticeable by the participant (and therefore might not influence their behavior or movement patterns), is not yet available publicly due to the presented hurdles. According to Stikic *et al.* [20] and later Cleland *et al.* [8], we distinguish between 6 or 7, respectively, different methods and 2 environments (online/offline) of labeling data, the methods are (1) Indirect Observation, (2) Self-Recall, (3) Experience Sampling, (4) Video/Audio Recordings, (5) Time Diary, (6) Human Observer, (7) Prompted Labeling. Cruz *et al.* [21] uses 4 different categories to classify data labeling approaches, these are **(1) temporal (when)** - is the label conducted during or after the activity, **(2) annotator (who)** - is the label given by the individual itself or by an observer, **(3) scenario (where)** - is the activity labeled in a controlled (e.g laboratory) or uncontrolled (in-the-wild) environment, and **(4) annotation mechanism (how)** - is the activity labeled manually, semi-automatically or fully-automatically. All labeling methods have their own benefits, costs and come with a trade-off between required time and label accuracy. However, not every method is suitable for long-term and in-the-wild recording data. Reining *et al.* [22], evaluated the annotation performance between 6 different human annotators of a MoCap (Motion Capturing) and IMU HAR Dataset for industrial deployment. They came to the conclusion that annotations are moderately consistent when subjects labeled the data for the first time. However, annotation quality improved after a revision by a domain expert. In the following, we would like to go into more detail on what we consider to be the most important labeling methods for the specific field of activity recognition.

2.1 Annotation Methods in Activity Recognition

Self-Recall methodologies are generally called methods in which study participants have to remember an event in the past. This methodology is used, for instance, in the medical field (e.g. in the diagnosis of injuries [23]), but also frequently in studies in the field of long-term activity recognition. Van Laerhoven *et al.* [24] used this method during a study in which participants were asked to label their personal daily data at the end of the day. They noticed that the label quality depends heavily on the participant's recall and can therefore be very coarse. During a study conducted by Tapia *et al.* [25], every 15 minutes a questionnaire was triggered in which participants needed the answer multiple choice questions about which of 35 predefined activities were recently performed.

App Assisted Labeling: Cleland *et al.* [8] presented in 2014 the so called Prompted labeling. An approach that is already used by commercial smartwatches like the Apple Watch[3]. In this study user's were asked to set a label for a time period which has been detected as an activity right after the activity stops. Akbari *et al.* [7] leverages freely available Bluetooth Low Energy (BLE) information broadcasted by other nearby devices and combines this with wearable sensor data in order to detect context and direction changes. The participant is asked to set a new label whenever a change in the signal is detected. Gjoreski *et al.* [26] published in 2017 the SHL dataset which contains versatile labeled multimodal sensor data that has been labeled using an Android application that asked the user to set a label whenever they detected a position change via GPS. Tonkin *et al.* [27] presented a smartphone app that was used in their experimental smart home environment with which study participants were able to either use voice-based labeling, select a label from a list of activities ordered by the corresponding location or scan NFC tags that were installed at locations in the smart house. Similar to Tonkin *et al.* [27], Vaizmann *et al.* [14] developed an open-source mobile app for recording sensor measurements in combination with a self-reported behavioral context (e.g. driving, eating, in class, showering). 60 subjects participated in their study. The study found that most of the participants preferred to fill out their past behavior

[3] `https://www.apple.com/watch/`

through a daily journal. Only some people preferred to set a label for an activity that they are about to do. Schröder *et al.* [28] developed a web-based GUI which can either be used on a smartphone, tablet, or a PC to label data recorded in a smart home environment. However, it is important to mention that, According to Cleland *et al.* [8], the process of continually labeling data becomes laborious for participants and can result in a feeling of discomfort.
Unsupervised Labeling is a methodology that uses clustering algorithms to first categorize new samples without deciding yet to which class a sample belongs. Leonardis *et al.* [29] presented in 2002 the concept of finding multiple subsets of eigenspaces where, according to Huynh [30], each of them corresponds to an individual activity. Huynh uses this knowledge to develop the eigenspace growing algorithm, whereby, *growing* refers to an increasing set of samples as well as to increasing the so-called *effective dimension* of a corresponding eigenspace. Based on the reconstruction error (when a new sample is projected to an eigenspace), the algorithm tries to find the best-fitting representation of a sample with minimal redundancy. Hassan *et al.* [31] recently published a methodology that uses the Pearson Correlation Coefficient to map very specific labels of a variety of datasets to 4 meta labels (inactive, active, walking, and driving) of the ExtraSensory Dataset [14].
Human-in-the-Loop (Labeling) is a collective term for methodologies that integrates human knowledge into their learning or labeling process. Besides of being applied in HAR research, such techniques are often used in Natural Language Processing (NLP) and according to [32] the NLP community distinguishes between entity extraction [33, 34], entity linking [35], Q&A tasks [36] and reading comprehension tasks [37].

Active Learning is a machine learning strategy that currently receives a lot of attention in the HAR community. Such strategies involves a Human-in-the-Loop for labeling purposes. In the first step the learning algorithm automatically identifies relevant samples of a dataset which are posteriorly queued to be annotated by an expert. Incorporating a human guarantees high quality labels which directly leads to a better performing classifier. Whether a sample is determined to be relevant, and as well the decision to whom it may get presented for annotation purposes are the main focus of research in this field. Bota *et al.* [38] presents a technique that relies on specific criteria defined by 3 different uncertainty-based selection functions to select samples that will be presented to an expert for labeling and then be propagated throughout the most similar samples. Adaimi *et al.* [39] benchmarks the performance of different Active Learning strategies and compared them, with regard to 4 different datasets with a fully-supervised approach. The authors came to the conclusion that Active Learning needs only 8% to 12% of the data to reach similar or even better results than a fully-supervised trained model. These results suggest that presenting pre-selected samples to a human for labeling purposes can reduce the amount of data needed to train a machine learning classifier significantly due to the increased quality of the labels. Miu *et al.* [40] presented a system which used the Online Active Learning approach published by Scully [41] to bootstrap [42] a machine learning classifier. The publication presented a smartphone app that asked the user right after finishing an activity, which activity has been performed. Afterwards a small subset of the labeled data was used to bootstrap a personalized machine learning classifier.

3 Methodology

3.0.1 Hardware

Participants wore the commercial open-source smartwatch Bangle.js Version 1 with our open-source firmware[4] installed. The device comes with a Nordic 64MHz nRF52832 ARM Cortex-M4 processor with Bluetooth LE, 64kB RAM, 512kB on-chip flash, 4MB external flash, a heart rate monitor, a 3D accelerometer, and a 3D magnetometer. Our firmware only uses the 3D accelerometer and provides the user with the basic functions of a smartwatch, like displaying the time and counting steps. The data is recorded with 25Hz, a sensitivity of ±8g and saved on the devices' memory with a delta compression algorithm. Therefore, we are able to save up to 8-9 hours (depending on how much of the data could be compressed) of data with the given parameters. The smartwatch stops recording as soon as the memory is full. At the end of the day, the participants need to upload their daily data and program the starting time for the next day using our upload web-tool[5].

3.1 Study Setup

Our study is conducted with 11 participants, from which 10 are male and 1 is female. The participants are between 25 and 45 years old. Out of 11 participants, 6 are researchers in the field of signal processing and are used to read and work with sensor data. Participants were selected among acquaintances and colleagues. The study was conducted over a period of 2 weeks while participants wore an open-source smartwatch on their wrist of choice. During the two week study, the participants were instructed to use 4 different labeling methods in parallel, In the first week they were

[4]Our smartwatch firmware is made publicly available at: `https://github.com/kristofvl/BangleApps/tree/master/apps/activate`

[5]Our web-tool is made publicly available at: `https://ubi29.informatik.uni-siegen.de/upload/`

asked to use the ① *in situ button*, ② *in situ app*, and ③ *pure self-recall* methods. In the beginning of the 2nd week we expanded the number of annotation methods with the ④ *time-series recall*. This annotation method combines the activity diary with a graphical visualization of the participants daily data.

① The Bangle.js smartwatch has 3 mechanical buttons on the right side of the case. These buttons are programmed to record the number of consecutive button presses per minute. The total number of button presses is stored with the given timestamp and can therefore be used to mark the beginning and end of an activity in the time-series.

② In addition, the participants were asked to track their activities with the smartphone app Strava. Strava is an activity tracker that is available for Android and iOS and freely downloadable from the app stores. The user can choose from a variety of predefined labels and start recording. Recording an activity starts a timer that runs until the user stops it. The time as well as the GPS position of the user during the activity is tracked and saved locally.

③ The *pure self-recall* methods consist of writing an activity diary on a daily basis at the end of the day. The participants were explicitly told that they should only write down the activities that they still remember 2 hours after the measurement stopped.

④ The *time-series recall* method can be seen as a combination of an activity diary and graphical representation of the raw sensor data. For visualization and labeling purposes, we provided the participants an adapted version of the MAD-GUI. The GUI has been published by Ollenschläger *et al.* [9] in 2022 and is a generic open-source Python package. Therefore, it can be integrated into one's own project. Our adaptions to the package are available for download from a GitHub repository[6]. It contains changes to the data loader, the definition of available labels and color settings for displaying the 3D raw data. **Annotation Guidelines:** The participants were briefed to note daily returning activities (sports or activities of daily living) that are performed longer than 10 minutes roughly 2 hours after the recording stops. The name of the activity was chosen individually by the participant. Participants decided individually at what time of the day the recording will start on the next day. Each of these annotation methods represents a layer of annotation that is used for the visual, statistical, and deep learning evaluation. Figure 1 illustrates the overall concept.

3.2 Statistical Analysis

The labels were statistically analyzed based on their consistency using the Cohen κ score as well as the number of missing annotations across all methods. The Cohen κ score describes the agreement between two annotation methods, which is defined as follows $\kappa = (p_0 - p_e)/(1 - p_e)$ (see [43] and [22]). Where p_0 is the observed agreement ratio and p_e is the expected agreement if both annotators assign labels randomly. The score basically shows how uniform two different annotators labeled the same data. For calculation purposes, an implementation provided by Scikit-Learn [44], was used. Furthermore, missing annotations across methods are measured as the percentage of missing or incomplete annotations. The annotations of all methods were first compared with each other and matched based on the given time indications. Annotations that could not be assigned or were missing were marked accordingly and are the base for calculating this indicator, Figure 4 visualizes this. We used a similar representation as [45] to visualize the matches among labeling methods. In this study, the authors compared genome annotations labeled by different annotators with regard to their error score between different annotators.

3.3 Effects on Deep Learning Performance

The deep learning analyses is performed using the DeepConvLSTM architecture [11] which is based on a Keras implementation of [46]. We did not perform hyperparameter tuning because it would involve a considerable amount of additional workload, since we trained 64 models independently during the evaluation. We therefore decided to opt-out the architecture with regards to efficiency rather than optimal classification results. Additionally, we don't expect that the actual experiment - evaluating different annotation methods - would benefit from hyperparameter tuning or gain any significant information and insights. Instead, we use the default hyperparameters provided by the authors. These are depicted in the Figure 2.

Furthermore, we reduce the number of LSTM-Layers to one and instead increase the number of hidden units of the only LSTM-Layer to 512. According to [10], this modification decreases the runtime up to 48 % compared to a two-layered DeepConvLSTM while significantly increasing the overall classification performance on 4/5 publicly available datasets: [47], [16], [1], [48], [49]. LSTM-Layers in general are important if the dataset contains sporadic activities [6]. However, our dataset does not and our evaluation aims to identify long periods of periodic activities, like walking or running. For this reason, we can conclude that additional LSTM layers are not needed. The implementation of [46] includes BatchNormalization-Layers after each Convolutional-Layer, as well as a MaxPooling for the transition between the last

[6] `https://github.com/ahoelzemann/mad-gui-adaptions/`

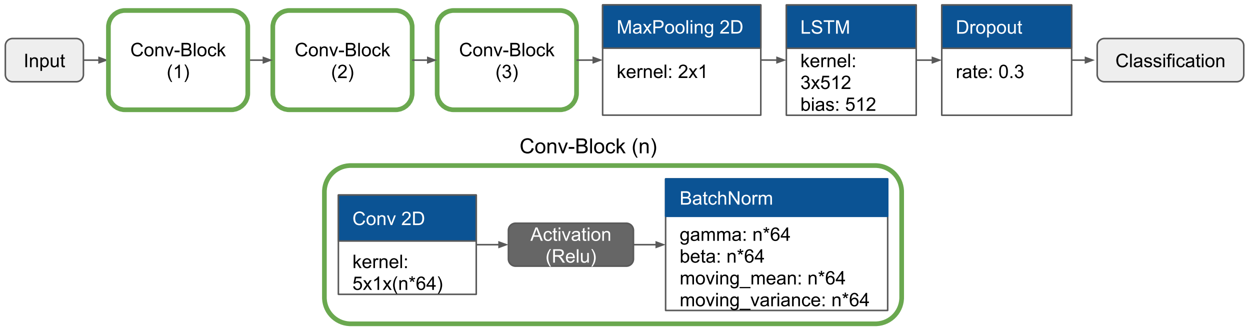

Figure 2: The architecture consists of an **Input** Layer with the kernel-size 10 (window_size) x 10 (filter_length) x 3 (channels). The data is passed into 3 concatenated **convolutional blocks**, followed by a **MaxPooling** (kernel 2x1) where 50% of the data is filtered. The convolutional block consists of a convolutional layer with a variable kernel size of 5x1x(n*64) following a ReLu activation function and a BatchNorm-Layer. We decided two use a single LSTM-Layer with the size of 512 units, as mentioned by [10], which is followed by a Dropout-Layer that filters 30% of randomly selected samples of the window.

convolutional block and LSTM-Layer and a Dropout-Layer before the classification. The BatchNormalization-Layers are included in order to speed up training time and to avoid the negative impact of the internal covariate shift, please see [50] for more information.

Before training the neural network we apply two preprocessing steps to the data. These are an upsampling to a constant 25 Hz due to small deviations in the device sampling-rate as well as a rescaling of the accelerometer data between -1 and 1.

Leave-One-Day-Out Cross-Validation: Figure 3 shows the train- and test-setting for the deep learning model. For every study participant and every week, a personalized model has been trained and tested. We adapted the leave-one-subject-out strategy to fit our needs. Instead of using one participant for testing, we used one day of the week for testing and trained on every other day. The dataset consists of a major *void*-class and a small number of samples per activity class and participant. In order to counteract an above-average large *void*-class, we trained our model with balanced class weights.

Not limiting the participants in their choice of daily performed activities as well as well as not specifying predefined activity labels resulted in very unique sets of activities per study participant. Due to these circumstances we cannot expect that it is possible to train a model which is capable to generalize across participants and days. Every participant comes with their personal specific patterns to perform an activity. Furthermore, due to the in-the-wild recording setup, the intra-class differences [51] for comparatively simple activities, *walking* or *running* can be significant. The impact of different labeling methods are therefore expected to be more present, and hence more visible, in a personalized than it would in a generalized model.

Postprocessing & Classification: We classified the data based on a sliding window with a length of 2 seconds (50 samples). However, we are looking for longer periods of reoccurring activities and decided therefore to apply a jumping window of 5 minutes that applies a majority vote to the given time period. The activity with the most instances in that 5 minute period is set for the whole window.

4 Results

Our participants were asked to annotate activities carried out in their daily life performed for more than 10 min. We didn't limit the participants to a predefined set of classes. They rather decided independently which labels they would like to set for certain time periods. After normalizing the labels, e.g. changing "going for a walk" to "walking" etc., the set of labels as given by the participants contains the following 23 (22 + void) different labels: *laying, sitting, walking, running, cycling, bus_driving, car_driving, vacuum_cleaning, laundry, cooking, eating, shopping, showering, yoga, sport, playing_games, desk_work, guitar_playing, gardening, table_tennis, badminton, horse_riding.* Every sample that wasn't specifically labeled is classified as *void*.

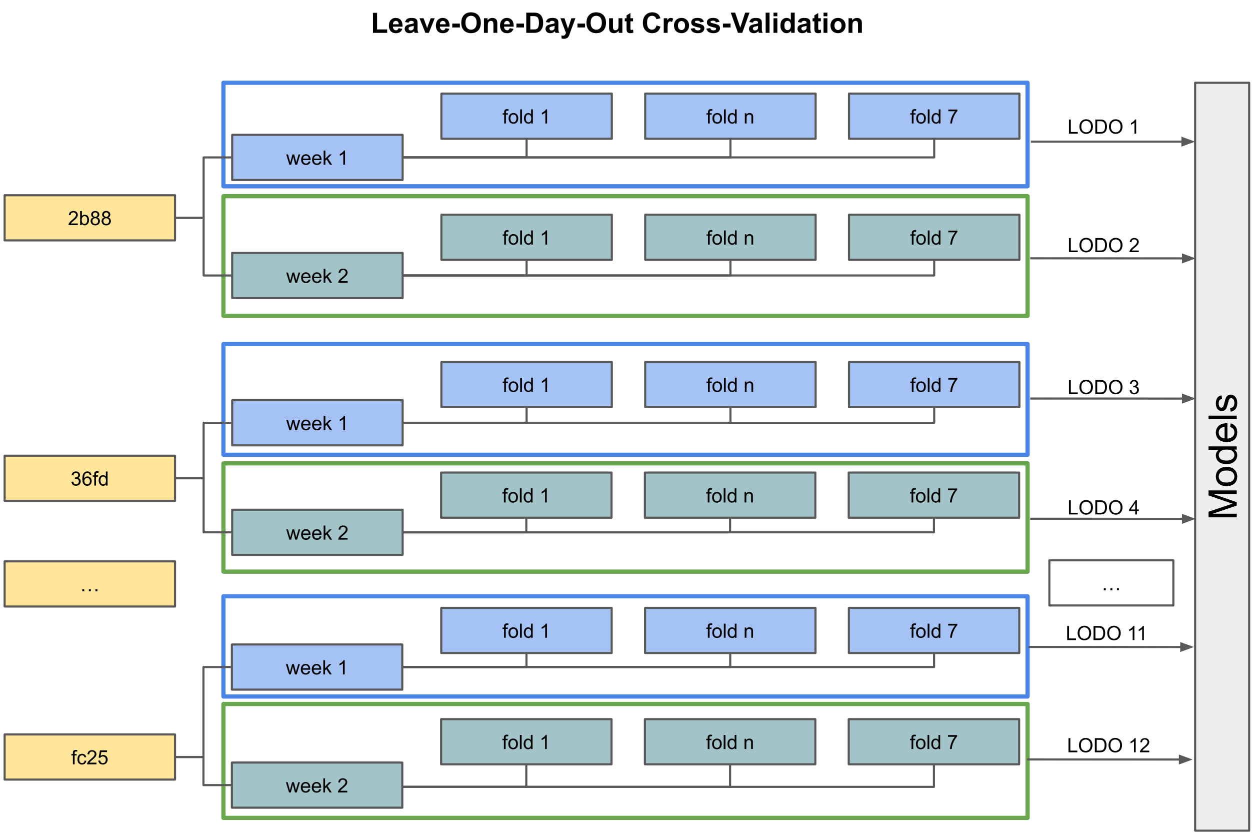

Figure 3: Leave-One-Day-Out Cross validation. The models are personally trained for every participant and are not intended to generalize across all study participants. Instead, a generalization across all days of one week is desired.

4.1 Missing Annotations and Consistency Across Methods

Missing Annotations and the consistency of labels set over the course of one week varied greatly depending on the study participant. However, tendencies with regards to specific methods are observable. Method (1), pressing the situ button on the smartwatch's case, was not consistently used by every participant. Furthermore, this method carries the risk that either setting one of the two markers (start or end) is forgotten. An annotation where one marker is missing becomes therefore obsolete. The app assisted annotation method (2), for which we used the app Strava, is well accepted among the participants that agreed with using a third-party software. However, 4 participants, namely *74e4*, *90a4*, *d8f2* and

Table 1: Missing annotations across all labeling methods (in %) of both weeks. The columns contain the Subject-ID of all participants. The last column shows the average percentage of missing annotations across all every labeling methods, for all participants.

						Week 1						
Subject	**2b88**	**36fd**	**74e4**	**90a4**	**834b**	**4531**	**a506**	**d8f2**	**eed7**	**f30d**	**fc25**	**Avg.**
(1) ***in situ button***	40	70.59	79.41	52.18	36.37	50	26.32	96.15	45.46	0	26.81	40.95
(2) ***in situ app***	13.30	5.89	97.06	100	5.00	0	36.84	92.30	22.73	100	4.35	43.40
(3) ***pure self-recall***	6.67	0	0	0	4.55	0	31.58	0	4.55	0	0	4.30
						Week 2						
(1) ***in situ button***	23.08	73.33	92.00	82.14	8.33	76.47	0	95.33	61.11	0	27.78	49.05
(2) ***in situ app***	61.54	6.67	100	89.29	8.33	35.30	100	79.16	33.33	100	11.11	56.79
(3) ***pure self-recall***	0	0	8.57	17.88	4.17	35.30	0	12.50	5.56	0	5.56	8.14
(4) ***time-series recall***	0	0	22.88	39.29	0	0	0	16.67	0	0	5.56	7.67

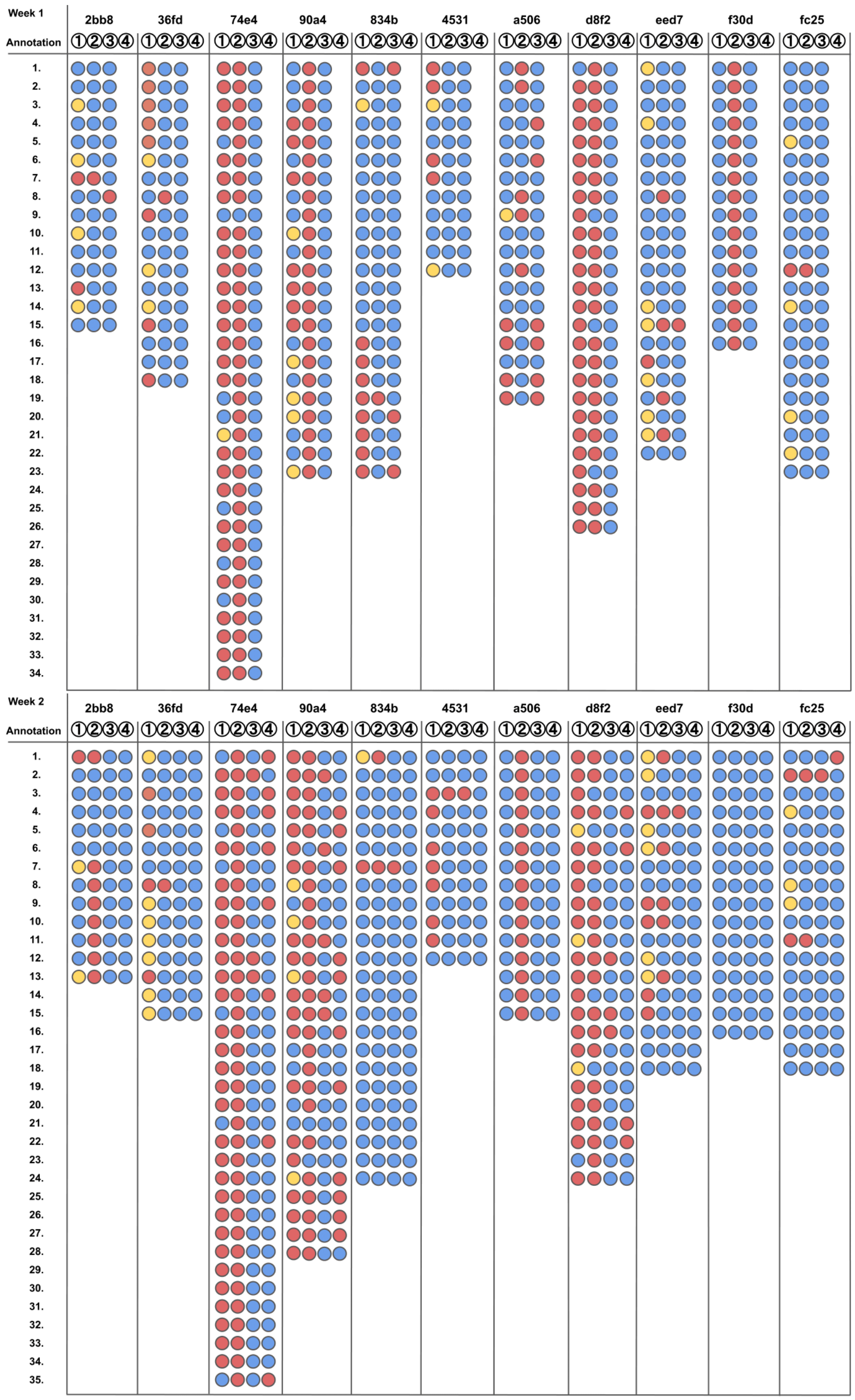

Figure 4: Missing annotations across all study participants and both weeks. The Y-axis shows the total number of annotations of one specific participant for the corresponding week. The color codes are as followed: Annotation is missing, Annotation is partially missing (start or stop time), Annotation is complete. The figure design is inspired by [45], Figure 1.

f30d did not use the app continuously or refused to use it completely due to concerns regarding their private data. Strava is a commercial app, which is freely available for download on the app stores, but it collects certain users' meta data. In order to label a time period with Strava, the participant needs to (1) take the smartphone in the hand, (2) open the app, (3) start a timer, set a label, and (4) end the timer. This procedure contains significantly more steps than other methods. Therefore, the average value of missing annotations results in 46.40% (week 1) and 56.79% (week 2). One participant found the annotation process in general very tedious and therefore dropped out of the study. These data have been excluded from the dataset and the evaluation. Method ③ *pure self-recall*, writing an activity diary, got well accepted by every participant. As Figure 4 shows and the results in Table 1 proofs, it is overall the most complete annotation method with an average amount of missing annotations of 4.30 % for the first and 8.14 % for the second week. By introducing the MAD-GUI, participants were able to inspect their personal daily data and get insights on how patterns of specific classes look like and label them interactively. With an average amount of missing annotations of 7.67%, this method became the most complete during the second study week. Table 2 shows the resulting Cohen κ scores. Due to the constraint that only one labeling method can be compared to a second one and since, according to Table 1, the most consistent annotation methods are ③ *pure self-recall* and ④ *time-series recall*, we used these methods as our baseline and compared them with every other method used in the study. The second column indicates the comparison direction. The abbreviations used in this column are defined as follows: (③ C/W ①) *pure self-recall* compared with *in situ button*, (③ C/W ②) *pure self-recall* compared with *in situ app* and (③ C/W ④) *pure self-recall* compared with *time-series recall*. The direction (④ C/W ③) is not explicitly included since Cohens κ is bidirectional and both directions result in the same score. The score indicates how similar two annotators, or in our study labeling methods, are to each other. The resulting score is a decimal value between -1.0 and 1.0, where -1.0 means that the two annotators differ at most and 1.0 means complete similarity. 0.0 denotes that the target method was not used on that specific day.

Comparing the ③ *pure self-recall* method with the ① *in situ button* and ② *in situ app* method we can see that the final results for weeks 1 and 2 are proximate to one another. ③ *Pure self-recall* compared with the ④ *time-series recall* results in the highest similarity of 0.52. The comparison between the ④ *time-series recall* and the ① *in situ button* as well as the ② *in situ app* assisted annotations result in higher similarity than the prior comparison of ③ *pure self-recall* vs. both methods ① and ②. This means that subjects rather agree to the timestamps of the *in situ* methods than to a self-written activity diary as soon as they can visually inspect the accelerometer data.

4.2 Visual Time-Series Analysis

Figure 5 shows exemplary the time-series of the sixth day of every participant's second week. The four bars that are visible above the accelerometer data are the labels set by the participants for every layer. The order is from bottom to top: ① *in situ button*, ② *in situ app*. ③ *pure self-recall*, and ④ *time-series recall*. Examples of labels that differs with regards to the applied labeling method are marked with red boxes. The x-axis of every subplot represents roughly 8-9 hours of data. Most of the day was not labeled and is therefore categorized as *void*. However, such long periods often contain shorter periods of other activities, like *walking*. This makes it difficult to define a distinguishable *void*-class, which results in false positive classifications of non-void samples. Figure 5 visually shows that each individual participant labels his or her data very subjectively. The long green-labeled time periods of participant *74e4* represent the class *desk_work*. The only other participant that used this label is *90a4*. Since each of the study participants work in an office environment and thus conclusively work at a desk, we can assume that the same class is classified as *void* for all other study participants. This intra-class and inter-participant discrepancy becomes a problem whenever a model shall be trained that is supposed to generalize across individuals. To reduce these side effects and focus on the experiment itself, we decided to evaluate on personalized models that take weekly data from participants into account.

The *in situ button* annotation is empty for 5 participants: *eed7*, *36fd*, *74e4*, *90a4* and *d8f2*. Labels are only partially set or missing entirely for this annotation method and we therefore assume that participants tending to forget to press the button on the smartwatch. Both tables, 1 and 2, support this assumption, as this labeling method shows a high percentage of missing annotations as well as a low Cohen κ score of 0.36% (week 1) and 0.39% (week 2). The *pure self-recall* method ③, visible on the 2nd upper layer, is often misaligned compared to the in situ methods as well as the *time-series recall* method ④. Participants tend to round up or down the start- and stop-time in steps of 5 or 10 minutes. For example, the annotations in Figure 5 given by the subjects *2b88*, *834b* or *f30d*, show such incorrectly annotated data. The pink color represents the class *walking*. With a closer look on the corresponding time-series data, one can see that the *in situ button* annotation (bottom layer) and *time-series recall* annotation (top layer) rather belongs to the typical periodically pattern of walking than the period labeled by *pure self-recall*. A consistent reliable performance in all labeling methods can only be observed at the participants *4531* and *fc25*. Other participants like *eed7*, *36fd*, *74e4* or

Table 2: Average similarity between annotation methods according to the Cohan κ score for both study weeks. The columns are ordered subject-wise. The last column shows the average across all participants for one study week. The Direction column indicates in which the direction the Cohan κ score is calculated and needs to be interpreted as follows: ① *in situ button*, ② *in situ app*, ③ *pure self-recall*, ④ *time-series recall*, (C/W) compared with.

Week, Day	**Direction**	**2b88**	**36fd**	**4531**	**74e4**	**834b**	**90a4**	**a506**	**d8f2**	**eed7**	**f30d**	**fc25**	**Avg.**
1, 1	③ C/W ①	0.32	0.0	0.0	0.0	0.69	0.35	0.79	0.22	0.0	0.23	0.58	
	③ C/W ②	0.69	0.85	0.69	0.0	0.76	0.0	0.0	0.0	0.90	0.0	0.49	
1, 2	③ C/W ①	0.64	0.69	0.0	0.09	0.85	0.05	0.51	0.0	0.55	0.47	0.74	
	③ C/W ②	0.64	0.68	0.84	0.05	0.86	0.0	0.50	0.0	0.93	0.0	0.73	
1, 3	③ C/W ①	0.0	0.62	-0.03	0.0	0.39	0.56	0.53	0.0	0.05	0.53	0.51	
	③ C/W ②	0.86	0.0	-0.03	0.0	0.38	0.0	0.44	0.0	0.28	0.0	0.54	
1, 4	③ C/W ①	0.38	0.30	0.91	0.08	0.63	0.03	0.80	0.0	0.66	0.80	0.0	
	③ C/W ②	0.99	0.69	0.90	0.0	0.74	0.0	0.57	0.69	0.80	0.0	0.0	
1, 5	③ C/W ①	0.33	0.33	0.0	0.04	0.39	0.32	0.93	0.0	-0.03	0.93	0.87	
	③ C/W ②	0.32	0.83	0.0	0.0	0.37	0.0	0.96	0.0	-0.31	0.0	0.89	
1, 6	③ C/W ①	0.0	0.0	0.75	0.07	0.0	0.34	0.67	0.0	0.42	0.99	0.84	
	③ C/W ②	-0.14	0.96	0.71	0.0	0.15	0.0	-0.07	0.41	0.52	0.0	0.84	
1, 7	③ C/W ①	0.30	0.0	0.56	0.04	0.0	0.525	0.99	0.0	0.29	0.90	0.49	
	③ C/W ②	0.78	0.15	0.69	0.0	0.10	0.0	0.43	0.0	0.42	0.0	0.77	
2, 1	③ C/W ①	0.30	0.56	0.36	0.10	0.51	0.0	0.88	0.0	0.41	0.89	0.85	
	③ C/W ②	0.45	0.77	0.37	0.0	0.57	-0.02	0.0	0.63	0.51	0.0	0.81	
	③ C/W ④	0.85	0.76	0.56	0.10	0.48	0.11	0.90	0.78	0.74	0.86	0.46	
	④ C/W ①	0.39	0.43	0.48	0.43	0.74	0.0	0.98	0.0	0.58	0.82	0.58	
	④ C/W ②	0.53	0.61	0.45	0.0	0.70	0.18	0.0	0.71	0.57	0.0	0.55	
2, 2	③ C/W ①	0.82	0.21	0.91	0.05	0.47	0.03	0.70	0.0	0.29	0.74	0.36	
	③ C/W ②	0.84	0.75	0.93	0.0	0.90	0.0	0.0	0.87	0.59	0.0	0.81	
	③ C/W ④	-0.02	0.82	0.62	0.09	0.84	0.09	0.70	0.83	0.56	0.85	0.46	
	④ C/W ①	-0.02	0.11	0.66	0.38	0.47	0.77	1.0	0.0	0.43	0.70	0.45	
	④ C/W ②	-0.02	0.83	0.63	0.0	0.92	0.0	0.0	0.96	0.71	0.0	0.46	
2, 3	③ C/W ①	0.70	0.44	0.0	0.0	0.62	0.0	0.90	0.0	0.28	0.99	0.68	
	③ C/W ②	0.66	0.44	0.98	0.0	0.72	0.0	0.0	0.47	0.36	0.0	0.82	
	③ C/W ④	0.68	0.54	0.62	0.91	0.49	-0.18	0.90	0.86	0.39	0.98	0.58	
	④ C/W ①	0.91	0.53	0.0	0.0	0.77	0.0	1.0	0.0	0.88	0.96	0.41	
	④ C/W ②	0.74	0.53	0.82	0.0	0.74	0.0	0.0	0.60	0.79	0.0	0.65	
2, 4	③ C/W ①	0.45	0.0	0.83	0.0	-0.02	0.05	0.64	0.0	0.23	0.67	0.78	
	③ C/W ②	0.14	0.85	0.83	0.0	-0.02	0.0	0.0	0.0	0.26	0.0	0.87	
	③ C/W ④	0.17	0.86	0.84	0.90	-0.02	0.0	0.64	0.84	0.38	0.86	0.92	
	④ C/W ①	0.71	0.0	0.86	0.0	0.80	0.47	1.0	0.0	0.68	0.66	0.76	
	④ C/W ②	0.82	0.51	0.86	0.0	0.80	0.0	0.0	0.0	0.50	0.0	0.85	
2, 5	③ C/W ①	0.59	0.0	0.0	0.0	0.28	0.09	0.60	0.0	0.0	0.73	0.95	
	③ C/W ②	0.0	0.41	0.77	0.0	0.28	0.01	0.0	0.54	0.54	0.0	0.92	
	③ C/W ④	0.48	0.47	0.76	0.16	0.26	0.09	0.59	0.82	0.40	0.43	0.47	
	④ C/W ①	0.5	0.0	0.0	0.0	0.82	0.94	0.99	0.0	0.0	0.38	0.46	
	④ C/W ②	0.0	0.34	0.89	0.0	0.83	0.60	0.0	0.49	0.70	0.0	0.44	
2, 6	③ C/W ①	0.48	0.0	0.90	0.0	0.39	0.0	0.73	0.0	0.0	0.20	0.86	
	③ C/W ②	0.0	0.85	0.96	0.0	0.39	0.02	0.0	0.55	0.83	0.0	0.87	
	③ C/W ④	0.47	0.86	0.92	0.77	0.30	0.0	0.72	0.83	0.86	0.74	0.86	
	④ C/W ①	0.98	0.0	0.95	0.0	0.47	0.0	0.99	0.0	0.0	0.46	0.83	
	④ C/W ②	0.0	0.88	0.89	0.0	0.62	0.69	0.0	0.43	0.95	0.0	0.82	
2, 7	③ C/W ①	0.0	0.0	0.0	0.0	0.30	0.0	0.73	0.40	0.43	0.91	0.86	
	③ C/W ②	0.0	0.41	0.76	0.0	0.30	0.0	0.0	0.0	0.14	0.0	0.86	
	③ C/W ④	0.96	0.47	0.72	0.0	0.24	-0.01	0.42	0.79	0.33	0.92	0.84	
	④ C/W ①	0.0	0.0	0.0	0.0	0.79	0.0	0.67	0.46	0.71	0.94	0.78	
	④ C/W ②	0.0	0.80	0.93	0.0	0.78	0.0	0.0	0.0	0.46	0.0	0.78	
Avg. Week 2	③ C/W ①	**0.48**	**0.17**	**0.43**	**0.02**	**0.36**	**0.02**	**0.74**	**0.0**	**0.23**	**0.73**	**0.76**	**0.39**
	③ C/W ②	**0.30**	**0.64**	**0.80**	**0.0**	**0.45**	**0.0**	**0.0**	**0.44**	**0.46**	**0.0**	**0.85**	**0.36**
	③ C/W ④	**0.51**	**0.68**	**0.72**	**0.12**	**0.37**	**0.01**	**0.70**	**0.82**	**0.33**	**0.81**	**0.66**	**0.52**
	④ C/W ①	**0.50**	**0.15**	**0.42**	**0.42**	**0.69**	**0.31**	**0.94**	**0.07**	**0.47**	**0.70**	**0.61**	**0.48**
	④ C/W ②	**0.30**	**0.64**	**0.78**	**0.0**	**0.78**	**0.21**	**0.0**	**0.46**	**0.67**	**0.0**	**0.65**	**0.41**

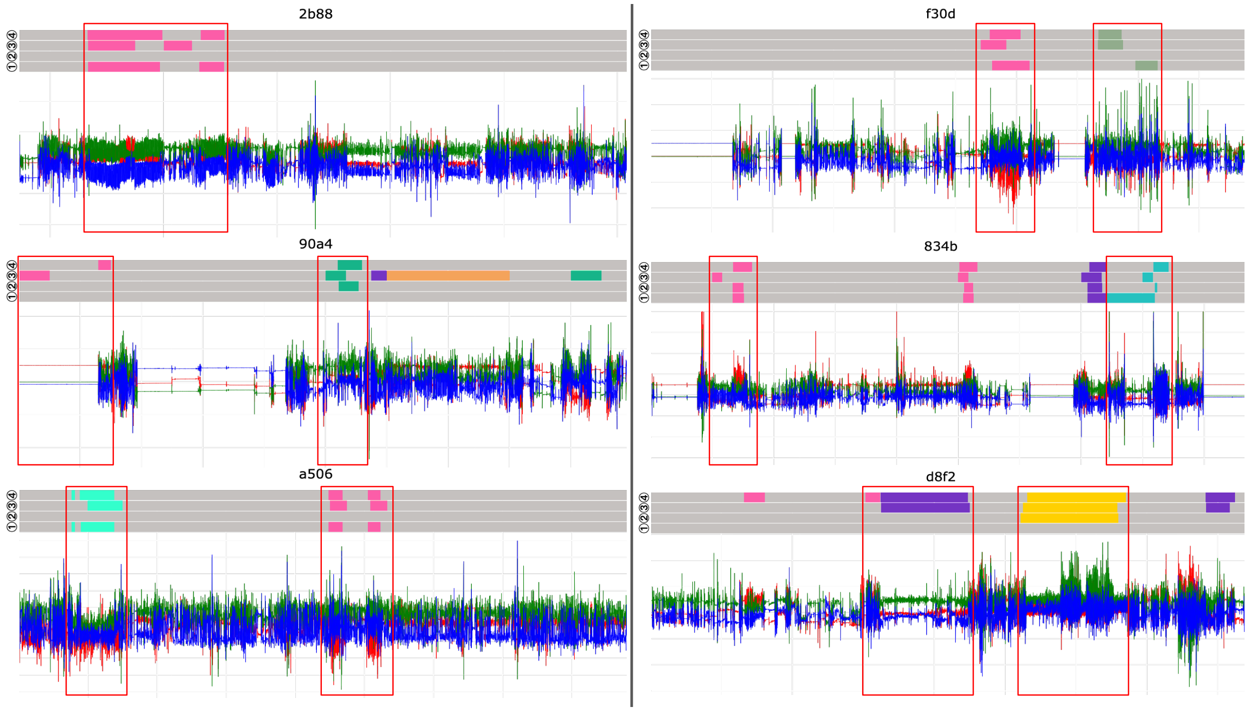

Figure 5: Visualization of participants' accelerometer data on the sixth day in the second week of the study, together with annotations set by them. The four layers in the upper part of every participant's daily data represent the four annotation methods. The order is from bottom to top: ① *in situ button*, ② *in situ app*, ③ *pure self-recall* and ④ *time-series recall*.

a506 are very precise in their annotations across methods, but are missing at least one layer of labels. The complete collection of visualizations is available in our dataset repository[7]

4.3 Effects on Classification

The results of our deep learning evaluation[8] suggest that the annotation method chosen can have a crucial impact on the classification ability of a trained deep learning model. Depending on the chosen methodology, the average F1-score results differ by up to 8%. In the first week, the in situ methodologies, button ① and app ②, generally perform better than the *pure self-recall* diary ③. Study participants mostly correctly estimated the duration of an activity, but tend to round up or down the start and end times. The in situ methods are up to 8% better than the *pure self-recall*, although the amount of annotated data available, due to missing annotations, is significantly lower than for other methods. Although, we work with a dataset recorded in-the-wild, the deep learning results generally show a high F1-score. This is untypical for such datasets, but can be explained by the fact that the majority of the daily data are assigned to the *void* class. This leaves proportionally only a few samples that are crucial for determining the classification performance.

Despite the fact that the number of available annotations that have been labeled by the study participants using the *time-series recall* method ④ is significantly high with 92.33%, the average F1 score is 1.1% lower (89.00%) than the results reached with the App Assisted method (90.1%). To understand this result it is crucial to look at Table 3 in detail and take meta-information about the participants into account. The participants mostly used their diary as a mnemonic aid for the graphical annotation method and tried to identify the corresponding periods in the acceleration data. The results of subjects *2b88*, *a506* and *eed7* show that the performance of the classifier could be increased with graphical assistance. However, the F1-Score of *2b88* is 0.01% below the F1 score of the *in situ app* assisted annotation method ②. These subjects have in common that they are already trained in interpreting acceleration data due to their prior knowledge and thus assign samples to specific classes more precisely. Subjects *fc25*, *4531*, and *834b*, on the other hand, do not have prior knowledge. Apart from subject *834b*, the deep learning results show that presenting a visualization to

[7] `https://doi.org/10.5281/zenodo.7654684`

[8] Detailed results for every participant included in our deep learning evaluation can be accessed online on the Weights & Biases platform: `https://tinyurl.com/4vxvfaed`.

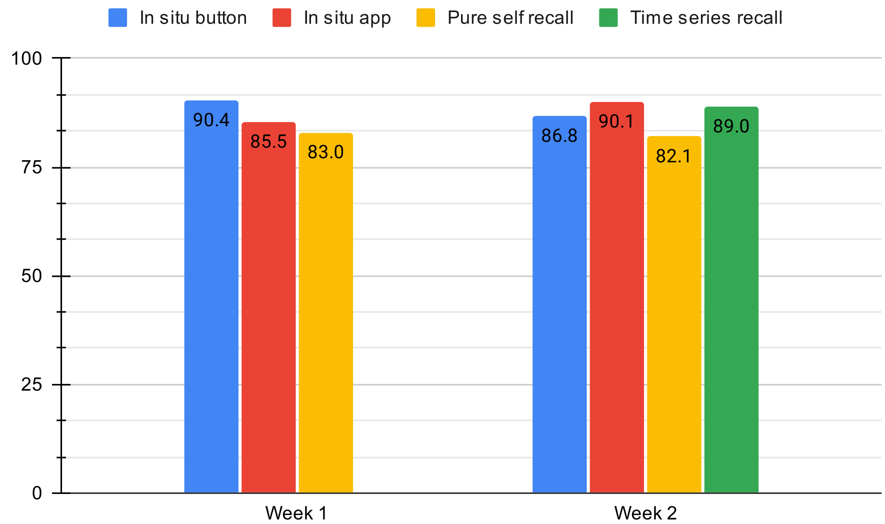

Figure 6: The overall mean F1-Scores for the Leave-One-Day-Out Cross Validation across all participants. In the first week the participants used methods ① - ③. In the second week we introduced method ④.

Week 1							
Subject	**2b88**	**a506**	**eed7**	**fc25**	**4531**	**834b**	**Average**
① ***in situ button***	0,91	0,92	0,89	0,89	0,91	0,91	90,4
② ***in situ app***	0,92	0,60	0,91	0,84	0,93	0,92	85,5
③ ***pure self-recall***	0,78	0,76	0,83	0,86	0,86	0,89	83,0
Week 2							
① ***in situ button***	0,88	0,92	0,90	0,92	0,86	0,72	86,8
② ***in situ app***	0,91	na	0,92	0,94	0,85	0,88	90,1
③ ***pure self-recall***	0,81	0,86	0,82	0,94	0,83	0,67	82,1
④ ***time-series recall***	0,90	0,91	0,95	0,86	0,86	0,86	89,0

Table 3: In detail representation of the final F1 scores for every annotation methodology and week per study participant. The average F1-Scores are graphically visualized in Figure 6.

an untrained participant rather harmed than helped the classifier. If one looks at the visualizations of day 1 & 6, week 2 of *fc25*, see Figure 5 and 7, the labels set by the subject with the help of the graphical interface, it is comprehensible that this study participant tended to be rather confused by the graphical representation and therefore labeled the data incorrectly.

5 Discussion

In our 2-week long-term study, we recorded the acceleration data of 11 participants using a smartwatch and analyzed it visually, statistically, and using deep learning. The findings of the visual and statistical analysis were confirmed by the deep learning result. They show that the underlying annotation procedure is crucial for the quality of the annotations and the success of the deep learning model.

The *in situ button* method ① offers accuracy but brings the risk that the setting of a label is forgotten entirely or incompletely set. However, this method can be combined with additional on-device feedback or a smartphone app, so that a greater accuracy and consistency of the annotation can be achieved. This involves a considerable implementation effort, which many scientists avoid because such projects, although of their significant value to the community, attract little attention in the scientific world. The use of existing, but often commercial, software and hardware is all too often

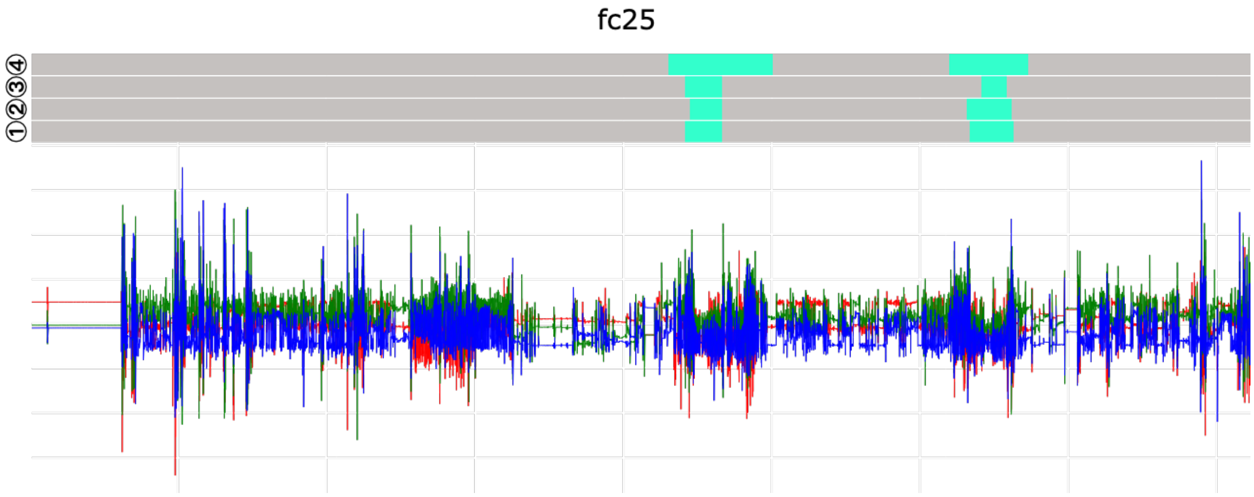

Figure 7: Visualization of the 1st day in week 2 of subject *fc25*. Differences can be seen in the upper annotation layer (④ *time-series recall*), exhibiting larger differences regarding the annotated start- and stop times compared to other methods.

accompanied by a loss of privacy. As our research has shown in passing, many users therefore shy away from using such products.

Through our investigation on the consistency of annotations between methodologies, we were able to show that participants in our study seem to prefer to write an activity diary (*pure self-recall* method ③). This finding corresponds to what Vaizmann *et al.* [14] already points out. However, this method has the disadvantage that it can be imprecise, which is evident in the visualization of the data and annotations. Similarly, the activity diary methodology performed the least reliably among all methodologies, which has been confirmed by the deep learning model. Since the deep learning results using the *in situ app* annotations ③ are almost similar to the results given by the *time-series recall* ④, even though the number of labeled samples is lower, it raises the question if a smaller set of high-quality annotations is more valuable for a classifier than a larger set of annotated data that comes with imprecise labels. This could mean that in future works we can reduce the amount of necessary training samples drastically if a certain annotation quality can be assured. However, this needs to be confirmed by further investigations.

Some participants reported that they found the support provided by the visual representation of the data helpful. The resulting Cohen κ scores strengthen this impression, since the F1-scores are much higher when we compare the *time-series recall* with both in situ methods vs. the *pure self-recall*. This indicates that as soon as the participants received a visual inspection tool, they rather tend to annotate data at similar time periods as through the in situ methods, since they are able to easily identify periods of activity that roughly correspond to the execution time they remember. Our participants reported similar preferences, which led us to the conclusion that a digital diary that includes a data visualization could combine the benefits of both annotation methods.

However, the study also showed that participants can find it difficult to interpret the acceleration data correctly and thus set inaccurate annotations. As our trained models show, this also has a strong influence on the classification result. If such a tool is to be made available to study participants, it must be ensured that they have the necessary knowledge and tools to be able to interpret these data. Thus, to assure the success of future long-term and real-world activity recognition projects, prior training of the study participants regarding the data interpretation is of crucial importance if a data visualization is supposed to be used. Apart from trying to solve annotation difficulties during the annotation phase itself, we can also partially counter wrong or noisy classified data by using machine learning techniques like Bootstrapping, see Miu *et al.* [40] or using a loss function that specifically tries to counteract this problem, such as [5, 52]. By using Bootstrapping, the machine or deep learning classifier is initially trained by a small subset of high-confident labels and further improved by using additional data. However, this technique comes with the trade-off that whenever wrong labeled data is introduced as training data, the error will get propagated into the model. An effect that sooner or later occurs as long as the annotation methodologies themselves are not further researched. Other machine learning techniques that can work with noisy labels, see [53], are already successfully tested for Computer Vision problems and can, in theory, be adopted for Human Activity Recognition. However, earlier research has shown that not every technique that is applicable in other fields is also applicable on sensor based data [46].

5.1 Discussing different Annotation Biases

We need to take into account that a number of biases could have been introduced due to the chosen annotation guidelines and tools. For example, the usage of in situ annotation methods during the day can have a positive effect on the self-recall capabilities of a participant at the end of the day. However, the comparison of consistencies across methods does not confirm that this effect indeed occurred. Every study participant showed an almost complete overall profile of self-recall annotations, even though the person has not used or has incomplete in-situ annotations, see Fig. 4. However, deeper investigations are needed in order to be able to understand such effects better.

Yordanova *et al.* [18] lists the following 3 biases for sensor-based human activity data: Self-Recall bias [23], Behavior bias [3] and the Self-Annotation bias [18]. We showed that indeed a time-deviation bias (which can be seen as a self-recall bias) has been introduced to annotations created with the *pure self-recall* method ③, and that such a bias effects the classifier negatively. However, visualizing the sensor data can counter this effect because it was easier for participants to detect active phases in hindsight.

A behavior bias can be neglected, because the participants were not monitored by a person or video camera during the day and the minimalistic setup of one wrist-worn smartwatch does not influence ones behavior since the wearing comfort of such a device is generally perceived as positive [54]. A self-annotation bias, a bias that occurs if the annotator labels their data in an isolated environment and cannot refer to an expert in order to verify an annotation, did occur as well. With the deep learning analysis we were able to show that the classifier was less negatively impacted by this bias than by time-deviation bias.

6 Conclusion

We argue that the annotation methodologies for benchmark datasets in Human Activity Recognition do not yet capture the attention it should. Data annotation is a laborious and time-consuming task that often cannot be performed accurately and conscientiously without the right tools. However, there is a very limited number of tools that can be used for this purpose and often they do not pass the prototype status.

Only a few scientific publications, such as [22], that focuses on annotations and their quality. Although the use of proper annotated data drastically affects the final capacities of the trained machine or deep learning model. Therefore, we consider our study to be important for the HAR community, as it analyzes this topic in greater depth and thus provides important insights that go beyond the current state of science. Table 4 summarizes the advantages and disadvantages of every method. To guarantee high-quality annotations in future studies, especially in an uncontrolled real-world

Table 4: Comparison of advantages and disadvantages of all annotation methods used in this study.

Methodology	Advantages	Disadvantages
① *in situ button*	- Easy to implement and use - Can be improved with feedback mechanisms	- Participants tend to forget pressing a button - Many incomplete annotations that become unusable for the classifier
② *in situ app*	- Tracking apps are already widely used and accepted, therefore low acceptance threshold - Can be improved with feedback mechanisms or additional smartphone functionalities - List of possible annotations can be expanded with minimum effort - Participants tend to set very precise annotations.	- Data and privacy concerns if a commercial app is used. - Participants often forgot to set an annotation, especially when they were unfamiliar with tracking apps. - Implementation workload may be very high
③ *pure self-recall*	- Easy to use even without technical knowledge (a handwritten diary) - Most accepted method in our experiment - Annotations are very consistent	- Can be very imprecise - Only suitable for coarse activity labels and activities that were performed for long periods of time, like *walking* or *running*
④ *time-series* recall	- Visualization of data helps participants to set annotations more accurate than using the pure self-recall method ③	- Available tools are often in the state of a prototype and need additional developments and adjustments and are therefore not impromptu usable. - Participants need to be trained to be able to interpret sensor data.

environment where a video recording is not available for the subsequent acquisition of ground truth, further research in this field is necessary. The methodologies used for annotating activity data need to be challenged and further developed.

The combination of a (handwritten) diary with a correction aided by a data visualization in hindsight shows the best results in terms of consistency and missing annotations and provides accurate start and end times. However, this combination results in additional work for the study participants and therefore, remains a trade-off between additional workload and annotation quality.

Lesson Learned. During this study we gained insights about the effects of different annotation methods on the reliability and consistency of annotations and finally on the classifier itself, but also about training deep learning models on data recorded in-the-wild. In this chapter we would like to share these insights in order to help other researchers performing their experiments more successful. With regards to Table 4, we are able to narrow down specific study setups that either benefit more from self-recall or in situ annotation methods.

1. Due to the good acceptance and the low workload for study participants we can recommend a self-recall method for studies where label precision is not the highest priority and rough estimations of activities are sufficient.
2. According to our study, we can increase the label precision of the self-recall method with an additional software that visualizes the raw data, e.g. [9]. We recommend to consider the implementation of such a module and providing this software to participants together with an introduction on how to interpret sensor data. According to [14] the self-recall method can also be effectively improved by introducing server guesses of activities or visually organizing the day chronologically.
3. In situ annotations result generally in more precise labels. However, the label process is more labor intensive than a self-recall method, since it can take a lot of time and often includes many steps to set the label. We argue, that smaller studies with participants that agree with performing such laborious work can benefit from this method.
4. Such a system needs to be implemented carefully and with a holistic concept in order to not be seen as a burden by the participants [55].
5. Annotating data with commercial apps, like Strava, are negligible due to data and privacy concerns.
6. In situ annotation can have the same benefits as an app solution. However, only if researchers have access to the programming interface of the recording device and can implement additional features that helps participants to not forget setting a label.

As part of our annotation guidelines, we gave our study participants the opportunity to name their activities as they wished. Therefore, we were forced to simplify certain activities. In order to be able to create a real-world dataset which contains complex classes or even classes that consists of several subclasses, more elaborated annotation methods and tools must be developed. We believe that with the current available resources the hurdle lies very high for such datasets to be annotated accurately.

Our study includes people who cycle to work in their daily work routine, others commute by public transport or work in a home office environment. Thus, each study participant has his or her own personal set of daily repetitive activities. Due to the nature of our dataset as one recorded in a real-world and long-term scenario, the number of labeled samples is rather small and given labels very participant dependant. This mix of factors creates a bias in the dataset and we concluded that a cross-participant train-/test-strategy is not appropriate for our study design and would not give meaningful insights, since every study participant has their own set of unique activities which are too different in nature and hardly generalizable. Therefore, for certain studies the commonly known and accepted Leave-One-Subject-Out Cross-Validation is not suitable.

Declarations

Conflict of Interest

On behalf of all authors, the corresponding author states that there is no conflict of interest.

Funding

This project is funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – 425868829 and is part of Priority Program SPP2199 Scalable Interaction Paradigms for Pervasive Computing Environments.

Ethics

The studies involving human participants were reviewed and approved by Ethics Committee of the University of Siegen (ethics vote #ER_12_2019). The participants provided their written informed consent to participate in this study and were briefed prior to the data capture about the study goals.

Availability of data and materials

The datasets generated during and/or analysed during the current study are available in the Zenodo repository: `https://doi.org/10.5281/zenodo.7654684`.

Code availability

All code is made publicly available on our GitHub repository `https://github.com/ahoelzemann/annotationMatters`.

Consent to participate

Informed consent was obtained from all individual participants included in the study.

Consent for publication

The participants have consented to the submission of the manuscript to the journal.

Authors' contributions

Alexander Hoelzemann has performed the implementation and implemented all studies and visualizations, **Kristof Van Laerhoven** has guided this work and assisted in the methodologies. Both authors have contributed substantially to the writing of this manuscript.

References

[1] Daniel Roggen, Alberto Calatroni, Mirco Rossi, Thomas Holleczek, Kilian Förster, Gerhard Tröster, Paul Lukowicz, David Bannach, Gerald Pirkl, Alois Ferscha, et al. Collecting complex activity datasets in highly rich networked sensor environments. In *2010 Seventh international conference on networked sensing systems (INSS)*, pages 233–240. IEEE, 2010.

[2] Sakorn Mekruksavanich and Anuchit Jitpattanakul. Recognition of real-life activities with smartphone sensors using deep learning approaches. In *2021 IEEE 12th International Conference on Software Engineering and Service Science (ICSESS)*, pages 243–246. IEEE, 2021.

[3] Kenzie B Friesen, Zhaotong Zhang, Patrick G Monaghan, Gretchen D Oliver, and Jaimie A Roper. All eyes on you: how researcher presence changes the way you walk. *Scientific Reports*, 10(1):1–8, 2020.

[4] Kunlun Zhao, Junzhao Du, Congqi Li, Chunlong Zhang, Hui Liu, and Chi Xu. Healthy: A diary system based on activity recognition using smartphone. In *2013 IEEE 10th international conference on mobile Ad-Hoc and sensor systems*, pages 290–294. IEEE, 2013.

[5] Nagarajan Natarajan, Inderjit S Dhillon, Pradeep K Ravikumar, and Ambuj Tewari. Learning with noisy labels. *Advances in neural information processing systems*, 26, 2013.

[6] Marius Bock, Alexander Hoelzemann, Michael Moeller, and Kristof Van Laerhoven. Investigating (re) current state-of-the-art in human activity recognition datasets. *Frontiers in Computer Science*, 4, 2022.

[7] Ali Akbari, Jonathan Martinez, and Roozbeh Jafari. Facilitating human activity data annotation via context-aware change detection on smartwatches. *ACM Transactions on Embedded Computing Systems (TECS)*, 20(2):1–20, 2021.

[8] Ian Cleland, Manhyung Han, Chris Nugent, Hosung Lee, Sally McClean, Shuai Zhang, and Sungyoung Lee. Evaluation of prompted annotation of activity data recorded from a smart phone. *Sensors*, 14(9):15861–15879, 2014.

[9] Malte Ollenschläger, Arne Küderle, Wolfgang Mehringer, Ann-Kristin Seifer, Jürgen Winkler, Heiko Gaßner, Felix Kluge, and Bjoern M Eskofier. Mad gui: An open-source python package for annotation and analysis of time-series data. *Sensors*, 22(15):5849, 2022.

[10] Marius Bock, Alexander Hölzemann, Michael Moeller, and Kristof Van Laerhoven. Improving deep learning for har with shallow lstms. In *2021 International Symposium on Wearable Computers*, pages 7–12, 2021.

[11] Francisco Ordóñez and Daniel Roggen. Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition. *Sensors*, 16:115, 01 2016.

[12] Fuqiang Gu, Mu-Huan Chung, Mark Chignell, Shahrokh Valaee, Baoding Zhou, and Xue Liu. A survey on deep learning for human activity recognition. *ACM Computing Surveys (CSUR)*, 54(8):1–34, 2021.

[13] Eugen Berlin and Kristof Van Laerhoven. Detecting leisure activities with dense motif discovery. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, pages 250–259, 2012.

[14] Yonatan Vaizman, Katherine Ellis, Gert Lanckriet, and Nadir Weibel. Extrasensory app: Data collection in-the-wild with rich user interface to self-report behavior. In *Proceedings of the 2018 CHI conference on human factors in computing systems*, pages 1–12, 2018.

[15] Edison Thomaz, Irfan Essa, and Gregory D Abowd. A practical approach for recognizing eating moments with wrist-mounted inertial sensing. In *Proceedings of the 2015 ACM international joint conference on pervasive and ubiquitous computing*, pages 1029–1040, 2015.

[16] Timo Sztyler and Heiner Stuckenschmidt. On-Body Localization of Wearable Devices: An Investigation of Position-Aware Activity Recognition. In *IEEE International Conference on Pervasive Computing and Communications*, pages 1–9, 2016.

[17] Hristijan Gjoreski, Mathias Ciliberto, Lin Wang, Francisco Javier Ordonez Morales, Sami Mekki, Stefan Valentin, and Daniel Roggen. The university of sussex-huawei locomotion and transportation dataset for multimodal analytics with mobile devices. *IEEE Access*, 6:42592–42604, 2018.

[18] Kristina Y. Yordanova, Adeline Paiement, Max Schröder, Emma Tonkin, Przemyslaw Woznowski, Carl Magnus Olsson, Joseph Rafferty, and Timo Sztyler. Challenges in annotation of user data for ubiquitous systems: Results from the 1st ARDUOUS workshop. *CoRR*, abs/1803.05843, 2018.

[19] Alexander Hoelzemann, Henry Odoemelem, and Kristof Van Laerhoven. Using an in-ear wearable to annotate activity data across multiple inertial sensors. In *Proceedings of the 1st International Workshop on Earable Computing*, pages 14–19, 2019.

[20] Maja Stikic, Diane Larlus, Sandra Ebert, and Bernt Schiele. Weakly supervised recognition of daily life activities with wearable sensors. *IEEE transactions on pattern analysis and machine intelligence*, 33(12):2521–2537, 2011.

[21] Dagoberto Cruz-Sandoval, Jessica Beltran-Marquez, Matias Garcia-Constantino, Luis A Gonzalez-Jasso, Jesus Favela, Irvin Hussein Lopez-Nava, Ian Cleland, Andrew Ennis, Netzahualcoyotl Hernandez-Cruz, Joseph Rafferty, et al. Semi-automated data labeling for activity recognition in pervasive healthcare. *Sensors*, 19(14):3035, 2019.

[22] Christopher Reining, Fernando Moya Rueda, Friedrich Niemann, Gernot A Fink, and Michael ten Hompel. Annotation performance for multi-channel time series har dataset in logistics. In *2020 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, pages 1–6. IEEE, 2020.

[23] Giulietta Valuri, Mark Stevenson, Caroline Finch, Peter Hamer, and Bruce Elliott. The validity of a four week self-recall of sports injuries. *Injury Prevention*, 11(3):135–137, 2005.

[24] Kristof Van Laerhoven, David Kilian, and Bernt Schiele. Using rhythm awareness in long-term activity recognition. In *2008 12th IEEE International Symposium on Wearable Computers*, pages 63–66. IEEE, 2008.

[25] Emmanuel Munguia Tapia, Stephen S Intille, and Kent Larson. Activity recognition in the home using simple and ubiquitous sensors. In *International conference on pervasive computing*, pages 158–175. Springer, 2004.

[26] Hristijan Gjoreski, Mathias Ciliberto, Francisco Javier Ordoñez Morales, Daniel Roggen, Sami Mekki, and Stefan Valentin. A versatile annotated dataset for multimodal locomotion analytics with mobile devices. In *Proceedings of the 15th ACM Conference on Embedded Network Sensor Systems*, pages 1–2, 2017.

[27] Emma L Tonkin, Alison Burrows, Przemysław R Woznowski, Pawel Laskowski, Kristina Y Yordanova, Niall Twomey, and Ian J Craddock. Talk, text, tag? understanding self-annotation of smart home data from a user's perspective. *Sensors*, 18(7):2365, 2018.

[28] Max Schröder, Kristina Yordanova, Sebastian Bader, and Thomas Kirste. Tool support for the online annotation of sensor data. In *Proceedings of the 3rd International Workshop on Sensor-based Activity Recognition and Interaction*, pages 1–7, 2016.

[29] Aleš Leonardis, Horst Bischof, and Jasna Maver. Multiple eigenspaces. *Pattern recognition*, 35(11):2613–2627, 2002.

[30] Duy Tâm Gilles Huynh. Human activity recognition with wearable sensors. *Technische Universität Darmstadt*, pages 59–65, 2008.

[31] Iqbal Hassan, Abtahi Mursalin, Robin Bin Salam, Nazmus Sakib, and HM Zabir Haque. Autoact: An auto labeling approach based on activities of daily living in the wild domain. In *2021 Joint 10th International Conference on*

Informatics, Electronics & Vision (ICIEV) and 2021 5th International Conference on Imaging, Vision & Pattern Recognition (icIVPR), pages 1–8. IEEE, 2021.

[32] Xingjiao Wu, Luwei Xiao, Yixuan Sun, Junhang Zhang, Tianlong Ma, and Liang He. A survey of human-in-the-loop for machine learning. *Future Generation Computer Systems*, 2022.

[33] Anna Lisa Gentile, Daniel Gruhl, Petar Ristoski, and Steve Welch. Explore and exploit. dictionary expansion with human-in-the-loop. In *The Semantic Web: 16th International Conference, ESWC 2019, Portorož, Slovenia, June 2–6, 2019, Proceedings 16*, pages 131–145. Springer, 2019.

[34] Shanshan Zhang, Lihong He, Eduard Dragut, and Slobodan Vucetic. How to invest my time: Lessons from human-in-the-loop entity extraction. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2305–2313, 2019.

[35] Jan-Christoph Klie, Richard Eckart de Castilho, and Iryna Gurevych. From zero to hero: Human-in-the-loop entity linking in low resource domains. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 6982–6993, 2020.

[36] Eric Wallace, Pedro Rodriguez, Shi Feng, Ikuya Yamada, and Jordan Boyd-Graber. Trick me if you can: Human-in-the-loop generation of adversarial examples for question answering. *Transactions of the Association for Computational Linguistics*, 7:387–401, 2019.

[37] Max Bartolo, Alastair Roberts, Johannes Welbl, Sebastian Riedel, and Pontus Stenetorp. Beat the ai: Investigating adversarial human annotation for reading comprehension. *Transactions of the Association for Computational Linguistics*, 8:662–678, 2020.

[38] Patrícia Bota, Joana Silva, Duarte Folgado, and Hugo Gamboa. A semi-automatic annotation approach for human activity recognition. *Sensors*, 19(3):501, 2019.

[39] Rebecca Adaimi and Edison Thomaz. Leveraging active learning and conditional mutual information to minimize data annotation in human activity recognition. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 3(3):1–23, 2019.

[40] Tudor Miu, Paolo Missier, and Thomas Plötz. Bootstrapping personalised human activity recognition models using online active learning. In *2015 IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing*, pages 1138–1147. IEEE, 2015.

[41] D Sculley. Online active learning methods for fast label-efficient spam filtering. In *CEAS*, volume 7, page 143, 2007.

[42] Steven Abney. Bootstrapping. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 360–367, 2002.

[43] Ron Artstein and Massimo Poesio. Inter-coder agreement for computational linguistics. *Computational linguistics*, 34(4):555–596, 2008.

[44] scikit learn. Cohen's kappa - scikit-learn, 2022. `https://scikit-learn.org/stable/modules/generated/sklearn.metrics.cohen_kappa_score.html`, Last accessed on 2022-10-02.

[45] Steven E Brenner. Errors in genome annotation. *Trends in Genetics*, 15(4):132–133, 1999.

[46] Alexander Hoelzemann and Kristof Van Laerhoven. Digging deeper: towards a better understanding of transfer learning for human activity recognition. In *Proceedings of the 2020 International Symposium on Wearable Computers*, pages 50–54, 2020.

[47] Jorge-L Reyes-Ortiz, Luca Oneto, Albert Samà, Xavier Parra, and Davide Anguita. Transition-aware human activity recognition using smartphones. *Neurocomputing*, 171:754–767, 2016.

[48] Philipp M Scholl, Matthias Wille, and Kristof Van Laerhoven. Wearables in the wet lab: a laboratory system for capturing and guiding experiments. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 589–599, 2015.

[49] Allan Stisen, Henrik Blunck, Sourav Bhattacharya, Thor Siiger Prentow, Mikkel Baun Kjærgaard, Anind Dey, Tobias Sonne, and Mads Møller Jensen. Smart devices are different: Assessing and mitigatingmobile sensing heterogeneities for activity recognition. In *Proceedings of the 13th ACM conference on embedded networked sensor systems*, pages 127–140, 2015.

[50] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015.

[51] Andreas Bulling, Ulf Blanke, and Bernt Schiele. A tutorial on human activity recognition using body-worn inertial sensors. *ACM Comput. Surv.*, 46(3), 2014.

[52] Zhiheng Ma, Xing Wei, Xiaopeng Hong, and Yihong Gong. Bayesian loss for crowd count estimation with point supervision. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6142–6151, 2019.

[53] Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. Learning from noisy labels with deep neural networks: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

[54] Debajyoti Pal, Suree Funilkul, and Vajirasak Vanijja. The future of smartwatches: assessing the end-users' continuous usage using an extended expectation-confirmation model. *Universal Access in the Information Society*, 19:261–281, 2020.

[55] Kristina Yordanova. Challenges providing ground truth for pervasive healthcare systems. *IEEE Pervasive Computing*, 18(2):100–104, 2019.