Bootstrapping Personalised Human Activity Recognition Models Using Online Active Learning

Tudor Miu
School of Computing Science
Newcastle University
Newcastle-upon-Tyne, UK
Email: t.a.miu@ncl.ac.uk

Paolo Missier
School of Computing Science
Newcastle University
Newcastle-upon-Tyne, UK

Thomas Plötz
Open Lab
School of Computing Science
Newcastle University
Newcastle-upon-Tyne, UK

***Abstract*—In Human Activity Recognition (HAR) supervised and semi-supervised training are important tools for devising parametric activity models. For the best modelling performance, typically large amounts of annotated sample data are required. Annotating often represents the bottleneck in the overall modelling process as it usually involves retrospective analysis of experimental ground truth, like video footage. These approaches typically neglect that prospective users of HAR systems are themselves key sources of ground truth for their own activities. We therefore propose an Online Active Learning framework to collect user-provided annotations and to bootstrap personalized human activity models. We evaluate our framework on existing benchmark datasets and demonstrate how it outperforms standard, more naive annotation methods. Furthermore, we enact a user study where participants provide annotations using a mobile app that implements our framework. We show that Online Active Learning is a viable method to bootstrap personalized models especially in live situations without expert supervision.**

I. Introduction

One of the key promises of Weiser's vision of pervasive computing has been the prospect of disappearing technologies that "weave themselves into the fabric of everyday life until they are indistinguishable from it" [1]. Tremendous progress has already been made towards making this vision a reality where smart environments, living labs, and especially mobile computing now constitute the central paradigm of this third generation of computing [2]. As an enabling technology, automatic inference of the context and especially of the activities humans are engaged in – typically referred to as Human Activity Recognition (HAR) – plays a central role in the majority of ubiquitous and mobile computing applications.

Supervised training methods play an important practical role in Human Activity Recognition research. Sample data in combination with manual ground truth annotations (labels) are used for parameters estimation in order to derive probabilistic models of activities of interest. Whilst this method works well in principle, there are major drawbacks associated to it. Mainly, acquiring the necessary annotations (class labels) to construct the training examples can be problematic, because it requires expert judgement on observed people's activities. Standard procedures focus on annotating datasets that were collected in the lab or in instrumented environments, which is, however, often laborious, time-consuming and therefore not appropriate for large volumes of data. Even when the process can be automated, privacy and ethical considerations may limit the researcher's ability to observe a person's activity. Furthermore, in mobile settings it may not even be possible to suitably instrument the environment to collect the necessary observations.

In this paper we explore an alternative approach of accumulating annotated training sets for bootstrapping personalised parametric activity models. We include prospective users of HAR systems into the training process and have them generate limited sets of annotations for their own activities as they unfold. We do not use video footage as a source of ground truth and, instead, rely on the user's short-term memory to provide annotations for her own activities. Specifically, we adopt an *online* annotation methodology whereby a user's stream of activities is monitored continuously in real-time and the user's HAR model is personalised with parsimoniously acquired annotations.

This is a challenging problem, chiefly because the user's memory has limited power of recall. As Eisen et al. [3] show, remembering long sequences of items or events is unreliable, so all annotation requests are aimed only at the latest identified activity. Implicitly, having access to only one potential annotation at any time makes it difficult to identify the annotations that are expected to bring the greatest improvement to the user's HAR model. Therefore, an annotation decision heuristic should not only operate on a stream of activities and have access to only the latest activity, but it should also outperform Random Selection (RS) – randomly asking for annotations without consideration to HAR model performance.

As an annotation decision heuristic, we employ Online Active Learning (OAL). It operates on a stream of activities and seeks to identify annotations which outperform Random Selection. We incorporate OAL into a machine learning framework into which it is possible to plug in concrete algorithms suitable for different contexts and types of data. We instantiate the framework to handle non-periodic and periodic data and we evaluate the performance of OAL using publicly available HAR datasets. Additionally, we apply an instance of the framework to a realistic user-based case study so that we collect genuine annotations from users.

Analysis on public datasets show that, compared to Random Selection, Online Active Learning improves model accuracy by up to 5% for non-periodic activities and by up to 8.5% for periodic activities. Results from our user-based case study show that acquiring annotations using Online Active Learning and other complementary techniques results in accuracy improvements of $38-47\%$ over a simplistic *strawman* classifier. We did not enact Random Selection in our user study because, as we show, it would have been resulted in the loss of 43% of

annotations for rare activities.

In a stream-based online setting, improving the personalised model accuracy over Random Selection is a difficult problem. *Offline* techniques, such as pool-based Active Learning [4], have been used in related settings to identify what annotations to request. However, as we show later in this paper, pool-based Active Learning, although common for HAR [5]–[9], is strictly not applicable to our mobile scenario due to the extremely limited number of potential annotations from which to choose at any one time. In spite of this limitation, we adapt to the stream-based nature of the data and apppy an Online Active Learning heuristic which attempts to optimise over the choice of annotations so that HAR model accuracy is improved over Random Selection.

Considering the challenges facing the problem of bootstrapping personalised HAR models from user-provided annotations using OAL, the contributions in this paper are four-fold:

1) Analysing an Online Active Learning annotation heuristic We propose an OAL annotation decision heuristic that operates over a data stream corresponding to ongoing activities. Similarly to other active learning approaches, our heuristic attempts to optimise model performance through informed decisions over what annotations are requested from the user. However, in contrast to previous applications of active learning to HAR, our heuristic does not need a long history of potential annotations. Instead, it works in the severely limited case when only the most recent activity is available for annotation. This ensures that annotations can be reported from the user's short-term memory and that the HAR model performance could be improved with respect to RS.

2) Designing a framework for bootstrapping activity recognisers using online active learning. We integrated our OAL annotation decision heuristic into a machine learning framework. The framework provides multi-stage processing, with the option of specifying concrete algorithm implementations for each step, depending on the type of data being monitored. The framework continuously monitors a user's activities and bootstraps a personalised model from user-provided annotations.

3) Evaluation through simulations. We use public HAR datasets to simulate the acquisition of user-provided annotations. We evaluate OAL on both non-periodic and periodic activities, using the challenging Opportunity dataset [10] and the USC-HAD [11] and PAMAP [12] datasets, respectively. In the case of periodic activities, we additionally adopt a method for activity segmentation, which exploits the repetitive nature of the movement to identify *segments* (contiguous subsequences that ideally span a single activity). Our results show that OAL constructs personalised models which exhibit superior accuracy over models constructed with RS: up to 5% for non-periodic activities and up to 8.5% for periodic activities.

4) Evaluation through a user study. We developed a mobile app (on an Android smart phone) to support an experimental user study, which was used to test OAL in the field, with the help of a panel of volunteer participants. The app implemented an instance of our proposed framework: it interacted with the user and collected genuine user annotations which were used to learn a personalised parametric model of the user's activities throughout the session. Encouraged by the performance gains in our simulations, we use the user study to demonstrate the feasibility of OAL. Results show that the personalised model outperforms a strawman model on accuracy by up to $38 - 47\%$. We did not enact RS because it would have missed a substantial amount of annotations for rare activities, namely 43% of annotations collected using OAL.

II. Background and Related Work

Our work integrates a set of techniques into an autonomous system that engages with a user to obtain annotations for their activities and that correspondingly bootstraps the user's personalized activity model. To do this, we relate to and distinguish our work from several directions of research.

A. Self-Provided Annotations

In HAR research, typically, while movement data is collected through sensors, the participants are observed by a researcher who annotates the data or by examining retrospective video footage of the participants (e.g., [10], [13]–[15]). Often these methods come with additional technical challenges such as synchronising accelerometer and video streams [16]. However, these methods do not involve the users in the annotation process. Instead, van Kasteren et al. [17] have used a headset to allow the users to self-annotate sensor data using voice recognition. The users were asked to provide labels using spoken words for their activities as they happened. The authors report near errorless voice recognition, but Hoque et al. [18] have shown that, in a different context, the precision for some labels can drop to 80%. The added layer of voice recognition may result in additional errors in the activity model. We want to avoid such errors and in this paper we suggest that annotations are collected using an unambiguous tap-only interface on a mobile device.

A self-reporting method, Ecological Momentary Assessment (EMA), described by Smyth and Stone [19], also known as Experience Sampling Method (ESM) according to Intille et al. [20], [21], is used in medical research to allow patients to report relevant symptoms, conditions or circumstances while they occur. Data integrity levels in EMA/ESM are high and Smyth and Stone [19] argue this may be due to the timeliness with which user or patient input is given.

We take advantage of this timeliness and we propose an ESM-style annotation process where the user takes ownership of annotating some of their own activities as they happen. We assume the users' short-term memory is a reliable source of ground truth for their activities. In addition, we continually monitor the user's context and identify which annotations are likely to improve the model more than Random Selection.

B. Pool-Based Active Learning

Users should only be asked to annotate limited amounts of just the most relevant data; otherwise it can lead to reduced user compliance. For example, in a bid to obtain sufficiently many user-provided annotations for supervised model building and evaluation, Intille et al. [20] generated annotation requests every 15 minutes over two weeks. The resulting level of user compliance was very low and the authors believe this is due to the excessive disruption that competes with normal

living. In our approach, we propose that annotation requests are informed by the user context, so that only the most beneficial activities are annotated by the user.

Active Learning, a semi-supervised learning methodology, serves to orchestrate the accumulation of training data in such a way that it improves the gains in recognition accuracy over random discovery of training data (Random Selection), according to Settles [4]. In HAR, many attempts focus on *pool-based* active learning – offline datasets are used and the annotation of data is simulated by revealing one or a few labels at a time from the entire dataset or from a large subset, as done by Rebetez et al. [22], Stikic et al. [5], Longstaff et al. [6], Alemdar et al. [7], Bagaveyev and Cook [8] and Liu et al. [9]. A heuristic function examines the input datasets and identifies the most promising data instance to annotate. In an offline setting, a good choice of the heuristic function and a comprehensive view of large parts or the whole dataset promise good optimality in choosing what activities to annotate. However, from a user perspective, this places unrealistic expectations on the user memory. In reality, people cannot be expected to precisely remember individual activities which took place in the distant past or the associated exact start and end times of these activities. Asking the user to annotate these would lead to unreliable annotations. Overall, we conclude offline pool-based approaches are not compatible with our online scenario.

C. Stream-Based Active Learning

Simulations that operate on datasets of annotations curated by researchers and experts can afford offline pool-based Active Learning or similar approaches. In reality, in many cases, activities unfold sequentially as a strean, so it is logical to consider an online stream-based annotation approach, like Miu et al. [23] or Abdallah et al. [24]. For example, Miu et al. [23], propose to generate annotations based on a priorly established schedule, but not according to a criterion that seeks to optimise HAR model performance, so this is not active learning. In this paper, our annotation decisions are not only online, but also informed by the context. This means that annotation decisions are made on the spot regarding whether the latest activity is suitable for annotation.

Abdallah et al. [24] propose an online stream-based active learning strategy where each annotation decision is aimed at clusters of potentially multiple activities. While the authors provide curation techniques that remove most of the outliers to keep only the predominant label in a cluster, the activities under consideration are not very diverse. For our proposed online scenario, we too employ an online stream-based annotation approach to collect personalised annotations. However, in contrast, we propose to direct annotation requests at individual activities and we evaluate the system against a more diverse set of activities. Additionally, we show that our Online Active Learning method registers performance gains over soliciting annotations at random. To this end, we use an existing Online Active Learning technique already elaborated for spam classification by Sculley [25] and apply it to HAR.

D. Interrupting Users

Another direction of research seeks to understand how appropriate it is to interrupt a user at a given time. For example, Pejovic and Musolesi [26] propose using a non-disruptive method of modelling the suitability of interruption using a multidimensional mobile phone trace including current time, accelerometer features and location. In an online setting, the authors report large variations in precision and recall, but also large discrepancies between the two. This suggests that interruption models can suit a large spectrum of preferences: from users who are strict about not being interrupted outside their preferred intervals of time to users who prefer not to miss important notifications with less regard to when they happen. Similarly, Fogarty et al. [27] leverage context cues such as video footage to model the suitability for interruption. Using audio processing, computer vision-based techniques and retrospective manual annotation, the authors construct models of suitability for interruption. Using a different approach, Kapoor and Horvitz [28] used a desktop-based application that not only monitored application use and other contextual information, but also probed the user to continually adapt the interruption model.

This direction of research is complementary to ours. Their focus is on the user's sentiment towards disruption, while ours is on maximizing the performance of a personalized activity model by carefully selecting what sample data to ask the users to annotate.

E. Activity Segmentation

Numerous techniques on how to detect segment boundaries in data streams have been developed, but these do not fit our assumptions. For example, in environments instrumented with on/off sensors, Chua et al. [29], Krishnan and Cook [30] or Okeyo et al. [31] have exploited discrete sensor changes to segment activities. However, this is not applicable to our scenario because our sensing framework is based on continuous acceleration signals. Continuous signals have been segmented by Junker et al. [32] or Krishnan et al. [33], but those methods are not applicable because they assume a prior corpus of annotations to inform the segmentation stage. Similarly, extreme points based segmentation (as, for example, in [34]) is also not practical for us as we do not make any assumption about the structure of the underlying accelerometer data.

Instead, we draw inspiration from the video segmentation literature and adapt an online segmentation procedure, devised by Cooper [35], to periodic activities.

F. Bootstrapping New Models vs. Adapting Existing Models

The main direction of research in this paper is on bootstrapping personalised models without prior knowledge and only from user-provided annotations. Another approach would be to adapt existing population models to the target user, like Abdallah et al. [24]. However, this typically assumes the existence of a large corpus of annotated data for the current sensor configuration or choice of activities. The assumption may not always be true, for example, if the sensor configuration differs or other activities are of interest.

Alternatively, even though it is possible to adapt the annotated data from one sensor configuration to another, e.g. Roggen et al. [36] and Kurz et al. [37], this results in a loss of accuracy compared to what would be obtained from annotations for the existing configuration.

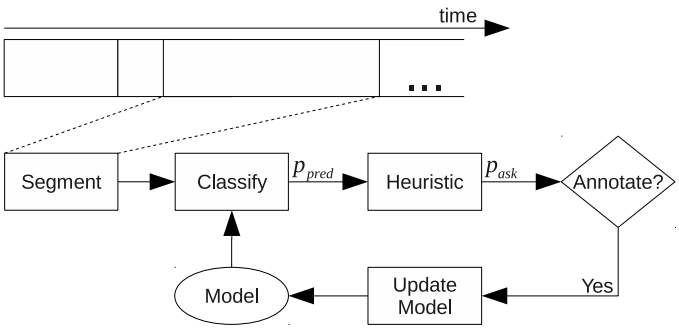


Fig. 1: Schematic of the Annotation Framework.

III. Online Active Learning Framework

In this section, we present our framework for bootstrapping personalized activity models from user-provided annotations. By "framework" we understand a multi-stage data processing pipeline with algorithm placeholders for every stage. Depending on the characteristics of the data, different framework instances can be created by plugging in concrete algorithms.

An annotation consists of two parts: (1) a *segment*, which is a contiguous sequence of sensor readings between a start and an end timestamp, and (2) a *label* denoting the activity of that segment. We suggest that annotations are obtained from user feedback, similarly to Intille et al. [20]. However, we draw cues from the continuously monitored user context to identify the segments which, if annotated, are expected to improve model performance more than randomly selected annotations.

Our proposed framework, illustrated in Fig 1, includes three stages: (1) a *segmentation* step which detects segments in a continuous stream of activity data, (2) a *classification* step for activity recognition and (3) an *annotation decision heuristic* step which discovers the annotations which are expected to improve model accuracy more than Random Selection.

The annotation decision heuristic is *Online Active Learning*, as employed by Sculley [25] for online spam classification. In our scenario, the heuristic relies on segments firstly being identified from a contiguous stream of activity data and then classified by the user's personalised probabilistic model/classifier. For each new segment, an annotation request will be issued with probability p_{ask} which is computed from the classification confidence associated to class predictions for that segment, as follows. The model classifies the current segment and generates a probability p^j_{pred} for each of the activity classes known to the model. $p_{conf} = \max_j p^j_{pred}$ is used as a measure of classification confidence. The probability p_{ask} of requesting an annotation for that segment as a function of classification confidence as follows:

$$p_{ask} = exp(-\gamma \cdot p_{pred}) \tag{1}$$

In Eq. 1, p_{ask} is monotonically decreasing with p_{pred}, which means that low classification confidences are mapped to high probabilities of asking for an annotation and vice-versa, with two-fold implications: (1) the segments the model struggles to classify are most likely to be annotated by the user and (2) the user is unlikely to be asked to annotate segments the model can already confidently classify. In Eq. 1, γ is a tunable parameter that controls the asking behaviour, which can be understood by examining what happens when γ is increased, as follows: Firstly, given a fixed p_{conf}, the probability of asking for an annotation decreases, which results in fewer annotation requests overall. Secondly, when p_{conf} decreases, the decline in asking probability is more pronounced with higher values of γ. Effectively, with an increased γ, segments with high confidence p_{conf} are more likely to be ignored, so the annotations will be focused more towards segments with low confidence.

By using only the latest identified segment, our heuristic supports *online* annotation of segments. This is different than offline pool-based active learning methods described in Section II-B, which typically require as input large histories of segments from which to choose annotations. These methods are not suitable for our scenario which necessitates annotation from the user's short-term memory.

Depending on the characteristics of the activity data, specific instances of the framework can be created by plugging in suitable algorithms for segmentation and classification. For example, Bulling et al. [38] review numerous machine learning procedures and algorithms (such as data preprocessing, feature extraction, model building) that have been used in HAR. In subsequent sections, we instantiate the framework to contexts involving non-periodic and periodic activities using analysis on public HAR datasets and also on live streams of periodic activities data generated from a field study.

We measure classification performance using the Weighted F-Score, as follows:

$$F = \sum_{i=1}^{N_C} 2w_i \frac{P_i R_i}{P_i + R_i}$$

where N_C is the number of activity classes, P_i is the precision, R_i is the recall of activity class i and $w_i = N_i / \sum N_i$ is the relative numerosity of class i in the test set.

To assess the effectiveness of Online Active Learning, we follow the practice suggested by Settles [4] and contrast the performance from our Online Active Learning heuristic with the learning performance from a more naive procedure – Random Selection – which we will use as a baseline in the next section. Random Selection does not use context to inform annotation requests and only triggers requests at random.

IV. Simulated Online Active Learning

We present learning simulations based on publicly available benchmark HAR datasets for non-periodic and periodic activities. We demonstrate the theoretical capabilities of our Online Active Learning framework for bootstrapping fully personalised HAR models in practical contexts. To this end, we evaluate the performances of personalized models bootstrapped with OAL and, separately, with Random Selection (of the annotations to be used for learning) and we test the hypothesis that OAL outperforms RS in terms of recognition accuracy. Results show that OAL improves over RS by up to 5% for non-periodic activities and by up to 8.5% for periodic activities.

We first present the evaluation results for the annotation method on the Opportunity dataset [10]. Opportunity, which

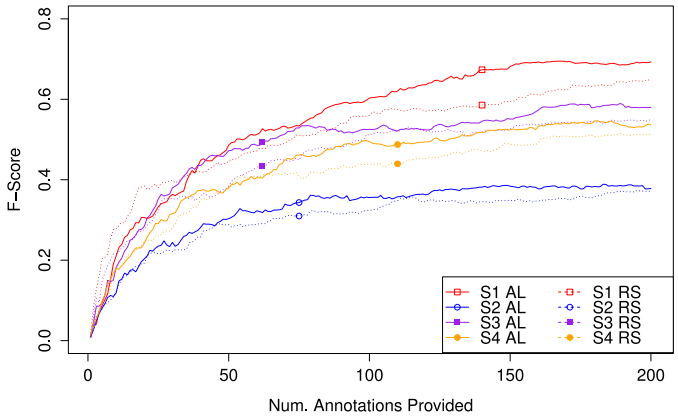


Fig. 2: Learning Curve for Opportunity; Legend: S1 - subject 1, AL - online active learning, RS - Random Selection

consists of non-periodic activities specific to daily routines, was collected with the aim of furthering state-of-the-art machine learning for activity recognition and presents a challenging benchmark dataset. Secondly, we present the evaluation results for the impact of our annotation framework on the USC-HAD [11] and the PAMAP [12] datasets, which consist of periodic activities typical of fitness contexts.

A. Non-periodic Activities

The Opportunity dataset [10] is a public state-of-the-art HAR dataset. It consists of 17 non-periodic activities of daily living and it is known as a challenging classification task.

1) Machine Learning: The machine learning algorithms used here exploit the temporal structure of non-periodic activities. Features were not extracted from the acceleration signals, but, instead, entire activities were classified based on their acceleration timeseries. A k-Nearest Neighbours [39] model was employed to distinguish between activities using Dynamic Time Warping [40] as a measure of dissimilarity between acceleration timeseries.

Data from five accelerometer positions (*upper right arm*, *lower right arm*, *upper left arm*, *lower left arm* and *back*) was used, similar to [22], [23]. In order to emphasize the effects of Online Active Learning versus Random Selection, an ideal segmentation procedure (one which identifies the exact boundaries of each activity) was assumed. We underlined previously that activity segmentation is a complicated research topic. However, for this simulation, we opted for a single major variable affecting performance – the annotation heuristic.

2) Simulation Procedure: Annotating from a continuous stream of activities was simulated by replaying data segments and presenting them to the annotation decision heuristic. Whenever an annotation was deemed necessary, according to Eq. 1, the ground truth label was revealed and the model was re-trained. Segments that were not annotated were made available for subsequent replay. We used the standard train-test split in Opportunity [10] and, in order to support our hypothesis that our framework can construct fully personalised models, the models were bootstrapped strictly on a per-user basis. Specifically, after each annotation, the user's model was evaluated against the test set associated to the same user, according to the Opportunity dataset specification.

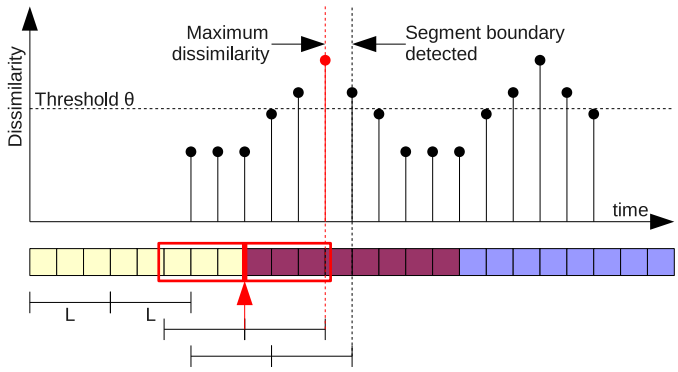


Fig. 3: Automatic Segmentation Strategy (Schematic)

3) Results: Fig. 2 shows that Online Active Learning outperforms Random Selection for all four subjects. When averaging across all subjects, performance is improved for 87.5% of the points on the learning curve and performance gains of up to 5% are registered.

B. Periodic Activities

We evaluated the applicability of our method on the publicly available USC-HAD [11] and PAMAP [12] datasets. The USC-HAD dataset consists of movement data collected about 12 activity classes from 14 participants. The PAMAP dataset consists of movement data collected about 12 activity classes from 9 subjects. The activities in both datasets are periodic ones, which are typical for healthcare and fitness applications.

1) Machine Learning: A sliding window procedure over the acceleration timeseries was used to generate frames of 5s (a common practice for periodic activities). For each frame, the following 9-dimensional feature vectors were extracted: *X axis mean*, *Y axis mean*, *Z axis mean*, *X axis variance*, *Y axis variance*, *Z axis variance*, *X and Y axis correlation*, *Y and Z axis correlation*, *Z and X axis correlation*. A Bootstrap Aggregator [41] with 30 Naive Bayes [42] base classifiers was used as a model builder. In our analysis, this model builder yielded superior performance over others commonly used in HAR (logistic regression, decision trees, k-Nearest Neighbours).

We took advantage of the relative uniformity of movement in periodic activities by including a segmentation procedure that operated on the sequence of consecutive monitored feature vectors. The procedure, adapted from the video segmentation literature [35], is illustrated in Fig. 3. Intuitively, it operates a sliding window over the stream of detected feature vectors. The feature vectors in the first half of the window are compared to those in the second half. If the degree of dissimilarity registered a local maximum above a fixed threshold, then a change in activity was deemed to have taken place. We did not use this segmentation procedure on the Opportunity dataset, because, for non-periodic activities, the assumption of uniformity across an entire activity does not generally hold.

More precisely, consider a window size $K = 2L$, with $L > 0$, covering the most recently produced feature vectors.

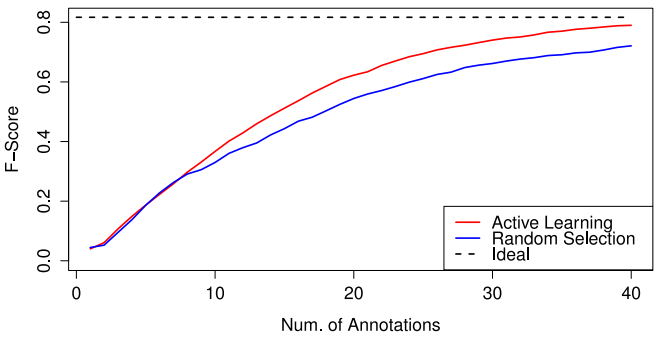


Fig. 4: Average Learning Curves for PAMAP

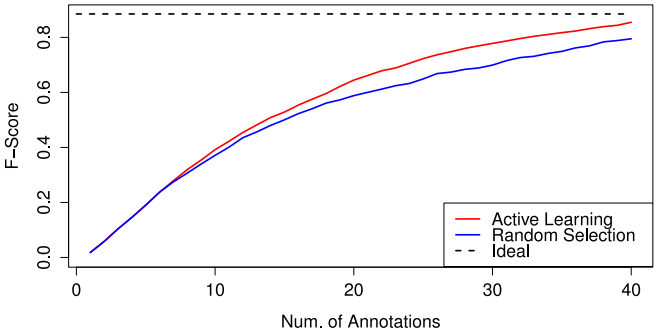


Fig. 5: Average Learning Curves for USC-HAD

The feature vectors indexed by $1, 2, ..., L$ represent the first half of the segmentation window and $L+1, L+2, ..., 2L$ the second half of the window. An *aggregate distance* is defined as the mean of the pair-wise dissimilarity between the vectors in the first half of the window and the vectors second half of the window. If the dissimilarity is greater than a predefined threshold θ, then a segment boundary is signalled between the frames indexed L and $L+1$. This means that the last feature vector of the current segment is L and the first feature vector of the new segment is $L+1$. The process is repeated with every new feature vector that becomes available. This segmentation procedure yields the sequence of segments, each of which, according to the Online Active Learning method, the user may be asked to annotate.

As a dissimilarity measure, all pairwise Euclidean distances between the feature vectors in $\{1, 2, ...L\}$ and those in $\{L+1, L+2, ..., 2L\}$ are averaged. Let $\{d_k\}_{k\in\mathbb{N}}$ be the sequence of average distances generated from the stream of feature vectors. A segment boundary is flagged between the frames causing d_k if d_k is a local maximum ($d_k > d_{k-1}$ and $d_k > d_{k+1}$) and d_k is above a fixed threshold θ ($d_k > \theta$).

The segmentation procedure is online because it continuously operates only on a recent sub-stream (the latest $2L$ feature vectors) in order to decide whether an activity segment has ended. A new segment is detected with a delay of $L+1$ frames, as shown in Fig. 3, so the horizon within which users are requested to provide annotations is limited to the duration of just a few frames.

2) Simulation Procedure and Results: Fully personalised HAR models are bootstrapped for every user in the dataset. For each user, a data stream is simulated by replaying contiguous sequences of frames from a randomly sampled activity. The automatic segmentation procedure operates over this input and outputs a sequence of segments which are candidates for annotation. Due to the limited sizes of the datasets, we limit the sizes of the replayed activities to 3-6 frames and stop after accumulating 200 frames (approx. 40 annotations).

Figs. 4 and 5 contrast the performance of Online Active Learning and Random Selection using the USC-HAD and the PAMAP datasets. For periodic activities, Online Active Learning registers clear improvement over Random Selection. For PAMAP, Online Active Learning scores improvements of up to 8.5% for 92.5% of the points on the learning curve, while for USC-HAD there are performance gains of up to 8% for 92.5% of the points. Additionally, because the learning curves are not saturated/flatlined, as in the Opportunity case, the figures also include the upper baseline ("Ideal" in the figures) of the F-Score that could be attained by using all the labelled data in the dataset. In both cases, Online Active Learning approaches this ideal level of accuracy more quickly than Random Selection.

For both the non-periodic and periodic cases, Online Active Learning is a justifiably useful annotation strategy because, as our results show, Online Active Learning outperforms Random Selection in terms of HAR model accuracy.

V. User Study of Online Active Learning

In this section we demonstrate the effects of applying the annotation framework to a naturalistic field study involving voluntary participants. A mobile app was used to process live streaming data from worn accelerometers and to collect annotations from users. We aim to show that, also in realistic conditions where users provide annotations without expert supervision, personalised activity models can be bootstrapped using our OAL framework and that model improvement is achieved using our approach. Specifically, we score OAL against a *strawman* classifier – a simplistic model that systematically outputs the predominant label in the training set. Results show that OAL outperforms a strawman classifier by $38-47\%$ in terms of recognition accuracy.

A. Accelerating Annotation Requests

As in the previous section, the user of the system is prompted to provide labels according to Eq. 1. The mechanism uses the confidence in prediction of a bootstrapped model to issue the probability p_{ask} of asking the user for annotation.

While Online Active Learning yields gains in recognition performance, the speed with which initial annotations are requested is very low. The problem arises with the initial annotated segment which results in a training set with a single label. At this stage, this training set leads the classifier to systematically predict that label for all new frames and with 100% confidence. The issue, which we call the *Ignorant Classifier Problem*, is illustrated in Fig. 6. Little diversity in the training set triggers classifier overconfidence which, in turn, causes very slow improvement. This behaviour can persists for many iterations afterwards unless more diverse labels are

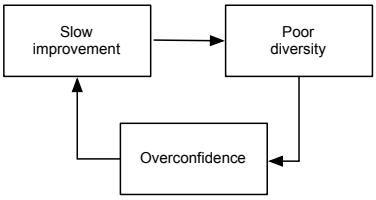


Fig. 6: The Ignorant Classifier Problem.

discovered, as pointed out by Sculley [25] and as observed by us in our simulations in Section IV. Therefore, initially, the classifier is misguided to confidently but incorrectly classify new segments and this makes Eq. 1 ineffective.

The *Ignorant Classifier* was not a problem in the simulations in the previous section because it was possible to cycle over numerous data points without generating annotation requests. Eventually, some annotations would occasionally be requested because the asking probability is nonetheless non-zero. The training set would eventually diversify and informed decisions would follow. However, initial overconfidence is problematic in the context of a realistic deployment because of the time constraints in our field experiments.

In order to address the Ignorant Classifier problem, we introduce another annotation heuristic called the Novel Activity Detector (NAD) which complements Online Active Learning. The NAD aims to increase label diversity in the early stages of learning. Using another annotation decision heuristic, the NAD generates annotation requests of its own in parallel to Online Active Learning. In our user study, we experimented with two NAD versions. The first version, the *Speculative* NAD, favours a high throughput of annotation requests, but it can be intrusive to users. The second version, the *Restrained* NAD, limits the number of annotation requests to one per activity class, but this version carries the risk of not discovering some labels.

1) Novel Activity Detector – Speculative Version: The initial NAD version used a Bag of 30 Naive Bayes classifiers. Normally, each Naive Bayes classifier would output a vector of *probability scores* for each label. These scores would be transformed into actual probabilities by scaling them such that they sum to 1. The unscaled probability scores are proportional to the scaled probabilities, so that they too are representative of the model's prediction confidence. However, for the 1-label Ignorant Classifier case, the confidence is always 100% and hides the potential variation of the corresponding unscaled probability score. We want to detect changes in confidence even if the training set contains only one label or very little label diversity. Therefore, we propose Eq. 2 as a NAD formula that uses the unscaled prediction confidence $p_{conf}^{unscaled}$ to generate its own probability p_{ask} of asking the user for an annotation. Because of the lack of scaling, the NAD works equally well for any number of classes known to the model, including for the one class case.

$$p_{ask} = exp(-\gamma \cdot \ln p_{conf}^{unscaled}) \tag{2}$$

The unscaled probability scores are extremely sensitive to the high dimensionality of the input space, where small input variability leads to huge variations in probability scores. To limit this variability, (1) the dimensionality of the input space was reduced by using only a subset of features (from the original set of features) and (2) a logarithmic factor was introduced to further reduce variability down to a manageable range.

The resulting asking probability is given by Eq. 2 which is linearly scaled to $[0, 1]$[1]. This annotation mechanism is similar to the main one used in Eq. 1. However, the Speculative NAD focuses high asking probabilities only in the region of very small unscaled probability scores.

We used a small dataset collected offline and concluded that $\gamma = 0.02$ would be a good value to highlight novel activities while ignoring known labels. However, the first participants who used the Speculative NAD had noted a large number of *sitting* activities they were asked to annotate. The cause of excessively many annotation requests was traced to the NAD which was too sensitive. The NAD would trigger annotation requests for known activities that were executed slightly differently even if this was natural variability and this is the cause of high throughput we noted earlier.

2) Novel Activity Detector – Restrained Version: We also experimented with a lower throughput NAD that did not engage users as often as the Speculative NAD. For the other half of the participants, we used a more restrained NAD mechanism of generating annotation requests. In this version, we leveraged the set of annotations collected from the participants who used the Speculative NAD and constructed a *population* model using a Nearest Neighbour classifier from the median feature vectors of each class – a training set of 9 points. Median values were used here because they are generally insensitive to outliers. Furthermore, a Nearest Neighbour classifier using such a small training set would still be able to deliver very fast online classifications. The second version of the NAD used this activity model to classify newly computed feature vectors. The NAD maintained a list of user-provided labels, but as classified by the population model. Namely, when the population model classifies a new activity a_i^{pop} which was not estimated before, then an annotation is requested for the current segment. Regardless of what label the user provides, say $a_i^{provided}$, the label a_i^{pop} is marked as annotated even if $a_i^{provided} \neq a_i^{pop}$. This ensures that the NAD never requests more than one annotation per activity class. Consequently, most annotation requests come from Online Active Learning. The Restrained version of the NAD, despite using a population model, still supports the bootstrapping of a fully personalized activity model, just as the Speculative NAD.

B. User Study

We now describe the experiment design of our user study. We set up a naturalistic case study that involved users in the annotation process and bootstrapped personalized activity models. The users were not supervised by anyone and annotations were provided using a phone app in a live manner, as the users executed the activities. For the panel of participants,

[1]The Weka implementation of the Naive Bayes classifier protects against numeric underflow by enforcing a minimum unscaled probability of 10^{-75}. This minimum value is used to scale p_{ask}.

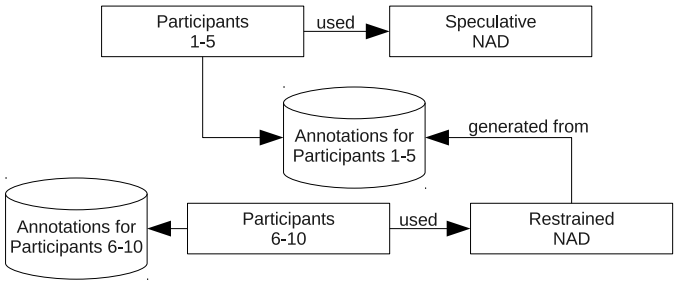


Fig. 7: NAD Usage Throughout the Experiment.

ten office workers were recruited to engage in sedentary and non-sedentary activities in their usual office environment.

The first five participants responded to annotation requests generated by Online Active Learning and the Speculative NAD. The data from this subset of participants was used to train the Restrained NAD which was used in conjunction with Online Active Learning by the last five of the participants, as detailed in Fig. 7.

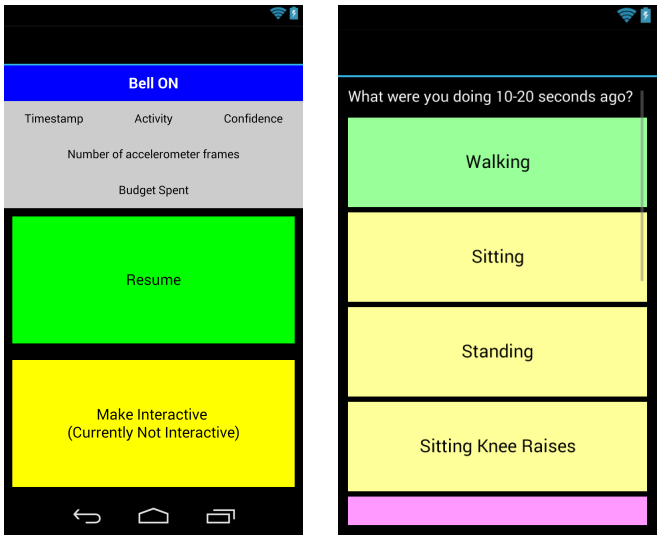
1) Activities: Nine light physical activities that could realistically be performed at the office were targeted for the user study: *sitting*, *standing*, *sitting knee raises*, *walking*, *squats*, *calf raises*, *torso side to side*, *torso twists* and *torso back to forward*. This is a diverse set of activities that is arguably suitable for an office environment. All activities require relatively little energy expenditure and, in retrospect, none of the participants mentioned any difficulty in performing them. Also, no special equipment or areas are needed and the setup is fully compatible with our mobile and online scenario.

Activity data was collected with WAX9 Bluetooth Low Energy accelerometers[2] placed in four locations on the participants' bodies: the right foot, the right lower leg, the right upper leg and the chest. These locations captured key movements for the proposed activities. The accelerometer data was transmitted wirelessly to an Android smartphone where app coordinated data processing and user interaction.

2) Protocol: In terms of participants, we recruited ten colleagues from our department, who were not affiliated with our research. We demonstrated the target activities and asked the participants to include $8-10$ repetitions of each activity in their daily routine at the office. Participants were informed that they could execute the activities in any order, at any time and could take breaks as they wished. The participants were not supervised while the experiment was under way in order to ensure that there was no interference in how the activities were performed or what or how annotations were provided. The participants were also informed that the app would not prompt or guide them to perform activities in any way, but rather would simply react to registered activities. This shows that the annotation framework is decoupled from the experimental protocol and, so, it could be applied in similar contexts, without the user having to observe a certain protocol.

The duration of the experiments was divided in two parts, each with its own annotation request mechanism. In the first part, only informed annotation requests were generated. In the second part, some random annotation requests were included in order to obtain additional annotations for performance evaluation purposes. In total, the participants were monitored by our mobile app for 55 hours and, during this time, they annotated 3 hours and 20 minutes worth of sensor data.

[2]http://axivity.com/product/5 Accessed 26.06.2015



(a) Main Screen (b) Interactive Annotation Screen

Fig. 8: App Screens

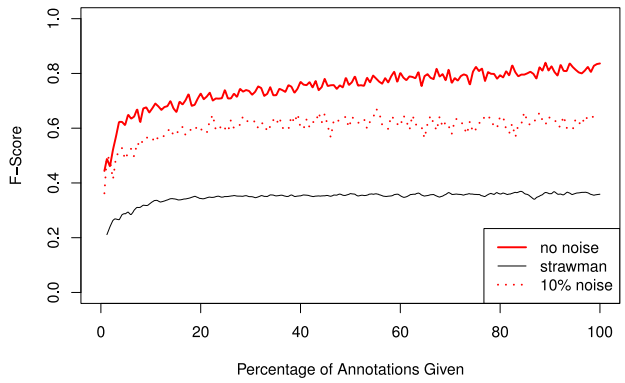
C. Mobile App

We implemented an Android mobile app that collected live streaming data from wireless accelerometers. Additionally, the app provided the user interface and learning machinery for bootstrapping fully personalised HAR models.

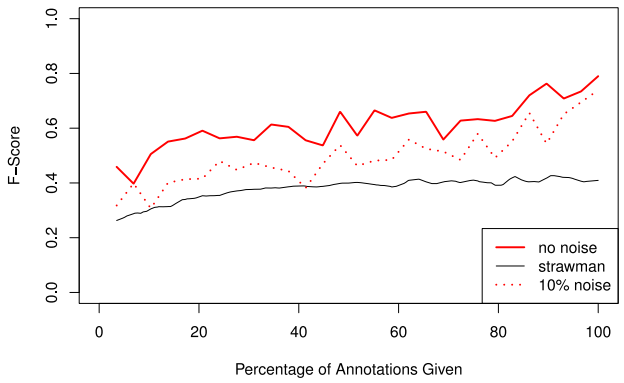
1) User Interface: The app implemented a straightforward interaction protocol supported by two Android activities, as shown in Fig. 8. The main activity (Fig. 8a), gave the user control over the behaviour of the app (such as audio feedback, pausing/resuming acceleration monitoring or enabling/disabling just the notification prompts) and basic monitoring information, such as the current timestamp, detected activity, confidence, etc.

When an annotation was deemed necessary, the annotation screen (Fig. 8b) was automatically presented to the user. The user could select the label for the newly delineated segment from a predefined menu of activities with a single tap. The annotation screen is presented for a maximum of 15 seconds before the annotation request is discarded and the user is no longer able to provide the annotation. This maximum delay with which an annotation can be provided ensures that the user recall is not stretched past a certain duration which could affect the user's memory.

2) Learning Machinery: The mobile app implements an online machine learning pipeline using Weka [43] and supports personalized model bootstrapping using our annotation framework. We integrate a very similar machine learning pipeline used in the simulations with periodic movements.



(a) Participants 1-5 (Speculative NAD)



(b) Participants 6-10 (Restrained NAD)

Fig. 9: Average Learning Curves

We apply a 5s sliding window with 50% overlap[3] over a live stream of acceleration data and, for each of the four triaxial accelerometers, we extract the same 9 features. The classifier is a Bootstrap Aggregator with 30 Naive Bayes base classifiers. The app runs the continuous segmentation procedure procedure described earlier and annotation decisions are made for each newly identified segment, using Eq. 1, the NAD and sometimes randomly, as previously explained. If a segment needs to be annotated, the user is prompted to provide a label using the screen depicted in Fig. 8b. Once the user provides a label, the model is updated with the newly annotated segment.

D. Results

We compare the performances of the bootstrapped model with two other models. Firstly, the performance of a *strawman* is considered – a simplistic model that systematically predicts the most prevalent label in the training set. Secondly, annotation noise was artificially added, by randomly altering 10% of the labels in the training set.

The contrast between the three learning curves is illustrated in Fig. 9 which shows that learning from user-provided annotations is substantially superior to simple strawman classification, outperforming it by $38 - 47\%$. In addition, because annotation noise worsens model performance, we conclude that user veracity when responding to annotation requests is essential to model bootstrapping.

In contrast to Section IV where we simulated annotations on public HAR datasets, we did not perform Random Selection in our user deployment. The reason was the very short overall duration of most activities relative to the duration of the *sitting* and *walking* activity as our participants' daily routine is naturally very sedentary. Analysis shows that 43% of the non-sitting and non-walking activities annotations (acquired using the combination of methods described earlier) would have been lost through Random Selection, if it had been enacted. For this reason, we did not use RS in our user study as a replacement for the OAL and NAD annotation decision heuristics.

Overall, our results show clear performance improvements of personalized models which are bootstrapped from user-provided annotations in a naturalistic deployment. Online Active Learning, combined with a Speculative NAD, can ensure the discovery of many activities, even if they are rare. A less engaging NAD, like the Restrained one, imposes less annotation effort on behalf of the user, but also attracts less annotation requests for rare activities which means that model personalisation is delayed, relative to a high throughput NAD.

VI. Conclusions and Further Work

In this paper we devised a method to collect annotations from HAR system users and to bootstrap personalised activity models solely from user-provided annotations. To this end, we designed an Online Active Learning framework for monitoring a user's stream of activities and identifying prospective annotations using a very limited horizon on time. We employed an online annotation methodology which relied on the users' short-term memory as the source of ground truth. Therefore, any annotation had to refer to the most recent activity so that the user can remember it. This means that more common offline approaches, such as pool-based active learning, are inapplicable to our stream-based scenario. However, despite the fundamental differences between our online method and previously existing offline approaches, our results show that personalised HAR models can be bootstrapped without expert supervision or retrospective analysis of data.

By constantly monitoring the user's activities, we have shown that it is possible to reason about the usefulness of each potential annotation so that the user is interrupted only with requests for highly critical annotations. Using public HAR datasets, we evaluated our framework in multiple scenarios concerning both non-periodic and periodic activities. Results show that attempting to maximize the performance gains from annotations using our Online Active Learning heuristic leads to performance gains compared to when naively requesting annotations using Random Selection.

[3]In order to better detect activity changes, which, unlike in the simulations earlier, do not necessarily happen exactly on window boundaries.

Additionally, we deployed our interactive machine learning pipeline within a naturalistic user study. The annotation process was accelerated with two Novel Activity Detectors that diversified the labels in the training set by issuing informed annotation requests even when classifier suffers from initial overconfidence. Results show that, even within a realistic deployment where users provide a limited number of annotations using just their short-term memory as the source of ground truth, personalized models register substantial performance improvement as annotations accumulate.

For further work, we are investigating segmentation strategies that are applicable to non-periodic activities and we intend to instantiate the framework in this case as well.

This work was supported by the Research Councils UK Digital Economy Programme [grant number EP/G066019/1 - SIDE: Social Inclusion through the Digital Economy]. The authors would like to thank Daniel Roggen for his contribution to the current work.

REFERENCES

[1] M. Weiser, "The computer for the 21st century," *Scientific American*, 1991.

[2] G. D. Abowd, "What next, Ubicomp? Celebrating an intellectual disappearing act." in *Proc. UbiComp*, 2012.

[3] M. L. Eisen, J. A. Quas, and G. S. Goodman, *Memory and Suggestibility in the Forensic interview (Personality and Clinical Psychology Series)*, 2001.

[4] B. Settles, "Active learning literature survey," University of Wisconsin, Madison, Tech. Rep., 2010.

[5] M. Stikic, K. Van Laerhoven, and B. Schiele, "Exploring semi-supervised and active learning for activity recognition," in *Proc. ISWC*, 2008.

[6] B. Longstaff, S. Reddy, and D. Estrin, "Improving activity classification for health applications on mobile devices using active and semi-supervised learning," in *Proc. PervasiveHealth*, 2010.

[7] H. Alemdar, T. van Kasteren, and C. Ersoy, "Using active learning to allow activity recognition on a large scale," in *Proc. AmI*, 2011.

[8] S. Bagaveyev and D. J. Cook, "Designing and evaluating active learning methods for activity recognition," in *Adjunct Proc. UbiComp*, 2014.

[9] R. Liu, T. Chen, and L. Huang, "Research on human activity recognition based on active learning," in *Proc. ICMLC*, 2010.

[10] R. Chavarriaga, H. Sagha, A. Calatroni, S. T. Digumarti, G. Tröster, J. D. R. Millán, and D. Roggen, "The Opportunity challenge: A benchmark database for on-body sensor-based activity recognition," *Pattern Recognition Letters*, 2013.

[11] M. Zhang and A. A. Sawchuk, "Usc-had: A daily activity dataset for ubiquitous activity recognition using wearable sensors," in *Proc. SAGAWARE*, 2012.

[12] A. Reiss and D. Stricker, "Creating and benchmarking a new dataset for physical activity monitoring," in *Proc. PETRA*, 2012.

[13] T. Ploetz, P. Moynihan, C. Pham, and P. Olivier, "Activity Recognition and Healthier Food Preparation," in *Activity Recognition in Pervasive Intelligent Environments*. Atlantis Press, 2010.

[14] C. Hooper, A. Preston, M. Balaam, P. Seedhouse, C. Pham, D. G. Jackson, T. Ploetz, and P. Olivier, "The French Kitchen: Task-Based Learning in an Instrumented Kitchen," in *Proc. UbiComp*, 2012.

[15] C. Ladha, N. Hammerla, P. Olivier, and T. Ploetz, "ClimbAX: Skill Assessment for Climbing Enthusiasts," in *Proc. UbiComp*, 2013.

[16] T. Ploetz, C. Chen, N. Y. Hammerla, and G. D. Abowd, "Automatic Synchronization of Wearable Sensors and Video-Cameras for Ground Truth Annotation - A Practical Approach," in *Proc. ISWC*, 2012.

[17] T. van Kasteren, A. Noulas, G. Englebienne, and B. Kröse, "Accurate activity recognition in a home setting," in *Proc. UbiComp*, 2008.

[18] E. Hoque, R. F. Dickerson, and J. A. Stankovic, "Vocal-diary: A voice command based ground truth collection system for activity recognition," in *Proc. Wireless Health*, 2014.

[19] J. M. Smyth and A. A. Stone, "Ecological Momentary Assessment Research In Behavioral Medicine," *Happiness Studies*, 2003.

[20] S. S. Intille, L. Bao, E. M. Tapia, and J. Rondoni, "Acquiring in situ training data for context-aware ubiquitous computing applications," in *Proc. CHI*, 2004.

[21] S. Intille, E. Tapia, J. Rondoni, J. Beaudin, C. Kukla, S. Agarwal, L. Bao, and K. Larson, "Tools for studying behavior and technology in natural settings," in *Proc. UbiComp*, 2003.

[22] J. Rebetez, H. F. Satizábal, and A. Perez-Uribe, "Reducing user intervention in incremental activity recognition for assistive technologies," ser. Proc. ISWC, 2013.

[23] T. Miu, P. Missier, D. Roggen, and T. Plötz, "On strategies for budget-based online annotation in human activity recognition," in *Adjunct Proc. UbiComp*, 2014.

[24] Z. S. Abdallah, M. M. Gaber, B. Srinivasan, and S. Krishnaswamy, "Adaptive mobile activity recognition system with evolving data streams," *Neurocomputing*, 2015.

[25] D. Sculley, "Online Active Learning Methods for Fast Label-Efficient Spam Filtering," in *Conference on Email and AntiSpam*, 2007.

[26] V. Pejovic and M. Musolesi, "Interruptme: Designing intelligent prompting mechanisms for pervasive applications," in *Proc. UbiComp*, 2014.

[27] J. Fogarty, S. E. Hudson, C. G. Atkeson, D. Avrahami, J. Forlizzi, S. Kiesler, J. C. Lee, and J. Yang, "Predicting human interruptibility with sensors," *ACM Trans. Comput.-Hum. Interact.*, 2005.

[28] A. Kapoor and E. Horvitz, "Experience Sampling for Building Predictive User Models : A Comparative Study," in *Proc. CHI*, 2008.

[29] S.-L. Chua, S. Marsland, and H. Guesgen, "Behaviour recognition from sensory streams in smart environments," in *Proc. AAAI*, 2009.

[30] N. C. Krishnan and D. J. Cook, "Activity recognition on streaming sensor data," *PMC*, 2014.

[31] G. Okeyo, L. Chen, H. Wang, and R. Sterritt, "Dynamic sensor data segmentation for real-time knowledge-driven activity recognition," *PMC*, 2014.

[32] H. Junker, O. Amft, P. Lukowicz, and G. Tröster, "Gesture spotting with body-worn inertial sensors to detect user activities," *Pattern Recognition*, 2008.

[33] N. C. Krishnan, P. Lade, and S. Panchanathan, "Activity gesture spotting using a threshold model based on adaptive boosting," in *Proc. ICME*, 2010.

[34] T. Ploetz, N. Hammerla, A. Rozga, A. Reavis, N. Call, and G. D. Abowd, "Automatic Assessment of Problem Behavior in Individuals with Developmental Disabilities," in *Proc. UbiComp*, 2012.

[35] M. Cooper, "Video segmentation combining similarity analysis and classification," *Proc. ICM*, 2004.

[36] D. Roggen, K. Förster, A. Calatroni, and G. Tröster, "The adarc pattern analysis architecture for adaptive human activity recognition systems," *Journal of Ambient Intelligence and Humanized Computing*, 2013.

[37] M. Kurz, G. Hölzl, A. Ferscha, A. Calatroni, D. Roggen, and G. Tröster, "Real-time transfer and evaluation of activity recognition capabilities in an opportunistic system," *machine learning*, 2011.

[38] A. Bulling, U. Blanke, and B. Schiele, "A tutorial on human activity recognition using body-worn inertial sensors," *ACM Comput. Surv.*, 2014.

[39] D. W. Aha, D. Kibler, and M. K. Albert, "Instance-based learning algorithms," *Machine Learning*, 1991.

[40] T. Giorgino, "Computing and visualizing dynamic time warping alignments in r: The dtw package," *Journal of Statistical Software*.

[41] L. Breiman, "Bagging predictors," *Machine Learning*, 1996.

[42] G. H. John and P. Langley, "Estimating continuous distributions in bayesian classifiers," in *Proc. UAI*, 1995.

[43] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The weka data mining software: An update," *SIGKDD Explor. Newsl.*, 2009.